Microblog Summarization Using Conversation Structures

LI, Jing

A Thesis Submitted in Partial Fulfilment

of the Requirements for the Degree of

Doctor of Philosophy

in

Systems Engineering and Engineering Management

The Chinese University of Hong Kong

July 2017

Thesis Assessment Committee

Professor MENG Mei Ling Helen (Chair)

Professor WONG Kam Fai William (Thesis Supervisor)

Professor CHENG Hong (Committee Member)

Professor Wan Xiaojun (External Examiner)

Abstract of thesis entitled:

Microblog Summarization Using Conversation Structures

Submitted by LI, Jing

for the degree of Doctor of Philosophy

at The Chinese University of Hong Kong in July 2017

Due to the widespread use of Internet, microblog is now a popular digital communication platform. However, large volume of microblog messages produced daily makes it difficult to understand the key information behind these messages. This thesis proposes *microblog summarization* models that can identify salient excerpts from microblog messages, digest them, and represent them in a succinct form for easy reading.

Microblog posts are *short*, *colloquial*, *unstructured*, and can cover a *variety of topics*. Traditional summarizers, relying heavily on textual information, are therefore ineffective for microblog summarization. Other information such as user authority and message popularity also do not necessarily indicate summary-worthy content.

To overcome these predicaments, we propose to *use conversation structures for microblog summarization.* We organize microblog messages as conversation trees based on *reposting* and *replying* relations, and extract the embedded discourse structures for summarization.

Due to the highly diverse information on microblog, intermediate topic representations have been proven useful to microblog summarization. We first cluster microblog messages into various topics by making use of coarse-grained "leader-follower" discourse information. And then, we summarize each topic cluster based on its embedded conversational structures. Focusing on summarization of a single conversation tree, we propose two summarization frameworks: 1) a random-walk based model leveraging on the coarse-grained "leader-follower" discourse structures, and 2) a weakly-supervised probabilistic model, which separates fine-grained discourse to distill summary-worthy content.

摘要

隨著互聯網的廣泛應用，如今，微博已經成為一個廣受歡迎的電子社交平臺。然而，由於每天產生的海量微博信息，使得用戶難而有效地理解其中表達的重要意義。此論文專註於微博摘要模型的研究，目的是能夠提取、理解海量微博中的重要信息，並以易於閱讀的方式呈現出來。

微博具有內容間短、語言貼近日常交流、數據非結構化的特點，並且內容包含多種多洋的主題。因此，嚴重依賴於文本內容信息的傳統的摘要模型，並不適用於於處理微博信息。

針對微博信息的多洋性, 以主題模型技術作為的信息去表達已被證實對自動摘要任務效果顯著。有見及此，我們首先從微博中挖掘出粗粒度的“領導者-追隨者”結構，然後將微博信息劃分為不同的主題。接下來，我們基於對話結構對每一個主題

的微博進行摘要。著眼於單課微博對話樹，我們提出兩種微博摘要的框架：1）結合粗粒度的“領導者-追隨者”對話結構的基於隨機遊走模型的算法；以及2）弱監督的概率模型，這種模型能夠刻畫細粒度的對話結構信息，並能夠同時學習情感信息和對摘要有用的內容信息。

Preface

Part of the research work in this thesis has been published in the peer-reviewed conference proceedings.

Chapter 3 has been publicized in Jing Li, Ming Liao, Wei Gao, Yulan He, Kam-Fai Wong: Topic Extraction from Microblog Posts Using Conversation Structures. ACL (1) 2016: 2114-2123 [63].

Chapter 4 has been publicized in Jing Li, Wei Gao, Zhongyu Wei, Baolin Peng, Kam-Fai Wong: Using Content-level Structures for Summarizing Microblog Repost Trees. EMNLP 2015: 2168-2178 [61].

Note that some details and results in this thesis and their previous publications vary due to different experiment settings.

Bibographical Sketch

Jing Li was born in Xiamen, Fujian, China. In July 2013, she obtained her B.S. degree from the Department of Machine Intelligence, Peking University, Beijing, China. One month later, she was enrolled as a PhD student in the Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong. During Jan – Apr, 2016, she was an intern at School of Engineering and Applied Science, Aston University, Birmingham, UK. And during Feb – May, 2017, she worked at the College of Computer and Information Science, Northeastern University, Boston, USA, as a visiting scientist. Jing has publicized papers in premier NLP conferences including ACL and EMNLP. She has also served as program committee member for top-tier conferences in the ACL community, such as EMNLP and EACL.

Acknowledgement

This thesis would never be possible without the support and help from my family, friends, colleagues, etc.

First and foremost, I am immensely lucky to have Prof. Kam-Fai Wong as my supervisor. Throughout my PhD study in CUHK, Kam-Fai always patiently taught me how to conduct research and encouraged me to dive into the most challenging and worthwhile research problems. He has gone out of his way to support my every opportunity in career despite of his busy schedule. I would never thank him enough.

I would like to extend my gratitude to my committee members: Prof. Helen Meng, Prof. Hong Cheng, and Prof. Xiaojun Wan. They generously shared their brilliant ideas and insightful suggestions with me. Without their tremendous help, this thesis would never have been accomplished.

Besides, I was fortunate to visit Aston University and Northeastern University, and got the chance to work with two incredibly excellent researchers, Yulan He and Lu Wang. During my visit, Yulan and Lu gave me sufficient freedom to explore whatever topics that interested me and was always ready to help. Their beneficial advice has helped me shape many of the ideas in this thesis. Also, my colleagues, friends, and collaborators in Aston and NEU contributed to make these two visits perfect. They are: Lei He, Kechen Qin, Xinyu Hua, Rui Dong, Ryan Muther, Nick Beauchamp, Sarah Shugars, Zhengxing Chen, Rundong Li, Jinghan Yang, Paola Rizzo, Chaima Jemmali, Elin Carstensdottir, Liwen Hou, Xiaofeng Yang, Yuan Zhong, etc.

In addition, I express my special appreciation to Wei Gao, who guided me to write my first research paper step by step. It was a great pleasure having Wei as my comrade to fight against those deadlines. Also, I want to thank my former and current colleagues in CUHK: Binyang Li, Lanjun Zhou, Zhongyu Wei, Junwen Chen, Baolin Peng, Ming Liao, Shichao Dong, Xingshan Zeng, Dingmin Wang, Yuqiu Qian, Gabriel Fung, Bo Li, Jing Ma, Huimin Wang, Hao Wei, Kangfei Zhao, Bei Shi, Piji Li, Yi Liao, Can Lu, Pengfei Liu, Xixin Wu, Shiyin Kang, etc. I really enjoyed the wonderful brain-stormings and exciting discussions with these fabulous and super smart people. They contributed, more or less, to the interesting ideas in this thesis. I also want to thank Ray Cheng from the General Office and Phyllis Leung from the Technical Support Team of the Department of SEEM, CUHK. Without them kindly taking care of every non-research matter, I would not have been able to focus on my research.

Moreover, I am grateful to Xinghong Hu, Qin Cao, Yishi Lin, Tian Wang, Shangzhu Wu, Yumeng Hao, Tian Zhang, Donghao Ren, Zhenhuang Wang, Zhen Liu, Jingshu Shen, Yanyi Ye, Jie Huang, Mengfei Ren, Yangfan Chen, etc. How lucky I am to be able to meet and have these people as my dear friends. Their moral encouragement plays an important role in my getting over obstacles throughout my PhD study.

Last but not least, I dedicate my deepest love and appreciation to the most important and beloved people in my life: my father, my mother, and my husband Runze Zhang. Their unlimited love and support is the most powerful strength that helps me get through all the difficulties when chasing after my dreams.

This thesis is dedicated to my family: Hongnian Li and Fang Liu, my parents, and Runze Zhang, my husband.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Since the last decade, we have witnessed the flourish of the Internet. It has broken the limitation of region, space, and time, and provides tremendous convenience to information exchange by revolutionizing the way we communicate. Recently, microblog, a social networking channel over the Internet, further accelerates communication and information broadcasting. Nowadays, microblog platforms, such as Twitter[1] and Sina Weibo[2], are im-

[1] twitter.com

[2] weibo.com

The story of Olivia Olivia is a typical office lady in Hong Kong. At June 13 midnight, Hong Kong Apple fans stayed up late to watch Apple WWDC 2016 and get a first look at Apple's latest iPhone 7. But Olivia failed to do that as she had to go to bed early. The next morning, she received an advertising microblog post, which said, "iPhone 7 is revealed!". She was curious about the details of WWDC 2016 that highlighted the key improvements of iPhone 7 over the previous models. She started to track the relevant microblog messages, but soon gave up as she was overwhelmed by the countless replicated and uninformative messages.
The story of Jian Yang Jian Yang was a first year CUHK student from the Mainland China. Once he settled down in Hong Kong, he joined a WeChat group that involves over 200 members related to Mainland alumni. The first message he received was "We are happy." Jian Yang was very confused: who were "we" and why were "we happy"? To find out the answer, he spent hours to trace the history chatting records.

Table 1.1: Two stories in big data era

portant outlets for people to share information and to voice opinions. They are widely used for real-life applications such as instant detection of breaking events [68, 89, 118], real-time ad-hoc microblog search [26, 64], user profiling [119], etc.

However, the explosive growth of microblog data far outpaces human beings' speed of reading and understanding. Table 1.1 tells two stories that commonly happen in our daily life. Although advanced big data technology renders large scale data analysis feasible, such as those established on microblogs, users now face the challenging problem of information explosion. This problem can seriously affect the effectiveness of many online applications, e.g., digital marketing [43] and stock prediction [11]. Thus, there is an urgent need for effective summarization systems to prevent users from superficial understanding of huge volume and unmanageable amount of user-generated social media content. Otherwise, wrong and even risky decisions would be made.

To help social media users distill useful information out of massive and noisy messages, this thesis focuses on the research problem of *automatic summarization of microblog posts*. In solving this problem, there are four challenges:

Challenge I: Short and informal language style. While many existing summarization methods, typically based on textual features like cosine similarity [77] and term frequency–inverse document frequency (TF-IDF) [28], have demonstrated their usefulness to process formal and well-written documents such as news reports and scientific articles, their effectiveness on microblog texts is still questionable [16]. This is because, unlike classical texts, microblog messages are typically short, informal, colloquial, and unstructured. Lack of contextual information and the problem of data sparseness render the unreliability of textual features and hence lead to the difficulty of microblog summarization. Novel summarization models are therefore required to fill the gaps.

Challenge II: Huge volume of messages. The increasing popularity of microblog services results in massive amount of messages. Take Sina Weibo, the most widely-used microblog website in China, as an example. In 2015, there were 100 million active users and 100 million messages on average each day.[3] This motivates our study of automatic summarization systems for representing massive messages in a concise and comprehensi-

[3] en.wikipedia.org/wiki/Sina_Weibo#cite_note-5

ble form [95]. In addition to huge volume of data, replicated and noisy microblog messages make it extremely time-consuming for human editors to annotate summary-worthy content. As a result, it is difficult to construct large-scale benchmark corpus for microblog summarization [16]. Without enough ground-truth summaries for model training, supervised models based on machine learning techniques, despite of its popularity in summarization applications, would suffer from the problem of over-fitting. For this reason, an effective microblog summarization model that does not rely on large-scale human written summaries for training is required.

Challenge III: Simple social signals are not enough. In order to improve summarization, prior research work incorporates social signals, e.g. author influence, message popularity, etc [16, 25, 73]. However, these features are not necessarily useful to signal important messages. For example, an influential celebrity may post a popular message without any summary-worthy content.

Challenge IV: Wide variety of topics. Microblog websites are generally open-domain. Due to various interests, users post, share, or comment on microblog messages that cover various topics including sports, politics, social issues, business, entertainment, etc. Automatically clustering massive and diverse microblog posts into topics are therefore a critical step before summarization [39]. A typical procedure is to first cluster posts by topics and then to summarize each topical cluster [14, 25, 74, 82, 99, 103]. This helps reduce redundancy, and yields well-structured and easy-to-read summaries. However, microblog topic clustering is another challenging task. Conventional topic clustering methods cannot perform well on short and informal microblog posts due to the sparseness of message-level word co-occurrence patterns and key words for topic representation. Therefore, to effectively summarize microblog messages, additional efforts should be made to improve microblog topic clustering.

To overcome the above challenges, we propose to use *conversation structures* and extract *discourse* therein to recognize summary-worthy content.

Microblog conversations are composed by *reposting* and *replying* messages. They are commonly used for free interactions with messages, e.g., information sharing, feeling expression, etc. Reposting and replying on microblogs enable us to build up a tree-structured multi-party conversational message cluster. This practically resembles a conversation tree. Structurally, nodes of a conversation tree represent messages and edges represent reposting and replying relations. By organizing messages as conversation trees, we can effectively enrich contextual information, connect related messages, and alleviate data sparseness exhibited in microblog posts.

If we consider a conversation tree as a single document and the messages therein as sentences in the document, we can borrow the concept of *discourse* to encode structural information embedded in microblog conversation trees. Discourse, such as elaboration, contrast, background, etc., is originally defined to capture the semantic or pragmatic connection of sentences in a document, and has been proven useful for identifying important sentences in conventional single document summarization [75, 76]. We argue that discourse structures embedded in microblog conversation trees can also help microblog summarization.

Though agreements exist that discourse analysis can similarly be applied to capture conversation structures. Due to the colloquial language style and complex interaction structures embedded in conversations, so far, researchers have not reached a consensus of exactly how to describe discourse structures in conversations. However, essential commonalities involves recognizing performative function of each utterance, namely, dialogue acts, as first-level conversation structures [98, 109]. Due to the

[O]	Immigration Ban Is One Of Trump's Most Popular Orders So Far.
[R1]	I love you Mr. President! This is really a good order 😀
[R2]	good order??! you are terribly wrong! this is racialism! Not all Muslims are bad!
[R3]	I feel sad for those poor guys... 😭

Table 1.2: A snippet of microblog conversation path about Trump's immigration ban.

short nature of microblog posts, we assume a message as an utterance and follow the paradigm of dialogue acts to describe discourse as message-level annotations, such as "statement" and "response", which indicate functions and pragmatic roles of microblog messages in context of conversation trees.

In general, discourse describes what role each message plays in the discussion flow of a conversation tree. Different discourse roles vary in probabilities to contain summary-worthy content, which covers the key focus of the conversation. For example, in the conversation path displayed in Table 1.2, message [R2] doubts the assertion of "immigration ban is good", and raises new discussion focus on "racialism". This in fact serves as a more important summary candidate than message [R1], which simply *responds* to its parent. For this reason, in this thesis we propose to identify messages with "good" discourse roles that describe key focuses and salient topics of a microblog conversation tree. This enables us find "good" summary candidates.

We first explore microblog topic extraction based on coarse-grained discourse in conversation trees through "leader-follower" relations (Chapter 3). Based on the results of topic extraction, we produce informative summary to each topic cluster using conversation structures. We propose two models based on: 1) straightforward coarse-grained "leader-follower" discourse (Chapter 4), and 2) fine-grained discourse via jointly modeling content and sentiment (Chapter 5).

1.2 Contributions

The contributions of this thesis lie in two research areas of Natural Language Processing (NLP) for microblog texts: *topic ex-*

traction and *text summarization.*

1.2.1 Topic Extraction from Microblog Posts

Topic models can derive an intermediate topic representation for given documents and have been considered useful to many downstream tasks including summarization [86].

Owing to their fully unsupervised manner and ease of extension, Bayesian topic models, e.g., Probabilistic Latent Semantic Analysis (pLSA) [42] and Latent Dirichlet Allocation (LDA) [9], have achieved huge success over the past decade. Nevertheless, ascribing to their reliance on document-level word co-occurrence patterns, the progress is still limited to formal conventional documents such as news reports and scientific articles.

The key to improve performance of topic models on short and informal microblog messages is to enrich the context and to alleviate data sparseness. We propose to use the structures of conversations. For each conversation tree, we capture a coarse-grained “leader-follower” discourse structure by differentiating messages as *leader messages* and *follower messages*. Leader messages, or *leaders* for short, shift the conversation focus to different topics or raise key aspects of previously focused topics, e.g., [R2] in Table 1.2. Leaders generally contain salient words in topic description, such as “racialism” and “Muslims”. On the other hand, follower messages, or *followers* for short, do not introduce any new information but simply echo topics of their parents, e.g., [R1] and [R3] in Table 1.2, which follow what have been raised by the leaders and often contain non-topic words, such as “love” and “sad”. Though it is difficult to define leaders and followers precisely, like the concept of summary (or relevant document in information retrieval). According to our empirical study in Section 3.5.1, human annotators can differentiate them reasonably well given a conversation path. In particular, a leader message can be posted by an opinion leader, i.e., a user who constantly provides opinions effective to others [59].

But the author of a leader message should not necessarily be an opinion leader. For example, a message that initiate a sub-topic based on a specific topic raised by an opinion leader can also serve as a leader message.

We present a novel topic model that incorporates "leader-follower" discourse for both topic assignments and topical word identification. In addition, we have publicized a large real-world microblog dataset containing over 60K conversation trees for the task of microblog topic extraction.[4] More details are described in Chapter 3.

1.2.2 Microblog Summarization

The research of automatic text summarization can be traced back to 1950s [22, 95]. Nowadays, automatic summarization techniques have already been applied to many real-life applications like the reddit bot "autotldr"[5]. However, the effective-

[4] www1.se.cuhk.edu.hk/lijing/data/microblog-topic-extraction-data.zip

[5] www.reddit.com/user/autotldr/

ness of existing summarization systems is undermined due to the short and informal nature of microblog messages, which leads to the severe problems of data sparseness and lack of context.

This thesis proposes an innovative solution for microblog summarization. It makes use of the *discourse structures provided by microblog conversations.* In this way, we can enrich contextual information, which is used to facilitate identification of summary-worthy content. Discourse structure is traditionally referred to functional relations of sentences within a coherent document. Previous work has shown that inter-sentence discourse relations can indicate salient content for single-document summarization [75].

A message in a conversation tree is analogous to a sentence in a document. We focus on summarizing one single conversation tree comprised of an original post (as root) and all its reposts and replies. Another well-known branch of microblog summarization is real-time microblog summarization (RTS) in

Text Retrieval Conference (TREC).[6] Our work is different from RTS in the following ways: 1) our *input* is a microblog conversation tree, while the input of RTS is a query and its related microblog stream; 2) we *output* summaries for generic purpose, while RTS generates summaries for specific information of interest requested by a user.

Chapter 3 shows that coarse-grained discourse, i.e., *"leader-follower" structure*, is useful to topic extraction. Intuitively, effectively differentiating leader messages, which raise new information, and follower messages, which mainly contain uninformative response, helps filter out unimportant noise and moves one step closer to finding good summary candidates. In Chapter 4, we explore the usefulness of the *coarse-grained "leader-follower" discourse* structure for microblog summarization.

Chapter 5 introduces a novel *fine-grained discourse* based summarization approach. We additionally capture sentiment-

[6] `trecrts.github.io/`

specific information because of its prevalence on microblog platforms [82], and use shifts of sentiment to detect discourse roles. For example, a message expressing a different sentiment from its parent is likely to “doubt” on previous discussions, e.g., [R2] in Table 1.2. And messages with “doubt” discourse usually raise controversy, potentially lead discussions in descendents, and therefore tend to contain summary-worthy content. A weakly-supervised probabilistic model is proposed for microblog summarization. It jointly infers representations of *discourse*, *sentiment*, and *content* with minimal supervision from emoji lexicon.

Our automatic microblog summarization are unique in the following ways:

Microblog posts organized as conversation trees. We propose a brand new concept of representing microblog posts as conversation trees by connecting microblog posts based on *reposting* and *replying* relations. Conversation tree structure helps enrich context, alleviate data sparseness, and in turn improve summarization.

Coarse-grained and fine-grained conversation discourse for microblog summarization. We propose to use *coarse-grained* and *fine-grained* discourse structures embedded in the conversation trees for summarization. *Coarse-grained discourse* is represented by distinguishing two message-level discourse roles: *leaders* and *followers*. We present a random-walk based summarization framework incorporating the outputs of CRF-based leader detection model (Chapter 4). *Fine-grained discourse* is latent clusters of discourse words inferred simultaneously with sentiment and content components in a weakly supervised manner (Chapter 5).

Public corpus for microblog summarization. We have released a real-world microblog corpus[7] that contains 10 conversation trees on popular Chinese microblog Sina Weibo[8], which

[7] `www1.se.cuhk.edu.hk/lijing/data/repost\_tree\_summ.zip`

[8] Sina Weibo has a similar market penetration as Twitter according to Fobes news:

is constructed following the previous settings reported in Chang et al. [16].[9] Each conversation tree has more than 12K messages on average and covers discussions about social issues, breaking news, jokes, celebrity scandals, love, and fashion, which matches the official list of typical categories for microblog posts released by Sina Weibo.[10] For each conversation tree, the corpus contains three human-generated summaries as reference. This corpus, to the best of our knowledge, being the only publicly available dataset of its kind so far, would be beneficial to future research in microblog summarization.

□ **End of chapter.**

China's Weibos vs US's Twitter: And the Winner Is?

[9] The corpus of Chang et al. [16] is not publicly available.

[10] `d.weibo.com/`

Chapter 2

Background Study

This thesis builds on diverse steams of previous work in lines of *discourse analysis*, *topic modeling*, and *text summarization*.

2.1 Discourse Analysis

Discourse defines the semantic or pragmatic relations between text units and reflect of the architecture of textual structure. This section reviews the prior research of traditional discourse schema for a single document (Section 2.1.1) and discourse extension to represent conversation structure (Section 2.1.2). And in Section 2.1.3, we describe the existing discourse-based summarization models and highlight the difference of our work to previous related research.

2.1.1 Traditional View of Discourse: Structural Art Behind a Coherent Document

It has been long pointed out that a *coherent* document is not simply a collection of independent and isolated sentences. Every two successive sentences are never happened to be juxtaposed. Instead, extra-sentential factors and intra-sentential information together tells the full story. Literally, a *coherent* document is like a well-structured house. Every piece of text units (which can be clause, sentence, or paragraph) therein is tightly connected with each other, and is meaningful only be understood in context. Thus, theoretically, how to understand and compute the structure of a *coherent* document becomes the key in discourse processing.

Linguists have striven to the study of discourse analysis in the ever since Ancient Greece [4]. In 1970s and 80s, a series of important work came out and shaped the modern concept of *discourse* [45], which depicts connections between text units, and reveals the structural art behind a *coherent* documents.

Rhetorical Structure Theory (RST) [76] was one of the most influential discourse theories. According to its assumption, a *coherent* document can be represented by text units at different levels (e.g., clauses, sentences, paragraphs) in hierarchical structure of *tree*. In particular, the minimal units in RST, i.e., leaves of the tree structure, are defined as sub-sentential clauses, namely, Elementary Discourse Units (EDUs). Adjacent units are linked by rhetorical relations, e.g., condition, comparison, elaboration, etc.

Based on RST, early work employs hand-coded rules for automatic discourse analysis [78, 112]. Later, thanks to the development of large-scale discourse corpus, e.g., RST corpus [12], Graph Bank corpus [121], and Penn Discourse Treebank (PDTB)

[91], data-driven and learning-based discourse parsers that exploit various features via manual design [5,30,31,52,70,107,110] and representative learning [48,62] became popular.

In particular, the discourse learning method presented in Chapter 3 and 4 is based on manually-crafted features (Section 3.2). And in Chapter 5, we use representative learning to capture discourse information, where features are extracted purely from data.

2.1.2 Extending Discourse Analysis to Understand Conversation Structure

Internet has revolutionized the way we communicate and facilitated the emergence of multifarious online communication platforms, e.g., emails, forums, and microblogs. This brings a constant flood of information exchange in a form similar to conversations. This leads to the demand of automatic conversation analysis technique. The first step is discourse analysis for conversation structure [109].

Although discovering hierarchical discourse structures, e.g., RST [76], have been proven possible for formal and edited documents (see Section 2.1.1), existing discourse parsers mostly focus on the detection of *dialogue acts* (DA), a useful first level conversational discourse structure, because of the complex structure and informal language embedded in conversations. Specifically, a DA represents the shallow discourse role that captures illocutionary meanings of a utterance, e.g., "statement", "question", "agreement", etc [109].

