VIRAL TOPIC PREDICTION AND DESCRIPTION IN MICROBLOG SOCIAL NETWORKS

BIAN JINGWEN

(B.S., PKU, CHINA)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
SCHOOL OF COMPUTING
NATIONAL UNIVERSITY OF SINGAPORE

2015

DECLARATION

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

Bian Jingwen Aug 20, 2015

ACKNOWLEDGEMENTS

The five years of Ph.D in NUS will definitely be one of the most important durations in my life. In retrospect, many people provided their grateful help during my tough moments. Here, I would like to take this opportunity to thank them.

In the first place, I would like to show my greatest gratitude to my supervisor, Prof. Chua Tat-Seng. Being a respectable and responsible professor, he provided great help during my research process. He supported me a lot when my research stuck, and his insightful suggestions guided me to the right direction. I would also express my sincerely appreciation to my doctoral committee, Prof. Min-Yen Kan and Prof. Anthony K. H. TUNG. Their constructive feedback and comments have been significantly helpful in shaping this thesis.

Also, I am very grateful for the harmonious environment in my lab. I would like to thank Dr. Yang Yang and Dr. Zhang Hanwang, who generously provided useful suggestions and help to my work. Besides, I feel very lucky to be labmates with the following people: Geng Xue, Song Xuemeng, Chen Jingyuan, Zhao Na, Nie Liqiang, Wang Fanglin, Wang Xiangyu, Gao Yue. It is them that made the research life less boring.

My special thanks give to my flat-mates and other friends: Shen yanyan, Wang Guanfeng, Liu Xiao, Chen Chao, Yang Jing, Yang Jie and He Lian. Being my best friends, the company of them made my life more colorful. Having the chance to know them is the most wonderful thing happened to me in Singapore.

Finally, I would like to express my thanks to my beloved parents, for their selfless love and endless support. They teach me to be kindhearted, and to cherish all the good things in the wold. I love them more than words can say.

CONTENTS

SUMMARY

Microblogging services have revolutionized the way people exchange information, and have emerged as an essential forum for people to air their views on topics of common interests. Therefore, monitoring and analyzing the rich and continuous flow of user-generated contents in microblog networks can yield unprecedentedly valuable information, which would not have been available from traditional media outlets. In particular, microblogs naturally unfold events occurring in the real-world. By monitoring on "viral topics" in microblog networks, *i.e.*, topics that receive a large volume of discussion as well as a large number of participants within a short period, we can make microblog a valuable source of information for individuals and organizations to stay informed of "what is happening and hot now". In this thesis, we aim to carry out a thorough study on viral topics in microblog networks. Specifically, the main aim of our study is to design and develop a viral topic monitoring system for microblog networks, which is able to predict, detect and summarize viral topics.

First, we investigate the prediction of viral microblogs by learning the influences among users in a microblog network. This component targets at predicting whether a piece of microblog will become viral, and which part of the network will participate in propagating this message. To facilitate the prediction ability, we firstly define three types of influences that will affect a user's decision on whether to perform a diffusion action, and propose a diffusion-targeted influence model to differentiate and quantify various types of influence. The problem of diffusion prediction is then modelled as factorizing a user's intention to transmit a microblog into these influences. In this way, a prediction model is achieved, which is able to predict the virality of incoming new microblogs.

Second, we explore the problem of microblog tracking. Due to the existence of celebrity effect, advertising needs and zombie accounts, a large portion of the viral messages predicted in the previous component are not topic-related, which cannot lead to sufficient follow-up discussions. Therefore, we further propose the second component, where the previously predicted viral microblogs are utilized to monitor on the incoming microblog stream. In this component, a novel dictionary learning based method is proposed for tracking an individual microblog. This component aims to filter out non-topic microblogs, and detect the occurrence of viral topics in the early stage.

Finally, we examine how to conduct summarization for the predicted viral topics, which are in the form of a collection of related microblogs, with too much information to be presented. To express the contents concisely and make the topic readable to people, in this step, a multimedia topic summarization scheme is proposed. Given a collection of microblogs related to a topic, this scheme is able to automatically generate a summary for this topic, which contains both textual and visual information.

Through extensive experiments conducted on the large-scale real-world datasets, the experimental results have demonstrated that our study could yield significant gains in providing users with timely and concise information about the occurrence of incoming viral topics.

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

Recent years have seen a transformation in the type of content available on the web. During the first decade of the web's prominence–from the early 1990s onwards–most online content resembled traditional published material: the majority of the web users were consumers of content, created by a relatively small amount of publishers. With the rapid and continuous development of Internet technology, more and more web applications begin to be utilised by users in an interactive way. Therefore, the role of an Internet user is constantly changing: from being only a browser, player, or consumer as in traditional way, to a participator, developer, producer and aggregator. More and more information and contents on the Internet are generated by users, which promote the development of a new type of media: *Social Media*. Unlike traditional media, *e.g.*, newspaper, magazine, and TV, social media is defined as "a group of Internet-based applications that build on the ideological and technological

foundations of Web 2.0, and that allow the creation and exchange of User Generated Content (UGC)" [63]. After the development and enrichment through all these years, the current social media family includes many substantial, valuable and diverse service types, including blogs (*e.g.*, Blogger), social networking sites (*e.g.*, Facebook), collaborative projects (*e.g.*, Wikipedia), video and photo sharing communities (*e.g.*, YouTube and Flickr), and social bookmarking sites (*e.g.*, Pinterest and Tumblr) [4].

Aside from these famous and mature web platforms listed above, a new type of social media service, *microblog*, emerges recently. Microblog provides a light-weight communication platform that enables users to broadcast and share information about their activities, options and status [60]. Ever since the launch of the first microblogging site, microblog has drawn much attention from the public. Due to its convenience and openness (the social relation in microblog is one direction, and not restricted to real-world friend relationship), microblog has become one of the most important channels for people to get information and share their views about the events occurring in the real world. Currently, the number of active microblog users is very huge, which results in tremendous volume of microblog posts everyday, covering various types of contents. In this thesis, we particularly focus on one important phenomenon of microblog: the viral topic, which is a topic that attracts much attention from many users, and receives a considerable number of related posts in a short period. Specifically, we are interested in the *prediction* of promising viral topics, the early-stage *detection* of the microblog posts which are related to the viral topics, as well as the *summarization* for the multimedia contents of these topics.

This chapter first introduces the background of microblogging social networks and viral topics in Section 1.1, followed by the motivation of our thesis in

Section 1.2. In Section 1.3, we will introduce the research problems and describe our strategies in tackling these challenges. Finally, we summarize our research contributions and organization in Section 1.4 and Section 1.5, respectively.

1.1 Background

Microblog is a service that allows the users to publish short messages and broadcast these messages to other users of the service, which integrates all the functions of blog, social networking service and instant messaging software. The appeal of microblog is both its immediacy and portability. Microblog posts are brief (typically 140 to 200 characters) and can be written or received with a variety of channels, including web, mobile and client program. In the earlier years, most of the microblog broadcasts are posted as text. Nowadays, the popularity of cell phones makes it convenient to incorporate multimedia contents (e.g., image, audio and video) into microblog posts, causing the content of microblogs to become more and more multimedia. With microblog, it becomes possible that everyone can express his/her opinions publicly and every user might be followed by anyone else.

With the characteristics of having low barriers for entry, concise contents and strong real-time and interactivity, microblog has quickly moved into the mainstream. Take Twitter as an example. It was first launched in July 13 of 2006. Currently, it is one of the top 10 most visited Internet sites. Figure 1.1 shows the timeline with the amount of monthly active Twitter users worldwide from 2010 to 2015. As we can see, at the beginning of the 2013, Twitter has already surpassed 200 million monthly active users per quarter. Several other statistics of Twitter by the time of July 2014 are listed in Figure 1.2, which

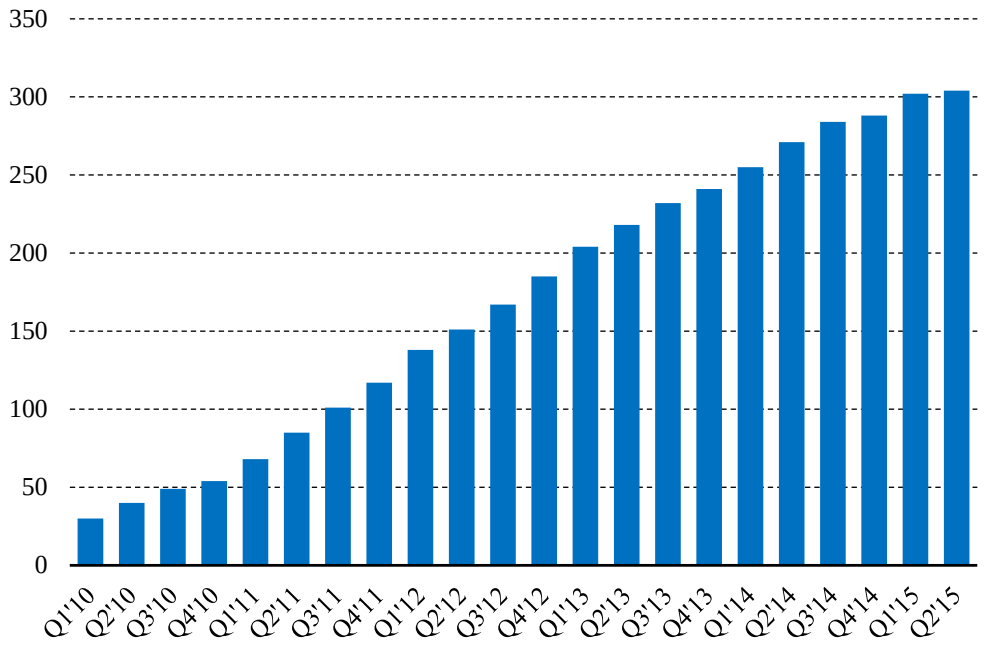


Figure 1.1: Number of monthly active Twitter users worldwide from 1st quarter 2010 to 2nd quarter 2015 (in millions) [1].

Twitter Company Statistics	Data
Total number of active registered Twitter users	645,750,000
Number of new Twitter users signing up everyday	135,000
Number of unique Twitter site visitors every month	190 million
Average number of tweets per day	58 million
Number of Twitter search engine queries every day	2.1 billion
Percent of Twitter users who use their phone to tweet	43%
Number of active Twitter users every month	115 million
Number of tweets that happen every sencond	9,100
Number of days it takes for 1 billion tweets	5 days

Figure 1.2: Statistics of Twitter by the time of July 2014.

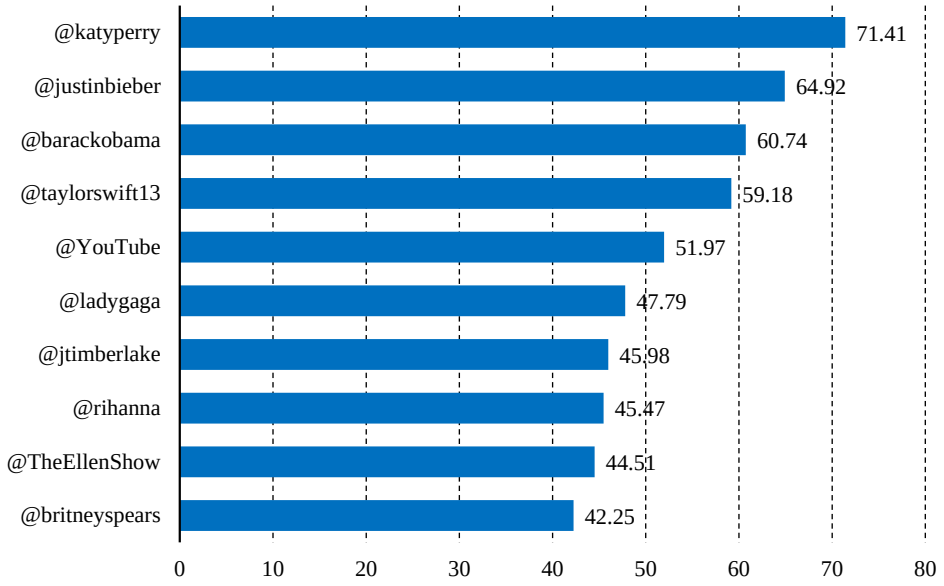


Figure 1.3: Twitter accounts with the most followers worldwide as of June 2015 (in millions) [2].

provide us evidence of the signification of microblogging service. In addition to the tremendous number of users and data volume, the user quality is also remarkable. A lot of celebrities have opened their accounts on Twitter. In the United States, for example, Presidential Barack Obama blogged regularly from the campaign trail. Traditional media organizations, including The New York Times and BBC, have already started sending headlines and links in microblog posts. Figure 1.3 and Figure 1.4 list the follower numbers of the 10 most popular Twitter accounts and 10 most popular retailers, respectively. From these numbers, we can gain an intuitive sense of the important role microblog plays in gathering people from all over the world and making it possible for the users and organizations to spread their updates extensively towards their followers without barrier and delay.

The above listed facts present a clear current state of microblog. To understand the special facets that promote the fast development of microblog, we

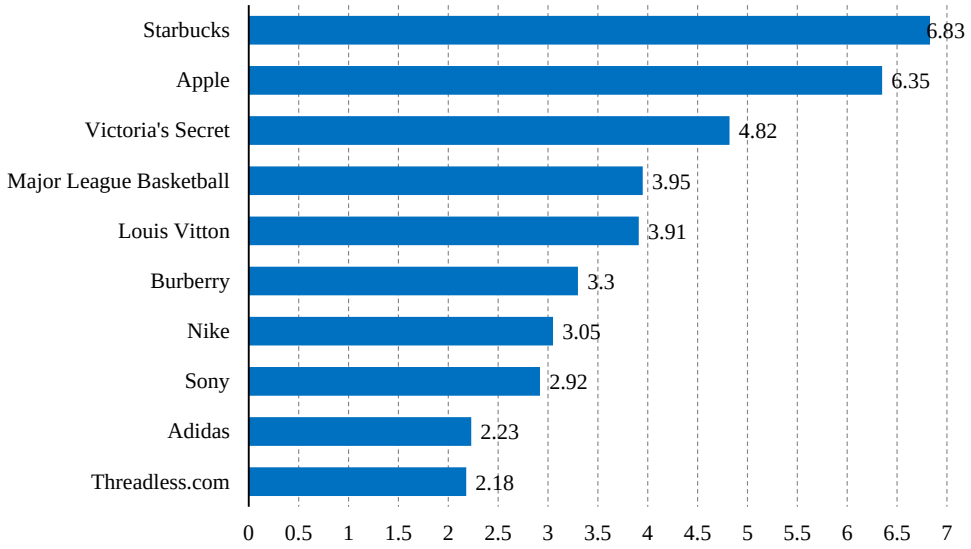


Figure 1.4: Most popular retailers on Twitter in 2014, based on number of followers (in millions) [3].

summarize its key characteristics into the following two types of factors:

- **Content Factor**: frequent brief updates about personal life activities, real-time news information and people-based RSS feed.
- **Technology Factor**: brevity, mobility and pervasive access, and broadcast nature.