Automatic dialogue act taggers have been traditionally trained in a supervised way depending on the pre-defined tag inventory and annotated data [6, 19, 109]. In particular, the CRF-based leader detection model in Chapter 3 and 4 is a special DA tagger based on DA inventory with only two tags, i.e., "*leader*" and "*follower*".

However, DA definition is generally domain-specific and manually designed by experts. The data annotation process is slow and expensive leading to the limitation of available data for training [24, 51, 53, 98]. These issues are pressing in the Internet era where new domains of conversations and even new dialogue act tags are boomed [51, 98]. For this reason, researchers proposed unsupervised or weakly supervised dialogue act taggers that identify indicative discourse word clusters based on probabilistic graphical models [21, 51, 98]. In particular, the discourse detection module of Chapter 5 falls into this category.

2.1.3 Discourse and Summarization

NLP researchers have confirmed that discourse structures could improve summarization. The empirical study by Louis et al. [75] compares the impact of structural discourse and non-discourse features on the task of extractive single document summarization, and reports that the discourse structure could best indicate salient summary candidates and could also be complementary to non-discourse features for summarization.

In the context of conversation summarization, previous work has also shown that pre-detected DAs are useful for identifying summary-worthy contents in conversations from emails [87], forums [8], meetings [85, 116], etc.

The above systems are however ineffective for microblog summarization. The main reason is that DA definition is typically domain-dependent. It is problematic to use DA inventory designed for other conversation domain, like meetings, to capture discourse structure of microblog conversations [98].

For this reason, in Chapter 3 and 4, we propose new tagset, i.e., *"leader"* and *"follower"*, to reflect coarse-grained conversation discourse for microblogs. Moreover, the above prior work ignores the error propagation from discourse tagger to summarization, which is an issue we addressed (see Chapter 4). In Chapter 5, we infer representations of fine-grained discourse in a weakly-supervised manner without reliance to either manually crafted tags or annotated data.

2.2 Topic Modeling

The last decade has witnessed the huge success of topic models. It can automatically discover word clusters describing latent "topics" representations from texts. Section 2.2.1 gives a brief introduction of Latent Dirichlet Allocation (LDA) proposed by Blei et al. [10], which forms the bases of many topic models including the models presented in Chapter 3 and 5. Section 2.2.2 compares the our topic model in Chapter 3 and related work of microblog topic modeling. Section 2.2.3 discusses how previous work utilizes representations captured by topic modeling for summarization.

2.2.1 LDA: Springboard of Topic Models

Topic models aim to discover the latent semantic information, i.e., topics, from texts and have been extensively studied. One of the most popular and well-known topic models is Latent Dirichlet Allocation (LDA) [10].

Suppose that each document d is a mixture of topics θ_d, and each topic z is captured by a word mixture ϕ_z, then the writing procedure of a document can be described as the repeat of the following two steps: (1) The writer first picks a topic z_0 from the topic mixture of the document; and (2) from the word mixture of z_0, he selects a word w_0 and writes it on the paper. Now observed a collection of documents and words in them, how can we guess the topic mixture θ_d of each document d and the word mixture of ϕ_z for each topic z so as to maximize the probability of seeing these documents (and their words)?

In the above writing process, LDA assumes seeing a topic in (1) and a word in (2) as a face of a k-sided fair die occurring in a independent die-rolling experiment, and thereby represents each topic mixture θ_d and each word mixture ϕ_z as multinomial distribution [10].

In parameter estimation, because calculating the integral of the marginal likelihood is intractable, to facilitate model inference, LDA encodes conjugate prior for multinomial distributions, i.e., Dirichlet distribution parameterized by α (for document-topic distribution θ_d) and β (for topic-word distribution ϕ_z).

Specifically, one of the most widely applied methods for learning parameters of LDA and its extensions is collapsed Gibbs Sampling [36], which is also adopted in the posterior inference of Chapter 3 and 5. It considers the assignments to hidden multinomial variables, e.g., seeing a topic z_0 in every step (1) of LDA's writing procedure, as the states in a Marcov chain. And the transition matrices are defined as the conditional probabilities given a complete assignment of all other hidden variables. The smoothing effect of positive Dirichlet parameters, e.g., α and β in LDA, ensure that every number in the transition matrix falls into the interval $(0, 1)$ and that stationary distribution of the Markov process exists and is unique. Therefore, when the

Markov chain converges, we can infer the multinomial distributions based on states of the hidden variables.

More details about conjugacy of Dirichlet and multinomial distributions, and the Gibbs sampling steps can be found in Gregor Heinrich's tutorial [41].

LDA plays an important role in semantic representation learning research and serves as the springboard of many famous topics models, e.g., HLDA [9], Author-Topic Model [100], etc. Besides "topic" modeling, it has also inspired *discourse* [21, 51, 98] or *sentiment* [49, 66, 67] detection without or with weak supervision, which is the basis of Chapter 5. In particular, Lazaridou et al. [58] simultaneously explores discourse and sentiment in a multi-task Bayesian model. However, none of them jointly exploits discourse, sentiment, and content for summarization, which is an issue that Chapter 5 tackles.

2.2.2 Topic Modeling on Microblog Posts

Though many topic models have been shown effective in extracting topics from conventional documents, prior research has demonstrated that standard topic models, essentially relying on document-level word co-occurrences, are unsuitable for processing microblog messages as severe data sparsity exhibited in short and informal texts [44, 117]. Therefore, how to enrich and exploit context information becomes a main concern. Weng et al. [119], Hong et al. [44] and Zhao et al. [129] first heuristically aggregate messages posted by the same user or sharing the same words before applying classic topic models to extract topics. However, such a simple strategy poses some problems. For example, it is common that a user has various interests and posts messages covering a wide range of topics. Ramage et al. [96] and Mehrotra et al. [80] used hashtags as topical labels to train supervised topic models. However, these models depend on large-scale hashtag-labeled data for model training, and their performance is inevitably compromised when facing unseen topics irrelevant to any hashtag in the training data. This problem exists because of the rapid change and wide variety of topics in social media.

SATM [93] combined short texts aggregation and topic induction into a unified model. But in their work, no prior knowledge is given to ensure the quality of text aggregation. This can therefore affect the performance of topic inference. In Chapter 3, we organize microblog messages as conversation trees based on reposting and reply relations, which is a more advantageous message aggregation strategy.

Another line of research tackled the word sparseness problem by modeling word relations instead of word occurrence patterns in documents. For example, the Gaussian Mixture Topic Model (GMTM) [108] utilized word embeddings to model the distributional similarities of words and then inferred clusters of words represented by word distributions using Gaussian Mixture Model (GMM), which captures the notion of latent topics. Nevertheless, GMTM heavily relies on meaningful word embeddings that require a large volume of high-quality external resources for training.

Biterm Topic Model (BTM) [125] directly explores unordered word-pair co-occurrence patterns in each individual message. Our model in Chapter 3 learns topics from aggregated messages based on conversation trees, which naturally provide richer context since word co-occurrence patterns can be captured from multiple relevant messages involved in the same conversation.

2.2.3 Topic Modeling and Summarization

Researchers have confirmed that the topic representation captured by topic models is useful to summarization [86]. Specifically, there are two different goals of using topic models in existing summarization systems: (1) to separate summary worthy content and non-content background (general information) [13,38,46], and (2) to cluster sentences or documents into topics, and summaries are then generated from each topic cluster for minimizing redundancy [79, 101, 105].

Focusing on summarization of a single conversation tree, our model in Chapter 5 lies in the research line of (1). In the future, if facing multiple conversation trees, it is necessary to follow (2) to cluster microblog posts before summarization, which can be processed by microblog topic models like that in Chapter 3.

2.3 Text Summarization

2.3.1 Conventional Summarization

The research of automatic text summarization has a history of over half a century [22, 95]. The goal of text summarization is to automatically produce a succinct summary for one or more documents that preserves important information [95].

Generally, text summarization techniques can be categorized into extractive and abstractive methods [22]. Extractive approaches focus on how to identify salient contents from original texts whereas abstractive approaches aim at producing grammatical summaries by text generation.

The summarization methods in in this thesis falls into extractive summarization category. In fact, most microblog summarization systems adopt extractive approaches because microblog posts are informal and noisy. This makes it difficult to generate grammatical summaries. Section 2.3.2 gives a more detailed discussion.

Here we discuss some most representative methods in text summarization research.

Graph-based methods. They are built upon the PageRank algorithm [28,83,88]. The input sentences are represented as vertices of a complete graph, and edges reflect text similarities, e.g., cosine similarity, between two connected sentences. By ranking the sentences similar to PageRank, graph-based method can select the top-ranking sentences as summary that have the highest information coverage to the rest of the sentences. Graph-based methods are easy to extend, thus have many variations. For example, DivRank [81] adds reinforcement factors so as to reduce the redundancy in top-ranking vertices (sentences), which serves as the basis of our summarization method in Chapter 4.

Integer programming (IP) based methods. The key of IP-based methods is to design the objective function, which generally encodes how much information covered by the produced summary, and the constraints, which restrict the summary length [34, 60, 123]. IP-based methods enable linguists to define what a "good" summary is. For example, TopicSum [38] recognizes a "good" summary by minimizing the KL divergence between the unigram distribution of the generated summary with an pre-induced content distribution, which is the basis of the summary extraction step in Chapter 5 (see Section 5.3).

Machine learning (ML)-based methods. The developments of machine learning triggers the popularity of ML-based methods. A general procedure is to cast summarization into a binary classification problem and train supervised machine learning (ML) models, e.g., SVM and CRF, combining various features [20,33,35,75,104,122,130]. We do not utilize ML-based methods for microblog summarization because of their dependence on large-scale gold-standard summaries for training, which is difficult to obtain for microblog.

2.3.2 Microblog Summarization

Recently, the development of social media has made microblog summarization a hot topic. Most prior work is on event-level or topic-level summarization, which follows the strategy (2) described in Section 2.2.3. Typically, the first step is to cluster posts into sub-events [14, 25, 103] or sub-topics [74, 82, 99], and then the second step generates the summary for each cluster.

Some work tried to apply conventional extractive summarization models directly, e.g., LexRank [28], MEAD [94], TF-IDF [47], Integer Linear Programming [71, 111], etc. Sharif et al. [102] casts the problem into optimal path finding on a phrase reinforcement graph. However, these general summarizers were found not suitable for microblog messages due to their informal and noisy nature [16]. Researchers have also considered social signals, e.g., user following relations and retweet count [25, 73], and reported such features useful to summarize microblog posts. This thesis studies microblog summarization by leveraging conversational discourse structure to enrich context of messages.

Chang et al. [16] summarizes Twitter conversation trees by combining user influence signals into a supervised summarization framework. Our summarization work is different from theirs in the following ways: (1) They treat a context tree as a stream of tweets, and we consider conversation tree structure for summarization; (2) They rely on user interactions to calculate *user* *influence* for extracting salient messages, and we focus on how to utilize coarse-grained and fine-grained discourse structure embedded conversation trees; 3) Our summarization modules are unsupervised. Therefore, ground-truth summaries are not required for training.

□ **End of chapter.**

Chapter 3

Conversation Structures and Topic Extraction from Microblog Posts

Conventional topic models are ineffective for topic extraction from microblog messages. Because the lack of structural and contextual information among the posts renders poor message-level word co-occurrence patterns. In this chapter, we organize microblog posts as conversation trees based on reposting and replying relations. By doing so, we enrich context information to the alleviate data sparseness problem. We propose a model that generates words according to topic dependencies derived from the conversation structures. Specifically, we propose a novel "leader-follower" discourse structure by differentiating two types of messages: (1) *leader messages*, which initiate key aspects of previously focused topics or shift the focus to different topics, and (2) *follower messages* that do not introduce any new information but simply echo topics from the messages that they repost or reply to. Our model explicitly captures the different extents that *leader* and *follower* messages contain the key topical words, thus further enhances the quality of the induced topics. For evaluation, we construct two annotated corpora, one for leader detection, and the other for topic extraction. Experimental results confirm the effectiveness of our method.

3.1 Introduction

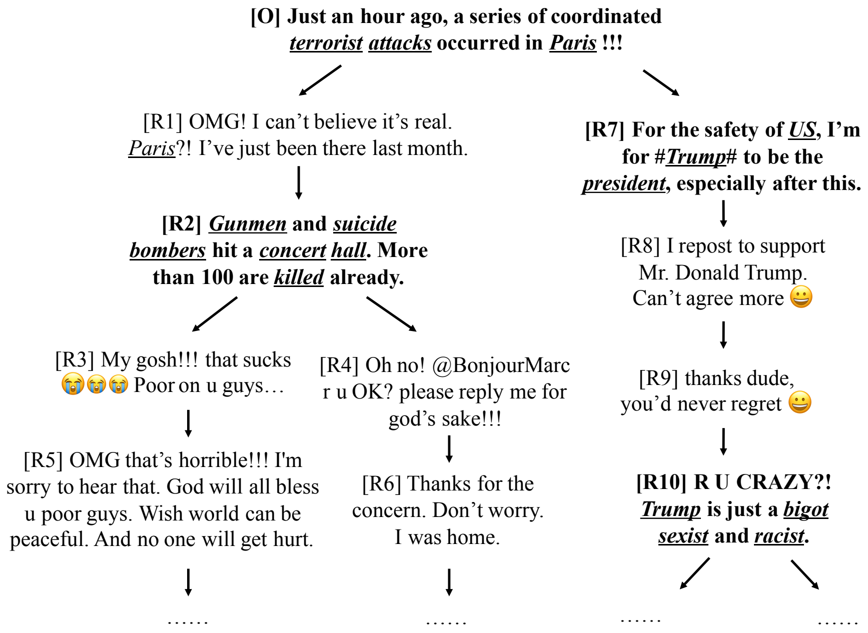
The increasing popularity of microblog platforms results in a huge volume of user-generated short posts. Automatically modeling topics out of such massive microblog posts can uncover the hidden semantic structures of the underlying collection, which is useful to many downstream applications such as microblog summarization [39], user profiling [119], event tracking [68], etc.

Popular topic models, like Probabilistic Latent Semantic Analysis (pLSA) [42] and Latent Dirichlet Allocation (LDA) [10], model the semantic relationships between words based on their co-occurrences in documents. They have demonstrated their success in conventional documents such as news reports and scientific articles, but perform poorly when directly applied to short and colloquial microblog content due to the severe sparsity in microblog messages [44, 117].

A common way to deal with short text sparsity is to aggregate short messages into long pseudo-documents. Most of the work heuristically aggregates messages based on authorship [44, 129], shared words [119], or hashtags [80, 96]. Some works directly take into account the word relations to alleviate document-level word sparseness [108, 125]. More recently, a self-aggregation-based topic model called SATM [93] was proposed to aggregate texts jointly with topic inference.

However, we argue that the existing aggregation strategies are suboptimal for modeling topics in short texts. Microblogs allow users to share and comment on messages with friends through reposting or replying, similar to our everyday conversations. Intuitively, the conversation structures not only enrich context, but also provide useful clues for identifying relevant topics. This is nonetheless ignored in previous approaches. Moreover, the occurrence of non-topical words, such as emotional, sentimental, functional and even meaningless words, are very common in microblog posts, which may distract the models from recognizing topic-related key words and thus fail to produce coherent and meaningful topics.

We propose a novel topic model by utilizing the conversation structures in microblogs. We link microblog posts using reposting and replying relations to build conversation trees. Particu-



[O] Just an hour ago, a series of coordinated *terrorist* *attacks* occurred in *Paris* !!!

[R1] OMG! I can't believe it's real. *Paris*?! I've just been there last month.

[R2] *Gunmen* and *suicide* *bombers* hit a *concert* *hall*. More than 100 are *killed* already.

[R3] My gosh!!! that sucks 😭😭😭 Poor on u guys…

[R4] Oh no! @BonjourMarc r u OK? please reply me for god's sake!!!

[R5] OMG that's horrible!!! I'm sorry to hear that. God will all bless u poor guys. Wish world can be peaceful. And no one will get hurt.

[R6] Thanks for the concern. Don't worry. I was home.

[R7] For the safety of *US*, I'm for #*Trump*# to be the *president*, especially after this.

[R8] I repost to support Mr. Donald Trump. Can't agree more 😀

[R9] thanks dude, you'd never regret 😀

[R10] R U CRAZY?! *Trump* is just a *bigot* *sexist* and *racist*.

[O]: the original post; [Ri]: the i-th repost or reply; Arrow lines: reposting or replying relations; Dark black posts: leaders to be detected; Underlined italic words: key words indicating topics

Figure 3.1: An example of conversation tree.

larly, the root of a conversation tree refers to the original post and its edges represent the reposting or replying relations.

Figure 3.1 illustrates an example of a conversation tree, in which messages can initiate a new topic, e.g., [O] and [R7], or raise a new aspect (subtopic) of the previously discussed topics, e.g., [R2] and [R10]. These messages are named as *leaders*, which contain salient content in topic description, e.g., the italic and underlined words in Figure 3.1. The remaining messages, named as *followers*, do not raise new issues but simply respond to their reposted or replied messages following what has been raised in their ancestors and often contain non-topical words, e.g., *OMG*, *OK*, *agree*, etc.

We first detect leaders and followers across paths of conversation trees using Conditional Random Fields (CRF) trained on annotated data. The detected leader and follower information is then incorporated as prior knowledge into our proposed topic model.

Our experimental results show that our model, which captures parent-child topic correlations in conversation trees and generates topics by considering messages being leaders or followers separately, is able to induce high-quality topics and outperformed a number of competitive baselines in experiments.

In summary, our contributions in this chapter are three-fold:

- We propose a novel topic model, which explicitly exploits the topic dependencies contained in conversation structures to enhance topic assignments.

- Our model differentiates the generative process of topical and non-topical words, according to the message where a word is drawn from being a *leader* or a *follower*. This helps the model distinguish the topic-specific information from background noise.

- Our model outperformed state-of-the-art topic models when it was evaluated on a large real-world microblog dataset containing over 60K conversation trees.[1]

3.2 CRF-based Leader Detection Model

Before topic extraction, we first organize microblog posts as conversation trees based on reposting and replying relations among the messages.[2] To identify key topic-related content from collo-

[1] `http://www1.se.cuhk.edu.hk/lijing/data/microblog-topic-extraction-data.zip`

[2] Reposting and replying relations are straightforward to obtain by using microblog APIs from Twitter and Sina Weibo.

quial texts, we differentiate the messages as *leaders* and *followers*, which describes the coarse-grained conversation discourse named as “leader-follower” structures.

A simple way to detect leaders on conversation trees is to directly apply a binary classifier like SVM on each individual message. However, these models assume that messages in conversation trees are independent instead of effectively leveraging abundant contextual information along the conversation tree paths. For instance, [R5] covering rich content may be misclassified as a leader message contextual information is not taken into account. But if we look into its context, we can find that [R5] talks about similar things as [R3], then [R3] classified as a follower indicates the higher chance of [R5] being a follower rather than a leader. The shows the importance of using contextual information in leader detection.

We extract all root-to-leaf paths within a conversation tree structure and detect leaders across each path. We formulate leader detection on conversation tree paths as a sequence tagging problem and utilize a state-of-the-art sequence learning model CRF [56]. By doing so, we can take advantage of the power of CRF in maximizing the likelihood of global label sequences. We adopt CRF rather than other competitive context-sensitive models like SVM^{hmm} [3] mainly due to its probabilistic nature. The predicted probabilities by CRF can provide critical chances for the following summarization procedures to reduce the impact of errors made by leader detection model on summarization.

We map a conversation tree path with n microblogs $(m_1, m_2, \cdots, m_n)$ to a training instance (X, Y). Let $X = (x_1, x_2, \cdots, x_n)$ represents an observed sequence, where x_i denotes the observed feature vector extracted from the i-th microblog m_i, and $Y = (y_1, y_2, \cdots, y_n)$ where y_i is a binary variable indicating whether m_i is a leader or not. CRF defines the discriminative function as a joint distribution over Y given X

as follows:

$$P(Y|X;\theta) \propto \exp\left(\sum_{i,j} \lambda_j f_j(y_i, y_{i-1}, X) + \sum_{i,k} \mu_k g_k(y_i, X)\right)$$

where f_j and g_k are the fixed feature functions, $\theta = (\lambda_1, \lambda_2, ...;$ $\mu_1, \mu_2, ...)$ are the parameters indicating the weights of features that can be estimated following maximum likelihood procedure in the training process. The prediction is done based on dynamic programming. More details can be found in [56]. Table 3.1 lists the features we use for leader detection.

CRF combines both historical and future information for prediction so as to maximize the likelihood of the global label sequences. For this reason, we may encounter the problem of label conflict, i.e., the predictions for the same node in context of different paths might be different. Therefore, we obtain the posterior probability of each node being a leader or follower by averaging the different marginal probabilities of the same node

Lexical features
of terms: the number of terms in m_i
POS: the part-of-speech of each term in m_i
Type of sentence: whether m_i contains a question mark or an exclamation
Microblog-specific features
of emoji: the number of emoji in m_i
of hashtags: the number of hashtags in m_i
of urls: the number of URLs in m_i
of mentions: the number of mentions, or @userName, in m_i
Path-specific features
Sim to neighbors: Cosine similarity between m_i and m_{i+d} where $d \in \{\pm 1, \pm 2, \pm 3\}$
Sim to root: Cosine similarity to the root microblog in conversation tree path

Table 3.1: Features used for leader detection

over all the tree paths that passes through the node. The obtained probability distribution is then considered as the observed prior variable input into our topic model.

3.3 Topic Model and "Leader-follower" Conversation Structures

In this section, we describe how to extract topics from a microblog collection utilizing conversation tree structures.

Intuitively, the emergence of a leader results in potential topic shift. It tends to weaken the topic similarities between leaders and their predecessors. For example, [R7] in Figure 3.1 transfers the topic to a new focus, thus weakens the tie with its parent. We can simplify our case by assuming that followers are topically responsive only up to (hence not further than) their nearest ancestor leaders. Thus, we can dismantle each conversation tree into forest by removing the links between leaders and their parents and produce a set of subgraphs like [R2]–[R6] and [R7]–[R9] in Figure 3.1. We then model the internal topic dependencies within each subgraph by inferring the parent-child topic transition probabilities that satisfy the first-order Markov properties. It is in a similar way as estimating the transition distributions of adjacent sentences in strTM [115]. At topic assignment stage, the topic of a follower will be assigned by referring to its parent's topic and the transition distribution that captures topic similarities of followers to their parents (see Section 3.3.1).

In addition, every word in the corpus is either a topical or

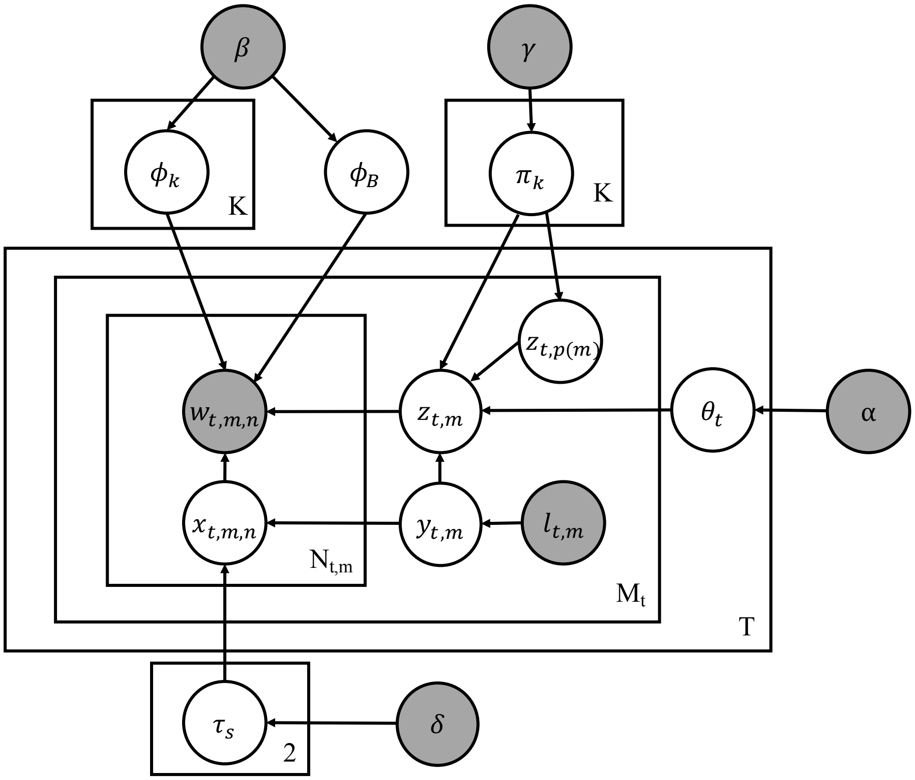


Figure 3.2: Graphical model of our topic model that explores "leader-follower" conversation structures.

non-topical (i.e., background) word, which highly depends on whether it occurs in a leader or a follower message.

Figure 3.2 illustrates the graphical model of our generative process.

3.3.1 Topic Modeling

Formally, we assume that the microblog posts are organized as T conversation trees. Each tree t contains M_t message nodes and each message m contains $N_{t,m}$ words in the vocabulary. The vocabulary size is V. There are K topics embedded in the corpus represented by word distribution $\phi_k \sim Dir(\beta)$ $(k = 1, 2, ..., K)$. Also, a background word distribution $\phi_B \sim Dir(\beta)$ is included to capture the general information, which is not topic specific. ϕ_k and ϕ_B are multinomial distributions over the vocabulary. A tree t is modeled as a mixture of topics $\theta_t \sim Dir(\alpha)$ and any message m on tree t is assumed to contain a single topic $z_{t,m} \in \{1, 2, ..., K\}$.

(1) Topic assignments. The topic assignments of our model is inspired by Griffiths et al. [37], which combines syntactic and semantic dependencies between words. Our model integrates the outcomes of leader detection with a binomial switcher

$y_{t,m} \in \{0, 1\}$ indicating whether m is a leader ($y_{t,m} = 1$) or a follower ($y_{t,m} = 0$), for each message m on the tree t. $y_{t,m}$ is generated by its leader probability $l_{t,m}$, which is the posterior probability output from the leader detection model and serves as an observed prior variable.

According to the notion of leaders, they initiate key aspects of previously discussed topics or signal a new topic shifting the focus of its descendant followers. So, the topics of leaders on tree t are directly sampled from the topic mixture θ_t.

To model the internal topical correlations within the subgraph of conversation tree consisting of a leader and all its followers, we capture parent-child topic transitions $\pi_k \sim Dir(\gamma)$, which is a distribution over K topics. $\pi_{k,j}$ denotes the probability of a follower assigned topic j when the topic of its parent is k. Specifically, if message m is sampled as a follower and the topic assignment to its parent message is $z_{t,p(m)}$, where $p(m)$ indexes the parent of m, then $z_{t,m}$ (i.e., the topic of m) is generated from topic transition distribution $\pi_{z_{t,p(m)}}$. In particular, since the root of a conversation tree has no parent and can only be a leader, we make the leader probability $l_{t,root} = 1$ to force its topic only to be generated from the topic mixture of tree t.

(2) Topical and non-topicalal words. We separately model the distributions of leader and follower messages emitting topical and non-topicalal words with τ_0 and τ_1, respectively. τ_0 and τ_1 both are drawn from a symmetric Beta prior parametererized by δ. Specifically, for each word n in message m on tree t, we add a binomial background switcher $x_{t,m,n}$ controlled by whether m is a leader or a follower, i.e., $x_{t,m,n} \sim Bi(\tau_{y_{t,m}})$. $x_{t,m,n}$ indicates that n is: 1) a topical word and to be generated from the topic-word distribution $\phi_{z_{t,m}}$ ($z_{t,m}$ is the topic of m), if $x_{t,m,n} = 0$; or 2) a background word and to be drawn from background word distribution ϕ_B modeling non-topical information, if $x_{t,m,n} = 1$.

(3) Generation process. To sum up, conditioned on the

```
• Draw $\theta_t \sim Dir(\alpha)$
• For message $m = 1$ to $M_t$ on tree $t$
    – Draw $y_{t,m} \sim Bi(l_{t,m})$
    – If $y_{t,m} == 1$
        ∗ Draw $z_{t,m} \sim Mult(\theta_t)$
    – If $y_{t,m} == 0$
        ∗ Draw $z_{t,m} \sim Mult(\pi_{z_{t,p(m)}})$
    – For word $n = 1$ to $N_{t,m}$ in $m$
        ∗ Draw $x_{t,m,n} \sim Bi(\tau_{y_{t,m}})$
        ∗ If $x_{t,m,n} == 0$
            · Draw $w_{t,m,n} \sim Mult(\phi_{z_{t,m}})$
        ∗ If $x_{t,m,n} == 1$
            · Draw $w_{t,m,n} \sim Mult(\phi_B)$
```

Table 3.2: Generation process of a conversation tree t

hyper-parameters $\Theta = (\alpha, \beta, \gamma, \delta)$, Table 3.2 describes the generation process of a conversation tree t.

3.3.2 Inference for Parameters

We use collapsed Gibbs Sampling [36] to carry out posterior inference for parameter learning. The hidden multinomial variables, i.e., message-level variables (y and z) and word-level variables (x) are sampled in turn, conditioned on a complete assignment of all other hidden variables. Here we give the core formulas in the sampling steps.

We first define the notations of all variables needed by the formulation of Gibbs sampling, which are described in Table 3.3.2. In particular, the various C variables refer to counts excluding the message m on conversation tree t.

—l—m12cm—

$C^{LB}_{s,(r)}$ # of words with background switchers assigned as r and occurring in messages with leader switchers s.

$C^{LB}_{s,(\cdot)}$ # of words occurring in messages whose leader switchers are s, i.e., $\sum_{r\in\{0,1\}} C^{LB}_{s,(r)}$.

$N^{B}_{(r)}$ # of words occurring in message (t,m) and with background switchers assigned as r.

$N^{B}_{(\cdot)}$ # of words in message (t,m), i.e., $N^{B}_{(\cdot)} = \sum_{r\in\{0,1\}} N^{B}_{(r)}$.

$C^{TW}_{k,(v)}$ # of words indexing v in vocabulary, sampled as topic (non-background) words, and occurring in messages assigned topic k.

$C^{TW}_{k,(\cdot)}$ # of words assigned as topic (non-background) word and occurring in messages assigned topics k, i.e., $C^{TW}_{k,(\cdot)} = \sum_{v=1}^{V} C^{TW}_{k,(v)}$.

$N^{W}_{(v)}$ # of words indexing v in vocabulary that occur in message (t,m) and are assigned as topic (non-background) word.

$N^{W}_{(\cdot)}$ # of words assigned as topic (non-background) words and occurring in message (t, m), i.e., $N^{W}_{(\cdot)} = \sum_{v=1}^{V} N^{W}_{(v)}$.

$C^{TR}_{i,(j)}$ # of messages sampled as followers and assigned topic j, whose parents are assigned topic i.

$C^{TR}_{i,(\cdot)}$ # of messages sampled as followers whose parents are assigned topic i, i.e., $C^{TR}_{i,(\cdot)} = \sum_{j=1}^{K} C^{TR}_{i,(j)}$.

$I(\cdot)$ An indicator function, whose value is 1 when its argument inside () is true, and 0 otherwise.

$N^{CT}_{(j)}$ # of messages that are children of message (t, m), sampled as followers and assigned topic j.

$N^{CT}_{(\cdot)}$ # of message (t, m)'s children sampled as followers, i.e., $N^{CT}_{(\cdot)} = \sum_{j=1}^{K} N^{CT}_{(j)}$

$C^{TT}_{t,(k)}$ # of messages on conversation tree t sampled as leaders and assigned topic k.

$C^{TT}_{t,(\cdot)}$ # of messages on conversation tree t sampled as leaders, i.e., $C^{TT}_{t,(\cdot)} = \sum_{k=1}^{K} C^{TT}_{t,(k)}$

$C^{BW}_{(v)}$ # of words indexing v in vocabulary and assigned as background (non-topical) words

---

$C^{BW}_{(\cdot)}$ # of words assigned as background (non-topical) words, i.e., $C^{BW}_{(\cdot)} = \sum_{v=1}^{V} C^{BW}_{(v)}$

For each message m on a tree t, we sample its leader switcher $y_{t,m}$ and topic assignment $z_{t,m}$ according to conditional probability distribution in Eq. (3.1).

$$\begin{aligned}
&p(y_{t,m} = s, z_{t,m} = k | \mathbf{y}_{\neg(t,m)}, \mathbf{z}_{\neg(t,m)}, \mathbf{w}, \mathbf{x}, \mathbf{l}, \Theta) \\
\propto &\frac{\Gamma(C^{LB}_{s,(\cdot)} + 2\delta)}{\Gamma(C^{LB}_{s,(\cdot)} + N^{B}_{(\cdot)} + 2\delta)} \prod_{r \in \{0,1\}} \frac{\Gamma(C^{LB}_{s,(r)} + N^{B}_{(r)} + \delta)}{\Gamma(C^{LB}_{s,(r)} + \delta)} \\
\cdot &\frac{\Gamma(C^{TW}_{k,(\cdot)} + V\beta)}{\Gamma(C^{TW}_{k,(\cdot)} + N^{W}_{(\cdot)} + V\beta)} \prod_{v=1}^{V} \frac{\Gamma(C^{TW}_{k,(v)} + N^{W}_{(v)} + \beta)}{\Gamma(C^{TW}_{k,(v)} + \beta)} \\
\cdot &g(s, k, t, m)
\end{aligned} \tag{3.1}$$

where $g(s, k, t, m)$ takes different forms depending on the value of s:

$$
\begin{aligned}
& g(0,k,t,m) \\
=& \frac{\Gamma(C^{TR}_{z_{t,p(m)},(\cdot)} + K\gamma)}{\Gamma(C^{TR}_{z_{t,p(m)},(\cdot)} + I(z_{t,p(m)} \neq k) + K\gamma)} \\
\cdot& \frac{\Gamma(C^{TR}_{k,(\cdot)} + K\gamma)}{\Gamma(C^{TR}_{k,(\cdot)} + I(z_{t,p(m)} = k) + N^{CT}_{(\cdot)} + K\gamma)} \\
\cdot& \prod_{j=1}^{K} \frac{\Gamma(C^{TR}_{k,(j)} + N^{CT}_{(j)} + I(z_{t,p(m)} = j = k) + \gamma)}{\Gamma(C^{TR}_{k,(j)} + \gamma)} \\
\cdot& \frac{\Gamma(C^{TR}_{z_{t,p(m)},(k)} + I(z_{t,p(m)} \neq k) + \gamma)}{\Gamma(C^{TR}_{z_{t,p(m)},(k)} + \gamma)} \cdot (1 - l_{t,m})
\end{aligned}
$$

and

$$
g(1,k,t,m) = \frac{C^{TT}_{t,(k)} + \alpha}{C^{TT}_{t,(\cdot)} + K\alpha} \cdot l_{t,m}
$$

For each word n in m on t, the sampling formula of its background switcher is given in Eq. (3.2).

$$
\begin{aligned}
& p(x_{t,m,n} = r | \mathbf{x}_{\neg(t,m,n)}, \mathbf{y}, \mathbf{z}, \mathbf{w}, \mathbf{l}, \Theta) \\
\propto& \frac{C^{LB}_{y_{t,m},(r)} + \delta}{C^{LB}_{y_{t,m},(\cdot)} + 2\delta} \cdot h(r,t,m,n)
\end{aligned} \tag{3.2}
$$

where

$$h(r,t,m,n) = \begin{cases} \frac{C^{TW}_{z_{t,m},(w_{t,m,n})}+\beta}{C^{TW}_{z_{t,m},(\cdot)}+V\beta} & \text{if } r=0 \\ \frac{C^{BW}_{(w_{t,m,n})}+\beta}{C^{BW}_{(\cdot)}+V\beta} & \text{if } r=1 \end{cases}$$

3.4 Experiments on Leader Detection

In this experiment, we evaluated the performance of CRF model with our manually-crafted features for leader detection task.

3.4.1 Data Collection and Experiment Setup

Data collection. We first crawled 1,300 different conversation trees using the public PKUVIS toolkit [97]. Given an original microblog post, i.e., root of conversation tree, the toolkit can automatically crawl its complete conversation tree. For each tree, we randomly selected one path and further formed a dataset with 1,300 conversation tree paths. This ensures no two paths shares the same root, thus the dataset can cover a wide variety of contextual information.

Data annotation. We invited three annotators and asked them to independently annotate each message as a leader or a follower in the context of its conversation tree path. The average Cohen's Kappa of each two of the three annotators was 0.52, which is considered as good agreement [32]. We then used the labels agreed by at least two annotators as the ground truth. The training and test process of the leader detection models were conducted on this corpus.

Comparison. We compared the performance of CRF-based leader detection model with three baselines:

RC: Random Classifier as a weak baseline; LR and SVM: two state-of-the-art point-wise supervised models Logistic Regression and Support Vector Machine, respectively;

Implementation and experiment setup. We applied LibLinear toolkit [29] to implement LR and SVM with linear kernel. SVM^{hmm} was implemented by SVM^{struct} toolkit [50]. And the implementation of CRF was based on CRF++.[3] For all the baselines, we used features listed in Table 3.1. The evaluation metrics were precision, recall, and F1 score for the detected leaders. In particular, because of the probabilistic nature of LR and CRF, they yield the probability of a message being a leader or follower instead of making hard decisions like SVM does. Though as a binary classification problem, the best heuristic cutoff of leader probability should be 0.5, because under this circumstance we can always pick up the message annotation (as leader or follower) that has the highest probability. The best empirical cutoff can be different owing to the actual distributions of data and annotation. We tuned the cutoff of leader classification for LR and CRF in 5-fold cross-validation on training set based on F1 scores in 5-fold cross validation (with 1 fold as development set), and obtained the best empirical cutoff of leader probability for LR and CRF were 0.5 and 0.35, respectively.

[3] `http://taku910.github.io/crfpp/`

Models	Cross-validation			Held-out		
	Prec	Rec	F1	Prec	Rec	F1
Random	29.8	49.5	37.3	31.6	49.6	38.6
LR	69.8	69.1	69.4	73.1	67.4	70.1
SVM	**73.9**	64.5	68.9	**74.2**	65.4	69.5
CRF	71.3	**77.8**	**74.4**	66.7	**78.1**	**72.0**

Table 3.3: The performance of leader detection (%)

Other hyper-parameters were also tuned to the same extent in this way.

3.4.2 Experiment Results

Table 3.3 shows the comparison result of 5-fold cross validation on 1,000 conversation tree paths and held-out experiment on 300 complete fresh paths.

We observed that context-sensitive model CRF achieved the best F1 scores. It outperformed LR and SVM by at least 5.7% and 2.7% in cross-validation and held-out evaluation, respectively. This indicates the effectiveness of incorporating structural information for leader detection.

Features	Cross-validation			Held-out		
	Prec	Rec	F1	Prec	Rec	F1
Lexical only	67.4	76.1	71.5	63.8	75.8	69.3
Microblog only	47.6	65.0	54.9	47.1	60.5	53.0
Path only	68.9	78.2	73.2	65.6	77.5	71.1
Lexical + Microblog	65.7	79.9	72.1	61.4	78.4	68.9
Lexical + Path	69.8	77.2	73.3	65.9	76.9	71.0
Microblog + Path	66.6	**82.2**	73.6	62.3	**81.3**	70.5
Full model	**71.3**	77.8	**74.4**	**66.7**	**78.1**	**72.0**

Table 3.4: The performance of CRF model with different feature combinations for leader detection (%)

3.4.3 Feature Analysis

We investigate the effectiveness of different features for leader detection. Table 3.4 reports the performance of CRF model with different combinations of features.

From the experimental results, we have the following observations:

- Path-specific features are more effective than lexical and microblog-specific features in identifying leader messages. This is because leaders and followers are defined in context of a conversation tree. For this reason, contextual information along the conversation paths are useful for leader detection.

- The full model achieved the best performance, which means that combining lexical, microblog, and path specific features are effective to detect leader messages.

3.5 Experiments for Topic Extraction

To evaluate our model, we conducted experiments on large-scale real-world microblog datasets collected from Sina Weibo.

3.5.1 Data Collection and Experiment Setup

Data collection. Because the content of posts are often incomplete and informal, it is difficult to manually annotate topics at large scale. Therefore, we follow Yan et al. [125] to utilize hashtags led by '#', which are manual topic labels provided by users, as ground-truth topical categories of microblog messages. We collected the real-time trending hashtags on Sina Weibo and utilized the hashtag-search API[4] to crawl the messages match-

[4] `http://open.weibo.com/wiki/2/search/topics`

Month	# of trees	# of messages	Vocab size
May	10,812	38,926	6,011
June	29,547	98,001	9,539
July	26,103	102,670	10,121

Table 3.5: Statistics of our three datasets for topic evaluation

ing the given hashtag queries. In the end, we built a corpus containing 596,318 posts during May 1 – July 31, 2014. This dataset is publicly available.[5]

To examine the performance of models with diverse topic distributions, we split the corpus into 3 datasets, each containing messages of one month. Similar to Yan et al. [125], for each dataset, we manually selected 50 frequent hashtags as topics, e.g. #mh17, #worldcup, etc. The experiments were conducted on the subsets of posts with the selected hashtags. Table 3.5 shows the statistics of the three datasets used in our experiments.

Comparison. We considered the state-of-the-art topic models on short texts in comparison.

[5] `http://www1.se.cuhk.edu.hk/~lijing/data/microblog-topic-extraction-data.zip`

BTM: Biterm Topic Model[6] [125] directly models topics of all word pairs (biterms) in each message, which outperformed LDA, Mixture of Unigrams model, and the model proposed by Zhao et al. [129], which aggregated messages by authorship to enrich context.

SATM: A general unified model proposed by Quan et al. [93] that aggregates documents and infers topics simultaneously. We reimplemented SATM and examined its effectiveness specifically on microblog data.

GMTM: To tackle word sparseness, Sridhar et al. [108] utilized Gaussian Mixture Model (GMM) to cluster word embeddings generated by a log-linear word2vec model.[7]

We also compared our FULL MODEL that combines everything in Section 3.3 with its variants:

LEADER ONLY model can be considered as a degeneration that assumes all messages are leaders. Topics assigned to all

[6] `https://github.com/xiaohuiyan/BTM`

[7] `https://code.google.com/archive/p/word2vec/`

messages can only be generated from the topic distributions of the conversation trees they are on. Analogous to Zhao et al. [129], where they aggregated messages posted by the same author, LEADER ONLY model aggregates messages from one conversation tree as a pseudo-document. Additionally, it includes a background word distribution to capture non-topical words controlled by a general Beta prior without differentiating leaders and followers.

FOLLOWER ONLY model is another variant that considers all messages as followers. Topics assigned to all messages can only be generated based on topic transitions from their parents. In particular, strTM [115] utilizes a similar model to capture the topic dependencies of adjacent sentences in a document. Following strTM, we add a dummy topic $\mathrm{T}_{\mathrm{start}}$ emitting no word to the "pseudo parents" of root messages. Also, we add the same background word distribution to capture non-topical words as LEADER ONLY model does.

Hyper-parameters and preprocessing. For our FULL MODEL, we fixed $\alpha = 50/K$, $\beta = 0.1$, following the common practice in previous work [36, 93]. Since there is no analogue of γ and δ in prior work, where γ controls topic dependencies of follower messages to their ancestors, and δ controls the different tendencies of leaders and followers to cover topical and non-topical words. We tuned γ and δ by grid search on a large development set containing around 120K posts and obtained $\gamma = 50/K$, $\delta = 0.5$.

For the variants LEADER ONLY and FOLLOWER ONLY model, the parameter settings were kept the same as our FULL MODEL, since they are its variants. Their background switchers were parameterized by symmetric Beta prior on 0.5, following Chemudugunta et al. [17].

For state-of-the-art topic models BTM, SATM and GMTM, their hyper-parameters were set according to the best hyper-parameters reported in their original papers.

We evaluated topic models with two sets of K, i.e., the number of topics. One is $K = 50$, to match the count of hashtags following Yan et al. [125], and the other is $K = 100$, which is much larger than the "real" number of topics.

We preprocessed the datasets before topic extraction in the following steps: 1) Used FudanNLP toolkit [92] for word segmentation, stop words removal, and POS tagging for Chinese Weibo messages; 2) Generated a vocabulary for each dataset and removed words occurring less than 5 times; 3) Removed all hashtags in texts before inputting them to models, since the models are expected to extract topics without knowing the hashtags, which served as ground-truth topics in our experiment.

We ran Gibbs samplings (in BTM, SATM, LEADER ONLY, FOLLOWER ONLY and FULL MODEL), and EM algorithm (in GMTM) with 1,000 iterations to ensure convergence.

3.6 Experimental Results

Topic model evaluation is inherently difficult. In previous work, perplexity is a popular metric to evaluate the predictive abilities of topic models given held-out dataset with unseen words [10]. However, Chang et al. [15] have demonstrated that models with high perplexity do not necessarily generate semantically coherent topics in human perception. Therefore, we conducted objective and subjective analysis on the coherence of produced topics.

3.6.1 Objective Analysis

The quality of topics is commonly measured by coherence scores [84], assuming that words representing a coherent topic are likely to co-occur within the same document. However, due to the severe sparseness of short text posts, we modify the calculation of commonly-used topic coherence measure. In the objective evaluation, we calculate topic coherence based on word co-occurrences

N	Models	May		June		July	
		K=50	K=100	K=50	K=100	K=50	K=100
10	**State-of-the-art**						
	BTM	**-26.7**	-28.9	-27.8	-25.5	-25.4	-25.2
	SATM	-30.6	-29.9	-23.8	-23.7	-24.3	-27.5
	GMTM	-40.8	-40.1	-44.0	-44.2	-41.7	-40.8
	Our models						
	Leader only	-27.9	-30.5	-24.0	-23.8	-23.9	-26.1
	Follower only	-29.9	-30.8	-24.0	-24.1	-24.4	-26.4
	Full model	-28.4	**-26.9**	**-19.8**	**-23.4**	**-22.6**	**-25.1**
15	**State-of-the-art**						
	BTM	-69.6	-71.4	-58.5	-60.3	-59.1	-63.0
	SATM	-74.3	-73.0	-54.8	-60.4	-61.2	-65.3
	GMTM	-96.4	-93.1	-100.4	-105.1	-94.6	-94.9
	Our models						
	Leader only	-71.9	-76.4	-55.3	-60.4	-61.2	-66.2
	Follower only	-76.4	-74.1	-57.6	-62.2	-58.1	-61.1
	Full model	**-67.4**	**-65.2**	**-52.8**	**-57.7**	**-55.3**	**-57.8**
20	**State-of-the-art**						
	BTM	-125.2	-131.1	-109.4	-115.7	-115.3	-120.2
	SATM	-134.6	-131.9	-105.5	-114.3	-113.5	-118.9
	GMTM	-173.5	-169.0	-184.7	-190.9	-167.4	-171.2
	Our models						
	Leader only	-138.8	-138.6	-102.0	-115.0	-115.8	-119.7
	Follower only	-134.0	-136.9	-104.3	-112.7	-111.0	-117.3
	Full model	**-120.9**	**-127.2**	**-101.6**	**-106.0**	**-97.2**	**-104.9**

Table 3.6: Coherence scores for different topic models. Higher is better. K:# of topics; N: # of top words ranked by topic-word probabilities

in messages tagged with the same hashtag, named as hashtag-document, assuming that those messages discuss related topics.[8]

Specifically, we calculate the coherence score of a topic given

[8] We sampled posts and their corresponding hashtags in our evaluation dataset and found only 1% mismatch.

the top N words ranked by likelihood as below:

$$\mathcal{C} = \frac{1}{K} \cdot \sum_{k=1}^{K} \sum_{i=2}^{N} \sum_{j=1}^{i-1} log \frac{D(w_i^k, w_j^k) + 1}{D(w_j^k)}, \tag{3.3}$$

where w_i^k represents the i-th word in topic k ranked by $p(w|k)$, $D(w_i^k, w_j^k)$ refers to the count of hashtag-documents where word w_i^k and w_j^k co-occur, and $D(w_i^k)$ denotes the number of hashtag-documents that contain word w_i^k.