Due to the above factors, monitoring and analyzing the rich and continuous flow of user-generated contents in microblog networks can yield unprecedentedly valuable information, which would not have been available from traditional media outlets.

1.2 Motivation

According to the analysis of the characteristics of microblog, the user generated contents in microblogging networks provide abounding valuable information,

which is real-time, all-embracing, and easy to spread to huge number of users. Therefore, microblog becomes an excellent channel for individuals, corporations, and government organizations to monitor "what is happening now". For the monitoring task, the first step is to select what type of topics to monitor. Previous works in this area either focus on monitoring specific organization (the topics to be monitored are defined as all topics related to the organization) [26], or specific type of events (*e.g.*, the occurrence of earthquakes [31]). In this thesis, we differ from previous works by defining our monitoring task as universal viral topics, *i.e.*, the topics which receive large volume of discussion as well as a large number of participants, while not restricted to any specific organizations, communities, or types of events. Although viral topics have gained extensive attention, most of the previous works can only detect viral topics that have already received a large number of reposts. In our monitoring task, we would like to take one step ahead of the outbreak detection: start to monitor a viral topic when it shows the potential to become viral. Therefore, we are facing with two problems: (1) how to predict which topics will become viral, and (2) how to present the viral topics to the user. We will discuss these two problems in detail below.

Viral Topic Prediction. Innumerable topics appear in microblogging streams every day. However, most of these topics will not gain much attention from the users and vanish very quickly, while only a tiny portion of them will become viral and worth monitoring. From the view of building a practical viral topic monitoring system, it is very important if there exist a mechanism to predict which topics have high probabilities to become viral in the near future. By doing this, the system could avoid spending too much resources on those ordinary topics, thus focusing more on those valuable trending ones.

From the view of providing services, predicting the forthcoming viral topics is useful in many ways. For instance, many business and administrative decision makers can gain benefits from viral topic prediction. To promote business, companies can design advertisement strategies which conform to the contents of the forthcoming viral topics, thereby taking advantage of the trend of the viral topics to promote the exposure of their advertisements. News media may publish news stories in the early stage to make greater impact to attract more readers. Other applications include online traffic management and server bandwidth adjustment. All the listed scenarios remind us of the usefulness and necessity of investigating the problem of viral topic prediction.

Multimedia Topic Summarization. Once we successfully predicted the occurrence of a viral topic, a subsequent question arises: how to present this topic to the users. At this stage, a viral topic exists in the form of a collection of many related microblog posts. Although the related posts collection can provide cues of the existence of this topic, this form of information is not user-friendly, since it contains too much detailed information for the readers to browse. Without effective summarization mechanism, the users will be confronted with incomplete, irrelevant and duplicate information. This makes it difficult to capture the essence of a topic and possible to miss information indicating a valuable sub-direction. Therefore, it would be of great benefit if an effective mechanism can be provided for summarizing the microblog collection to provide a user-friendly description for the predicted viral topics.

Different from traditional documents that contain only textual objects, microblog posts are comprised of contents of various media types, such as images and video links. Such high proportion of multimedia contents are potentially precious resources for generating a comprehensive summary. The benefit of in-

corporating different media types into summarization is three-fold: 1) In many cases, images contain essential information which could not be completely expressed by the microblog texts. Therefore, the visual information is of great significance for summarizing the topic and remedying the descriptive power of short texts. In addition, when the emphasis of a topic lies in the visual part, it will not be meaningful if only textual summary is generated. 2) Multimedia contents can facilitate subtopic discovery. Intuitively, given a viral topic, multimedia contents from different subtopics should have lower visual similarity while those within the same subtopic should have higher visual similarity. Thus, discriminative information embedded in visual information of multimedia contents can be exploited as critical cues for subtopic discovery. 3) Incorporating concrete multimedia exemplars into summarization can assist users to gain a more visualized understanding of interesting topics. Therefore, instead of generating a text-based summary, a visualized summary with multiple media types is preferred for better describing the content of the predicted viral topic.

1.3 Strategies

One straightforward method to predict whether a topic will become viral is to predict the amount of microblogs related to this topic. However, this solution is intractable because of the following two reasons: (1) To monitor a viral topic as early as possible, prediction of its virality should be performed in the very early stage, when the amount of posts related to this topic is very small as compared to the whole data stream. In this case, it will be very difficult to detect the existence of this topic from such small amount of data. (2) Even if the viral topics can be successfully detected in the early stage, due to the large number

of users, it is still very challenging to predict how many users will participate in the propagation process of this topic, thus even more difficult to predict the trend of this topic.

Due to the above two challenges, we adopt an indirect method to predict viral topics, which takes advantages of the repost mechanism of microblog. According to our observation, a viral topic in the microblog stream is usually led by some viral microblogs (*i.e.*, microblog posts that receives a large amount of reposts). In other words, at the very early stage, the outbreak of a topic is mostly caused by the large amount of reposts of a few microblogs. And in the following step, more and more individual discussions and comments join in, further promoting the propagation of this topic. Based on this observation, we propose a two-phase solution for viral topic prediction: predicting viral microblogs first, then detecting viral topics based on the prediction results.

As illustrated in Figure 1.5, our system comprises three main components, including viral microblog prediction, microblog tracking, and multimedia topic summarization. Specifically, given a microblog social network and the microblog stream, our system first predicts whether the incoming microblog has the potential to become viral in the near future. After this step, a collection of potential viral microblogs are obtained. However, not all the microblogs can be viewed as a "topic". A viral microblog is very likely to be just a single hot post and may not receive much subsequent discussion. Therefore, in the following phase, the system will track the obtained potential viral microblogs and filter out those "single microblogs", meanwhile detecting and tracking viral topics. After these two phases, the viral topics are obtained, together with the microblog posts related to these topics. Taking the related microblog collection as input, the last component summarizes these posts and automatically

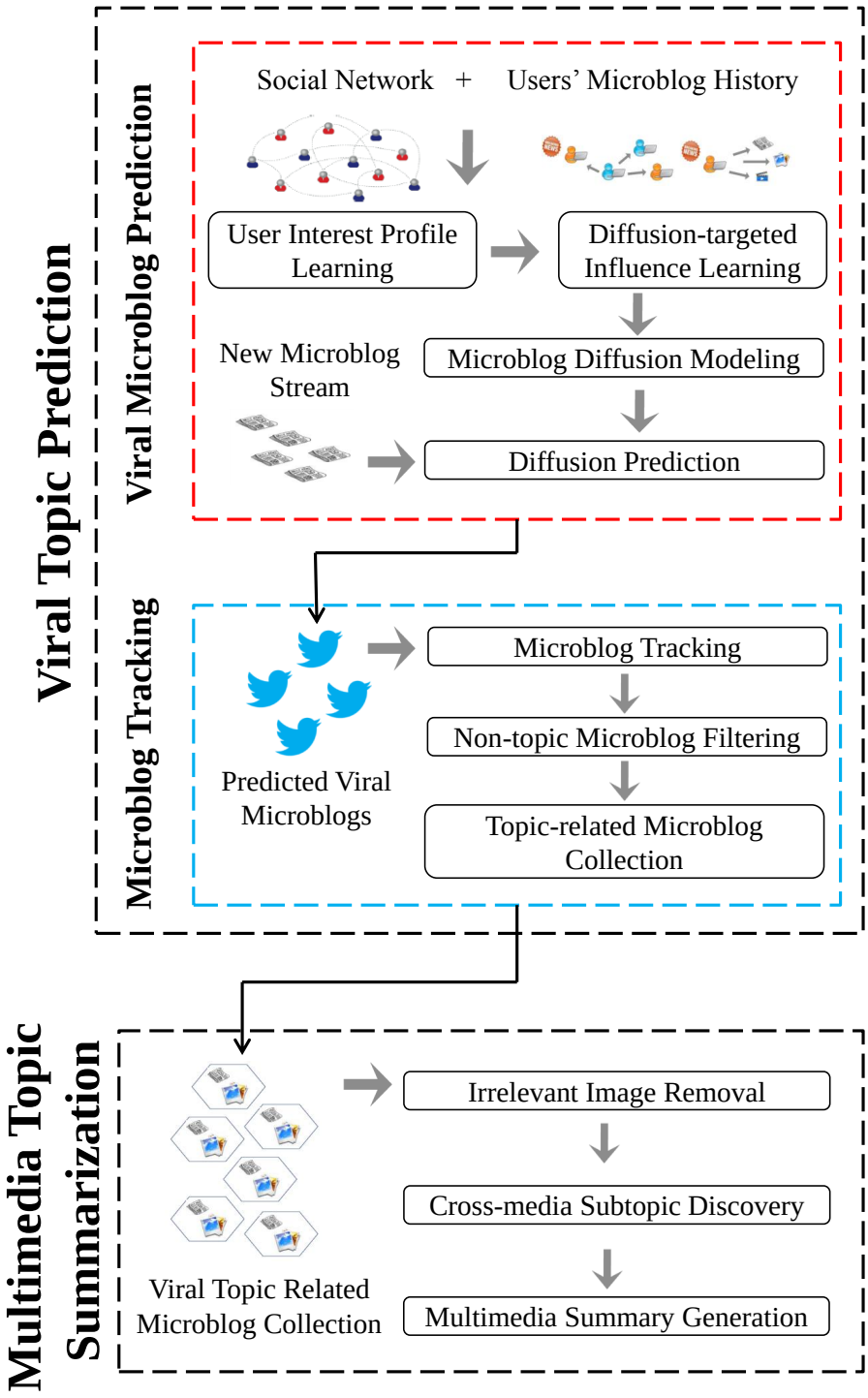


Figure 1.5: Architecture of the proposed viral topic monitoring system.

generates a multimedia summary. The generated summary is viewed as a presentation for the detected viral topic, which include both textual description, as well as image illustration.

1.3.1 Viral Microblog Prediction

In order to predict the propagation trend of a microblog, we first analyse the motivation behind repost actions. A user's action of propagating a microblog from a friend may be affected by various factors. Generally, there are three factors that contribute to a diffusion action: 1) the content of the microblog is in accordance with this user's interest; 2) the microblog is posted by this user's close friend, and his repost action is due to social needs; and 3) the information is epidemic (e.g., a piece of breaking news), and his propagation action is a result of conformity behavior (i.e., the act of matching attitudes, beliefs, and behaviors to group norms. These factors exhibit different types of influences exerted on a user from different sources: his friends, his interests, and the information content.

For this component, we propose a scheme to quantify the three types of influences and adopt the learned influences to model and predict users' diffusion actions. In the off-line part, each user's interest profile is firstly learnt from the past microblogging history. Next, we propose a diffusion-targeted influence model to quantify various influences a user receives. A factor graph model is elaborated to categorize and analyse the three types of influences introduced above. Finally, given the history diffusion action set, we learn how the various influences could affect a user's decision of whether to perform a particular diffusion action. The learned weighting configuration for various influences will

eventually contribute to the online prediction of future diffusion status with regard to a new microblog.

1.3.2 Microblog Tracking

As discussed previously, many viral microblogs are "single microblogs" and do not contribute to viral topics. For example, due to the effect of idolatry, many microblogs posted by celebrities will receive a large amount of reposts, thus very likely to be predicted as potential viral microblogs. However, many of them are pointless, or cannot be viewed as topics. In order to obtain the desired topics, we propose to monitor the forthcoming microblog stream to extract topic-related posts. Based on this motivation, we propose a method to track related microblogs in the forthcoming data stream for given microblog targets. The microblog tracking problem is modeled as an evolutional dictionary learning method that jointly track all microblog targets in a unified framework. With this method, we expect to learn a dictionary of atoms as a compact summary representation for all the tracking targets, such that the microblogs can be approximately represented by a linear combination of a few atoms. Considering the evolvement of the tracking contents, a dictionary-transition matrix is introduced to capture the relationship between the dictionaries of two adjacent time slot. By doing this, both the information about the current batch of data and the accumulated history are taken into consideration.

1.3.3 Multimedia Topic Summarization

It is non-trivial to integrate textual and visual information to generate comprehensive summaries due to the following critical challenges: (1) For microblog,

the inconsistency of textual part and visual part in the same microblog is very common, we are faced with irrelevant microblog texts and/or images in the input data stream. (2) Intrinsic correlations between textual and visual information are not well explored, which may exert negative influence on each other.

In order to address the above challenges, we propose a multimedia viral topic summarization framework to generate a holistic visual summary from the microblog posts with multiple media types. Specifically, the proposed framework comprises three stages: removal of irrelevant data, cross-media subtopic discovery and multimedia summary generation. First, we devise a data cleansing approach to automatically eliminate those irrelevant/noisy images. An effective spectral filtering model is exploited to estimate the probability that an image is relevant to a given topic. In the second stage, we propose a novel cross-media probabilistic model, termed *Cross-Media-LDA* (CMLDA), to jointly exploit the microblogs of multiple media types for discovering subtopics. Finally, based on the cross-media distribution knowledge of all the discovered subtopics, we generate a holistic visualized summary for the topics by pinpointing both the representative textual and visual samples in a joint fashion.

1.4 Research Contributions

The main contributions of this thesis can be summarized as follows:

- **A viral topic monitoring framework**. We developed a viral topic monitoring framework to address the problem of "what to monitor" in microblog streams. Specifically, our framework targets at general viral topics, which are not restricted to preselected organizations, or types of

events. In addition, instead of detecting viral topics after the topic has already accumulated a large number of posts, we aim at monitoring the viral topics from the early stage. Our framework provides a solution to the problem of viral topic prediction, which has not been addressed by previous works.