Table 3.6 shows the values of $\mathcal{C}$ scores for topics produced on the three evaluation datasets (May, June and July), and the top 10, 15, 20 words of topics were selected for evaluation. A higher scores indicates better coherence in the induced topic.

We have the following observations:

- GMTM gave the worst coherence scores. This may be ascribed to its heavy reliance on relevant large-scale high-quality external data, without which the trained word embedding model failed to capture meaningful semantic features for words. Therefore, it could not yield coherent topics.

- Leader only and Follower only models produced competitive results compared to the state-of-the-art models. This indicates the effectiveness of using conversation structures to enrich context, which helps generate topics of reasonably good quality.

- The coherence of topics generated by our Full model outperformed all the baselines on the three datasets, most of time by large margins and was only outperformed by BTM on the May dataset when $K = 50$ and $N = 10$. The generally higher performance of Full model is due to three reasons: 1) It effectively identifies topics using the conversation tree structures, which provide rich context information; 2) It jointly models the topics of leaders and the topic dependencies of follower messages on a conversation tree. Leader only and Follower only models, each only considering one of these factors, performed worse than our Full model; 3) Our Full

MODEL separately models the probabilities of leaders and followers containing topical and non-topical words, while the competitors only model the general background information regardless of the different message types. This implies that leaders and followers do have different capacities in covering key topical words or background noise, which is useful to identify salient words for topic representation.

3.6.2 Subjective Analysis

To evaluate the coherence of induced topics from human perspective, we invited two annotators to subjectively rate the quality of every topic (by displaying the top 20 words) generated by different models on a 1-5 Likert scale. A higher rating indicates better quality of topics. The Fless's Kappa of annotators' ratings measured for various topic models on different datasets given $K = 50$ and 100 range from 0.62 to 0.70, which indicates substantial agreements [57].

Table 3.7 shows the overall subjective ratings. We noticed that humans preferred topics produced given $K = 100$ to $K = 50$, though coherence scores gave generally better grades to models for $K = 50$, which matched the number of topics in ground truth. This is because models more or less mixed more common words when K is larger. Coherence score calculation (Eq. (3.3)) penalizes common words that occur in many documents, whereas humans could somehow "guess" the meaning of topics based on the rest of words thus gave relatively good ratings. Nevertheless, annotators gave remarkably higher ratings to our FULL MODEL than baselines on all datasets regardless of K being 50 or 100, which confirmed that our FULL MODEL effectively yielded high-quality topics.

To present a more detailed analysis, Table 3.8 lists the top 20 words about "MH17 crash" induced by different models when $K = 50$.[9] We have the following observations:

[9] The topic generated by GMTM is not shown because we cannot find a relatively coherent topic describing "MH17". As shown in Table 3.6 and 3.7, the topic coherence scores of GMTM were the worst.

Model	May		June		July	
	K=50	K=100	K=50	K=100	K=50	K=100
State-of-the-art						
BTM	3.04	3.26	3.40	3.37	3.15	3.57
SATM	3.08	3.43	3.30	3.55	3.09	3.54
GMTM	2.02	2.37	1.99	2.27	1.97	1.90
Our models						
Leader only	3.12	3.41	3.42	3.44	3.03	3.48
Follower only	3.05	3.45	3.38	3.48	3.08	3.53
Full model	**3.40**	**3.57**	**3.52**	**3.63**	**3.55**	**3.72**

Table 3.7: Subjective ratings of topics. K: # of topics.

BTM	SATM	Leader only	Follower only	Full model
香港 入境处 家属 证实 男子 护照 外国 消息 坠毁 马航 报道 联系 电台 客机 飞机 同胞 确认 事件 霍家 直接	马航 祈祷 安息 生命 逝者 世界 艾滋病 恐怖 广州 飞机 无辜 默哀 远离 事件 击落 公交车 中国人 国际 愿逝者 真的	香港 微博 马航 家属 证实 入境处 客机 消息 曹格 投给 二胎 选项 教父 滋养 飞机 外国 心情 坠毁 男子 同胞	乌克兰 航空 亲爱 国民 绕开 飞行 航班 领空 所有 避开 宣布 空域 东部 俄罗斯 终于 忘记 公司 绝望 看看 珍贵	乌克兰 马航 客机 击落 飞机 坠毁 导弹 俄罗斯 消息 乘客 中国 马来西亚 香港 遇难 事件 武装 航班 恐怖 目前 证实
Hong Kong, immigration, family, confirm, man, passport, foreign, news, crash, Malaysia Airlines, report, contact, broadcast station, airliner, airplane, fellowman, confirm, event, Fok's family, directly	Malaysia Airlines, prey, rest in peace, life, dead, world, AIDS, terror, Guangzhou, airplane, innocent, silent tribute, keep away from, event, shoot down, bus, Chinese, international, wish the dead, really	Hong Kong, microblog, family, confirm, immigration, airliner, news, Grey Chow, vote, second baby, choice, god father, nourish, airplane, foreign, feeling, crash, man, fellowman	Ukraine, airline, dear, national, bypass, fly, flight, airspace, all, avoid, announce, airspace, eastern, Russia, finally, forget, company, disappointed, look, valuable	Ukraine, Malaysia Airlines, airliner, shoot down, airplane, crash, missile, Russia, news, passenger, China, Malaysia, Hong Kong, killed, event, militant, flight, terror, current, confirm

Remarks:
– The 2nd row: original Chinese words.
– The 3rd row: English translations.

Table 3.8: The extracted topics describing MH17 crash.

- BTM, based on word-pair co-occurrences, mistakenly grouped “Fok’s family” (a tycoon family in Hong Kong), which co-occurred frequently with “Hong Kong” in other topics, into the topic of “MH17 crash”. “Hong Kong” is relevant here because a Hong Kong passenger died in the MH17 crash.

- The topical words generated by SATM were mixed with words relevant to the bus explosion in Guangzhou, since it aggregated messages according to topic affinities based on the topics learned in the previous step. For this reason, SATM aggregated together mistakenly the messages about bus explosion and MH17 crash, both pertaining to disasters, and thus generated spurious topic results.

- Both Leader only and Follower only models generated topics containing non-topical words like “microblog” and “dear”. This means that without distinguishing leaders and followers, it is difficult to filter out non-topical words. The topic quality of Follower only model nevertheless seems better than Leader only model, which implies the usefulness of exploiting topic dependencies of messages in conversation structures.

- Our Full model not only produced more semantically coherent words to represent the topic, but also revealed some important details, e.g., MH17 was shot down by a missile.

3.7 Conclusion

This chapter has proposed a novel topic model by considering the conversation tree structures of microblog messages. By rigorously comparing our proposed model with a number of competitive baselines on large-scale real-world microblog datasets, we have demonstrated the effectiveness of using conversation structures to help extract topics embedded in short and colloquial microblog messages. Based on the topic clustering results produced in this chapter, we aim to summarize each topic cluster based on conversation structures. In Chapter 4, we will leverage the “leader-follower” discourse structures presented in this chapter to summarization framework.

□ **End of chapter.**

Chapter 4

Microblog Summarization and Coarse-grained "Leader-follower" Structures

In Chapter 3, we showed that "leader-follower" discourse structures embedded in conversation trees are useful to topic modeling. In this chapter, we incorporate this type of discourse structures to microblog summarization and explore how it helps summarization for a single conversation tree.

A microblog conversation tree provides strong clues on how an event develops. To help social media users capture the main clues of events on microblog websites, we propose a novel conversation tree summarization framework by effectively differentiating two kinds of messages on conversation trees called *leaders* and *followers*, which are derived from content-level discourse structure indicated by content of messages together with conversation relations formed by reposting and replying behaviors. To this end, following Chapter 3, we use Conditional Random Fields (CRF) model to detect leaders across conversation tree paths. We then present a variant of random-walk-based summarization model to rank and select salient messages based on the result of leader detection. To reduce the error propagation cascaded from leader detection, we further improve the framework by enhancing the random walk with sampling-based adjustment steps. The sampling steps are based on leader probabilities given by CRF-based leader detection module. The results of thorough experiments demonstrate the effectiveness of our proposed model.

4.1 Introduction

Microblog platforms have become the center for reporting, discussing, and disseminating real-life issues. They allow users to *reply* to messages to voice their opinions. Also users can *repost with commentary* for not only share messages with their following users, but also extending content of the original microblog post. Because a single post is generally too short to cover the main clues of an event, microblog users cannot easily capture the key information from received posts due to the lack of contextual information. On the other hand, reposting and replying messages, namely conversation messages, can provide valuable contextual information to the previous posts, such as their background, development, public opinions, etc. However, a popular post usually attracts a large volume of conversation messages. It is impractical for users to read all of them and to fully understand their content.

Microblog conversation summarization aims to produce succinct summaries to help users better understand the main clues discussed in a conversation. It automatically extracts salient information from massive conversation messages of the original posts.

An intuitive approach is to directly apply existing extractive summarizers based on the unstructured and plain microblog content. However, the short and informal nature of microblog posts renders the lack of structures in each individual message, As a result, it is difficult for conventional extractive summarizers to identify salient messages. Chang et al. [16] proposed to summarize Twitter conversation trees by leveraging modeling user influence. However, the messages posted by influential users might not be salient summary candidates necessarily. For instance, celebrities might simply reply with nothing important. Also, modeling user influence accurately requires tremendous historical user interaction data external to the tree being summarized. Also, such kind of information cannot be used directly for summarizing microblog conversations.

In this chapter, we propose a novel microblog conversation summarization framework based on discourse structure derived from message content and conversation relations (reposting and replying relations), rather than user-specific influence signals. The conversation relations connect the conversation messages and form a cohesive body as a tree structure called *conversation tree*. The root represents the original post and the edges denote conversation relations. Conversation structures have been shown helpful to microblog topic extraction (see Chapter 3). Here we explore their usefulness to summarization.

We use the similar "leader-follower" conversation structures proposed in Chapter 3, which distinguishes two different messages on conversation tree, i.e., *leaders* and *followers*. As mentioned in Chapter 3, a *leader* is referred to as a message on conversation tree covering salient new information, which can lead further comments or discussions in its descendant conversation messages; And a *follower* is referred to as a message that contains no comment, simply repeats, or naively responds to its ancestor leader message, thus providing no important information. Also, we showed that leaders are more likely to cover topical words, which describe key focus of the conversation, and followers tend to contain non-topical words, which are background noise.

From the perspective of summarization, *leaders* would be more important than *followers* since *leaders* are supposed to capture the main clues or aspects describing how event evolves. The first step of our summarization system is to effectively distinguish *leaders* and *followers*. We follow the leader detection step in Section 3.2 to detect leaders across conversation tree paths, which provides rich contextual information owing to the tree structure. We use sequence tagging model Conditional Random Fields (CRF) to infer how likely it is each conversation message being a leader or follower. Then we incorporate leader detection result into an unsupervised summarization model based on random walk. Our model uses content similarities between messages and considers their possibilities of being leaders to rank conversation messages for summary extraction. Furthermore, we improve the framework by enhancing the random walk to reduce the error propagation from the leader detection module. By comparing with the competitive summarization models on large-scale microblog corpus, the experimental results confirm the effectiveness of our proposed framework. The corpus has been released for future research of microblog summarization.[1]

4.2 "Leader-follower" Structures and Microblog Summarization

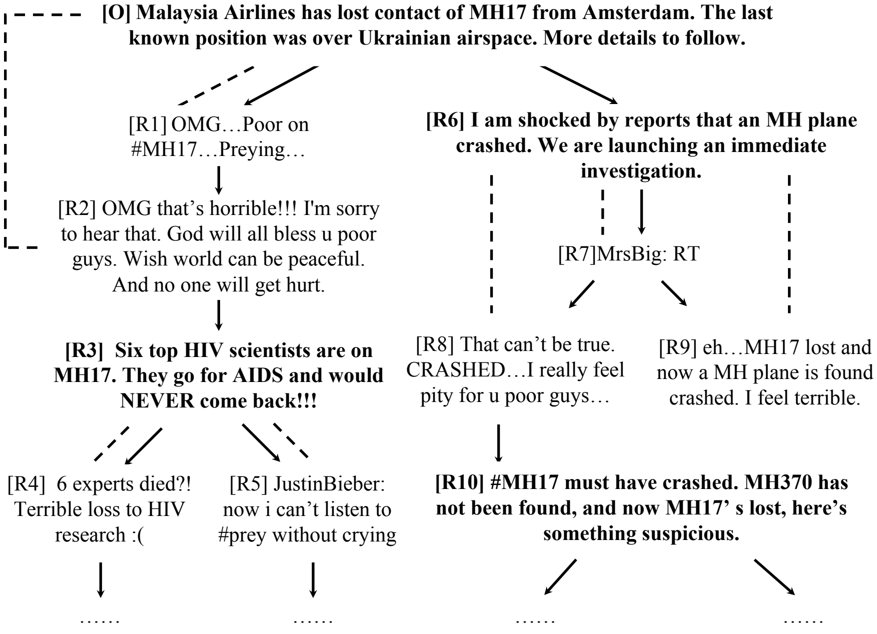
Messages on a conversation tree have different quality and importance for event description, which should be differentiated

[1] http://www1.se.cuhk.edu.hk/lijing/data/repost_tree_summ.zip

properly for summarization. This section explains “leader-follower” conversation structures, i.e., the differentiation of leader and follower messages in context of conversation structures, from the aspect of summarization.

Figure 4.1 illustrates an example of a conversation tree. As shown in this figure, a *leader* message contains content that brings essential information increase, such as a new clue about MH17 reported by [R6], and potentially triggers a new round of information propagation via attracting follower messages to focus on the raised clue, e.g., [R7], [R8], and [R9]. As the conversation tree grows, it also happens that some new conversation messages join in, following the clue raised by one of their ancestors, but further extend it by mentioning something new. For this reason, some of these messages may evolve into new leaders, such as [R10].

Intuitively, identifying leaders effectively makes one step closer to obtaining a summary. Because leaders generally in-


[O] Malaysia Airlines has lost contact of MH17 from Amsterdam. The last known position was over Ukrainian airspace. More details to follow.
[R1] OMG…Poor on #MH17…Preying…
[R2] OMG that's horrible!!! I'm sorry to hear that. God will all bless u poor guys. Wish world can be peaceful. And no one will get hurt.
[R3] Six top HIV scientists are on MH17. They go for AIDS and would NEVER come back!!!
[R4] 6 experts died?! Terrible loss to HIV research :(
[R5] JustinBieber: now i can't listen to #prey without crying
[R6] I am shocked by reports that an MH plane crashed. We are launching an immediate investigation.
[R7]MrsBig: RT
[R8] That can't be true. CRASHED…I really feel pity for u poor guys…
[R9] eh…MH17 lost and now a MH plane is found crashed. I feel terrible.
[R10] #MH17 must have crashed. MH370 has not been found, and now MH17' s lost, here's something suspicious.
……
……
……
……

[O]: the original post; [Ri]: the i-th repost or reply; Solid arrow lines: reposting or replying relationship; Dotted lines: hidden leader-follower relationship; Dark black posts: leaders to be detected.

Figure 4.1: An example of microblog conversation tree.

troduce new information and potentially lead discussion in descendents. Therefore leaders have higher probability of contain summary-worthy information that describe key focus of the conversation than followers, which simply respond to what leaders talk about.

Following Section 3.2, we extract all root-to-leaf paths on conversation trees and use the state-of-the-art sequence learning model CRF [56] to detect the leaders. The CRF model for leader detection was trained on our corpus with all the messages annotated on the tree paths. And the posterior probability of each node being a leader or follower is obtained by averaging the different marginal probabilities of the same node over all the tree paths that passes through the node. We determine a message as a leader if its average marginal probabilities being a leader in context of different paths exceeds 40%, which is the best empirical cutoff obtained for leader detection. Details were described in Section 3.4.1.

4.3 LeadSum Summarization Model

Let $T = (V, E)$ represent a conversation tree to be summarized, where V is a set of nodes representing microblog messages, and $E = \{(u, v) | v \text{ reposts or replies } u\}$ is the edge set denoting reposting and replying relations. This section describes how to rank nodes in V and produce summaries for conversation trees.

Enlightened by the general ranking algorithm DivRank [81], we propose an unsupervised summarization model called LeadSum that selects true and salient leaders into summaries based on a variant of random walk that jointly considers content similarities and conversation relations of messages. We first present a basic LeadSum model, which assumes leader detection is perfect. We then enhance it to a soft LeadSum model that reduces the impact of leader detection errors on the summarization.

4.3.1 Basic-LeadSum Model

Due to the nature of leaders, they generally cover more important content than followers do. Thus our first summarizer selects content only from detected leaders. For the leaders detected in a conversation tree T, we build a similarity graph among leaders denoted as $G_L = (V_L, E_L)$, where $V_L = \{v \in V | v \text{ is a detected leader}\}$ is the vertex set and $E_L = \{(u, v) | u \in V_L, v \in V_L, \text{ and } u \neq v\}$ is the edge set. The weight for any edge (u, v) represents the content similarity between u and v, for which we use cosine similarity.

DivRank [81] is a generic graph ranking model that aims to balance high information coverage and low redundancy in top ranking vertices. These are also two key requirements for choosing salient summary-worthy content [65, 72]. Based on DivRank, we present a model to rank and select salient messages only from leader set V_L to form a summary. Since this model simply assumes perfect leader detection, it is therefore named Basic-LeadSum.

Similar to DivRank [81], the transition probability at the t-th iteration of random walk is given in Eq. (4.1).

$$p_t(u \to v) = (1 - \mu) \cdot p_0(v) + \mu \cdot \frac{p_0(u \to v) N_{t-1}(v)}{Z(u)} \tag{4.1}$$

and $Z(u)$ is the normalizing factor:

$$Z(u) = \sum_{w \in V_L} p_0(u \to w) N_{t-1}(w) \tag{4.2}$$

where $p_0(u \to v)$ is the organic transition probability that represents the content similarity between u and v; $N_{t-1}(v)$ denotes the times vertex v is visited up to the $(t-1)$-th iteration; $p_0(v) = \frac{1}{|V_L|}$ refers to random jumping probability similar to that in PageRank; and μ is the damping weight set as 0.85 following the settings in most PageRank-based models. The probability of traveling to leader v can accumulate as its weight increases during random walk, and leaders already having high weight would "absorb" weights from other leaders highly similar to it, thus avoids redundancy.

For any $v \in V_L$, the update function in ranking process at the t-th iteration $R_t(v)$ is formulated in Eq. (4.3).

$$R_t(v) = \sum_{u \in V_L} p_t(u \to v) R_{t-1}(u) \tag{4.3}$$

It has been proved that this Markov chain is ergodic. Thus it is able to converge to a stationary distribution [81], which determines the final rankings for leaders.

4.3.2 Soft-LeadSum Model

As a two-step summarization system, the performance of LeadSum relies on the outputs of leader detection. This might be error-prone due to the following reasons: 1) Followers misidentified as leaders participating in leader ranking brings risks to extract real followers into summary; 2) Leaders misclassified as followers may leave out strong summary candidates.

To reduce such error propagation problem, we enhance Basic-LeadSum by proposing an even-length random walk with adjustment steps that sample from leader probabilities given by CRF-based leader detection module. This enhanced model is named as SOFT-LEADSUM.

Different from BASIC-LEADSUM, every message on conversation tree T, no matter detected as a leader or a follower, participates in the ranking process of SOFT-LEADSUM. In other words, in the random walk, the visitor wanders on a complete graph $G = (V, E')$. Its vertex set V is identical to conversation tree T. This makes it possible to select true leaders misclassified as followers by leader detection module into summary. And $E' = \{(u, v) | u \in V, v \in V, \text{ and } u \neq v\}$ represents the edge set defined analogous to BASIC-LEADSUM.

However, allowing all messages to participate in ranking also increases the risk of selecting real followers. To avoid this problem, SOFT-LEADSUM runs two types of random walks on G, namely WALK-1 and WALK-2. In WALK-1, the visitor moves based on content similarities between messages, which follows transition probabilities similar to equation (4.1), and is specifically given by Eq. (4.4).

$$p_t(u \to v) = (1-\mu) \cdot \frac{1}{|V|} + \mu \cdot \frac{p_0(u \to v) N_{t-1}(v)}{Z(u)} \qquad (4.4)$$

where $u, v \in V$, $p_0(u \to v)$ is proportional to content similarity between u and v similar to Basic-LeadSum, and $Z(u)$ is the normalizing factor.

WALK-2 attempts to avoid selecting true followers via a sampling process, whose result determines that the visitor stays or moves to another vertex on G. Suppose the current vertex being visited is u, we then sample from $p_L(u)$, i.e., the probability of u being a leader. Practically, $p_L(u)$ is estimated with the average of u's marginal probabilities as a leader over all root-to-leaf paths passing through u on T, which is outputted by the leader detection module.

If u is sampled to be a leader, we claim that leader detection is correct and the visitor stays; otherwise, u is sampled as a follower, indicating that leader detection module misclassified u, so the visitor should go to the leader of u. Here we assume that a follower u's leader is its nearest ancestor leader on T as shown by the dotted lines in Figure 4.1. Based on such simplification, we let the visitor trace back one by one along the path of T from u to root, and sample from their leader probabilities until a node v is sampled as a leader. Then, we say that v as u's leader.

Formally, for any u's ancestor v, the probability of v being u's leader is described in Eq. (4.5).

$$\begin{aligned} & Pr\{v \text{ is } u\text{'s leader}\} \\ =& p_L(v)(1 - p_L(u) - \sum_{w \in \mathcal{P}(v,u)} Pr\{w \text{ is } u\text{'s leader}\}) \\ =& p_L(v) \prod_{w \in \mathcal{P}(v,u) \bigcup \{u\}} (1 - p_L(w)) \end{aligned} \tag{4.5}$$

where $\mathcal{P}(v, u)$ is the set of nodes between v and u on the v-to-u

path of conversation tree, i.e., $\mathcal{P}(v,u) = \{w \in V | w$ is v's descendant and u's ancestor on $T\}$. In particular, we assume that $p_L(r) = 1$ so as to stop the sampling when the visitor arrives at root r.

Therefore, transition probabilities of WALK-2 can be calculated by Eq. (4.6).

$$q(u \rightarrow v) = \begin{cases} p_L(v) & \text{if } v = u; \\ Pr\{v \text{ is } u\text{'s leader}\} & \text{if } v \text{ is } u\text{'s ancestor;} \\ 0 & \text{otherwise} \end{cases} \tag{4.6}$$

Algorithm 1 shows the ranking process of Soft-LeadSum, during which the visitor walks on G alternately following WALK-1 and WALK-2. The fact that WALK-1 is ergodic ensures the convergence of the algorithm. In implementation, we empirically set max iteration $N = 1000$, which is large enough to ensure convergence. The algorithm can also stop in advance when the converging condition is met, i.e., the change of Euclidean difference of ranking scores for three consecutive iterations are all less than 1e-6.

Algorithm 1 Algorithm of Soft-LeadSum

```
Input: $T$, $G$, $\mu$=0.85, max iteration $N$, length cut-off $n$
Output: Summary with $n$ microblog messages
For all $v \in V$, initialize $R_0(v) = p_0(v) = \frac{1}{|V|}$
Initialize WALK-1's transition probabilities $p_0(u \rightarrow v)$ with normalized cosine similarity between $u$ and $v$.
Calculate WALK-2's transition probabilities $q(u \rightarrow v)$ by equation (4.5) and (4.6).
Initialize current_walk="WALK-1"
for $t = 1$ to $N$ and not converged do
    for all $v \in V$ do
        if current_walk=="WALK-1" then
            Update $p_t(u \rightarrow v)$ by equation (4.4)
            Update $R_t(v) = \sum_{u \in V} R_{t-1}(u) \cdot p_t(u \rightarrow v)$
            Set current_walk="WALK-2"
        end if
        if current_walk=="WALK-2" then
            Update $R_t(v) = \sum_{u \in V} R_{t-1}(u) \cdot q(u \rightarrow v)$
            Set current_walk="WALK-1"
        end if
    end for
end for
Sort all $v \in V$ by $R_N(v)$ in descending order
Pick the top-$n$ messages as summary
```

Soft-LeadSum can reduce the impact of errors made by leader detection on summarization due to the following two reasons: 1) It allows all messages to participate in ranking process, thus permits those true leaders leaving out by leader detection module to be selected into summary; 2) With WALK-2 sampling from leader probabilities, it also reduces the risk of including real followers into summary.

4.4 Data Collection and Evaluation Metrics

Collection of microblog conversation data. There is no public editorial conversation tree dataset for summarization. Therefore, we build a corpus for summarization evaluation by following our previous work Chang et al. [16]. We selected 10 hot original posts, crawled their conversation trees, and invited human editors to write summaries for each conversation tree as gold-standard reference.[2]

Though comparing with many other corpora in NLP and IR community, this corpus is relatively small. However, it is typ-

[2] Chang et al. [16] doesn't release their dataset.

ically difficult and time-consuming for human editors to write summaries for conversation trees because of their massive nodes and complex structures [16]. The editors could hardly reconstruct the conversation trees even though they went through all the message nodes.

The 10 original posts we used in experiments were posted on Sina Weibo during January 2nd – July 28th 2014 and cover topics that match the official list of general post category released by Sina Weibo.[3]. We then used the PKUVIS toolkit [97] to crawl the complete conversation trees given the corresponding original posts. Table 4.1 shows the statistic information about the conversation tree corpus.[4] Note that this conversation tree corpus has no overlap with the conversation tree path dataset used for training leader detection models (see Section 3.4.1).

Reference summaries and evaluation metrics. We invited three experienced editors whose native language are all

[3] `d.weibo.com/`

[4] All descriptions are English translations of the root microblogs originally in Chinese.

Name	# of nodes	Height	Description of root post
Tree (I)	21,353	16	A girl quit HKU, re-applied universities for her dream, and received admission from PKU.
Tree (II)	9,616	11	A cute German boy complained hard schoolwork in Chinese High School.
Tree (III)	13,087	8	Movie “Tiny Times 1.0” won high grossing in criticism.
Tree (IV)	12,865	8	TV show “I am A Singer” stated the resinging of singer G.E.M conformed to rules.
Tree (V)	10,666	8	Crystal Huang clarified her love affair.
Tree (VI)	21,127	11	Brazil 1:7 Germany in World-Cup semi-final.
Tree (VII)	18,974	13	A pretty girl pregnant with a second baby graduated with her master degree.
Tree (VIII)	2,021	18	Girls appealed for sexual equality in college admission.
Tree (IX)	9,230	14	Terror attack in Kunming railway station.
Tree (X)	10,052	25	Top HIV researchers were killled in MH17 crash.

Table 4.1: Description of conversation tree summarization corpus

Chinese to write summaries for each conversation tree. To ensure the quality of reference summaries, we first extracted a list of frequent nouns from each conversation tree and generalized 7 to 10 aspects based on the nouns list. This provided a high-level overview of a conversation tree to the editors. Our guideline asked the editors to read all messages ordered sequentially on a conversation tree. For every message, its entire conversation tree path was also provided as supplementary contextual information. When finished reading, editors wrote down one or two sentences to summarize each aspect in the list.

We evaluated the performance of our summarization method by both objective and subjective analysis. In objective evaluation, we used ROUGE metric [69] as benchmark, which is a widely-applied standard for evaluating automatically produced summaries based on N-gram overlapping between a system-generated summary and a human-written reference. Specifically, precision, recall, and F1 score of ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-SU4 served as our evaluation metrics. In subjective analysis, we invited two native Chinese speakers, different from the three editors who wrote reference summaries, to read and subjectively rates the produced summaries according to their informativeness, conciseness and readability.

In our human-generated summaries, the average inter-annotator-agreement by ROUGE-1 is 0.431, which means each pair of manual summaries have no more than 50% words overlap on average even written under aspect constraints. This confirms that microblog conversation tree summarization is generally a difficult task. In particular, in the evaluation for each tree summary, we computed the average ROUGE scores between the system-generated summary and the three human-written summaries.

This annotated microblog corpus is publicized for future research on microblog summarization.[5]

Baselines and comparisons. We considered baselines that rank and select messages by 1) LENGTH; 2) POPULARITY (# of reposts and replies); 3) USER influence (# of authors' followers); 4) text similarities to other messages using LEXRANK [28].

We also considered CHANG ET AL. [16], a state-of-the-art and fully *supervised* summarizer. It is based on Gradient Boosted Decision Tree (GBDT) algorithm with manually-crafted features that capture text, popularity, temporal and user signals. In particular, without the interaction data with external users, we use users' follower count to approximate the user influence.

[5] http://www1.se.cuhk.edu.hk/~lijing/data/repost_tree_summ.zip

GBDT implementation is based on RankLib,[6] and as a supervised method, Chang et al. [16] is evaluated based on 10-fold cross validation.

In addition, we compared the our models with its two variants: 1) LeadProbSum, a simple variant of LeadSum that ranks messages simply by their marginal probabilities as leaders in decreasing order; 2) DivRank, a direct application of Mei et al. [81] to rank all messages unaware of leaders and followers. A similar model is also reported in Yan et al. [124]. Following their work, we set the damping weight as 0.85.

Data preprocessing and hyper-parameters. Before summarization, we preprocessed the evaluation corpora in the following three steps: 1) Use FudanNLP toolkit [92] for word segmentation of Chinese microblog messages; 2) Filter out non-Chinese characters; 3) For Basic and Soft LeadSum, we used the CRF-based leader detection model 3.2 to classify messages

[6] `http://sourceforge.net/p/lemur/wiki/RankLib/`

as leaders and followers. The leader detection module was implemented by using CRF++,[7] and was trained on the dataset described in Section 3.4.1. The training set was composed of 1,300 conversation paths and achieved state-of-the-art 73.7% and 70.5% F1-score of classification accuracy in 5-fold cross-validation and held-out evaluation, respectively.

In particular, in experiment, we applied cosine similarities to represent content similarities encoded in edges weights of DivRank, Basic-LeadSum, and Soft-LeadSum.

Post-processing. In testing phase, we dropped out messages that have >= 0.8 cosine similarity with any higher-ranked message to reduce redundancy. And the top-10 ranked messages were picked up to form a summary for each conversation tree.

[7] taku910.github.io/crfpp/

4.5 Experiment Results

In this experiment, we evaluated end-to-end performance of our basic and soft LeadSum summarization models by comparing them with competitive microblog summarizers.

4.5.1 ROUGE Comparison

Table 4.2 shows the result of overall comparisons. Note that the results here are different from those reported in its earlier publication Li et al. [61]. Because the ROUGE scores here were given by ROUGE 1.5.5.[8], while Li et al. [61] uses Dragon toolkit [132] for ROUGE calculation.[9] Though the scores were different, the trends reported here remain the same with Li et al. [61]. We changed the experiment setting to be consistent with the experiment setting in Chapter 5. We have the following observations.

Simple features are not enough. The poor performance of all

[8] github.com/summanlp/evaluation/tree/master/ROUGE-RELEASE-1.5.5

[9] dragon.ischool.drexel.edu/

Models	Len	ROUGE-1			ROUGE-2		
		Prec	Rec	F1	Prec	Rec	F1
Baselines							
Length	95.4	19.6‡	**53.2**	28.1‡	5.1‡	**14.3**	7.3‡
Popularity	27.2	33.8	25.3‡	27.9‡	8.6	6.1‡	6.8‡
User	37.6	32.2	34.2‡	32.5	8.0	8.9‡	8.2†
LexRank	25.7	**35.3**	22.2‡	25.8‡	11.7	6.9‡	8.3‡
State-of-the-art							
Chang et al. [16]	68.6	25.4†	48.3	32.8	7.0	13.4	9.1
Our models & variants							
DivRank	31.1	28.0	25.2‡	25.4‡	6.4†	5.7‡	5.7‡
LeadProSum	60.1	26.0	43.4	31.9†	6.1‡	10.1‡	7.4‡
Basic-LeadSum	85.5	21.2	40.6	29.7‡	6.0‡	10.2	7.4‡
Soft-LeadSum	58.6	27.3	45.4	**33.7**	7.6	12.6	**9.3**

Models	Len	ROUGE-L			ROUGE-SU4		
		Prec	Rec	F1	Prec	Rec	F1
Baselines							
Length	95.4	16.4‡	**44.4**	23.4‡	6.2‡	**17.2**	8.9‡
Popularity	27.2	28.6	21.3‡	23.6‡	10.4	7.6‡	8.4‡
User	37.6	28.0	29.6‡	28.2	9.8	10.6‡	10.0†
LexRank	25.7	**30.6**	18.8‡	22.1‡	**12.3**	7.5‡	8.8‡
State-of-the-art							
Chang et al. [16]	68.6	21.6	41.1	27.9	8.3	16.0	10.8
Our models & variants							
DivRank	31.1	24.1	21.5‡	21.7‡	8.3	7.5‡	7.6‡
LeadProSum	60.1	22.1	37.0	27.1†	27.1‡	13.2†	9.6‡
Basic-LeadSum	85.5	19.8‡	33.3	24.4‡	7.6‡	12.8	9.4‡
Soft-LeadSum	58.6	23.3	38.6	**28.7**	8.8	14.7	**10.9**

Remarks:

–Len: count of Chinese characters in the extracted summary.

–Prec, Rec and F1: average precision, recall and F1 ROUGE measure over 10 conversation trees (%).

–Notions † and ‡ means the improvement of our full model over the corresponding summarizer is significant at level 0.1 ($p < 0.1$) and level 0.05 ($p < 0.05$) based on one-tailed pairwise t-test.

Table 4.2: ROUGE comparison of summarization models

baselines demonstrates that microblog summarization is a challenging task. It is not possible to trivially rely on simple features like length, message popularity, user influence, or text similarities to identify summary-worthy messages because of the severe colloquiality, noise, and redundancy exhibited in microblog texts.

Leaders tend to contain summary-worthy content. By simply ranking messages based on their marginal probabilities as leaders, LEADPROSUM achieved close performance with state-of-the-art supervised model CHANG ET AL. [16], which can learn from manually crafted features and human-written summaries. Also, using CRF-based leader detection model to filter out detected followers, BASIC-LEADSUM remarkably outperformed DIVRANK, which ranks messages without differentiating leader and follower messages. These confirm that leaders do contain salient content and should be should be differentiated from followers for summarization.

Sampling steps in WALK-2 of Soft-LeadSum *was useful.* Soft-LeadSum significantly outperformed Basic-LeadSum measured by F1 scores of all types of ROUGE. This implies that sampling steps in the enhanced random walk of Soft-LeadSum can successfully reduce the impact of leader detection error on summarization.

Soft-LeadSum *framework was effective.* Soft-LeadSum outperformed all the unsupervised competitors with a large margin on all sets of ROUGE F1 scores. The one-tailed pairwise t-test on ROUGE F-1 indicates that all the improvements over the competitors were significant at >= 90% confidence level except for the User baseline on ROUGE-1 and ROUGE-L. Without relying on any gold-standard summaries for training, the performance of Soft-LeadSum was competitive, and even slightly better than Chang et al. [16], which relies on full supervision from human-generated summaries. This confirmed that our framework was capable of producing informative summaries for microblog conversation trees.

4.5.2 Human Evaluation

We also conducted human evaluations for informativeness (Info), conciseness (Conc) and readability (Read) of extracted summaries. Two native Chinese speakers were invited to read the output summaries and subjectively rated on a 1-5 Likert scale in 0.5 units. A higher rating indicates better quality. Their overall inter-rater agreement achieved Krippendorff's α of 0.71, which indicates reliable results [55]. Table 4.3 shows the average ratings by two raters and over ten conversation trees.

In human evaluation, our SOFT-LEADSUM model produced summaries achieving remarkably higher ratings than all competitors in informativeness, conciseness, and readability. This demonstrates that our proposed summarization method can produce high-quality summaries popular to humans.

Models	Info	Conc	Read
Baselines			
LENGTH	2.33	2.93	2.28
POPULARITY	2.38	2.35	3.05
USER	3.13	3.10	3.75
LEXRANK	3.05	2.70	3.03
State-of-the-art			
CHANG ET AL. [16]	3.43	3.50	3.70
Our models & variants			
DIVRANK	2.78	3.36	3.53
LEADPROSUM	3.25	3.28	3.53
BASIC-LEADSUM	3.23	3.25	3.38
SOFT-LEADSUM	**3.70**	**3.90**	**4.15**

Table 4.3: Overall human ratings on summaries

4.6 Conclusion

This chapter presents a study for microblog conversation tree summarization. Its output can provide important clues for event analysis on microblog platforms. Conventional work considering only plain text streams is insufficient for summarizing noisy conversation trees. We propose a novel summarization system based on "leader-follower" discourse structures proposed in Chapter 3, which effectively differentiates leader and follower messages on conversation trees. Firstly, a leader detection model categorizes each message on conversation tree path as a leader or a follower. Then, a random-walk variant summarization model called LeadSum ranks and selects salient microblog messages on the basis of leader detection results. To reduce errors cascaded from leader detection module, we enhance LeadSum to an even-length random walk by sampling from leader probabilities for improving summarization. Based on real-world microblog post dataset, the objective and subjective experimental results confirmed that our proposed framework can outperform non-trivial methods for conversation tree summarization.

In this chapter, we have proven that detecting leaders and followers, which are coarse-grained discourse derived from conversation structures, is useful to microblog summarization. In Chapter 5, we propose fine-grained discourse structures, and explore their usefulness in indicating summary-worthy content.

□ **End of chapter.**

Chapter 5

A Joint Microblog Summarization Model Based on Sentiment, Content, and Fine-grained Discourse

In Chapter 4, we have proven that coarse-grained discourse is useful to summarization. This chapter aims to study how to effectively exploit fine-grained discourse for recognizing summary-worthy content.

Although there is good progress in extracting discourse structures from some conversation tasks, e.g., meetings and emails, the domain-specific nature of discourse analysis renders directly applying discourse inventory designed for other conversation domains to microblog conversations ineffective. Because discourse analysis including fine-grained conversation discourse on microblog is a new research topic, we attempt to discover discourse components purely from data without relying on any pre-defined discourse inventory.

We present a weakly supervised probabilistic model for microblog conversation summarization by jointly exploring representations of discourse, sentiment, and content. With minimal supervision from emoji lexicon, our model exploits sentiment shifts to detect message-level discourse function in the context of conversations. It also discovers clusters of discourse words that are indicative of summary-worthy content. In automatic evaluation on a large-scale microblog corpus, our joint model significantly outperformed state-of-the-art methods on ROUGE F1. Human evaluation also shows that our system summaries are competitive in informativeness, conciseness, and readability.

Qualitative analysis on model outputs indicates that our model induced meaningful representations for discourse and sentiment.

5.1 Introduction

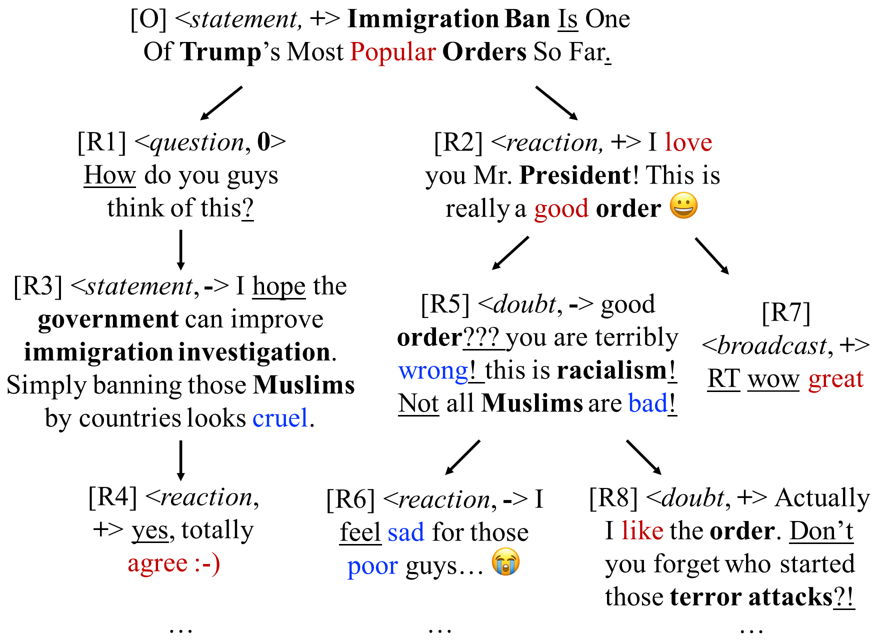
Microblogs have become a popular outlet for online users to share information and voice opinions on diverse topics. Users frequently form discussions on issues of interests by reposting messages and replying to others. Those conversations provide a valuable resource for instant detection of trendy topics [68, 89, 118] and understanding public discourse on controversial issues [90].

However, the large volume of messages and the often complex interaction structures make it impossible for human to read through all conversations, identify gist information, and make sense out of it.

Automatic summarization methods have been proposed to construct concise summaries for lengthy conversations on microblogs to capture the informative content [16, 61] (see Chapter 4 also). Previous work mainly focuses on understanding the content of the conversations through topic modeling approaches [18, 63, 96], and largely ignores the prevalent sentiment information and the rich discourse structure among user interactions.

Here we argue that *reliable estimation of the summary-worthy content in microblog conversations requires additional consideration of sentiment and discourse.* For instance, Figure 5.1 illustrates a snippet of Twitter conversation with replying structures on the topic of "Trump administration's immigration ban". [1] We can observe three major components from the conversation: 1) *discourse*, indicated by underlined words, that describes the intention and pragmatic roles of messages in conversation structures, such as making a "statement" or asking a "question"; 2) *sentiment*, reflected by positive (red) and negative (blue) words

[1] Table 1.2 actually denotes the [O] → [R2] → [R5] → [R6] path of this tree.


[O] <statement, +> Immigration Ban Is One Of Trump's Most Popular Orders So Far.
[R1] <question, 0> How do you guys think of this?
[R2] <reaction, +> I love you Mr. President! This is really a good order 😀
[R3] <statement, -> I hope the government can improve immigration investigation. Simply banning those Muslims by countries looks cruel.
[R5] <doubt, -> good order??? you are terribly wrong! this is racialism! Not all Muslims are bad!
[R7] <broadcast, +> RT wow great
[R4] <reaction, +> yes, totally agree :-)
[R6] <reaction, -> I feel sad for those poor guys… 😭
[R8] <doubt, +> Actually I like the order. Don't you forget who started those terror attacks?!
…
…
…

[O]: the original post; [Ri]: the i-th repost or reply; arrow lines: reposting or replying relations; *italic words* in <>: discourse role of the message; +, 0, or - in <>: the sentiment of message is positive, neutral, or negative, respectively; underlined words: words indicating discourse role; **bold words**: content words representing discussion focus; red and blue words: positive and negative sentiment words.

Figure 5.1: A sample Twitter conversation tree on "Trump administration's immigration ban".

including emoji (e.g., 😀),[2] that expresses users' attitudes; and 3) *content*, represented by bold words, captures the topics and focus of the conversation, such as "racialism" and "Muslims".

[2] Emoji symbols are added by users, encoded in unicodes, and rendered as pictures of facial expressions.

As can be seen, the content words are usually mixed with sentiment words and discourse function words. A summarization model will thus benefit from the separation of content words from sentiment-specific or discourse-specific information. However, previous efforts on discourse or sentiment analysis on microblogs mostly rely on pre-defined inventory of discourse relations (e.g., dialogue acts) [114, 126, 127], or sentiment polarity [2,7], to train supervised classifiers, which requires significant human efforts.

To address the above problems, we present a novel probabilistic model, which jointly infers the word representations for *discourse*, *sentiment*, and *content* in a weakly supervised manner. While prior work investigates the use of either discourse or sentiment in microblog summarization [61, 82, 127, 131], to the best of our knowledge, we are the first to explore their joint effect for summarizing microblog conversations. Importantly, our joint model of discourse, sentiment, and content in microblog conversations only requires minimal supervision from a small emoji lexicon to inform the sentiment component. Based on the inference results, representative messages that capture the critical content of discussions will be extracted for use in the conversation summary.

Empirical results with ROUGE [69] show that conversation summaries generated by our joint model contain more salient information than state-of-the-art summarization models based on supervised learning. Human evaluation also indicates that our system summaries are competitive in the aspects of informativeness, conciseness, and readability. Qualitative analysis in Section 5.6 further shows that our model is able to yield meaningful clusters of words that are related to manually crafted discourse and sentiment categories.

5.2 The Joint Model of Discourse, Sentiment, and Content

We assume that the given corpus of microblog posts is organized as T conversation trees based on reposting and replying relations.

Each tree t contains M_t microblog messages and each message m has $N_{t,m}$ words in vocabulary. The vocabulary size is V. We separate four components, i.e., *discourse*, *sentiment*, *content* and *background* underlying conversations, and utilize four types of word distributions to represent them.

At the corpus level, $\delta_d \sim Dir(\mu^{disc})$ $(d = 1, 2, ..., D)$ represents the D *discourse* roles embedded in corpus. $\sigma_p \sim Dir(\mu_p^{pol})$ $(p \in \{\mathrm{POS}, \mathrm{NEG}\})$ exploits sentiment polarities, i.e., the positive (POS) and negative (NEG) sentiment.[3] In addition, we add

[3] Sentiment words that indicate neutral sentiment are sparse in microblog messages. Modeling neutral sentiment words would bring the problem of data sparseness and affect the performance of unsupervised and weakly supervised models like ours. Therefore, in this chapter, we assume sentiment words can only indicate positive and negative polarity.

a *background* word distribution $\beta \sim Dir(\mu^{back})$ to capture general information (e.g., common words), which cannot indicate discourse, sentiment, or content.

For each conversation tree, $\gamma_t \sim Dir(\mu^{cont})$ describes tree-specific *content* that captures core focus or topic of the conversation, based on which summary messages are extracted (see Section 5.3).

δ_d, σ_p, β and γ_t are all multinomial word distributions over vocabulary size V.

5.2.1 Message-level Modeling

For each message m on tree t, our model assigns two types of message-level multinomial variables to it, i.e., $d_{t,m}$ representing its *discourse* role and $s_{t,m}$ reflecting its *sentiment* category.

(1) Discourse Assignments

Our discourse detection is inspired by Ritter et al. [98] that exploits the discourse dependencies derived from reposting and replying relations to help discourse assignments. For example, a "doubt" message is likely to start controversy thus triggers another "doubt", e.g., [R5] and [R8] in Figure 5.1.

Assuming that the index of m's parent is $pa(m)$, we use transition probabilities $\pi_d \sim Dir(\lambda)$ $(d = 1, 2, ..., D)$ to explicitly model discourse dependency of m to $pa(m)$. π_d is a distribution over D discourse roles and $\pi_{d,d'}$ denotes the probability of m assigned discourse d' given the discourse of $pa(m)$ being d. Specifically, $d_{t,m}$ (discourse role of each message m) is generated from discourse transition distribution $\pi_{d_{t,pa(m)}}$ where $d_{t,pa(m)}$ is the discourse assignment on m's parent.

To create a unified generation story, we place a pseudo message emitting no word before the root of each conversation tree and assign dummy discourse indexing $D + 1$ to it. π_{D+1}, defined as discourse transition from pseudo messages to tree roots, in fact models the probabilities of different discourse roles as conversation starter.

(2) Sentiment Assignments

We assume there are S distinct labels for message-level sentiment and each message m is assigned a sentiment label $s_{t,m} \in \{1, 2, ..., S\}$.

Since discourse can indicate sentiment shifts in conversations, which is useful to sentiment assignments. For example, "broadcast" message [R7] in Figure 5.1 keeps the sentiment of its parent because "broadcast" is supposed to be a purely sharing behavior without changing sentiment. We utilize $\psi_{d,s} \sim Dir(\xi)$ to capture the parent-child sentiment shifts given discourse of the child message where $\psi_{d,s,s'}$ means the probabilities of current message assigned as sentiment s' conditioned on its discourse as d and the sentiment of its parent as s. In particular, $s_{t,m}$ (the sentiment of each message m) is generated from $\psi_{d_{t,m},s_{t,pa(m)}}$ where $pa(m)$ being the index of m's parent.