- **A viral topic prediction mechanism**. We proposed a two-phase approach for predicting the near-future viral topics. Specifically, we address the following two sub-problem: predicting viral microblogs from microblog streams, followed by the tracking of microblogs for early-stage topic detection.

- **A multimedia topic summarization approach**. We proposed a multimedia summarization method to summarize the collection of microblog posts related to a viral topic, which features the exploration of the intrinsic correlations among different media types for enhancing the summarization performance. With this method, a holistic visualized description can be generated, presenting to the users a comprehensive overview of the viral topic.

1.5 Organization

The following of the thesis is organized as follows. Chapter 2 provides a brief literature review of the broad domain of topic detection and automatic summarization. In Chapter 3, we present the viral microblog prediction component. Through the view of information diffusion, we study the users' action of propagating microblog messages, and learn the social influence which affects the

propagation actions to predict users' future action. Later in Chapter 4, the method for microblog tracking is introduced in detail. In Chapter 5, we depict the multimedia topic summarization component by giving a detailed introduction of our multimedia viral topic summarization method. Finally, Chapter 6 concludes the thesis and points the future potential research directions.

CHAPTER 2

Literature Review

In this chapter, we offer a comprehensive literature review in the domain of topic detection and automatic summarization. For topic detection, we first briefly introduce the methods and systems for traditional media, then present their recent developments when applying to the microblog domain. Later, researches related to automatic summarization will be introduced. Specifically, we will start from the original methods designed for traditional media, *i.e.*, text collections, then move to the summarization for images, and finally review a few recent works for microblog summarization.

2.1 Topic Detection

2.1.1 Topic Detection in Traditional Media

In this section, the techniques for topic detection in traditional media are introduced. According to the features, these methods can be classified into document-pivot techniques and feature-pivot techniques. The former methods detect topics by grouping similar documents into clusters based on their similarity, and the latter methods focus on detecting bursty keywords as the sign of emerging topics. Next, these two type of techniques are introduced in detail.

Document-Pivot Techniques

The history of topic detection can be traced back to the Topic Detection and Tracking (TDT) program [6]. The target of this program is to provide technology for news monitoring tools to keep users updated about the news and new developments with information from multiple sources of traditional news media. Originally, three main tasks are considered in TDT: information segmentation, topic detection, and topic tracking. These tasks attempt to segment text into cohesive stories first, detect unforeseen topics from these texts, and also keep track of the development of a previously topic.

According to TDT, the objective of topic detection is to discover new or previously unidentified topics, which has been broadly divided into two categories: retrospective topic detection, which focuses on discovering previously unidentified topics from accumulated historical collections, and new topic detection, which involves the discovery of new topics from live streams in real time. Clustering-based algorithms have been mainly employed for both tasks.

Retrospective topic detection (RTD) involves iterative clustering algorithms

that require the entire document collection, to organize the documents into topic clusters. Hierarchical clustering methods have been widely used for this task, *e.g.*, the bottom-up hierarchical agglomerative clustering. This method starts by representing each single data point as an individual cluster. Next, based on the similarity of clusters, the closest clusters are merged until there is only one single cluster, or the structure satisfies certain termination criteria. There are some variations of the hierarchical clustering algorithms, and they have also been employed in TDT tasks [138, 16].

New topic detection (NTD) performs like a query-free retrieval task. Since there is no prior known information about the topic, it cannot be characterized into an explicit query. Different from RTD, in NTD, decisions about whether a document is related to an old or new topic must be provided when new document arrives. Therefore, the clustering process are usually based on some incremental methods, which can sequentially process the input documents. A topic can be merged with the most similar topic, or it is viewed as a new topic cluster if the similarity between this topic and the most similar one exceeds certain threshold. This type of approach usually has high requirements for the computing resources in a short time, therefore it may be unfeasible. However, some methods have been proposed to address the efficiency problem [80]. For example, a sliding time window can be introduced to limit the number of past documents to be compared, thus alleviating the resource requirements [138]. Such methods rely on the assumption that documents related to the same topic should usually be close to each other in the time domain. There are also other techniques proposed to improve the efficiency of the topic detection system, such as limiting the number of terms in each story, keeping limited number of terms, and processing the documents in parallel [80].

In the above document-pivot techniques, the textual similarity is adopted for grouping document clusters. The research for traditional textual documents takes the assumption that all the documents are relevant, while some of them are closer to each other and can be grouped into topics of interests. However, for microblog posts, this assumption is clearly violated. Microblog posts contain many noisy contents, and only a very small portion are related to topics [19]. Besides, because of the huge data volume, the traditional techniques cannot handle the scale and efficiency requirements of microblogs.

Feature-Pivot Techniques

Different from document-pivot techniques, feature-pivot techniques attempt to discover the bursty activity (a sharp increase in certain features) in text streams, and view these bursty activities as the sign of an emerging topic [130, 56, 49]. Therefore, a topic is represented by the bursty keywords, *i.e.*, the words with a sudden increasing count. The underlying assumption is that when a new topic occurs, the words related to this topic would be used frequently. Feature-pivot techniques usually analyze the distribution of certain features, and the topics are discovered with groups of bursty features. In a seminal work [66], an infinite-state automaton was proposed to model the document arrival time to discover the high intensity bursts. The frequencies of individual words are saved in the states, and the transition in different stages occurs when the word frequency changes significantly, which is shown as a signal of bursts. In [40], Fung *et al.* adopted a binomial distribution to model the appearances of words. The bursty words are identified by a pre-defined threshold, and the bursty topics are discovered by grouping of bursty features. Spectral analysis was applied by He *et al.*[56], where discrete Fourier transformation was introduced to categorize

the different characteristics of topic features. Another important work is [116], where an online method was proposed to detect in news streams. For the frequency of each n-gram word, this method tests its statistical significance. In order to improve the efficiency for online detection, an incremental suffix tree was designed to decrease the requirements for time and space constraints.

Although the feature-pivot techniques are very successful for topic detection in traditional documents, it is not suitable to apply to microblog streams. The reason is that for microblog streams, the information is quite diversified, therefore, the occurrence of a burst may be the result of noisy data, while not relevant to a emerging topic.

2.1.2 Topic Detection in Microblog

The previous review of topic detection in traditional media provides a brief introduction to the developments of techniques and analyzes the reasons why they cannot be directly applied to microblog streams. In this section, the techniques proposed for topic detection in microblog streams are introduced. These techniques can be classified into unspecific and specific topic detection depending on the type of topics, and retrospective and new topic detection according to the detection task and target application. Next, these methods are introduced in detail.

Unspecified versus Specified Topic

According to the available information related to the topic of interest, we can classify topic detection into specified and unspecified techniques. Since there is no available information about the topic, the techniques for unspecified topic

detection usually detects the occurrence of a new topic based on the temporal signal in microblog streams. Typically, in these methods, the bursts are firstly discovered from microblog streams, and the bursts with similar trends are grouped into topics, which are classified into different topic categories. Specific topic detection techniques, on the other hand, usually depend on certain known information about the topic (*e.g.*, time, venue, name). Therefore, given these known information, traditional techniques of information retrieval and extraction can be applied to facilitate the detection process.

The nature of microblog posts is to reflect topics. Therefore, microblogs can be very useful for unspecific topic detection. Unspecific topics usually include emerging topics, breaking news, and general topics that attract the attention of a large number of microblog users. Unspecific topics are typically detected by exploiting the temporal patterns or signal of microblog streams. New topics of general interest usually present a burst of features in microblog streams, *e.g.*, a sudden increase of the number of some specific keywords. Posts with these features can then be grouped into trends [82]. However, microblog not only contains trending topics, but also includes abundant non-topic trends. Therefore, there is a need to discriminate trending topics from the non-topic trends. In addition, since the volume of microblog post is very huge, scalability and efficiency should also be considered.

Several works have been done to address the above scalability problem. In [109], the writers adopt a naïve Bayes classifier to filter the intended news from all the microblog posts, thus decreasing the number of documents to be processed. In another work, Petrovic *et al.* [95] proposed to improve the efficiency of the online NED approach, which was originally proposed for traditional news media. By adopting an adapted variant of the LSH method, they were able

to limit the documents to a small number, thus improving the scalable of the NED method to make it suitable for microblogs.

Other works try to leverage the microblog-specific features to improve the detection performance. For example, Phuvipadawat and Murata [96] adopted the clustering method for topic detection. Instead of using the entire microblog content, they only used proper noun terms, user names and hashtags to compute the similarity of microblog posts. In addition to these shallows features, Long *et al.* [79] proposed a new feature, "topical words", defined as words that are more popular than others with regarding to a topic. Topical words are extracted considering their frequency, their occurrence in hashtag, and their information entropy. With the facilitation of topic words, a maximum-weighted bipartite graph matching is employed to create topic chains, to detect the existence of new topics. In [101], although simple features are adopted, a new term weighting scheme is proposed, in which the sparse aspect, global weight and local weight are jointly modeled. From the aspect of implementation, this method is scalable and computationally preferable.

Recently, another research direction focuses on adopting dictionary learning and matrix factorization techniques for new topic detection. In [64], a dictionary learning based method was proposed. The overall framework is divided into two stages: determining novel documents from the stream first, then identifying cluster structure among the novel documents subsequently. While in a related work [106], an online non-negative matrix factorization framework with a temporal regularization was adopted. The temporal regularization is formulated by chaining together trend extraction with a margin-based loss function to penalize static or decaying topics.

Although various microblog-specific features have been adopted for un-

specific topic detection, these features only reflect the general characters of microblog posts. However, if the topics to be detected are specific topics, then we can get additional content information. Based on this idea, several works have been done for various applications. For example, Popescu and Pennacchiotti [98] proposed a system to identify controversial topics in Twitter, *i.e.*, the discussions containing opposing opinions. Lee and Sumiya [73] presented a geo-social local topic detection system based on a model to identify festival activities, by monitoring the crowding behaviors. Sakaki *et al.* [107] proposed a system to detect earthquakes and typhoons based monitoring the contents of microblog posts. In another work, Becker *et al.* [13] presented a general system for planned topics, where the simple rules and query expansion techniques were elaborated.

In conclusion, existing works for unspecific topic detection generally apply filtering components or adapt general features related to temporal information to address the scalability problem, and specific topic detection uses similar detection methods, where additional topic-specific information is adopted to collect and filter microblog posts.

New versus Retrospective Topic

Similar to topic detection from traditional media, we can also classify topics in microblog streams into new and retrospective considering the requirements of tasks and applications. In order to discover new topics in real time, techniques for the detection of new topics usually monitor the signals in microblog streams continuously. Therefore, these techniques are naturally suitable for the detection of unknown topics and breaking news occurring in real world. Generally, bursty topics in microblog steams usually reflect breaking news in the real

world. However, due to the social property of microblog networks, many non-topic trends which are unrelated to real-world events also arise in microblog streams. For instance, when NASA's Curiosity Rover landed on Mars, Bobak Ferdowsis hairstyle became viral in twitter. Although in the process of new topic detection, usually no assumption is imposed to the topic, this type of methods is not restricted to the detection of unspecified topics. On the other hand, new topic detection could also be applied to specific topics given related information or description. These information can be integrated into the new topic detection system to help the system better focus on the interested topics by performing filtering techniques on the data streams[107]. Some other examples of specific information include the controversy restriction [98] or geo-tagged information [73].

While most of the works focused on new topic detection, some recent research paid attention to the detection of retrospective topics from historical microblog posts. The ability of existing search engines for microblog data, such as the applications offered by Twitter and Google, is restricted to retrieving individual microblog posts given a query [84]. In order to find microblog messages related to a given query, the biggest challenges are the sparseness of the posts and the mismatch in the dynamically evolving vocabulary. For instance, the relevant microblog posts may not contain the query keywords, or the topic may be represented with different abbreviations and hashtags. Traditional query expansion techniques usually rely on the co-occurrence of words with query keywords. On the other hand, temporal and dynamic query expansion techniques should be exploited for the retrieval of topics from microblog data.

Our work focuses on unspecific new topic, and the existing techniques

which are most relevant to our work are the dictionary learning based methods. Therefore, in our later experimental evaluation (Chapter 4.5), only selected unspecific and new topic detection methods [82, 101, 64, 106] will be chosen as comparative baselines, while other specific topic detection or retrospective topic detection methods are beyond our scope.

2.2 Automatic Summarization

In this section, automatic summarization techniques are reviewed. Particularly, three areas of summarization will be introduced, including summarization for text, summarization for images, and summarization for microblog.

2.2.1 Text Summarization

The target of text summarization is to produce a concise summary of the most important information, given one or multiple documents as input. Writing a good abstract is often difficult even for people. Under this circumstances, it is very challenging to design a considerably good summarization system. In most cases, the current state-of-the-art methods rely on extraction of sentences from the original documents. Such extractive techniques allow the systems to focus on the approaches to choose sentences containing meaning contents, without the bother of text-to-text generation techniques. The extractive approach focuses on one important research question: how to determine which sentences are important for generating a summary.

General Framework

Generally, all extractive summarization systems include the following three components [89]: (1) representing the original input with intermediate information to capture the key contents of the documents, (2) assigning importance score to the sentences, and (3) selecting sentences based on their importance score to generate a summary.

1. Intermediate representation. The representation should possess the ability to reveal the important content. Two kinds of representation approaches have been proposed: topic representation approaches and indicator representation approach.

Topic representation approaches convert the text in the documents into an intermediate representation. Examples of this approach include:

- Frequency, tf-idf and topic word based approaches, where the original documents are represented according to the vocabulary.
- Lexical chain approaches, where the documents are represented by semantically related words discovered with the assistance of a thesaurus (*e.g.*, WordNet).
- Latent semantic analysis, where the documents are represented by the latent semantic factors.
- Bayesian topic models, there the documents are represented by a mixture of topics.

Indicator representation approaches adopt various importance indicators to represent sentences in the original documents, *e.g.*, the length of sentence, location, or the occurrence of some phrases.

2. Score sentences. Given the intermediate representation, the next step is to assign a score to indicate the importance of each sentence. In topic representation approaches, the score usually reflects how well a sentence can expresses the information of the document. While in indicator representation methods, the importance score of s sentence is commonly determined by the combination of evidence from various indicators, and machine learning techniques are commonly used to discover the weight of different indicators.

3. Select summary sentences. In the last step, the summarization system needs to determine the best choices of sentences to include in the final summary. The number of sentences can be pre-defined, where the system terminates after selecting the top n sentences that is most important. Besides, the *maximal marginal relevance approaches* [18] are also extensively used, where the system selects sentences iteratively, and the importance score of a sentence automatically changes considering its similarity to sentences that are already chosen. Other methods include *global selection approaches*, where the system focuses on selecting an optimal collection of sentences subject to certain constraints, *e.g.*, the maximization of overall importance, the minimization of redundancy, or the maximization of coherence.

Several other factors can also affect the ranking of the sentences, such as the context information related the user needs [7, 35], or the genre of a document [10, 88, 100].

As the three steps are relatively independent from each other, a summarization system can incorporate any combination of specific methods for each step. Next, these techniques are introduced in detail.

Topic Representation Approaches

1. Frequency-based approach. SumBasic [90] is one of the most famous systems which adopt frequency to select sentences. For each sentence in the input document, a weight is assigned as the average probability of all the words in the sentence. Then, this system selects sentences which contain the word with the current hight score. This selection strategy assumes that when selecting a sentence, a single word with the highest score can represent the most important content of the original document.

In addition to word probability, tf-idf has also been an important weighting method [108], which considers both the term frequency and the number of documents that contain this word. According to the definition of tf-idf, a topic word is descriptive if it appears frequently in a document, but rarely in other documents. Many systems incorporate tf-idf [37, 38, 42], such as the popular baseline centroid-based summarization system [99].

2. Latent Semantic Analysis. Latent semantic analysis (LSA) [34] is an unsupervised method which is designed to derive the implicit representation of the semantics of texts. LSA performs SVD on documents to discover the latent topics, and the documents are presented by the latent topics, where topics with low weight ignored as noise. LSA was originally adopted to the summarization task by Gong and Liu [48]. In this work, dimensionality reduction is performed such that the number of topics is the same as the number of expected sentences in the summary. For each topic, the sentence with the highest weight is included in the summary. This strategy treats each topic equally important. As a result, a summary may include sentences related to unimportant topics. In order to address this drawback, various extensions have been proposed. For example,

using the weight of each topic in determining the proportion of topics to include in the summary [117], or selecting sentences which include several important topics[118].

3. Bayesian Topic Models. As one of the most sophisticated approaches for topic representation, Bayesian topic model has gained increasing attention for summarization tasks [33, 55, 128, 20]. The topic model representations are very appealing, since they are able to capture the important information which are neglected in most other techniques. With the detailed topic representation, better summarization system can be developed to convey the similarities and differences among the different input documents. In these methods, sentences are scored and selected iteratively in a greedy process [55]. Each step selects the best sentence which minimize the KL divergence between the original text and the summary after this sentence is included.

4. Sentence Clustering and Domain-dependent Topics. The input to multi-document summarization consists of several articles, and different articles may contain sentences with similar information. This method treat clusters of similar sentences as topics, and sentence clustering approaches are exploited to discover repetition of sentences in different articles. A cluster is considered more important if there are more sentences in it. When producing the summary, one representative sentence is selected from each main cluster to minimize the possible redundancy in the summary.

For domain-specific articles, sentence clusters can be a good indicator about whether the topics are frequently discussed in the domain. In such a case, a Hidden Markov Model (HMM) can be trained [12, 41] to capture the "story flow", *i.e.*, the order of topics discussed. HMM models utilize the fact that information in various articles within a specific domain is usually presented

following a common information flow. For instance, news articles reporting earthquakes commonly introduce the location of the earthquake first, followed by the damage and rescue efforts. Such structure of "story flow" can be learned from multiple documents in the same domain.

Indicator Representation and Machine Learning for Summarization

Instead of interpreting the topics in the input, indicator representation approaches represent the text by indicators which can be directly used to rank the importance of sentences. Indicator representation approaches include graph-based methods and machine learning based methods.

1. Graph-based Methods. Graph-based algorithms have been shown to be effective. Basically, the nodes of the graph represent text elements (*i.e.* normally words or sentences),whereas edges are links between those text elements. Edges are assigned weights according to the similarity between the two elements, *e.g.*, cosine similarity or binary relationship. On the basis of the text representation as a graph, the idea is that the topology of the graph reveals interesting things about the salient elements of the text. Sentences that are related to many other sentences are likely to be central and would have high weight for selection in the summary.

One of the famous graph-based multi-document summarization system is LexRank [37]. In this graph, two sentences are connected if their similarity exceeds a predefined threshold. Random walk is performed on the graph to discover the central sentences. [85] analyzed several graph-based algorithms and evaluated their ability for summarization. Furthermore, in [126], an approach based on affinity graphs is suggested for both generic and query-focused multi-document summarization. Besides the above methods, a graph can be also

built using concepts identified with Wordnet [86].

2. Machine Learning-based Methods. Edmundson proposed the earliest work [36] which adopted machine learning techniques for summarization [69]. Rather than relying on a single representation, many different indicators can be combined. Machine learning techniques are then exploited to learn the weight of each indicator. Machine learning approaches offer great freedom to summarization, since the number of indicators is practically endless [92, 144, 39]. Some common features include the sentence position (first sentences of news are usually very informative), position in the paragraph (first and last sentences are usually important), the length of sentence, similarity to the document title, weights of the words in a sentence, named entities or key phrases, *etc.*

However, the supervised learning paradigm has an inherent problem that it relies on labeled data for training. A reasonable solution is to ask annotators to select sentences for summary [102], but it is time consuming and low in annotator agreements. Instead, many researchers worked with abstracts written by people, and concentrated on developing methods to automatic align human abstracts to the input to provide labeled summary data for machine learning [81, 62, 11].

Selecting Summary Sentences

In most summarization approaches, the sentences are chosen sentence by sentence. In order to avoid the inclusion of repetitive sentences, the checking for similarity between the chosen sentences is also employed during the selection process.

1. Greedy Approaches: Maximal Marginal Relevance. As a greedy

approach, Maximal Marginal Relevance (MMR) is widely adopted [18]. This approach creates summaries sentence-by-sentence. At each step, a greedy algorithm is constrained to select the sentence that is maximally relevant to the user query (or has highest importance score when a query is not available) and minimally redundant with sentences already included in the summary. MMR measures relevance and novelty separately and then a linear combination of the two is used to produce a single score for the importance of a sentence.

2. Global Summary Selection. Global optimization algorithms can be used to solve the new formulation of the summarization task, in which the best overall summary is selected. Given some constraints imposed on the summary, such as the maximization of informativeness, the minimization of repetition, and the consistency to the required summary length, the task would be to select the best summary. Dynamic programming algorithm can be applied to find approximate solution to this problem [83, 142, 143]. Global optimization approaches have been shown to outperform greedy selection algorithms in several evaluations using news data as input, and have been proved to be especially effective for extractive summarization of meetings [103, 46].

2.2.2 Image Summarization

An image summary is a set of photos that represent the most interesting visual content of an image collection. The purpose of an image summary is to quickly give a viewer an accurate impression of what visual aspects are captured inside an image collection. Generally, there are two different tasks for generating image summaries: (1) representative image selection, which selects a set of original images from the image collection without modification of the visual

content, and (2) visual synthesis, which generates new images that contains contents taken from the original collection.

The second kind of approach requires techniques for salient region detection, image segmentation, visual layout, *etc.* For example, in [105], Rother *et al.* summarized a set of images with a "digital tapestry", *i.e.*, a single image which act as a 'thumbnail' of the image collection. They synthesized a large output image from a set of input images, stitching together salient and spatially compatible blocks from the input image set. In a different work, Wang *et al.* [129] created a "picture collage": a 2D spatial arrangement of the all the images in the input set to maximize the visibility of salient regions. In both works, the set of images to appear have already been chosen, and the visual layout is to be determined, which is irrelevant to our study. In this section, we ignore issues of layout and focus on selecting the set of images to appear in the summary.

Summarization for Web Image Search Results

Existing web image search engines return a large quantity of image search results ranked by their relevance to the given query. Web users have to go through the list and look for the desired ones. This is a time consuming task since the returned results always contain multiple topics and these topics are mixed together. Things become even worse when one topic is overwhelming but it is not what the user desires.

A possible solution to this problem is to cluster the image search results into different groups with different topics. In traditional Content-Based Image Retrieval (CBIR), image clustering techniques are often used to provide a convenient user interface [104], which helps to make more meaningful representations of search results [27]. In [77], top result images were clustered based

on visual features so that the images in the same cluster are visually similar. Considering that global image features do not describe individual objects in the images precisely, [131] proposed to use region level image analysis. The problem is formalized as a salient image region pattern extraction problem. According to the region patterns, images are assigned to different clusters. In [125], the weight of visual features are dynamically appropriated according to the query to best capture the discriminative aspects of the resulting set of images that is retrieved. These weights are used in a dynamic ranking function that is deployed in a lightweight clustering technique to obtain a diverse ranking based on cluster representatives. Three clustering algorithms that target different objective are proposed, *i.e.*, folding, max-min and reciprocal election.

Besides visual information, textual and link information have also been used recently. In [132] and [43], a reinforcement clustering algorithm and a bipartite graph co-partitioning algorithm were proposed to integrate visual and textual features respectively. [17] proposed a hierarchical clustering framework. By considering the image and the block containing that image as a whole, this framework exploits three kinds of information (visual information, textual information and link information) to hierarchically cluster the image search results based on these image representations. It first adopts block-level link analysis to construct an image graph. Then, spectral clustering techniques are adopted to hierarchically cluster the top image search results based on visual representation, textual representation, and graph representation.

The effectiveness of these clustering approaches depends heavily on clustering performance and the quality of representative images of each cluster. For an online process, clustering of hundreds of images using high dimensional features is not efficient enough to be practical. Considering these drawbacks,

[61] proposed an efficient and effective algorithm to organize Web image search results into semantic clusters, which firstly identifies several semantic clusters related to the query and then assigns all the resulting images to corresponding clusters.

All above methods are cluster-based approach. In an alternative view, [136] treats automatic image summarization as the problem of dictionary learning for sparse coding, *e.g.*, the summary of a given image set can be treated as a sparse representation of the given image set (*i.e.*, sparse dictionary for the given image set). For a given semantic category, they build a sparsity model to reconstruct all its relevant images by using a subset of most representative images (*i.e.*, image summary), and a stepwise basis selection algorithm is developed to learn such sparse dictionary by minimizing an explicit optimization function.

Summarization using Community-contributed Knowledge

Community-contributed knowledge and resources are becoming commonplace, and represent a significant portion of the available and viewed content on the web. In these community datasets, landmarks and geographic elements enjoy a significant contribution volume. The abundant additional information (*e.g.*, tag data, location and temporal metadata) can greatly facilitate the process of visual summarization. Since 2003, a number of research efforts have considered geographic location information associated with photographs.

Some works take advantage of user generated tags. In [115], an unsupervised learning approach is proposed to summarize a visual scene by finding a set of canonical views. Given a set of images for a given scene, canonical views are generated by clustering images based on their visual properties. The clustering is performed using a greedy method that outperforms k-means for

this application. Once clusters are computed, an image browser is generated to explore scenes hierarchically. The researchers extract representative tags for each cluster given the photographs' tags on Flickr. Due to the large amount of noise in user tags, a function score is defined to measure the description ability of tags. [29] constructed a hierarchy of images using textual caption data. Co-occurrence between terms associated with image captions and a statistical relation called subsumption are used to generate term clusters which are organized hierarchically. [110] used a similar approach and built a subsumption based model on a Flickr tag set, demonstrating the potential to induce an ontology suitable for a browsing user interface.

Aside from user generated tags, many photographs are also connected to geo-referenced metadata describing the geographic location in which they were taken [87, 123]. Comparing with the noisy tags, the metadata of geographic location is much more accurate and precise, which makes it a quite valuable supplement for the summarization of landmark photographs. In [123], the authors described WWMX, a map-based system for browsing a global collection of geo-referenced photos. The WWMX system tries to handle clutter by consolidating multiple photograph markers into a single marker according to the zoom level. Several projects [87, 97] use geographic data to organize photo collections in novel ways, for example, by detecting significant topics and locations in a photo collection.

In the absence of location metadata, temporal metadata is also considered for the purpose of photo collection summarization. In [52], Graham *et al.* described an algorithm to heuristically select representative photos for a given time period in a personal collection, utilizing patterns in human photo-taking habits. Additional time-based work aimed to detect topics in personal

collections [30], which could be the basis for collection summarization.

[59] is a comprehensive work which considers a multitude of spatial, social and temporal metadata (such as where the photo was taken, by whom, at what time), as well as textual-topical patterns in the data (such as textual tags associated with the photo). This algorithm utilizes metadata-based heuristics that capitalize on patterns in users photographic behavior. Foremost among these heuristics is the premise that photographs taken at a location 'vote' for the presence of something interesting at that location. A modified version of this summarization algorithm serves as a basis for a new map-based visualization of large collections of geo-referenced photos, which visualize the data by placing highly representative textual tags on relevant map locations in the viewed region, effectively providing a sense of the important concepts embodied in the collection.

2.2.3 Microblog Summarization

Under the context of microblogging, new summarization tasks appears.

Summarization for Trending Topics

A huge number of topics and vast volume of microblogs are posted every day. To help people who read posts, many microblogging platforms provide the following two functions: searching for posts that contain a topic phrase and browsing a short list of popular trending topics. A user can perform a search for a topic and retrieve a list of the most recent posts that contain the topic phrase. The difficulty in interpreting the results is that the returned posts are only sorted by recency, not relevancy. Therefore, the user is forced to manually read through

the posts in order to understand what users are primarily saying about a particular topic. Facing this problem, following approaches have been proposed to generate summaries of microblog posts relating to the same trending topic.

In [111], Sharifi *et al.* developed an algorithm to summarize microblogs related to a user specified phrase. Taking a trending phrase as input, their model first collects a large number of posts containing the phrase. Summarization is then conducted on this collected microblog set. During the generation process, the Phrase Reinforcement algorithm is proposed, with the central idea that the summary should be composed of the most commonly used phrase that encompasses the topic phrase. After filtering the input sentences, the algorithm builds a graph which represent the common sequences of words that occur both before and after the topic phrase. With the built graph, the path with the most total weight is selected as the summary. Targeting the same problem, a hybrid TF-IDF summarization method is proposed in [112], where the TF-IDF is refined for the special case of microblogging.