Similar to discourse assignments described in (1), we put a pseudo message before each conversation root and assign a dummy sentiment $S+1$ to it. The sentiment of a root post is then only determined by its discourse assignment.

5.2.2 Word-level Modeling

In order to separately capture *discourse*, *content*, *sentiment* and *background* information, for each word n in message m and tree t, a quaternary variable $x_{t,m,n} \in \{\text{DISC}, \text{CONT}, \text{SENT}, \text{BACK}\}$ controls word n to fall into one of the four types: *discourse*, *content*, *sentiment* and *background* word.

(1) Discourse words (DISC) can indicate the discourse roles of messages, e.g., in Figure 5.1, "How" and question mark "?" reflect [R1] being discourse "question". Therefore, if $x_{t,m,n} = \text{DISC}$, i.e., n is assigned as a discourse word, word $w_{t,m,n}$ is generated from discourse word distribution $\delta_{d_{t,m}}$ where $d_{t,m}$ is discourse assignment to m.

(2) Content words (CONT) describe the core focus or topic of a conversation, such as "Muslim", "order" and "Trump" in Figure 5.1. When $x_{t,m,n} = \text{CONT}$, i.e., n is assigned as a content word, word $w_{t,m,n}$ is generated from content word distribution of tree t, i.e., γ_t.

(3) Sentiment words (SENT) reflect the overall sentiment of the corresponding message, e.g., "like" and "sad" in Figure 5.1. When n is assigned as a sentiment word ($x_{t,m,n} = \text{SENT}$), we further capture its sentiment polarity with a binary variable $p_{t,m,n} \in \{\text{POS}, \text{NEG}\}$ and generate word $w_{t,m,n}$ from sentiment polarity distribution $\sigma_{p_{t,m,n}}$. Since polarities of sentiment words can indicate message-level sentiment assignments, we bridge the message-level and word-level sentiments by multinomial distribution $\theta_s \sim Dir(\omega)$ $(s = 1, 2, ..., S)$ where $\theta_{s,p}$ refers to the probability of messages assigned sentiment label s containing positive ($p = \text{POS}$) or negative ($p = \text{NEG}$) words. The polarity of each sentiment word n, i.e., $p_{t,m,n}$, is then drawn from $\theta_{s_{t,m}}$ where $s_{t,m}$ being the sentiment of m.

Previous works have shown the usefulness of emoji (e.g. 😭 in Figure 5.1) in sentiment analysis [54, 128]. Therefore, instead of filtering out emoji, we consider them as special words and incorporate supervision from positive and negative emoji into Dirichlet prior of sentiment polarity distributions.

Inspired by how He et al. [40] combines prior from sentiment lexicon, the Dirichlet prior of positive word distribution σ_{POS}, is parameterized as

$$\mu^{pol}_{\text{POS},v} = \begin{cases} 0.95 & \text{if } v \text{ is a positive emoji} \\ 0.05 & \text{if } v \text{ is a negative emoji} \\ 0.5 & \text{if } v \text{ is not a emoji} \end{cases}$$

and negative sentiment prior is controlled by $\mu^{\vec{pol}}_{\text{NEG}} = \vec{1} - \mu^{\vec{pol}}_{\text{POS}}$.

(4) Background words (BACK) capture the general information that is not related to discourse, content, or sentiment information. When word n is assigned as a background word ($x_{t,m,n} = \text{BACK}$), word $w_{t,m,n}$ is drawn from background distribution β.

(5) Discourse and word generation. We assume that messages with different discourse roles vary in tendencies to contain discourse, content, sentiment, and background words. In addition, as mentioned in (3), emoji are more likely to be sentiment words than non-emoji words. So we add a binary variable $e_{t,m,n}$ to indicate whether $w_{t,m,n}$ is emoji ($e_{t,m,n} = 1$) or not ($e_{t,m,n} = 0$). The quaternary word type switcher $x_{t,m,n}$ is hence jointly controlled by the discourse of m ($d_{t,m}$) and the emoji switcher $e_{t,m,n}$, i.e., $x_{t,m,n} \sim Multi(\tau_{d_{t,m},e_{t,m,n}})$.

In addition, for all $d = 1, 2, ..., D$, our model gives higher prior for emoji to be sentiment words by defining Dirichlet prior ν of word type emitter τ as:

$$\nu_d = \begin{pmatrix} 0.25 & 0.25 & 0.25 & 0.25 \\ \\ 0.7 & 0.1 & 0.1 & 0.1 \end{pmatrix} \tag{5.1}$$

where the first and second row refers to parameters of Dirichlet prior for emoji and non-emoji words, respectively, and the first column means the prior parameters of emitting sentiment words (SENT).

5.2.3 Generation Process

In summary, Figure 5.2 illustrates our graphical model, and Table 5.1 shows the generation process of each message m on conversation tree t.

We use collapsed Gibbs Sampling [36] to carry out posterior inference for parameter learning. The hidden multinomial variables, i.e., message-level variables (d and s) and word-level variable (x and p) are sampled in turn, conditioned on a complete assignment of all other hidden variables, conditioned on

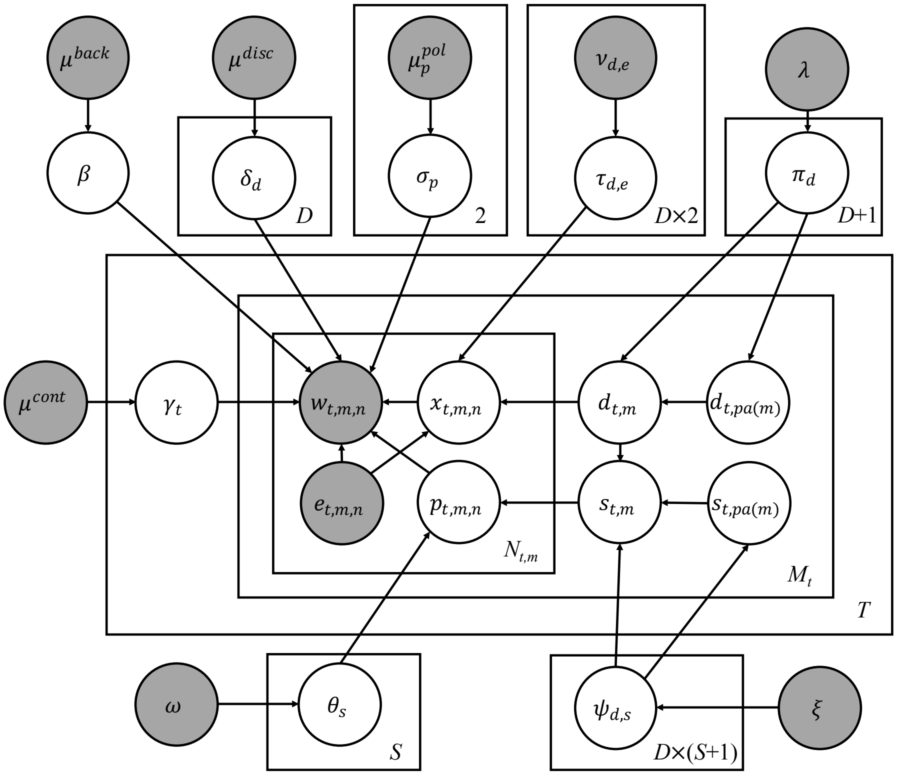


Figure 5.2: Our graphical model

- Draw discourse $d_{t,m} \sim Multi(\pi_{d_{t,pa(m)}})$
- Draw sentiment $s_{t,m} \sim Multi(\psi_{d_{t,m},s_{t,pa(m)}})$
- For word $n = 1$ to $N_{t,m}$
  - Draw switcher $x_{t,m,n} \sim Multi(\tau_{d_{t,m},e_{t,m,n}})$
  - If $x_{t,m,n} ==$ SENT
    - Draw polarity $p_{t,m,n} \sim Multi(\theta_{s_{t,m}})$
    - Draw word $w_{t,m,n} \sim Multi(\sigma_{p_{t,m,n}})$
  - If $x_{t,m,n} ==$ DISC
    - Draw word $w_{t,s,n} \sim Multi(\delta_{d_{t,m}})$
  - If $x_{t,m,n} ==$ CONT
    - Draw word $w_{t,m,n} \sim Multi(\gamma_t)$
  - If $x_{t,m,n} ==$ BACK
    - Draw word $w_{t,m,n} \sim Multi(\beta)$

Table 5.1: Generation process of a conversation tree t

a complete assignment of all other hidden variables and hyper-parameters $\Theta = (\mu^{disc}, \mu^{pol}, \mu^{cont}, \mu^{back}, \nu, \lambda, \omega, \xi)$.

5.2.4 Inference for Parameters

We first define the notations of all variables needed by the formulation of Gibbs sampling, which are described in Table 5.2.4. In particular, the various C variables refer to counts excluding the message m on conversation tree t. —c—m11cm—

e word-level emoji switcher. $e = 0$: non-emoji; $e = 1$: emoji;

x word-level word type switcher. $x = 0$: sentiment word (SENT); $x = 1$: discourse word (DISC); $x = 2$: content word (CONT); $x = 3$: background word (BACK).

$C^{DX}_{d,e,(x)}$ # of words with word type as x, emoji switcher as e, and occurring in messages with discourse d.

$C^{DX}_{d,e,(\cdot)}$ # of words with emoji switcher e that occur in messages whose discourse assignments are d, i.e., $\sum_{x=0}^{3} C^{DX}_{d,e,(x)}$.

$N^{DX}_{e,(x)}$ # of words occurring in message (t, m) and with word type assignment as x and emoji switcher e.

$N^{DX}_{(e,\cdot)}$ # of words in message (t, m) with emoji switcher e, i.e., $N^{DX}_{(e,\cdot)} = \sum_{x=0}^{3} N^{DX}_{(e,x)}$.

$\nu_{d,e,x}$ Dirichlet prior parameter for generating word type x given emoji e, i.e., the value in the $(e+1)$-th row and $(x+1)$-th colume in Eq. (5.2). In particular, we let $\nu_{1,e,x} = \nu_{2,e,x} \cdots = \nu_{D,e,x}$

$\nu_{d,e,(\cdot)}$ $\nu_{d,e,(\cdot)} = \sum_{x=0}^{3} \nu_{d,e,x}$

$C^{DW}_{d,(v)}$ # of words indexing v in vocabulary, assigned as discourse word, and occurring in messages assigned discourse d.

$C^{DW}_{d,(\cdot)}$ # of words assigned as discourse words (DISC) and occurring in messages assigned discourse d, i.e., $C^{DW}_{d,(\cdot)} = \sum_{v=1}^{V} C^{DW}_{d,(v)}$.

$N^{DW}_{(v)}$ # of words indexing v in vocabulary that occur in messages (t, m) and are assigned as discourse words (DISC).

$N^{DW}_{(\cdot)}$ # of words assigned as discourse words (DISC) and occurring in sentence (t, s), i.e., $N^{DW}_{(\cdot)} = \sum_{v=1}^{V} N^{DW}_{(v)}$.

$C^{SP}_{s,(p)}$ # of words assigned as sentiment words (SENT) with polarity p that occur in messages assigned sentiment s.

$C^{SP}_{s,(\cdot)}$ # of words assigned as sentiment words that occur in messages assigned sentiment s, i.e., $C^{SP}_{s,(\cdot)} = \sum_{p \in \{\text{POS},\text{NEG}\}} C^{SP}_{s,(p)}$

$N^{SP}_{(p)}$ # of words in message (t, m) assigned as sentiment words (SENT) with polarity p

$N^{SP}_{(\cdot)}$ # of words in message (t, m) assigned as sentiment words (SENT) words, i.e., $N^{SP}_{(\cdot)} = \sum_{p \in \{\text{POS,NEG}\}} N^{SP}_{(p)}$

$C^{DD}_{d,(d')}$ # of messages assigned discourse d' whose parent is assigned discourse d.

$C^{DD}_{d,(\cdot)}$ # of messages whose parents are assigned discourse d, i.e., $C^{DD}_{d,(\cdot)} = \sum_{d'=1}^{D} C^{DD}_{d,(d')}$.

$I(\cdot)$ An indicator function, whose value is 1 when its argument inside () is true, and 0 otherwise.

$N^{DD}_{(d)}$ # of messages whose parent is (t, m) and assigned discourse d.

$N^{DD}_{(\cdot)}$ # of messages whose parent is (t, m), i.e., $N^{DD}_{(\cdot)} = \sum_{d=1}^{D} N^{DD}_{(d)}$

$C^{DS}_{d,s,(s')}$ # of messages assigned as discourse d and sentiment s' whose parent assigned sentiment label s.

$C^{DS}_{d,s,(\cdot)}$ # of messages assigned as discourse d whose parent assigned sentiment label s, i.e., $C^{DS}_{d,s,(\cdot)} = \sum_{s'=1}^{S} C^{DS}_{d,s,(s')}$

$N^{DS}_{(d,s)}$ # of (t, m)'s children that are assigned discourse d and sentiment label s

---

$N^{DS}_{(d,\cdot)}$ # of (t, m)'s children that are assigned discourse d, i.e., $N^{DS}_{(d,\cdot)} = \sum_{s=1}^{S} N^{DS}_{(d,s)}$

$C^{BW}_{(v)}$ # of words indexing v in vocabulary and assigned as background words (BACK)

$C^{BW}_{(\cdot)}$ # of words assigned as background words (BACK), i.e., $C^{BW}_{(\cdot)} = \sum_{v=1}^{V} C^{BW}_{(v)}$

$C^{CW}_{t,(v)}$ # of words indexing v in vocabulary and assigned as content words (CONT) of tree t

$C^{CW}_{t,(\cdot)}$ # of words assigned as content words (CONT) of tree t, i.e., $C^{CW}_{t,(\cdot)} = \sum_{v=1}^{V} C^{CW}_{t,(v)}$

$C^{PW}_{p,(v)}$ # of words indexing v in vocabulary and assigned as sentiment words (SENT) with polarity p

$C^{PW}_{p,(\cdot)}$ # of words assigned as sentiment words (SENT) with polarity p, i.e., $C^{PW}_{p,(\cdot)} = \sum_{v=1}^{V} C^{PW}_{p,(v)}$

For each message m on tree t, we sample its discourse $d_{t,m}$ and sentiment $s_{t,m}$ according to the following conditional probability distribution:

$$\begin{aligned}
&p(d_{t,m} = d, s_{t,m} = s | \mathbf{d}_{\neg(t,m)}, \mathbf{s}_{\neg(t,m)}, \mathbf{w}, \mathbf{x}, \mathbf{p}, \mathbf{e}, \Theta) \qquad (5.2)\\
&\propto \prod_{e=0}^{1} \frac{\Gamma(C^{DX}_{d,e,(\cdot)} + \nu_{d,e,(\cdot)})}{\Gamma(C^{DX}_{d,e,(\cdot)} + N^{DX}_{(e,\cdot)} + \nu_{d,e,(\cdot)})} \prod_{x=0}^{3} \frac{\Gamma(C^{DX}_{d,e,(x)} + N^{DX}_{(e,x)} + \nu_{d,e,x})}{\Gamma(C^{DX}_{d,e,(x)} + \nu_{d,e,x})}\\
&\cdot \frac{\Gamma(C^{DW}_{d,(\cdot)} + V\mu^{disc})}{\Gamma(C^{DW}_{d,(\cdot)} + N^{DW}_{(\cdot)} + V\mu^{disc})} \prod_{v=1}^{V} \frac{\Gamma(C^{DW}_{d,(v)} + N^{DW}_{(v)} + \mu^{disc})}{\Gamma(C^{DW}_{d,(v)} + \mu^{disc})}\\
&\cdot \frac{\Gamma(C^{SP}_{s,(\cdot)} + 2\omega)}{\Gamma(C^{SP}_{s,(\cdot)} + N^{SP}_{(\cdot)} + 2\omega)} \prod_{p \in \{\text{POS}, \text{NEG}\}} \frac{\Gamma(C^{SP}_{s,(p)} + N^{SP}_{(p)} + \omega)}{\Gamma(C^{SP}_{s,(p)} + \omega)}\\
&\cdot \frac{\Gamma(C^{DD}_{d_{t,pa(m)},(\cdot)} + D\lambda)}{\Gamma(C^{DD}_{d_{t,pa(m)},(\cdot)} + I(d_{t,pa(m)} \neq d) + D\lambda)} \cdot \frac{\Gamma(C^{DD}_{d_{t,pa(m)},(d)} + I(d_{t,pa(m)} \neq d) + \lambda)}{\Gamma(C^{DD}_{d_{t,pa(m)},(d)} + \lambda)}\\
&\cdot \frac{\Gamma(C^{DD}_{d,(\cdot)} + D\lambda)}{\Gamma(C^{DD}_{d,(\cdot)} + I(d_{t,pa(m)} = d) + N^{DD}_{(\cdot)} + D\lambda)} \cdot \prod_{d'=1}^{D} \frac{\Gamma(C^{DD}_{d,(d')} + N^{DD}_{(d')} + I(d_{t,pa(m)} = d = d') + \lambda)}{\Gamma(C^{DD}_{d,(d')} + \lambda)}
\end{aligned}$$

$$
\begin{aligned}
&\cdot\frac{\Gamma(C^{DS}_{d,s_{t,pa(m)},(\cdot)}+S\xi)}{\Gamma(C^{DS}_{d,s_{t,pa(m)},(\cdot)}+I(s_{t,pa(m)}\neq s)+S\xi)}\cdot\frac{\Gamma(C^{DS}_{d,s_{t,pa(m)},(s)}+I(s_{t,pa(m)}\neq s)+\xi)}{\Gamma(C^{DS}_{d,s_{t,pa(m)},(s)}+\xi)}\\
&\cdot\Bigg(\prod_{d'=1}^{D}\frac{\Gamma(C^{DS}_{d',s,(\cdot)}+S\xi)}{\Gamma(C^{DS}_{d',s,(\cdot)}+N^{DS}_{(d',\cdot)}+I(d=d')\cdot I(s_{t,pa(m)}=s)+S\xi)}\\
&\cdot\prod_{s'=1}^{S}\frac{\Gamma(C^{DS}_{d',s,(s')}+N^{DS}_{(d',s')}+I(d=d')\cdot I(s_{t,pa(m)}=s=s')+\xi)}{\Gamma(C^{DS}_{d',s,(s')}+\xi)}\Bigg)
\end{aligned}
$$

For each word n in m on t, the sampling formula of its word type $x_{t,m,n}$ (as discourse (DISC), sentiment (SENT), content (CONT), and background (BACK)), and when $x_{t,m,n}$ == SENT (i.e., n being a sentiment word), its sentiment polarity $p_{t,m,n}$ is given as the following:

$$
\begin{aligned}
&p(x_{t,m,n}=x,p_{t,m,n}=p|\mathbf{x}_{\neg(t,m,n)},\mathbf{p}_{\neg(t,m,n)},\mathbf{d},\mathbf{s},\mathbf{w},\mathbf{e},\Theta)\\
\propto&\frac{C^{DX}_{d_{t,m},e_{t,m,n},(x)}+\nu_{d_{t,m},e_{t,m,n},x}}{C^{DX}_{d_{t,m},e_{t,m,n},(\cdot)}+\nu_{d_{t,m},e_{t,m,n},(\cdot)}}\cdot g(x,p,t,m)
\end{aligned}
\tag{5.3}
$$

where

$$
g(x,p,t,m)=\begin{cases}
\frac{C^{PW}_{p,(w_{t,m,n})}+\mu^{pol}}{C^{PW}_{p,(\cdot)}+V\mu^{pol}}\cdot\frac{C^{SP}_{s_{t,m},(p)}+\omega}{C^{SP}_{s_{t,m},(\cdot)}+2\omega} & \text{if } x == \text{SENT}\\
\frac{C^{DW}_{d_{t,m},(w_{t,s,n})}+\mu^{disc}}{C^{DW}_{d_{t,m},(\cdot)}+V\mu^{disc}} & \text{if } x == \text{DISC}\\
\frac{C^{CW}_{t,(w_{t,m,n})}+\mu^{cont}}{C^{CW}_{t,(\cdot)}+V\mu^{cont}} & \text{if } x == \text{CONT}\\
\frac{C^{BW}_{(w_{t,s,n})}+\mu^{back}}{C^{BW}_{(\cdot)}+V\mu^{back}} & \text{if } x == \text{BACK}
\end{cases}
$$

5.3 Summary Extraction

We extract messages from each conversation tree as its summary based on the content distribution γ_t produced by our model (see Section 5.2).

For each conversation tree t, we plug in its content word distribution γ_t produced by our model to the criterion proposed by Haghighi et al. [38]. The goal is to extract L messages forming a summary set E_t^* that closely match γ_t, which captures salient content of tree t and does not include background noise (modeled with β), discourse indicative words (modeled with δ_d), or sentiment words expressing positive or negative sentiment polarity in general (modeled with σ_p). Conversation summarization is cast into the following Integer Programming problem:

$$E_t^* = \arg \min_{|E_t|=L} KL(\gamma_t || U(E_t)) \tag{5.4}$$

where $U(E_t)$ represents the empirical unigram distribution of the candidate summary set E_t and $KL(P||Q)$ denotes the Kullback-Lieber (KL) divergence, i.e., $\sum_w P(w) \log \frac{P(w)}{Q(w)}$.[4]

Since globally optimizing Eq. (5.4) is exponential in the total number of messages in conversation, and is thus an NP problem. We utilize greedy approximation similar to Haghighi et al. [38] to obtain local optimal solutions. Messages are greedily added to a summary so long as they minimize the KL-divergence in the current step.

5.4 Data and Experiment Setup

Data and comparisons. The summarization evaluation was carried out on the same dataset used in Section 4.4. We also compared our model with the same baselines that rank and select messages by 1) LENGTH; 2) POPULARITY (# of reposts and replies); 3) USER influence (# of authors' followers); 4)

[4] To ensure the value of KL-divergence to be finite, we smooth $U(E_t)$ with μ^{cont}, which also serves as the smoothing parameter of γ_t (Section 5.2).

text similarities to other messages using LexRank [28].

We also considered two state-of-the-art summarizers in comparison: 1) Chang et al. [16], a fully *supervised* summarizers with manually crafted features; 2) Soft-LeadSum (see Section 4.3.2), a random walk variant summarizer incorporating outputs of *supervised* discourse tagger, and achieving the best performance in Section 4.5.

In addition, we compared our Full model, which combines everything in Section 5.2 with its variants that model partial information: 1) Cont only, which separates content and non-content (background) information and is equivalent to TopicSum [38]; 2) Sent+cont, which separates sentiment, content, and background components without discourse modeling. It also incorporates emoji-based prior in sentiment inferring. Message-level sentiment labels are generated from sentiment mixtures of conversation trees similar to Lin et al. [67]; 3) Disc+cont (w/o rel), which separates content, discourse, and background information but draw word-type switchers from word-type mixtures of conversation trees instead of relating to message discourse as in (5) of Section 5.2.2. This is an extension of Ritter et al. [98]; 4) CONT+DISC (W/ REL), which jointly models discourse, content, and background information without considering sentiment. Different from DISC+CONT (W/O REL), word type generation is related to message discourse ((5) of Section 5.2.2).

Hyper-parameters. For our models, i.e., FULL MODEL and all its variants, we set the count of discourse roles as $D = 6$ following the categorization of microblog discourse by Zhang et al. [127], and a total number of $S = 3$ message-level sentiment labels representing traditional sentence-level sentiment standard {positive, negative, neutral} [120], and considering any message can be categorized into one of these three classes.

We fixed smoothing parameters not described in Section 5.2 as $\mu^{back} = \mu^{cont} = \mu^{disc} = 0.01$, $\lambda = 50/D$, $\omega = \xi = 0.5$, chose the size of extracted summaries as $L = 10$ messages (the same as Section 4.4) and run Gibbs samplings for 1,000 iterations to ensure convergence.