The above methods only generate a single sentence as the summary. Since the posts related to a specified topic likely represent several subtopics or themes, it may be more appropriate to produce summaries that encompass the multiple themes rather than just having one post describe the whole topic. For this reason, Inouye *et al.* extended the work significantly to create summaries that contain multiple posts [58]. As an extension, [58] modified the methods in [111] and [112], making them suitable for multiple post selection. Besides, traditional multi-documents methods, *e.g.*, SumBasic, MEAD, LexRank and TextRank, are adopted to test their ability beyond document summarization.

A similar problem was proposed in [23] for context summarization. Given a single microblog as root microblog, they summarize the context information

(*i.e.*, microblogs with a reply relationship to the root) to help the users better understand the context of this microblog. Besides the textual content of microblogs, this methods also take the social relationship into account. Two kinds of user influence models are proposed, the pairwise user influence model and global user influence model. With the text based signals and user influence signals represented as features, a supervised learning framework is adopted which uses the Gradient Boosted Decision Tree algorithm to lean a non-linear model.

Storyline Summarization

Most of the time, a flat summarization is enough to provide the users sufficient information about the summarized topic. On the other hand, a storyline summarization is more necessary in some cases, *e.g.*, sport games [91, 22] or long-running news [75]. The work of [22] focused on repeated topics, such as sports, where different games share a similar underlying structure though each individual game is unique. By learning from previous games, this method is able to better summarize a recent or ongoing game topic. A two-step process is proposed: a modified Hidden Markov Model is firstly used to segment the topic storyline, depending on both the burstiness of the microblog stream and the word distribution used in microblog; later, several key microblogs are picked to describe each segment judged to be interesting enough to build the summary.

While the above method can only work on highly-structured topics, Lin *et al.* proposed a more general framework to automatically generate a skeleton of long-running topics [75]. A two-level framework is designed: firstly, a dynamic pseudo relevance feedback language model is designed to retrieval relevant microblogs to a given topic query; later, the problem of storyline generation on the retrieved microblogs is formulated as a graph-based optimization problem

and is solved by approximation algorithms of minimum-weight dominating set and directed Steiner tree. The finally generated storyline ensure both temporal continuity and content coherence.

Hybrid Summarization

Microblogging is regarded as a faster, first-hand source of information generated by massive users. The content diffused through this channel, although noisy, provides important complementary and sometimes even substitute to traditional news media reporting. The competing-complementary role between social media and traditional media become more and more evident recently in different scenarios. [44] summarized the given subject matter by jointly extracting important and complementary pieces of information across news media and Microblog. In this work, the authors propose a balanced complementary measure for the sentence-microblog pair by leveraging topic modeling approach based on a variant of cross-collection LDA (ccLDA). The summaries are generated by co-ranking the complementary sentences and microblogs at either side using random walk on a bipartite graph which reinforces the strength of connection between the pair.

In [141], the main task was to summarize the Web documents, and microblogging serves as a complementary information source by providing social context. They answered the following question: how to generate a summary for online documents by considering both the informativeness of sentences and interests of social users? This work considers the social influence and the information propagation for document summarization. This problem is formulated into a dual-wing graph model, simultaneously incorporates all resources in social context to generate high-quality summaries for online documents.

2.3 Summary

The reviewed literatures in the previous sections demonstrate that both topic detection and automatic summarization have achieved great success. The former mainly addresses the problem of detecting viral topics from traditional text streams and microblog streams. All these methods, either document-pivot techniques which depends on the clustering of similar texts, or feature-pivot techniques which rely on bursty signals, require the accumulation of a certain amount of texts. With these methods, an outbreaking topic cannot be detected until a large number of related microblog posts have engaged. This restricts their ability in response time. Besides, the previous topic detection methods treat microblog streams as special large-scale text streams, while neglecting one important character of microblog: the social feature. As a new type of social network, microblog not only performs as a channel for information diffusion, but also holds valuable social information. Since the posts are propagated through the social network, in addition to the contents of microblog posts, the network structure and the relationship between users should also be taken into consideration. This will help us to fully understand the performance of a viral topic. Facing the above two limitations, we propose a novel problem of detecting viral topics in the early stage, termed as viral topic prediction, and present method that brings the social factors into consideration. This work will be elaborated in Chapter 3 and Chapter 4.

The latter part of this chapter focuses on the problem of automatic summarization. As shown in the review, summarization for traditional text collection has achieved great success. There are also abundant work on image summarization and microblog summarization. Due to the application scenario, these

methods all focused on generating summaries for single media type, either text or image. The previous summarization technologies designed for microblog collections only take textual contents into consideration. However, with the development of microblog services, the proportion and importance of multimedia information are increasing, resulting in the necessity for cross-media summarization technology. To address this problem, we propose a framework targeting summarization for viral topic in microblog, where multimedia microblog information is taken into account. This work will be introduced in detail later in Chapter 5.

CHAPTER 3

Viral Microblog Prediction

3.1 Introduction

In recent years, information diffusion has drawn considerable research interest in computer science, and a variety of techniques and models have been developed to capture the information diffusion in online social networks [54]. Some researchers focus on building standard models to explain the general information diffusion process, such as the two seminal models, namely Independent Cascades (IC) model [47] and Linear Threshold (LT) model [65]. These models are useful for simulating the information flow in social networks. However, they cannot be directly applied to predict the diffusion process. Another research direction lies in detecting the outbreak of information cascades [74], which focuses on the cascades that have already broken out. In this chapter, we target a different problem: given a new microblog post, we intend to predict whether

it will become viral in the near future, and we also try to predict which users will participate in the future diffusion process of this microblog.

As introduced previously, a user's action of propagating a microblog from a friend may be affected by three types of factors, which exhibit different types of influences exerted on a user from different sources: his friends, his interests, and the information content. It is difficult to quantify these influences due to several challenges. The first challenge is how to differentiate these influences. In our scenario, only the diffusion action is observable, while the underlying influences that trigger this action are hidden. Therefore, it is impractical to directly infer the degree of different influences based on the performed diffusion actions. Second, in order to obtain the interest-oriented influence, we need to generate the user's interest profile, *i.e.*, what kind of content he is interested in, from his microblogging history. Nowadays, a growing proportion of microblogs contain multimedia information [14, 28], *e.g.*, both texts and images, and images could provide more information than the short texts contained in a microblog. How to discover the interest profile of a user from these multimedia contents remains a problem. A third challenge is in constructing a unified model that can jointly leverage various types of influences to model and predict the information diffusion process.

To solve the proposed problem and tackle the above challenges, we propose a novel scheme to quantify the three types of influences and adopt the learned influences to model and predict users' diffusion actions. Specifically, the proposed framework comprises three essential stages, as shown in Figure 3.1: user interest profile learning, diffusion-targeted influence learning, and microblog diffusion modeling and prediction. First, in order to learn a user's interest profile, we need to map the user's microblogs into the corresponding interest categories.

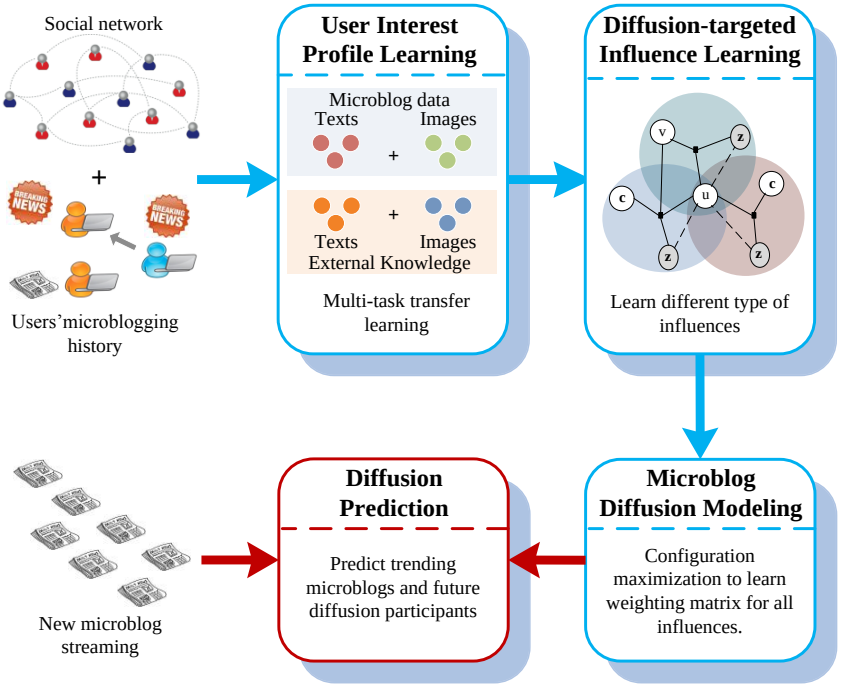


Figure 3.1: Overview of the viral microblog prediction framework.

We devise a classification approach, termed *Multi-Task Transfer Learning*, to jointly classify the multimedia microblogs posted by a user into various interest categories. In order to address the deficiency of labeled training data, we bring in external knowledge where labeled samples are easy to acquire, and the transfer learning technique is adopted to project data samples from different domains into the same embedding space. Meanwhile, a multi-task co-learning process is integrated for the classification task, which will benefit from the joint information shared by different media contents. In the second part, we propose a diffusion-targeted influence model to quantify various influences a user receives. Three types of influences are formally defined, and a factor graph model is elaborated to categorize and analyse these influences. Finally, given the history diffusion action set, we learn how the various influences affect a

user's decision of whether to perform a particular diffusion action. The learned weighting configuration for various influences will eventually contribute to the prediction of future diffusion status with regard to a new microblog.

3.2 Related Work

3.2.1 Influence analysis

Social influence is the behavioral change of a person because of the perceived relationship with other people, organizations and society. It has been a widely accepted phenomenon for decades, and many works have been done to demonstrate the existence of social influence in online social networks [8, 70, 53]. One important research direction is the problem of influence maximization. Given a network with influence estimates, influence maximization tries to select an initial set of users such that they will eventually influence the largest number of users. Kempe *et al.* introduced a fundamental work [65]. Following this, many other methods [25, 51, 24, 135] have been proposed to improve the efficiency. All the related works discussed above assume the influence probability on the edges are given as input, which is impractical for real-world problems. Some works have been done to infer the degree of influence from a given social network [50, 121, 9]. A probabilistic model was proposed in [50] to learn influence probabilities by mining past influence cascades. Tang *et al.* studied the topic-level social influence in [121], and a Topical Affinity Propagation (TAP) method was proposed to model this problem. Other works include the detection of influential users [134], influence measurement in Twitter [21], *etc.*

3.2.2 Outbreak detection

The target of outbreak detection is to select a set of nodes from a social network in order to detect the spread of a virus as fast as possible. Leskovec *dt al.* presented a general methodology for near-optimal sensor placement in [74]. By exploiting submodularity they developed an efficient algorithm much faster than the greedy algorithm. The work in [68] conducted evolutionary analysis in blog networks, and showed that the blogspace had been expanding in metrics of community structure and connectedness. The goal of the above works is to detect existing outbreaks, which is different from our target of predicting the outbreak of a microblog diffusion process before it happens. Recently, Cui *et al.* [32] raised the question of cascading outbreak prediction. Based on the historical cascade data, a data driven approach was proposed to select important nodes as sensors. The prediction is based on the cascading behaviors of these sensors. Although the problem is similar to ours, the above method could only predict whether a cascade will breakout, but could not provide more detailed information about the scale of the cascade or which of the users will participate in the future diffusion process. Besides, the limitation with the small number of sensors results in low recall in prediction performance. The models described in [119] and [120] aim at modeling and predicting users' social actions based on the past action history. However, since a model needs to be trained for each information diffusion process, and the training process requires a considerable number of actions, these models could not be adopted for the diffusion prediction of a relatively new microblog. Unlike these methods, our proposed framework could quantify general influence degree. The prediction model is trained with regards to the behavior of users while is not constrained

to any specific diffusion process. Therefore, our model can handle new incoming microblogs without the need to train a new model every time.

3.3 Problem Definition

We denote the social network as a directed graph $G = (V, E)$, where V is the user set and $E \in V \times V$ represents the social relationships between users. We denote a user u' as a friend of u if there is a edge $(u, u') \in E$, *i.e.*, u is a follower of u'. The basic action of a user is to post microblogs, which can either be original or reposted from friends. A microblog m contains two multimedia components: the textual part m^t and the visual part m^v (either m^t or m^v could be empty). We denote M_u as the set of all microblogs the user u has posted, and the overall microblog set as $M = \cup_{u \in V} M_u$. Next, we present the formal definitions of some terms used in this chapter.

Definition 1. **Interest Profile.** The microblogs could be related to various interest categories. We denote $\mathcal{C} = \{c_1, ..., c_{|\mathcal{C}|}\}$ as the collection of all interest categories. The interest profile of user u, $I(u)$, is a $|\mathcal{C}|$-dimensional vector which represents the user's interest distribution over all interest categories $\mathcal{C}$.

Definition 2. **Diffusion action.** The users in the social network interact with each other through reposting microblogs published by their friends. A diffusion action is defined as a triple $a = (u, u', m)$, representing the user u reposts a microblog with content m from his friend u'. Here, m can either be original microblog sent by u' or a microblog reposted by u' from his friend.

The input to our problem is the social network G, the past microblogging history of all users M, as well as the diffusion action set A which contains the past diffusion actions of all users. For a new incoming microblog m_{new}, we

intend to predict: (1) whether it will become viral in the near future, and (2) which of the users will participate in the diffusion process of this microblog.

3.4 User Interest Profile Learning

Generally, a user will show different level of interest and possess different level of expertise for various interest categories, *e.g.*, sports, music, history, *etc.* Therefore, the degree of influence he exerts to his friends, as well as his interest in propagating further information, will be different for various categories. As a result, learning the interest profile will be crucial for detailed analysis of a user's influence on his friends, his diffusion actions, and the prediction of his future action.

Given a user's historical microblogging data, the target of interest profile learning is to map the microblogs to the interest categories $\mathcal{C}$, which in essence is a microblog classification problem. Under the circumstance of microblogging, we are confronted with the following two problems:

- It is difficult to collect labeled training microblog data, as the data labeling task is tedious and expensive. Besides, as the users tend to talk about the latest trends, the contents of microblogs are highly dynamic and the data vocabulary changes continuously. Consequently, even if we can get some labeled data, they will quickly become out-dated over time.

- Microblog contains multimedia data which contains both text and image. Therefore, instead of traditional short-text classification task, the problem becomes cross-media classification where both textual and visual information should be incorporated to better capture the various

aspects of a microblog.

In order to address the first problem, we propose to include external knowledge into the training process to assist the classification of microblogs into the related interest categories. The well-edited articles from portal website (such as Sina.com[1]) are chosen as the external knowledge with the following reasons: 1) the articles in portal website are well-categorized, which means we can directly get the interest category labels for these articles; 2) the contents of these articles cover nearly all aspects; and 3) these articles contain rich multimedia information, which is appropriate for our cross-media classification problem. We denote the external knowledge as $E = \{(e^t, e^v)\}$, where each data sample includes the textual content e^t and the visual content e^v. Besides, we also have the $|E| \times |\mathcal{C}|$ label matrix Y of the external knowledge, with each element $y_{ij} \in \{-1, 1\}$ indicating whether the i-th data sample belongs to the j-th interest category.

Although the microblog domain and the external knowledge domain are relevant, their data distribution are different, which makes it infeasible to directly use external data as training samples. Domain adaption is a solution to this problem [140]. Domain adaption aims at solving a learning problem in the target domain by utilizing training data in the source domain, allowing data from the both domains to be transferred to the same embedded space. Traditional domain adaption problems usually target at a single media type, while the problem in our scenario contains two modalities. One naive solution is to apply domain adaption techniques on each media type separately, and then train two unrelated classifiers for text and image. However, the contents of the two media types are not isolated and there is interrelationship between

[1]http://www.sina.com.cn/

them. For example, the text and image contained in the same microblog data are usually related to the same topic. By applying the classification separately, these beneficial relationship will be ignored. With the above consideration, we propose the *Multi-Task Transfer Learning (MTTL)* model, which targets at the cross-media, domain-adaptive classification task.

3.4.1 The MTTL Model

Given the unlabeled microblog data M and the labeled external knowledge E, we target at jointly handling both the textual and visual classification tasks in microblogs. In each task, the external knowledge and microblog data need to be transferred to the same embedded space.

We first delineate two desirable properties for the transfer learning task, namely: 1) maximal alignment of distribution between the source and target domain data in the embedded space; and 2) preservation of the local geometry.

1) Objective 1: Distribution Matching. We employ transfer component analysis (TCA) [93] for transfer learning. Specifically, let the kernel matrices defined on the microblog domain, external knowledge domain, and cross-domain data in the embedded space be $K_{M,M}$, $K_{E,E}$ and $K_{M,E}$, respectively, and the kernel matrix defined on all the data be

$$K = \begin{bmatrix} K_{M,M} & K_{M,E} \\ K_{E,M} & K_{E,E} \end{bmatrix} \in \mathbb{R}^{(|M|+|E|)\times(|M|+|E|)} , \tag{3.1}$$

then TCA tackles the domain adaptation problem by minimizing the MMD

distance between the two domains:

$$\min_{Q} tr(Q^T K L K Q) \ , \tag{3.2}$$

where $Q \in \mathbb{R}^{(|M|+|E|)\times d}$ is the embedding matrix; d is the dimensionality of the embedding space; and $L_{ij} = 1/|M|^2$ if both x_i and $x_j \in M$, $L_{ij} = 1/|E|^2$ if both x_i and $x_j \in E$, and $L_{ij} = -1/(|M| \times |E|)$ otherwise.

2) Objective 2: Locality Preserving. We would like to preserve the local structures of both the microblog and external knowledge data, *i.e.*, if two data samples are close to each other in the original domain, this relationship should be preserved in the embedded space [139]. Let $\mathcal{G}$ be the k nearest neighbors graph of the original data with $g_{ij} = exp(-d_{ij}^2/\sigma^2)$ for $x_i, x_j \in M \cup E$ if x_i and x_j are in the same data domain and x_i belongs to the k nearest neighbor set of x_j, or vice versa, and $g_{ij} = 0$ otherwise. Let d_{ij} represents the distance of x_i and x_j, and the graph Laplacian matrix of $\mathcal{G}$ be $\mathcal{L}$. Note that after domain adaption using TCA, the data projection in the embedded space is $Q^T K$, where the i-th column $[Q^T K]_i$ provides the embedding coordinates of x_i. Hence, we minimize the following objective function for locality preserving:

$$\sum_{i,j} g_{ij} \left\| [Q^T K]_i - [Q^T K]_j \right\|^2 = tr(Q^T K \mathcal{L} K Q). \tag{3.3}$$

With the above two objectives, we are able to map the unlabeled microblog and the labeled external knowledge data into the same embedded space. In order to jointly learn both the textual and visual classifiers, we propose to

utilize the following multi-task model to explores the intrinsic correlation:

$$\min_{\{Q_t,W_t,b_t\}_{t=1}^2} \sum_{t=1}^{2} \|K_t Q_t W_t + \mathbf{1} b_t^T - Y_t\|_F^2 + \rho \|W\|_{2,1}$$
$$\text{s.t. } Q_t Q_t^T = I, \ t = 1, 2 \tag{3.4}$$

where $t = 1$ indicates the text classification task and $t = 2$ indicates the image classification task. $\{W_t, b_t\}$ are classification regression parameters. Q_t is the embedding matrix of the t-th task. The cross-media consistency is preserved by the $\ell_{2,1}$ regulation term $\|W\|_{2,1}$, where $W = [W_1, W_2]$ and $\|W\|_{2,1} = \sum_{j=1}^{d} \|w_j\|_2$ with w_j representing the j-th row of W.

Combining the three objectives in Eq.(3.2), (3.3) and (3.4), the final optimization problem for MTTL can be written as:

$$\min_{\{Q_t,W_t,b_t\}_{t=1}^2} \sum_{t=1}^{2} (tr(Q_t^T A_t Q_t) + \mu \|K_t Q_t W_t + \mathbf{1} b_t^T - Y_t\|_F^2) + \rho \|W\|_{2,1}$$
$$\text{s.t. } Q_t Q_t^T = I, \ t = 1, 2 \tag{3.5}$$

where $A_t = K_t L_t K_t + \delta K_t \mathcal{L}_t K_t$, and δ, μ and ρ are the balance parameters.

3.4.2 Optimization

The problem in Eq.(3.5) can be reformulated as

$$\min_{\{Q_t,W_t,b_t\}_{t=1}^2} \sum_{t=1}^{2} (tr(Q_t^T A_t Q_t) + \mu \|K_t Q_t W_t + \mathbf{1} b_t^T - Y_t\|_F^2) + \rho tr(W^T S W) \tag{3.6}$$

where S is a diagonal matrix with $S_{jj} = \frac{1}{2\|w_j\|_2}$. We design the following iteration strategy which includes two steps:

Step 1: Keep Q_t fixed, and update W_t and b_t. By setting the derivative of Eq.(3.6) w.r.t. b_t to zero, we obtain

$$b_t = \frac{1}{n_t}(Y_t - K_t Q_t W_t)^T \mathbf{1}, \tag{3.7}$$

where n_t is the number of training samples in the t-th task. Then substituting the derived b_t into Eq.(3.6) and setting the derivative w.r.t. W_t to zero, we get

$$W_t = (U_t^T U_t + \frac{\gamma}{\mu} S)^{-1} U_t^T V_t \tag{3.8}$$

where $U_t = H_t K_t Q_t$, $V_t = H_t Y_t$, and $H_t = I - \frac{1}{n_t}\mathbf{1}\mathbf{1}^T$ is the centering matrix.

Step 2: We update Q_t by fixing W_t and b_t. With W_t and b_t fixed, the objective function in Eq.(3.6) is reduced to

$$\begin{aligned} &\min_{Q_t} tr(Q_t^T A_t Q_t) + \mu \| K_t Q_t W_t + \mathbf{1} b_t^T - Y_t \|_F^2 \\ &\text{s.t. } Q_t Q_t^T = I, \ t = 1, 2. \end{aligned} \tag{3.9}$$

This optimization problem can be efficiently solved by the algorithm introduced in [133].

The learned transformation matrix and regression parameters could be adopted to classify new microblogs. By denoting k^m as the kernel vector of microblog m, the classification output of m is $l_t(m) = W_t^T Q_t^T k_t^m + b_t, \ t = 1, 2$, then the corresponding interest category of m, $C(m)$, is the category with largest classification output in either textual or visual domain. The interest profile of the user u is then defined as:

$$I(u)_c = \frac{|\{m \in M_u | C(m) = c\}|}{|M_u|}, \ c = 1, 2, ..., |\mathcal{C}|.$$

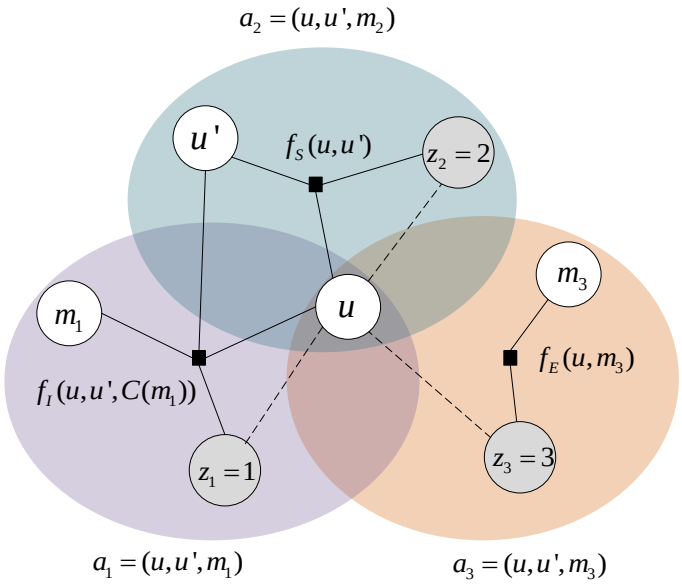


Figure 3.2: Graphical representation of the diffusion-targeted influence model. Three example diffusion actions of user u from the friend u' are shown as examples.

3.5 Diffusion-targeted Influence Learning

Generally, the reposting action of a user may be affected by the following three types of influences:

- **Interest-oriented Influence**: A user is likely to repost a microblog if the content is interesting to him. Consider a friend u' of u. If u' shares similar interests with u, or if u' is an expert for the interest category that u is interested in, then the friend u' is likely to have high influence on the user u.

- **Social-oriented Influence**: In this case, the repost action is based purely on the needs of social interaction. In other words, the action is triggered by the social influence exerted by his friend, instead of the

information contained in this microblog.

- **Epidemic-oriented Influence**: If a microblog is epidemic (*e.g.*, breaking news), it will be very probable to result in a repost action. In this situation, the influence is highly related to the epidemic-degree of this information rather than the content or the friend who post it.

Formally, let $F_I(u, u', c)$ denote the interest-oriented influence receive by user u from his friend u' related to the interest category c; $F_S(u, u')$ the social-oriented influence receives by u from u'; and $F_E(u, m)$ the epidemic-oriented influence of the microblog m. A user may receive different degree of influences from different friends according to either the closeness of their social connection, the friend's authority in the corresponding interest category, or simply the epidemic-degree of the disseminated information. In the following subsection, we elaborate a diffusion-targeted influence model, which efficiently differentiates and quantifies different types of influences.

3.5.1 Diffusion-targeted Influence Model

Consider a diffusion action $a = (u, u', m) \in A$. As aforementioned, this action could be triggered by three types of influences. We introduce a latent variable z, which indicates the source of the influence that leads to the diffusion action a. More precisely, $z = k, k \in \{1, 2, 3\}$ indicates that a is caused by the k-th type of influence. In order to infer the latent influence-indicator z for each diffusion action, as well as to quantify the degree of F_I, F_S and F_E, we propose the Diffusion-targeted Influence Model. Figure 3.2 shows an illustrative example of three diffusion actions between user u and his friend u'. Our model comprises the following three feature functions which models the three types of influences

F_I, F_S and F_E, respectively.

- **Interest-related feature function** $f_I(u, u', c)$, which contains two factors. The first factor is defined as the ratio between the number of microblogs that u reposted from u' in the interest category c and the total number of microblogs that belongs to interest category c posted by u'. The second factor refers to the weight of the interest category c in the interest profile of user u'. Intuitively, a higher weight represents a higher authority level in the corresponding interest category, which should cause larger influence. This function is defined as follows:

$$f_I(u, u', c) = \frac{|\{a = (u, u', m)|C(m) = c \wedge z_a = 1\}|}{|\{m \in M_{u'}|C(m) = c\}|} \times I(u')_c \ .$$

- **Social-related feature function** $f_S(u, u')$, which is defined as the ratio between the number of actions that u diffuses a microblog from u', over the total number of microblogs belonging to u':

$$f_S(u, u') = \frac{|\{a = (u, u', \cdot)|z_a = 2\}|}{|M_{u'}|} \ .$$

- **Epidemic-related feature function** $f_E(m)$, which is defined as the ratio between the number of friends who repost microblog m, over the total number of friends of user u:

$$f_E(m) = \frac{|\{a = (\cdot, u', m)|u' \in (N)(u)\}|}{|\mathcal{N}(u)|} \ ,$$

where $\mathcal{N}(u)$ denotes the friends of user u.

Typically, the target of this influence model is to best fit (reconstruct) the observation data, which is usually achieved by maximizing the likelihood function. With these feature functions, we define the objective likelihood function as:

$$\begin{aligned} P(Z) = & \frac{1}{R} \prod_{(u,u',m)\in A, z=1} f_I(u, u', C(m)) \\ & \times \prod_{(u,u',\cdot)\in A, z=2} f_S(u, u') \times \prod_{(\cdot,\cdot,m)\in A, z=3} f_E(m) \end{aligned} \tag{3.10}$$

where $Z = \{z_1, z_2, ..., z_{|A|}\}$ represents the hidden variables corresponding to all the actions in A, and R is a normalization factor. Figure 3.2 describes an illustration of this factorization. Each feature function (denoted in black box) is connected to the corresponding variables.