Preprocessing. Before summarization, we preprocessed the evaluation corpora in the following three steps: 1) Use FudanNLP toolkit [92] for word segmentation of Chinese microblog messages; 2) Generate a vocabulary and remove words occurring less than 5 times; 3) Annotate sentiment polarity of 60 most popular emoji as positive or negative based on Zhao et al. [128]; 4) Replace all mentions and links with "@" and "URL", respectively.

For our models, we only removed digits but left stop words and punctuation because: 1) stop words and punctuation can be useful discourse indicators, such as question marks and "what" suggesting "question" discourse; 2) we have background distribution β to separate useless general information not related to content, discourse, or sentiment, e.g., "do" and "it".

For baselines and the two state-of-the-art summarizers, we filtered out non-Chinese characters in preprocessing keeping the same as traditional settings, which is helpful to them.[5] Also, we did the same post-processing step in Section 4.4 for baselines and state-of-the-art models.

5.5 Summarization Evaluation

The evaluation in this chapter keeps the same as that in Section 4.5. We carried out automatic ROUGE evaluation (see Section 5.5.1) as objective analysis, and human ratings as subjective analysis (see Section 5.5.2).

5.5.1 ROUGE Comparison

In objective analysis, we evaluated the performance of summarizers using ROUGE scores [69] as benchmark, a widely used standard for automatic summarization evaluation based on over-

[5] We also conducted evaluations on the versions without this pre-processing step, and they gave worse ROUGE scores.

Models	Len	ROUGE-1			ROUGE-2		
		Prec	Rec	F1	Prec	Rec	F1
Baselines							
Length	95.4	19.6‡	**53.2**	28.1‡	5.1‡	**14.3**	7.3‡
Popularity	27.2	33.8	25.3‡	27.9‡	8.6	6.1‡	6.8‡
User	37.6	32.2	34.2‡	32.5	8.0	8.9‡	8.2†
LexRank	25.7	**35.3**	22.2‡	25.8‡	11.7	6.9‡	8.3‡
State-of-the-art							
Chang et al. [16]	68.6	25.4†	48.3	32.8	7.0	13.4	9.1
Soft-LeadSum	58.6	27.3‡	45.4	33.7‡	7.6‡	12.6†	9.3‡
Our models & variants							
Cont only	48.6	30.4†	40.4‡	33.6‡	9.2‡	12.0‡	10.0‡
Sent+cont	48.1	31.1	40.2‡	33.7‡	9.9‡	12.3‡	10.5‡
Disc+cont (w/o rel)	37.8	**38.1**	35.5‡	33.1†	**13.2**	11.5‡	10.8‡
Disc+cont (w/ rel)	48.9	32.3	41.3‡	34.0†	10.3†	12.5‡	10.5‡
Full model	52.7	32.5	44.9	**35.9**	11.1	**14.8**	**12.0**

Models	Len	ROUGE-L			ROUGE-SU4		
		Prec	Rec	F1	Prec	Rec	F1
Baselines							
Length	95.4	16.4‡	**44.4**	23.4‡	6.2‡	**17.2**	8.9‡
Popularity	27.2	28.6	21.3‡	23.6‡	10.4	7.6‡	8.4‡
User	37.6	28.0	29.6‡	28.2	9.8	10.6‡	10.0†
LexRank	25.7	**30.6**	18.8‡	22.1‡	**12.3**	7.5‡	8.8‡
State-of-the-art							
Chang et al. [16]	68.6	21.6	41.1	27.9	8.3	16.0	10.8
Soft-LeadSum	58.6	23.3‡	38.6	28.7‡	8.8‡	14.7	10.9‡
Our models & variants							
Cont only	48.6	26.3	34.9‡	29.0†	10.2†	13.8‡	11.3‡
Sent+cont	48.1	27.2	34.8‡	29.3†	10.5‡	13.5‡	11.3‡
Disc+cont (w/o rel)	37.8	**33.3**	30.7‡	28.6†	**13.3**	12.2‡	11.3‡
Disc+cont (w/ rel)	48.9	28.0	35.4‡	29.3†	10.9	14.0‡	11.5†
Full model	52.7	28.2	38.6	**31.0**	11.4	15.7	**12.6**

Remarks:

–Len: count of Chinese characters in the extracted summary.

–Prec, Rec and F1: average precision, recall and F1 ROUGE measure over 10 conversation trees (%).

–Notions † and ‡ means the improvement of our Full model over the corresponding summarizer is significant at level 0.1 ($p < 0.1$) and level 0.05 ($p < 0.05$) based on one-tailed pairwise t-test.

Table 5.2: ROUGE comparison of summarization models

lapping units between a produced summary and a gold-standard reference. Specifically, Table 5.2 reports ROUGE-1, ROUGE-2, ROUGE-L and ROUGE-SU4 outputted by ROUGE 1.5.5[6], which is the same as Section 4.5.1. In addition to that discussed in Section 4.5.1, we have the following observations:

It is important to capture sentiment-discourse and discourse-content relations. Compared with CONT ONLY model, DISC+CONT (W/O REL) and SENT+CONT model further separates components of discourse and sentiment from non-content information, respectively, without modeling sentiment-discourse ((2) of Section 5.2.1) and discourse-content ((5) of Section 5.2.2) relations. But they yielded close results and were all outperformed by DISC+CONT (W/ REL) and FULL MODEL.

Discourse can indicate summary-worthy content. Our DISC+CONT (W/ REL) model achieved the second best performance by exploring the relations between discourse and con-

[6] `github.com/summanlp/evaluation/tree/master/ROUGE-RELEASE-1.5.5`

tent. It yielded competitive and even slightly better performance than Soft-LeadSum that relies on supervised discourse tagger for summarization. This demonstrates that our model, without learning from gold-standard annotation, is able to capture discourse information that helps identify key content for summarization.

Sentiment is useful in discourse modeling and summarization. Exploring the effects of sentiment shifts on discourse detection ((2) in Section 5.2.1) significantly boosted the ROUGE F1 scores indicated by the comparison between Disc+cont (w/ rel) model and our Full model. This is because capturing sentiment shifts helps discourse induction and thus improves identifications of salient content and summarization.

Jointly modeling sentiment, discourse, and content helps summarization. The ROUGE F1 scores produced by our Full model were higher than all the competitors, significantly and by large margins. The generally higher performance of our Full

MODEL is because it effectively separates discourse, sentiment, and content components in conversation structure, and explores their joint effect for summarization.

5.5.2 Human Evaluation

Similar to Section 4.5.2, we conducted human evaluations on informativeness (Info), conciseness (Conc) and readability (Read) of extracted summaries. We invited the same two annotators to read the output summaries and subjectively rated on a 1-5 Likert scale and in 0.5 units. Their overall inter-rater agreement achieved Krippendorff's α of 0.73 in this experiment, which indicates reliable results [55]. Table 5.3 shows the average ratings by two raters and over ten conversation trees.

In informativeness assessment, we outperformed all competitors by large margins, which is consistent with the automatic evaluation results by ROUGE (Section 5.5.1).

In conciseness and readability assessments, SOFT-LEADSUM

Models	Info	Conc	Read
Baselines			
Length	2.33	2.93	2.28
Popularity	2.38	2.35	3.05
User	3.13	3.10	3.75
LexRank	3.05	2.70	3.03
State-of-the-art			
Chang et al. [16]	3.43	3.50	3.70
Soft-LeadSum	3.70	**3.90**	**4.15**
Our models			
Cont only	3.33	3.03	3.35
Sent+cont	3.25	3.30	3.68
Disc+cont (w/o rel)	3.25	3.15	3.55
Disc+cont (w/ rel)	3.35	3.28	3.73
Full model	**3.90**	3.73	**4.15**

Table 5.3: Overall human ratings on summaries

presented in Section 4.3.2 gave the best performance because it learned patterns from human annotated discourse that indicate concise and easy-to-ready messages. Without these prior knowledge, our model gave competitive performance. The reasons are: 1) When separating content components from discourse and sentiment information, it also filtered out irrelevant noise and distilled important information; 2) It can exploit the tendencies of messages with various discourse roles containing core contents, thus is able to identify "bad" discourse roles that bring redundancy or irrelevant noise disturbing reading experience.

Figure 5.5.2 displays the sample summary generated by our Full model from the conversation started by a report about a girl who quit HKU and took the risk of reapplying for universities to pursue her dream. The summary covers salient comments that helps understand public opinions towards the girl's decision.[7]

5.6 Qualitative Analysis on Discourse, Sentiment and Contents

This section qualitatively analyzes outputs of our Full model.

5.6.1 Sentiment Representations

Figure 5.3 illustrates the top 16 words ranked by polarity-word distributions. Our model tends to choose emoji in describing positive and negative sentiment polarity, which is affected by

[7] We only display part of the output.

Original post
...去年辽宁高考文科状元刘丁宁入读香港大学一个月后,放弃72万元全额奖学金，退学回到本溪高中复读只为追求更纯粹的国学，梦想进入北大中文系。今年，她以666分再次拿到辽宁省高考文科最高分...
... Last year, Dingning Liu who won the champion in College Entrance Exam was admitted to HKU. After one month, she quitted, giving up 720K HKD scholarship and went back to high school to reapply university for her dream of studying Chinese ancient civilization in PKU. This year, she's got 666 in College Entrance Exam and won the first place again ...
Summary replies or reposts
我也无法理解，上港大了还想怎样啊，让我们这等丝情何以堪...
I can't understand either. It was HKU! What did she want? This embarrassed us losers...
这种心态这种毅力，比她的选择本身更值得膜拜
It is her mentality and persistence that deserve admiration rather than her choice itself.
考试人才，到港大未必能适应吧。大陆变态社会和教育制度制造的神经病。还国学，病入膏荒了。
Nerd! Maybe she could not adapt to the life in HKU. This is just a nut produced by our deformed society and education. Chinese ancient civilization? She is really hopeless!
看过她上的天天向上，感觉真的是活在自己的世界里，对国学有很深层的热爱。人家在某一领域做的如此出色我就不知道底下那些嘲讽她又考不上港大或北大的人什么心态了
I've seen her on the TV show "Day Day Up". I think she lives in her own world and deeply loves Chinese ancient civilization. She has been so successful in one domain. I don't know why those people sneering at her. They could get in to neither HKU nor PKU.
好厉害！可是！为什么不本科读香港大学研究生再考去北大呢......浪费一年青春多可惜
Good for her! But, why not go to HKU for undergraduate and apply for graduate school in PKU? It is not worth wasting a whole year.

Table 5.4: Sample of the output of our FULL MODEL originally in Chinese and their English translations

Interpretation	$\theta_{s,\text{POS}}$ (%)	$\theta_{s,\text{NEG}}$ (%)
Positive	99.3	0.7
Neutral	50.1	49.9
Negative	0.1	99.9

Table 5.5: Message-level sentiment clusters.

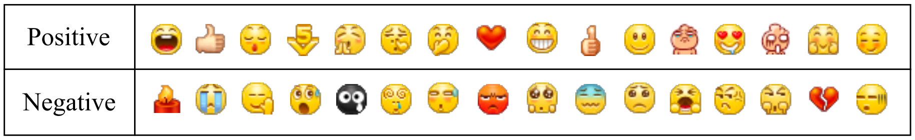

Positive
Negative

Figure 5.3: Word representations for positive and negative sentiment polarity

the emoji prior it incorporates.

To understand each message-level sentiment cluster s, Table 5.5 shows θ_s ((3) of Section 5.2.2), i.e., probabilities of s containing positive and negative words. Based on their different tendencies in containing positive and negative words, we interpret them as *positive*, *neutral* and *negative* sentiment.

5.6.2 Discourse Representations

Figure 5.4 displays the representative words (col 2) for each discourse cluster, i.e., top 30 words ranked by discourse-word distribution, and an example message assigned to it (col 3), both in their English translation versions.[8] In col 1 of both Figure 5.4 and Table 5.6 mentioned later, we use intuitive names in place of cluster numbers. These are based on our interpretations of the clusters, and provided to benefit the reader. We discuss each discourse cluster in turn:

Statement presents arguments and judgments. Messages in this cluster usually give reasons and conditions suggested by words “because”, “if”.

Doubt expresses strong opinions. Indicative words are “!”, “?” and “not”, etc.

Miscellaneous mixes content words like “music”, “fans”, etc., most of which are from a message posted by GEM, a HK singer. Many of her crazy fans copied the message in reposts or replies. Our model captures this abnormal repeating behavior and recognizes it as a special discourse.

Question represents users asking questions to followers, indi-

[8] We only put one English translation in col 2 for multiple Chinese words that share the same meaning.

cated by “?”, “how”, “what”, etc. Interestingly, “@” is also a popular word, which implies that users usually mention others in questions and expect answers from them.

Broadcast is for information sharing without adding new issues. It is dominated by words like “repost” and “forward”. Also prominent are “URL” and “hashtag” as quot in broadcasting.[9]

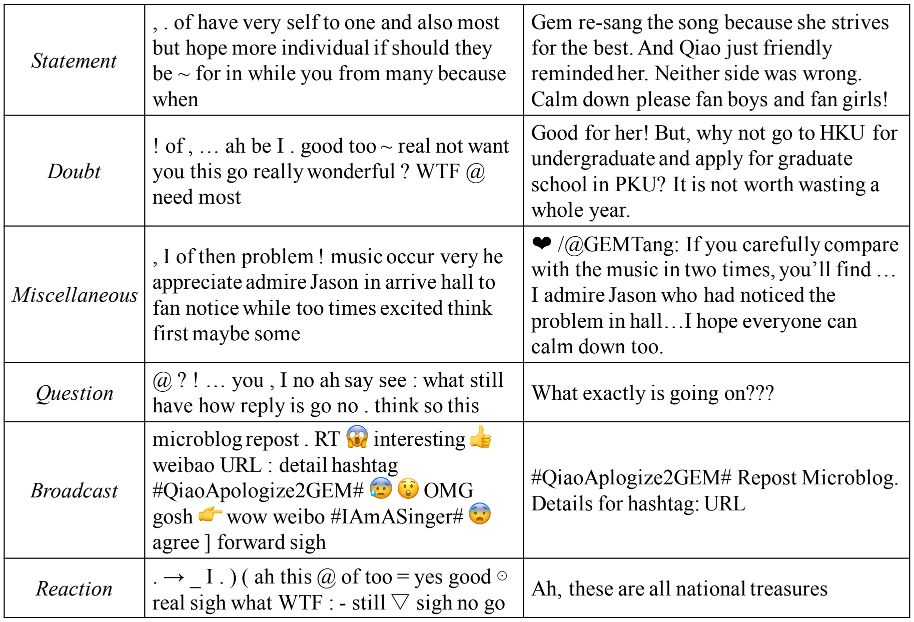
Reaction expresses non-argumentative opinions. Different from statement, it generally voices feelings and responses without detailed explanation (e.g., reasons). There are many symbols being part of emoticons that are scattered by word segmentation, such as “:-)”, “→_→” and “⊙ ▽ ⊙”.[10] This is because of the popularity of adding emoticons in *reaction*.

5.6.3 Discourse, Sentiment, and Summarization

Table 5.6 illustrates how discourse affects sentiment shifts. Col 2-4 show $\psi_{d,s,s}$, i.e., the probabilities of messages keeping sen-

[9] A hashtag contains link to other messages sharing the same hashtag

[10] Emoticons are typographic display in texts. They are different from emoji that are actually pictures. We don’t use emoticons as sentiment prior because they are user-defined thus have many variants and are harder to annotation.



Statement	, . of have very self to one and also most but hope more individual if should they be ~ for in while you from many because when	Gem re-sang the song because she strives for the best. And Qiao just friendly reminded her. Neither side was wrong. Calm down please fan boys and fan girls!
Doubt	! of , … ah be I . good too ~ real not want you this go really wonderful ? WTF @ need most	Good for her! But, why not go to HKU for undergraduate and apply for graduate school in PKU? It is not worth wasting a whole year.
Miscellaneous	, I of then problem ! music occur very he appreciate admire Jason in arrive hall to fan notice while too times excited think first maybe some	❤ /@GEMTang: If you carefully compare with the music in two times, you'll find … I admire Jason who had noticed the problem in hall…I hope everyone can calm down too.
Question	@ ? ! … you , I no ah say see : what still have how reply is go no . think so this	What exactly is going on???
Broadcast	microblog repost . RT 😱 interesting 👍 weibao URL : detail hashtag #QiaoApologize2GEM# 😰 😲 OMG gosh 👉 wow weibo #IAmASinger# 😨 agree] forward sigh	#QiaoAplogize2GEM# Repost Microblog. Details for hashtag: URL
Reaction	. → _ I .) (ah this @ of too = yes good ⊙ real sigh what WTF : - still ▽ sigh no go	Ah, these are all national treasures

Figure 5.4: Produced discourse clusters.

Discourse	Sentiment keeping(%)			# of msgs
	positive	neutral	negative	in summ.
Statement	73.4	10.7	51.0	34
Doubt	76.2	0.4	62.2	52
Miscellaneous	95.4	2.3	82.1	0
Question	46.6	6.4	65.4	13
Broadcast	97.2	0.0	99.9	1
Reaction	34.8	0.3	80.8	0

Table 5.6: Relations between discourse, sentiment, and summarization.

timents of parents conditioned on their discourse roles ((2) of Section 5.2.1).

Sina Weibo messages are dominant by subjective opinions, which results in the rareness of neutral (objective) messages. For subjective (positive and negative) messages, in general, users tend to follow the sentiments of their parents. However, our model inferred that discourse roles vary in extents to follow previous sentiment. For example, *broadcast*, a "sharing only" behavior, almost stays in the same polarity of their parents.

The last column of Table 5.6 shows the count of messages in different discourse clusters that are extracted into conversation summaries. It indicates that our model captured important relation between discourse and good summary messages thus tended to extract summaries from "*statement*" and "*doubt*" messages.

5.7 Conclusion

We have presented a summarization model on microblog conversations that allows joint induction of representations for fine-grained discourse, as well as sentiment and content components in a weakly supervised manner. By rigorously comparing our model with a number of competitive summarizers in automatic ROUGE evaluation and human assessment, we have demonstrated that our model can extract informative, concise, and easy-to-read summaries, and thereby proven the effectiveness of exploiting fine-grained discourse for microblog summarization.

□ **End of chapter.**

Chapter 6

Conclusions and Future Work

6.1 Conclusions

The growing popularity of microblog platforms results in large volume of user-generated data valuable to many real-life applications, e.g., event tracking and user profiling. However, excessive data inevitably leads to the challenge of information overload. To help microblog users separate the wheat from the chaff, this thesis presents different effective *microblog summarization* models, which automatically extract key information from massive and noisy microblog environment.

Natural Language Processing (NLP) researchers and engineers have been wrestling with the difficulties of microblog summarization for years. The challenges are mainly imposed by the intrinsic data sparseness, informal writing styles, and topic diversity nature of the input data. Moreover, simple social network features, e.g., user influence and message popularity, are not necessarily useful for summarization. All these reasons hinder the progress of automatic summarization research for microblog posts.

In this thesis, we propose a novel solution, which uses *conversation structures for microblog summarization.* We first organize microblog messages as conversation trees by the embedded *reposting* and *replying* relations, and then capture *discourse* information therein for summarization purpose.

For summarizing open-domain microblog posts, in Chapter 3, we extracted topics from the microblog posts. Document-level word co-occurrence features are critical to topic modeling. These features are, however, sparse in the short and colloquial microblog messages. To overcome this problem, we captured topic dependencies within conversation trees and proposed *coarse-grained discourse, i.e., "leader-follower" structure* to identify topical words.

Chapter 4 explored the usefulness of *coarse-grained "leader-follower" discourse* for microblog summarization. We focused on summarization of a single conversation tree, and presented a model based on random-walk algorithm that preferably select messages into summaries from leaders than from followers. Leaders were pre-detected using a CRF-based model.

Chapter 5 argued that fine-grained discourse was a better fit for summarization. A probabilistic model was proposed to induce fine-grained discourse simultaneously with sentiment and content representations. The model only required weak supervision with a small emoji lexicon. In summary generation, salient messages were extracted based on content representations. Automatic ROUGE evaluation on large-scale microblog corpus showed that our weakly-supervised model significantly outperformed state-of-the-art fully-supervised summarizers. Subjective human annotations also proved that our proposed model produced better summaries in terms of informativeness, conciseness, and readability than contemporary summarizers.

In summary, the contributions of this thesis are three folds:

- We have set a new direction of treating microblog messages as conversation trees for enriching contextual information.
- We have proposed coarse-grained and fine-grained discourse to capture microblog conversation structure, and have demonstrated their usefulness on microblog-oriented topic extraction and summarization.
- We conducted thorough empirical study of our proposed methods based on large-scale real-world microblog corpora and have released the corpora for future research.

6.2 Future Work

This section discusses possible future research topics aroused from this thesis. These topics are practically relevant to content analysis of social media, which is critical to the advancement of the world digital economy.

Summarization of multiple conversation trees. In this thesis, we focused on summarization of a single conversation tree, which can be analogous to single-document summarization. Nevertheless, a topic cluster may contain multiple conversation trees. For this reason, how to exploit connections and differences between different conversation trees for summarization similar to multi-document summarization is an interesting problem. Here we highlight four key questions to be answered in future research. 1) Are all conversation trees equally important in microblog summarization? 2) If not, how do we rank them? 3) Can discourse structures be useful for this purpose?

4) Also, how to capture shared information between different conversation trees to reduce redundancy in the final summary?

Discourse relations beyond parent-child pairs and hierarchical discourse parsing for microblog conversation trees. Section 2.1 suggested that discourse parsing theories, e.g., RST [76], could uncover the hierarchical discourse structure of documents and had been proven beneficial to single document summarization [75]. Till now, existing automatic discourse parser is still far from satisfactory, especially on conversations based on comprehensive structure and colloquial language style [1, 62]. For this reason, this thesis mainly focused on shallow discourse parsing confined to parent-child message pairs on a conversation tree. In the future, discourse relations beyond parent-child pairs and hierarchical discourse structures could be explored for revealing how messages are connected semantically, functionally, and logically in a conversation tree.

Conversation structure for other microblog-oriented

NLP tasks. Similar to text summarization, many NLP applications, e.g., information extraction and semantic parsing, over short and noisy microblog posts are also very ineffective [23, 27]. This thesis demonstrated the usefulness of contextual and discourse information provided by conversation structures to *microblog summarization* and *topic extraction*. We reckon that conversation discourse is also applicable to other microblog-oriented NLP applications. For example, previous work has pointed out that microblog sentiment analysis was more effective if conducted in conversation context instead of assuming independence of messages [113]. We believe that inter-message discourse relations extracted from conversation structures could improve microblog sentiment analysis. One evidence comes from Chapter 5 where we showed that different discourse roles, e.g., "doubt" and "broadcast", vary in tendencies to trigger sentiment shifts. Also, in meeting domain, discourse features have proven helpful to improve classification of sentiment polarity [106]. Using conversation discourse for microblog sentiment analysis therefore has huge potential and is worth further research.

□ **End of chapter.**

Bibliography

[1] S. D. Afantenos, E. Kow, N. Asher, and J. Perret. Discourse parsing for multi-party chat dialogues. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 928–937, 2015.

[2] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau. Sentiment analysis of twitter data. In *Proceedings of the Workshop on Language in Social Media (LSM 2011)*, June 2011.

[3] Y. Altun, I. Tsochantaridis, and T. Hofmann. Hidden markov support vector machines. In *Proceedings of the Twentieth International Conference on Machine Learning,*

ICML, pages 3–10, 2003.

[4] S. J. Bakker and G. C. Wakker. *Discourse Cohesion in Ancient Greek*, volume 16. Brill, 2009.

[5] J. Baldridge and A. Lascarides. Probabilistic head-driven parsing for discourse structure. In *Proceedings of the 9th Conference on Computational Natural Language Learning, CoNLL*, pages 96–103, 2005.

[6] S. Bangalore, G. D. Fabbrizio, and A. Stent. Learning the structure of task-driven human-human dialogs. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, ACL*, 2006.