3.5.2 Model Learning

We intend to find the optimal parameter configuration that maximizes the objective function in Eq.(3.10). We propose to use the sum-product algorithm [67] to infer the latent variables. Two update rules are defined, one for message sent from variable node to factor node:

$$\mu_{z\to f}(z) = \prod_{f'\sim z\backslash f} \mu_{f'\to z}(z) \ ,$$

and one for message sent from factor node to variable node:

$$\mu_{f\to z}(z) = \sum_{\sim\{z\}} \left(f(Z) \prod_{z'\sim f\backslash z} \mu_{z'\to f}(z') \right) \ ,$$

where μ is the passed message; $f' \sim z\backslash f$ represents that f' is a neighbor node of the variable z on the factor graph except factor f; $z' \sim f\backslash z$ indicates that

z' is a neighbor node of the factor f on the factor graph except variable z, and $\sim \{z\}$ represents all the variables in Z except z.

After the learning process, the interest-oriented influence $F_I(u, u', c)$, social-oriented influence $F_S(u, u')$ and epidemic-oriented influence $F_E(u, m)$ could be achieved by calculating $f_I(u, u', c)$, $f_S(u, u')$ and $f_E(u, m)$, respectively.

3.6 Microblog Diffusion Modeling and Prediction

After learning the influence between the users and their friends, our next target is to utilize these influences for microblog diffusion analysis and prediction. Let $h \in \{-1, 1\}$ indicates whether an action (u, u', m) is actually performed, *i.e.*, whether user u reposts the microblog m of his friend u'. We maximize the conditional probability of user actions given the input social network G and history action set $\mathcal{A}$, *i.e.*, $P_\theta(H|G, \mathcal{A})$. More precisely, for each action in $\mathcal{A}$, we construct a training instance. We target at finding the optimal parameter θ^* to maximize the joint conditional probability for all the actions.

Note that the diffusion action set A used for training in Section 3.5 contains only those performed actions. As the task here is to predict whether a user will perform a diffusion action, we also include unperformed diffusion actions as negative samples into the training set $\mathcal{A}$. Suppose u' post a microblog m at time $t_{u'}$, and u does not repost this microblog. Then we add the unperformed actions (u, u', m) into $\mathcal{A}$ if only $t_u - t_{u'} < \Delta$, where Δ is the threshold time interval, and t_u is the activation time stamp of the user u, *i.e.*, u performs certain activity at the time. The underlying reasons for choosing these unperformed

actions is as follows. If the interval between the posting time of a microblog post and the activation time is too large, then unperformed diffusion action may probably because the user misses the corresponding microblog. On the contrary, if the microblog is presented within the time duration $(t_u - \Delta, t_u)$ and this user does not repost this microblog, then we have good reason to believe that the unperformed diffusion action is actually caused by the lack of influence, and thus is suitable to be included to the training set.

In order to maximize the probability $P(H|G, \mathcal{A})$, we factorize the global probability as the product of several local factor functions. We adopt the influences learned in the previous stage as the input factors, and learn the weighting parameters. Integrating all the factors together, we obtain the following log-likelihood objective function:

$$\begin{aligned}\mathcal{O}(\theta) =& \log P_\theta(H|G, \mathcal{A}) \\ =& \sum_{i,j,d} \left(\sum_{a_k=(u_i,u_j,m)\in\mathcal{A}\wedge C(m)=c} \alpha_{ijc} g(h_k, F_I(u_i, u_j, c)) \right) \\ &+ \sum_{ij} \left(\sum_{a_k=(u_i,u_j,\cdot)\in\mathcal{A}} \beta_{ij} g(h_k, F_S(u_i, u_j)) \right) \\ &+ \sum_{i} \left(\sum_{a_k=(u_i,\cdot,m)\in\mathcal{A}} \gamma_i g(h_k, F_E(u_i, m)) \right) - \log R\end{aligned} \tag{3.11}$$

where $g(h_k, F(\cdot))$ acts as the feature functions to link the factors to the corresponding variables, which is defined as

$$g(h_k, F(\cdot)) = \begin{cases} F(\cdot) \text{ , if } h_k = 1, \\ 1 - F(\cdot) \text{ , if } h_k = 0. \end{cases} \tag{3.12}$$

α, β and γ are the factor weights, and R is a normalization factor which ensures that the distribution is normalized with the sum of the probabilities equals to 1. With the function defined in Eq.(3.11), the objective of the training process is to estimate an optimial parameter configuration of $\theta^* = \{\alpha^*, \beta^*, \gamma^*\}$ from the training set $\mathcal{A}$ that maximizes $\mathcal{O}(\theta)$. The learning process contains two steps: 1) compute the gradient for each parameter; and 2) optimize all parameters with gradient descents. Specifically, we first approximate the marginal distribution $P_\theta(h_k|G, \mathcal{A})$. With the marginal probabilities, the gradient of a parameter can be obtained by summing over all the corresponding factor functions. Next, we use a gradient descent method to solve the above problem.

Diffusion Prediction. Given a new microblog m_{new}, and the action set A_{new} consisting of all existing diffusion actions related to m_{new}, the learned influences and weighting parameters can be used to predict the future participants in disseminating this new microblog. In practice, it is meaningless to do prediction for every new microblog since only a small portion will finally break out according to the power-law of information cascades [32]. Therefore, we devise certain criteria for starting the monitoring and prediction, *e.g.*, we delay the prediction until the number of existing diffusion actions $|A_{new}|$ exceeds some minimum number N_{thres}.

The diffusion of microblogs through the social network fits the Independent Cascade (IC) model [65]. IC starts with an initial set of active nodes, and the process unfolds in discrete steps according to the following rule: when a node becomes active, it is given a single chance to activate each currently inactive neighbor. If it succeeds, the corresponding neighbor will become active and follow this rule to activate more neighbors. But whether or not this node succeeds, it cannot make further attempts to active its neighbors in the

subsequent rounds. This process runs until no more activations are possible. Following the IC model, for each active user (user that has already performed the diffusion action) u and each of his friend u', we predict whether the action $a_{new} = (u, u', m_{new})$ will be performed by predicting the corresponding indicator h according to:

$$\begin{aligned} h^* &= \arg\max_h \log P(h|G, \mathcal{A}) \\ &= \arg\max_h \Big(\alpha_{ijC(m_{new})} g(h, F_I(u_i, u_j, C(m_{new}))) \\ &\quad + \beta_{ij} g(h, F_S(u_i, u_j)) + \gamma_i g(h, F_E(u_i, m_{new})) \Big). \end{aligned}$$

The above prediction process simply assumes that the delay in receiving the information will not affect the diffusion action. In other words, no matter how long the message is received after the original posting time, the user will make the same diffusion decision. However, this assumption will not hold under the microblogging circumstance, where people have more intention in reposting fresh microblogs, and the outbreak of a microblog usually happens during a relatively short time period. To handle this problem, we propose to incorporate a time decay factor to the feature functions in Eq.(3.12) as:

$$g(h_k, F(\cdot)) = \begin{cases} \lambda^l F(\cdot) \text{ , if } h_k = 1 \\ 1 - \lambda^l F(\cdot) \text{ , if } h_k = 0 \end{cases}$$

where $0 < \lambda < 1$ is the decay parameter and l is the length of the diffusion path when information reaches the predicted user. This new feature function penalizes long pathes.

With the above prediction process, we are able to predict the future virality

of a microblog in terms of its estimated reposting number, as well as the users who will participate in the diffusion process.

3.7 Experiments

In this section, we present the experimental results for evaluating our proposed approach.

3.7.1 Dataset and Experimental Settings

We conduct the experiments on a real-world dataset collected from Tencent Weibo[2], one of the largest microblogging platforms in China. We crawled a network with around 2.62 million users and all the microblogs posted by them from July 1st to August 31th in 2013, which gives rise to a total number of 192.3 million microblogs. We could observe a very high percentage of diffusion actions in the collected dataset, in which 63% of these microblogs are reposted from friends. The statistics of this dataset is shown in Table 3.1. In this experiment we focus on the original microblogs, and predict how they will be diffused through this social network. In order to estimate the diffusion lifetime of a microblog, we calculated the average duration between the time a microblog is originally posted to the time that the repost number of this microblog reaches 90% of the total repost number. The average time is less than 4 days, which means most of the diffusion actions are performed within 4 days after a microblog is posted. According to this observation, we divided our dataset into two parts, the training set with microblogs posted in the first 50 days, and the testing set with microblogs posted in the following 8 days, while

[2]http://t.qq.com/

Table 3.1: Statistics of the dataset used for viral microblog prediction task.

	#Microblogs	#Original	#Repost	#Image	#Days
Whole dataset	192.3m	71.5m	120.9m	129.2m	62
Training set	154.7m	60.3m	94.5m	103.9m	50
Testing set	25.3m	9.4m	15.9m	9.7m	8

we exclude the original microblogs posted in the last 4 days from the testing set, as the diffusion process of these microblogs may not have been finished and could not provide a valid groundtruth for our evaluation. For the external knowledge required in the MTTL classification model, we crawled 0.65 million articles (with 0.83 million images) from 20 categories from Sina.com.

Before the evaluation, we first pre-processed the texts and images. Texts were firstly segmented, then stop words, low-frequency words, mentions and urls were removed from the text vocabulary. For visual feature extraction, scale-invariant feature transform (SIFT) descriptors were first extracted from each image. We then trained a code book of 1,000 visual words. With the trained codebook, each descriptor was quantized into a visual word. Each image was further represented as a 1,000-D bag-of-visual-words feature. The parameters in MTTL were empiricially set as follows: $\sigma = 1$, $\rho = 0.1$, $\mu = 0.1$, $\delta = 0.01$. The threshold time interval Δ is set to 10 minutes.

One important parameter which will influence the experiment performance is the decay factor λ, which reflects the simulation for time delay, *i.e.*, the time duration between the initial posting of a microblog and the time that it reaches the user. Figure 3.3 shows the influence of λ in affecting the performance of predicting viral microblogs in terms of F1 value (refer to the following subsection for experiment details). As we can see, a large λ fails to provide enough decay

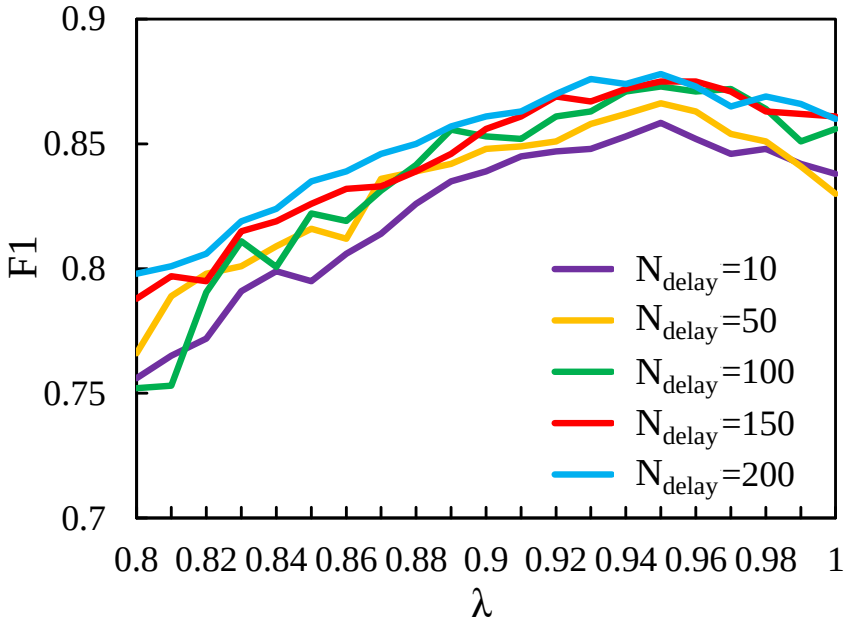


Figure 3.3: Influence of λ, in terms of F1 value of predicting viral microblogs.

ability, and a small λ causes too many potential diffusion actions to be rejected. Therefore, we adopt the optimal value of 0.95 for λ.

3.7.2 Predicting Viral Microblogs

Our first task is to predict whether a new coming microblog will become viral in the near future. Figure 3.4 shows the power-law distribution of the repost number in our dataset. We empirically define the viral microblog as one with repost number exceeding 1,000. This results in 168 viral microblogs in the testing set. We also randomly selected 832 non-viral microblogs with more than 200 repost number from the testing set. The prediction performance is measured for these 1,000 microblogs. We repeated the random selection 20 times, and the average result is reported. We compare our approach (denoted as TMP) to the following state-of-the-art methods:

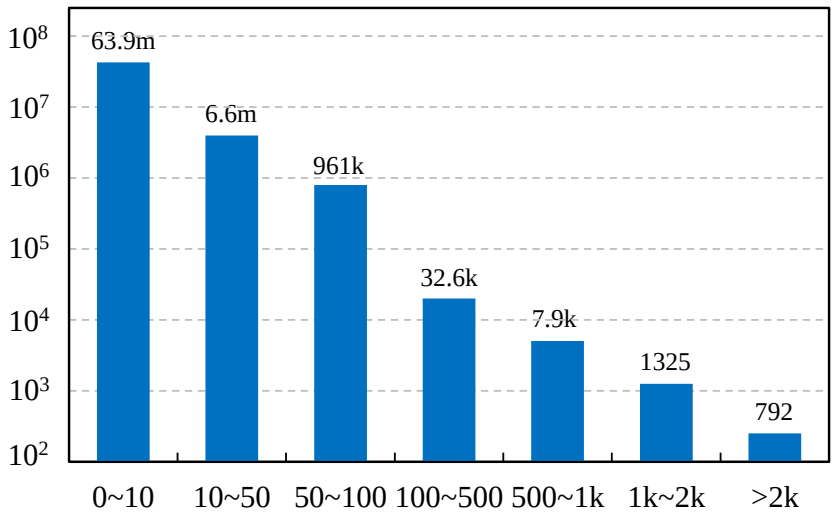


Figure 3.4: The distribution of repost number for all microblogs in our dataset.

- OSLOR [32]: The orthogonal sparse logistic regression model is a data-driven approach for cascading outbreak prediction. This method selects a set of nodes as sensors, and predicts the outbreaks based on the cascading behaviors of these sensors.