[7] L. Barbosa and J. Feng. Robust sentiment detection on twitter from biased and noisy data. In *Proceedings of the 23rd International Conference on Computational Linguistics, Posters Volume, COLING*, pages 36–44, 2010.

[8] S. Bhatia, P. Biyani, and P. Mitra. Summarizing online forum discussions - can dialog acts of individual messages help? In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014*, pages 2127–2131, 2014.

[9] D. M. Blei, T. L. Griffiths, M. I. Jordan, and J. B. Tenenbaum. Hierarchical topic models and the nested chinese restaurant process. In *Proceedings of the 17th Annual Conference on Neural Information Processing Systems, NIPS*, pages 17–24, 2003.

[10] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[11] J. Bollen, H. Mao, and X. Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8, 2011.

[12] L. Carlson, D. Marcu, and M. E. Okurovsky. Building a discourse-tagged corpus in the framework of rhetorical structure theory. In *Proceedings of the SIGDIAL 2001 Workshop, The 2nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 2001.

[13] A. Çelikyilmaz and D. Hakkani-Tür. A hybrid hierarchical model for multi-document summarization. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, ACL*, pages 815–824, 2010.

[14] D. Chakrabarti and K. Punera. Event summarization using tweets. In *Proceedings of the Fifth International Conference on Weblogs and Social Media*, pages 66–73, 2011.

[15] J. Chang, J. L. Boyd-Graber, S. Gerrish, C. Wang, and D. M. Blei. Reading tea leaves: How humans interpret topic models. In *Proceedings of the 23rd Annual Conference on Neural Information Processing Systems, NIPS*,

pages 288–296, 2009.

[16] Y. Chang, X. Wang, Q. Mei, and Y. Liu. Towards twitter context summarization with user influence models. In *Sixth ACM International Conference on Web Search and Data Mining, WSDM*, pages 527–536, 2013.

[17] C. Chemudugunta, P. Smyth, and M. Steyvers. Modeling general and specific aspects of documents with a probabilistic topic model. In *Proceedings of the 20th Annual Conference on Neural Information Processing Systems, NIPS*, pages 241–248, 2006.

[18] F. C. T. Chua and S. Asur. Automatic summarization of events from social media. In *Proceedings of the Seventh International Conference on Weblogs and Social Media, ICWSM*, 2013.

[19] W. W. Cohen, V. R. Carvalho, and T. M. Mitchell. Learning to classify email into "speech acts". In *Proceedings*

of the 2004 Conference on Empirical Methods in Natural Language Processing , EMNLP, pages 309–316, 2004.

[20] J. M. Conroy and D. P. O'Leary. Text summarization via hidden markov models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR*, pages 406–407, 2001.

[21] N. Crook, R. Granell, and S. G. Pulman. Unsupervised classification of dialogue acts using a dirichlet process mixture model. In *Proceedings of The 10th Annual Meeting of the Special Interest Group on Discourse and Dialogue, SIGDIAL 2009*, pages 341–348, 2009.

[22] D. Das and A. F. Martins. A survey on automatic text summarization. *Literature Survey for the Language and Statistics II course at CMU*, 4:192–195, 2007.

[23] L. Derczynski, D. Maynard, N. Aswani, and K. Bontcheva. Microblog-genre noise and impact on semantic annotation accuracy. In *Proceedings of the 24th ACM Conference on Hypertext and Social Media, HT*, pages 21–30, 2013.

[24] R. Dhillon, S. Bhagat, H. Carvey, and E. Shriberg. Meeting recorder project: Dialog act labeling guide. Technical report, DTIC Document, 2004.

[25] Y. Duan, Z. Chen, F. Wei, M. Zhou, and H. Shum. Twitter topic summarization by ranking tweets using social influence and content quality. In *Proceedings of the 24th International Conference on Computational Linguistics, COLING*, pages 763–780, 2012.

[26] Y. Duan, L. Jiang, T. Qin, M. Zhou, and H. Shum. An empirical study on learning to rank of tweets. In *Proceedings of the 23rd International Conference on Computational Linguistics, COLING*, pages 295–303, 2010.

[27] J. Eisenstein. What to do about bad language on the internet. In *Proceedings of Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics: Human Language Technologies, NAACL-HLT*, pages 359–369, 2013.

[28] G. Erkan and D. R. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Intell. Res. (JAIR)*, 22:457–479, 2004.

[29] R. Fan, K. Chang, C. Hsieh, X. Wang, and C. Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

[30] V. W. Feng and G. Hirst. A linear-time bottom-up discourse parser with constraints and post-editing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL, Volume 1: Long Papers*, pages 511–521, 2014.

[31] S. Fisher and B. Roark. The utility of parse-derived features for automatic discourse segmentation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, ACL*, 2007.

[32] J. L. Fleiss, B. Levin, and M. C. Paik. *Statistical methods for rates and proportions.* John Wiley & Sons, 2013.

[33] M. F. Fort, E. Alfonseca, and H. Rodríguez. Support vector machines for query-focused summarization trained and evaluated on pyramid data. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, ACL*, 2007.

[34] D. Galanis, G. Lampouras, and I. Androutsopoulos. Extractive multi-document summarization with integer linear programming and support vector regression. In *Proceedings of the 24th International Conference on Computational Linguistics, COLING*, pages 911–926, 2012.

[35] M. Galley. A skip-chain conditional random field for ranking meeting utterances by importance. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, EMNLP 2006*, pages 364–372, 2006.

[36] T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America*, 101 Suppl 1:5228–35, 2004.

[37] T. L. Griffiths, M. Steyvers, D. M. Blei, and J. B. Tenenbaum. Integrating topics and syntax. In *Proceedings of the 18th Annual Conference on Neural Information Processing Systems, NIPS*, pages 537–544, 2004.

[38] A. Haghighi and L. Vanderwende. Exploring content models for multi-document summarization. In *Proceedings of the 2009 Conference of the North American Chapter of the Association of Computational Linguistics: Human Language Technologies, NAACL-HLT 2009*, pages 362–370,

2009.

[39] S. M. Harabagiu and A. Hickl. Relevance modeling for microblog summarization. In *Proceedings of the 5th International Conference on Web and Social Media, ICWSM*, 2011.

[40] Y. He. Incorporating sentiment prior knowledge for weakly supervised sentiment analysis. *ACM Asian and Low-Resource Language Information Processing TALIP*, 11(2):4:1–4:19, 2012.

[41] G. Heinrich. Parameter estimation for text analysis. *University of Leipzig, Technical Report*, 2008.

[42] T. Hofmann. Probabilistic latent semantic indexing. In *In Proceedings of the 22nd Annual International, ACM SIGIR*, pages 50–57, 1999.

[43] B. Hollerit, M. Kröll, and M. Strohmaier. Towards linking buyers and sellers: detecting commercial intent on twitter.

In *Proceedings of the 22nd International World Wide Web Conference, WWW, Companion Volume*, pages 629–632, 2013.

[44] L. Hong and B. D. Davison. Empirical study of topic modeling in twitter. In *Proceedings of the first workshop on social media analytics*, pages 80–88, 2010.

[45] E. H. Hovy and E. Maier. Parsimonious or profligate: how many and which discourse structure relations. *Discourse Processes*, 1997.

[46] H. D. III and D. Marcu. Bayesian query-focused summarization. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, ACL*, 2006.

[47] D. Inouye and J. K. Kalita. Comparing twitter summarization algorithms for multiple post summaries. In *Pri-*

vacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), pages 298–306, 2011.

[48] Y. Ji and J. Eisenstein. Representation learning for text-level discourse parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL, Volume 1: Long Papers*, pages 13–24, 2014.

[49] Y. Jo and A. H. Oh. Aspect and sentiment unification model for online review analysis. In *Proceedings of the Forth International Conference on Web Search and Web Data Mining, WSDM*, pages 815–824, 2011.

[50] T. Joachims, T. Finley, and C. J. Yu. Cutting-plane training of structural svms. *Machine Learning*, 77(1):27–59, 2009.

[51] S. R. Joty, G. Carenini, and C. Lin. Unsupervised modeling of dialog acts in asynchronous conversations. In *Pro-*

ceedings of the 22nd International Joint Conference on Artificial Intelligence, IJCAI 2011, pages 1807–1813, 2011.

[52] S. R. Joty, G. Carenini, and R. T. Ng. A novel discriminative framework for sentence-level discourse analysis. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL*, pages 904–915, 2012.

[53] D. Jurafsky, E. Shriberg, and D. Biasca. Switchboard swbd-damsl shallow-discourse-function annotation coders manual. *Institute of Cognitive Science Technical Report*, pages 97–102, 1997.

[54] E. Kouloumpis, T. Wilson, and J. D. Moore. Twitter sentiment analysis: The good the bad and the omg! In *Proceedings of the Fifth International Conference on Weblogs and Social Media, ICWSM*, 2011.

[55] K. Krippendorff. *Content analysis: An introduction to its methodology*. Sage, 2004.

[56] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning, ICML*, pages 282–289, 2001.

[57] J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174, 1977.

[58] A. Lazaridou, I. Titov, and C. Sporleder. A bayesian model for joint unsupervised induction of sentiment, aspect and discourse representations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL, Volume 1: Long Papers*, pages 1630–1639, 2013.

[59] P. F. Lazarsfeld, B. Berelson, and H. Gaudet. *The peoples choice: how the voter makes up his mind in a presidential campaign.* New York Columbia University Press, 1948.

[60] C. Li, X. Qian, and Y. Liu. Using supervised bigram-based ILP for extractive summarization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL, Volume 1: Long Papers*, pages 1004–1013, 2013.

[61] J. Li, W. Gao, Z. Wei, B. Peng, and K. Wong. Using content-level structures for summarizing microblog repost trees. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 2168–2178, 2015.

[62] J. Li, R. Li, and E. H. Hovy. Recursive deep models for discourse parsing. In *Proceedings of the 2014 Con-*

ference on Empirical Methods in Natural Language Processing, EMNLP, pages 2061–2069, 2014.

[63] J. Li, M. Liao, W. Gao, Y. He, and K. Wong. Topic extraction from microblog posts using conversation structures. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*, 2016.

[64] J. Li, Z. Wei, H. Wei, K. Zhao, J. Chen, and K. Wong. Learning to rank microblog posts for real-time ad-hoc search. In *Proceedings of the 4th CCF converence on Natural Language Processing and Chinese Computing, NLPCC*, pages 436–443, 2015.

[65] L. Li, K. Zhou, G. Xue, H. Zha, and Y. Yu. Enhancing diversity, coverage and balance for summarization through structure learning. In *Proceedings of the 18th International*

Conference on World Wide Web, WWW, pages 71–80, 2009.

[66] K. W. Lim and W. L. Buntine. Twitter opinion topic model: Extracting product opinions from tweets by leveraging hashtags and sentiment lexicon. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, CIKM 2014, Shanghai, China, November 3-7, 2014*, pages 1319–1328, 2014.

[67] C. Lin and Y. He. Joint sentiment/topic model for sentiment analysis. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM*, pages 375–384, 2009.

[68] C. X. Lin, B. Zhao, Q. Mei, and J. Han. PET: a statistical model for popular events tracking in social communities. In *Proceedings of the 16th International Con-*

ference on Knowledge Discovery and Data Mining, ACM SIGKDD, pages 929–938, 2010.

[69] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Proceedings of the ACL-04 Workshop*, pages 74–81, 2004.

[70] Z. Lin, M. Kan, and H. T. Ng. Recognizing implicit discourse relations in the Penn discourse treebank. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 343–351, 2009.

[71] F. Liu, Y. Liu, and F. Weng. Why is sxsw trending?: exploring multiple text sources for twitter topic summarization. In *Proceedings of the Workshop on Languages in Social Media*, pages 66–75. Association for Computational Linguistics, 2011.

[72] H. Liu, H. Yu, and Z. Deng. Multi-document summarization based on two-level sparse representation model. In *Proceedings of the Twenty-Ninth Conference on Artificial Intelligence, AAAI*, pages 196–202, 2015.

[73] X. Liu, Y. Li, F. Wei, and M. Zhou. Graph-based multi-tweet summarization using social signals. In *Proceedings of the 24th International Conference on Computational Linguistics*, pages 1699–1714, 2012.

[74] R. Long, H. Wang, Y. Chen, O. Jin, and Y. Yu. Towards effective event detection, tracking and summarization on microblog data. In *Web-Age Information Management - 12th International Conference, WAIM*, pages 652–663, 2011.

[75] A. Louis, A. K. Joshi, and A. Nenkova. Discourse indicators for content selection in summarization. In *Proceedings of the SIGDIAL 2010 Conference, The 11th Annual Meet-*

ing of the Special Interest Group on Discourse and Dialogue, 24-15 September 2010, Tokyo, Japan, pages 147–156, 2010.

[76] W. Mann and S. Thompson. Rhetorical structure theory: towards a functional theory of text organization. *Text*, 8(3):243–281, 1988.

[77] C. D. Manning and H. Schütze. *Foundations of statistical natural language processing*. MIT Press, 2001.

[78] D. Marcu. The rhetorical parsing of unrestricted texts: A surface-based approach. *Computational Linguistics*, 26(3):395–448, 2000.

[79] K. McKeown, J. Klavans, V. Hatzivassiloglou, R. Barzilay, and E. Eskin. Towards multidocument summarization by reformulation: Progress and prospects. In *Proceedings of the Sixteenth National Conference on Artificial Intelli-*

gence and Eleventh Conference on Innovative Applications of Artificial Intelligence, pages 453–460, 1999.

[80] R. Mehrotra, S. Sanner, W. L. Buntine, and L. Xie. Improving LDA topic models for microblogs via tweet pooling and automatic labeling. In *Proceedings of the 36th International conference on research and development in Information Retrieval, ACM SIGIR*, pages 889–892, 2013.

[81] Q. Mei, J. Guo, and D. R. Radev. Divrank: the interplay of prestige and diversity in information networks. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,KDD*, pages 1009–1018, 2010.

[82] X. Meng, F. Wei, X. Liu, M. Zhou, S. Li, and H. Wang. Entity-centric topic-oriented opinion summarization in twitter. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data*

Mining, KDD, pages 379–387, 2012.

[83] R. Mihalcea and P. Tarau. Textrank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing , EMNLP*, pages 404–411, 2004.

[84] D. M. Mimno, H. M. Wallach, E. M. Talley, M. Leenders, and A. McCallum. Optimizing semantic coherence in topic models. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 262–272, 2011.

[85] G. Murray, S. Renals, J. Carletta, and J. D. Moore. Incorporating speaker and discourse features into speech summarization. In *Proceedings of the 2006 Conference of the North American Chapter of the Association of Computational Linguistics: Human Language Technologies, NAACL-HLT*, 2006.

[86] A. Nenkova and K. McKeown. A survey of text summarization techniques. In *Mining Text Data*, pages 43–76. Springer, 2012.

[87] T. Oya and G. Carenini. Extractive summarization and dialogue act modeling on email threads: An integrated probabilistic approach. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue, SIGDIAL 2014*, pages 133–140, 2014.

[88] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[89] B. Peng, J. Li, J. Chen, X. Han, R. Xu, and K. Wong. Trending sentiment-topic detection on twitter. In *Processings of the 16th International Conference on Computational Linguistics and Intelligent Text, CICLing, Part II*, pages 66–77, 2015.

[90] A. Popescu and M. Pennacchiotti. Detecting controversial events from twitter. In *Proceedings of the 19th ACM Conference on Information and Knowledge Management, CIKM*, pages 1873–1876, 2010.

[91] R. Prasad, N. Dinesh, A. Lee, E. Miltsakaki, L. Robaldo, A. K. Joshi, and B. L. Webber. The Penn discourse treebank 2.0. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2008, 26 May - 1 June 2008, Marrakech, Morocco*, 2008.

[92] X. Qiu, Q. Zhang, and X. Huang. Fudannlp: A toolkit for chinese natural language processing. In *Proceedings of Annual Meeting of the Association for Computational Linguistics, ACL*, pages 49–54, 2013.

[93] X. Quan, C. Kit, Y. Ge, and S. J. Pan. Short and sparse text topic modeling via self-aggregation. In *Proceedings*

of the 24th International Joint Conference on Artificial Intelligence, IJCAI, pages 2270–2276, 2015.

[94] D. R. Radev, T. Allison, S. Blair-Goldensohn, J. Blitzer, A. Çelebi, S. Dimitrov, E. Drábek, A. Hakim, W. Lam, D. Liu, J. Otterbacher, H. Qi, H. Saggion, S. Teufel, M. Topper, A. Winkel, and Z. Zhang. MEAD - A platform for multidocument multilingual text summarization. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation, LREC*, 2004.

[95] D. R. Radev, E. H. Hovy, and K. McKeown. Introduction to the special issue on summarization. *Computational Linguistics*, 28(4):399–408, 2002.

[96] D. Ramage, S. T. Dumais, and D. J. Liebling. Characterizing microblogs with topic models. In *Proceedings of the 4th International Conference on Web and Social Media, ICWSM*, 2010.

[97] D. Ren, X. Zhang, Z. Wang, J. Li, and X. Yuan. Weiboevents: A crowd sourcing weibo visual analytic system. In *IEEE Pacific Visualization Symposium, PacificVis*, pages 330–334, 2014.

[98] A. Ritter, C. Cherry, and B. Dolan. Unsupervised modeling of twitter conversations. In *Proceedings of the 2010 Conference of the North American Chapter of the Association of Computational Linguistics: Human Language Technologies, NAACL-HLT*, pages 172–180, 2010.

[99] K. D. Rosa, R. Shah, B. Lin, A. Gershman, and R. Frederking. Topical clustering of tweets. *Proceedings of the ACM SIGIR 3rd Workshop on Social Web Search and Mining*, 2011.

[100] M. Rosen-Zvi, T. L. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. In

UAI '04, Proceedings of the 20th Conference in Uncertainty in Artificial Intelligence, pages 487–494, 2004.

[101] G. Salton, A. Singhal, M. Mitra, and C. Buckley. Automatic text structuring and summarization. *Information Processing and Management*, 33(2):193–207, 1997.

[102] B. Sharifi, M.-A. Hutton, and J. Kalita. Automatic summarization of twitter topics. In *National Workshop on Design and Analysis of Algorithm*, 2010.

[103] C. Shen, F. Liu, F. Weng, and T. Li. A participant-based approach for event summarization using twitter streams. In *Proceedings of the 2013 Conference of the North American Chapter of the Association of Computational Linguistics: Human Language Technologies, NAACL-HLT*, pages 1152–1162, 2013.

[104] D. Shen, J. Sun, H. Li, Q. Yang, and Z. Chen. Document summarization using conditional random fields. In

Proceedings of the 20th International Joint Conference on Artificial Intelligence, IJCAI, pages 2862–2867, 2007.

[105] A. Siddharthan, A. Nenkova, and K. McKeown. Syntactic simplification for improving content selection in multi-document summarization. In *Proceedings of the 20th International Conference on Computational Linguistics, Proceedings of the Conference, COLING*, 2004.

[106] S. Somasundaran, J. Wiebe, and J. Ruppenhofer. Discourse level opinion interpretation. In *Proceedings of the 22nd International Conference on Computational Linguistics, COLING*, pages 801–808, 2008.

[107] R. Soricut and D. Marcu. Sentence level discourse parsing using syntactic and lexical information. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT*, 2003.

[108] V. K. R. Sridhar. Unsupervised entity linking with abstract meaning representation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT*, pages 1130–1139, 2015.

[109] A. Stolcke, K. Ries, N. Coccaro, E. Shriberg, R. A. Bates, D. Jurafsky, P. Taylor, R. Martin, C. V. Ess-Dykema, and M. Meteer. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational Linguistics*, 26(3):339–373, 2000.

[110] R. Subba and B. D. Eugenio. An effective discourse parser that uses rich linguistic information. In *Proceedings of the 2009 Conference of the North American Chapter of the Association of Computational Linguistics: Human Language Technologies, NAACL-HLT*, pages 566–574, 2009.

[111] H. Takamura, H. Yokono, and M. Okumura. Summarizing

a document stream. In *Advances in Information Retrieval - 33rd European Conference on IR Research, ECIR*, pages 177–188, 2011.

[112] H. L. Thanh, G. Abeysinghe, and C. R. Huyck. Generating discourse structures for written text. In *Proceedings of the 20th International Conference on Computational Linguistics, COLING*, 2004.

[113] A. Vanzo, D. Croce, and R. Basili. A context-based model for sentiment analysis in twitter. In *Proceedings of the 25th International Conference on Computational Linguistics, COLING: Technical Papers*, pages 2345–2354, 2014.

[114] S. Vosoughi and D. Roy. Tweet acts: A speech act classifier for twitter. In *Proceedings of the Tenth International Conference on Web and Social Media, ICWSM*, pages 711–715, 2016.

[115] H. Wang, D. Zhang, and C. Zhai. Structural topic model for latent topical structure analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, ACL*, pages 1526–1535, 2011.

[116] L. Wang and C. Cardie. Domain-independent abstract generation for focused meeting summarization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, Volume 1: Long Papers*, pages 1395–1405, 2013.

[117] X. Wang and A. McCallum. Topics over time: a non-markov continuous-time model of topical trends. In *Proceedings of the 12th International Conference on Knowledge Discovery and Data Mining, ACM SIGKDD*, pages 424–433, 2006.

[118] J. Weng and B. Lee. Event detection in twitter. In *Proceedings of the Fifth International Conference on Weblogs*

and Social Media, Barcelona, Catalonia, Spain, July 17-21, 2011, 2011.

[119] J. Weng, E. Lim, J. Jiang, and Q. He. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the 3rd International Conference on Web Search and Web Data Mining, WSDM*, pages 261–270, 2010.

[120] J. Wiebe, R. F. Bruce, and T. P. O'Hara. Development and use of a gold-standard data set for subjectivity classifications. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics, ACL*, 1999.

[121] F. Wolf and E. Gibson. Representing discourse coherence: A corpus-based study. *Computational Linguistics*, 32(2):249–287, 2005.

[122] K. Wong, M. Wu, and W. Li. Extractive summarization using supervised and semi-supervised learning. In *Pro-*

ceedings of the 22nd International Conference on Computational Linguistics, COLING, pages 985–992, 2008.

[123] K. Woodsend and M. Lapata. Multiple aspect summarization using integer linear programming. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL*, pages 233–243, 2012.

[124] R. Yan, L. Kong, C. Huang, X. Wan, X. Li, and Y. Zhang. Timeline generation through evolutionary trans-temporal summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 433–443, 2011.

[125] X. Yan, J. Guo, Y. Lan, and X. Cheng. A biterm topic model for short texts. In *Proceedings of the 22nd International World Wide Web Conference, WWW*, pages 1445–1456, 2013.

[126] E. Zarisheva and T. Scheffler. Dialog act annotation for twitter conversations. In *Proceedings of the 2015 Conference, The 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue, SIGDIAL*, pages 114–123, 2015.

[127] R. Zhang, W. Li, D. Gao, and O. You. Automatic twitter topic summarization with speech acts. *IEEE Trans. Audio, Speech & Language Processing*, 21(3):649–658, 2013.

[128] J. Zhao, L. Dong, J. Wu, and K. Xu. Moodlens: an emoticon-based sentiment analysis system for chinese tweets. In *Proceedings of The 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD*, pages 1528–1531, 2012.

[129] W. X. Zhao, J. Jiang, J. Weng, J. He, E. Lim, H. Yan, and X. Li. Comparing twitter and traditional media using topic models. In *Advances in Information Retrieval - 33rd*

European Conference on IR Research, ECIR, pages 338–349, 2011.

[130] L. Zhou and E. H. Hovy. A web-trained extraction summarization system. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technology HLT-NAACL*, 2003.

[131] X. Zhou, X. Wan, and J. Xiao. Cminer: Opinion extraction and summarization for chinese microblogs. *IEEE Transactions on Knowledge and Data Engineering*, 28(7):1650–1663, 2016.

[132] X. Zhou, X. Zhang, and X. Hu. Dragon toolkit: Incorporating auto-learned semantic knowledge into large-scale text retrieval and mining. In *Proceedings of 19th IEEE International Conference on Tools with Artificial Intelligence, ICTAI, Volume 2*, pages 197–201, 2007.