- ETL [26]: In the hot emerging topic learner, features are proposed for outbreak training and prediction. Although this method was originally used for emerging topic prediction, the proposed features and learning methods can also be applied to our task.

- PMP [57]: This method is able to predict popular microblogs in Twitter. Similarly, several features are defined and a multi-class classification method is adopted to predict the volume of retweets.

Furthermore, we also design two comparing methods by replacing the diffusion-targeted influence model of our framework with other influence learning model, and build the prediction model based the new types of influence. Specifically, the following influence models are adopted:

- TFG [121]: The topical factor graph model targets at quantifying the topic-level social influence between each pair of users.

- IPL [50]: The influence probabilities learning method adopts a probabilistic approached to assign each pair of users an influence probability.

The threshold number N_{thres} (as discussed in Section 3.6) is set in the following range: $\{10, 50, 100, 150, 200\}$. Precision, recall and F1 score are used as the evaluation measures. The results for our proposed TMP and the comparing methods are presented in Figure 3.5. From this figure, we have the following observations. 1) In terms of F1 measurement, TMP significantly outperforms the comparing methods. Larger threshold will benefit the prediction performance as more information about the diffusion process is available. 2) OSLOR achieves slightly better performance on precision as compared to TMP, however, the recall performance of OSLOR is far worse than that of TMP. OSLOR is designed to only monitor the most influential users, who are probable to trigger many reposting actions while inevitably less likely to participate in many outbreaking diffusion processes. In contrary, instead of the global influential measurement, our method models the local influences for each user and takes more factors into consideration in the prediction procedure, leading to more comprehensive results. 3) In general, the influence based methods, *i.e.*, TMP, TFG and IPL, perform better than feature based methods, *i.e.*, ETL and PMP. This is because those simple features defined on the small number of early participating users do not possess sufficient prediction ability. 4) By comparing TMP with the other two influence based methods, *i.e.*, TFG and IPL, the performance improvement demonstrates the effectiveness of our proposed influence model. While our proposed TMP characterizes the network in a more comprehensive

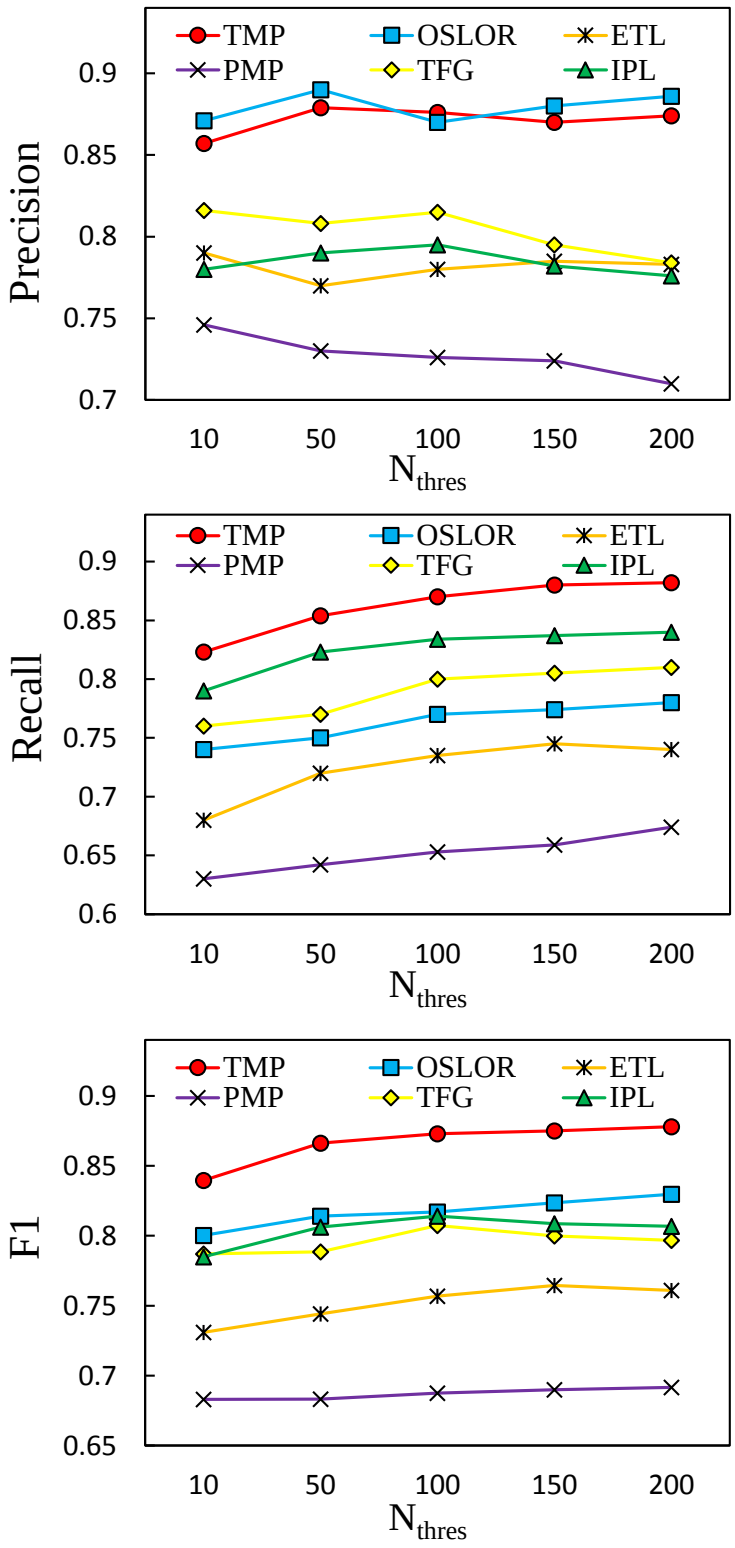


Figure 3.5: The results of predicting viral microblogs.

way, the other two influence models only model either the interest-related influence or pair-wise influence among users.

3.7.3 Predicting Diffusion Participants

The second task targets at predicting which users will participate in propagating a particular microblog. We compare our proposed TMP with the following methods:

- SVM: It uses the associated interest profile of users as well as the states of their neighbors to train a microblog classifier, which is then employed to predict the user actions.
- NTT-FGM [119]: The noise tolerant time-varying factor graph approach simultaneously models social network structure, user attributes and user action history for predicting the users' future actions.

Both of the two comparing methods need to train a prediction model, which requires a sufficiently large number of positive training samples to achieve satisfactory performance. We randomly select 1,000 microblogs whose final repost numbers exceed 200, and we want to predict all the diffusion participants when the retweet number of this microblog (denoted as # early participants) reaches 10, 50, 100, 150 and 200, respectively. The average prediction results for these 1,000 microblogs are presented in Figure 3.6. The results demonstrate the superiority of our proposed method under all evaluation measurements. In addition, our method also shows more stable prediction performance over different early participant numbers. The underlying reason is that our training procedure does not depend on these early diffusion activities. On the other hand, the two comparing methods need to train a prediction model for each

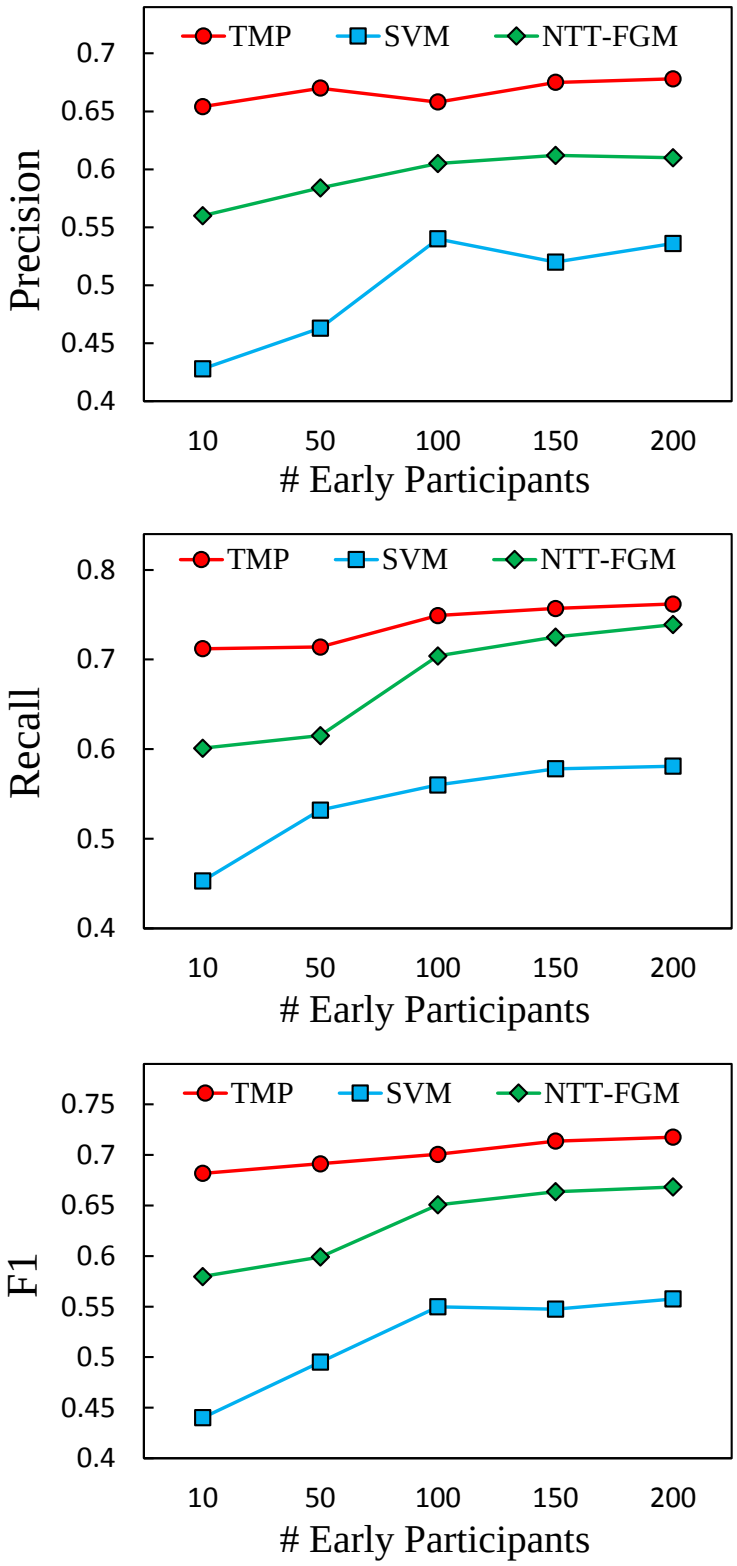


Figure 3.6: The results of predicting future diffusion participants.

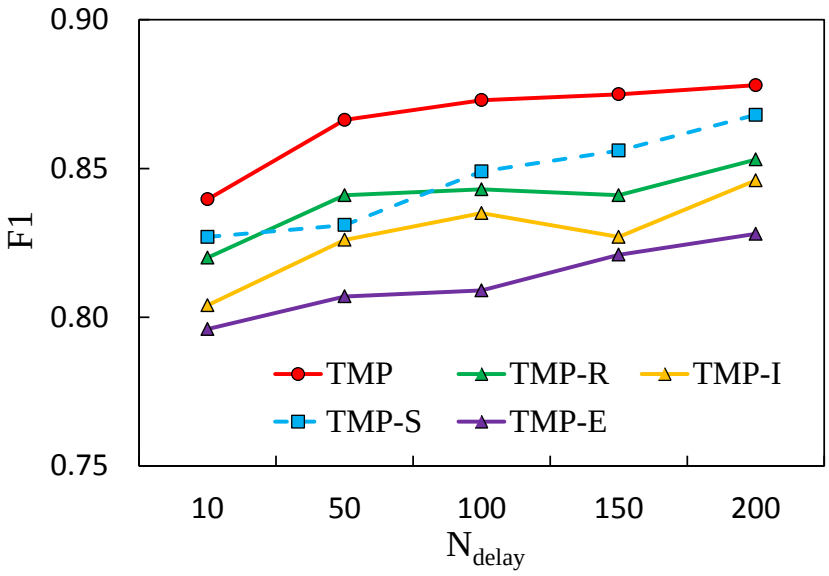


Figure 3.7: Effects of different components, in terms of F1 value of predicting viral microblogs.

microblog diffusion process, and the performance heavily relies on the number of training samples, *i.e.*, the early participants.

3.7.4 Component Contribution Analysis

In this part, we evaluate the effects of the three types of influences defined in our framework, namely, interest-oriented influence, social-oriented influence and epidemic-oriented influence; while the interest-oriented influence is closely related to the result of user interest profile learning. Specifically, the following cases are evaluated: 1) TMP-P, which replaces our interest profile learning component with LDA to generate topics and infer user interest profile. 2) TMP-I,TMP-S and TMP-E, which removes the interest-oriented influence, social-oriented influence and epidemic-oriented influence from our framework, respectively. The results of predicting viral microblogs in terms of F1 measurement are shown in Figure 3.7. From the figure, we can see that the perfor-

mance drops significantly by removing any of the proposed components. Specially, epidemic-oriented influence has the strongest correlation with the diffusion process of viral microblogs, hence removing it will decrease the prediction performance the most. Besides, by replacing our proposed interest profile learning component with other profile learning method, the performance decreases, which demonstrates the effectiveness of our model.

3.8 Summary

In this chapter, we presented a novel approach for predicting the viral microblogs and the subsequent diffusion actions in microblogging network. Specifically, we defined three types of influence, namely, interest-oriented influence, social-oriented influence, and epidemic-oriented influence, which jointly determine the user diffusion action. We devised a multi-task transfer learning model for identifying the interest categories of microblogs. A diffusion-targeted influence model was proposed for quantifying different types of influences. We formulated the diffusion prediction as a factorization of the user intention of reposting a microblog. Extensive experiments have been conducted on a real-world microblogging dataset to show the superiority of our proposed approach as compared to the state-of-the-art methods.

CHAPTER 4

Microblog Tracking

4.1 Introduction

In the microblog networks, the outbreak of a viral topic is usually led by a few outbreaking microblogs. Therefore, viral posts are indispensable for the detection of viral topics in the subsequent step. On the contrary, a viral post will not always lead to a viral topic. Due to the intrinsic "entertainment" character of most microblog users, we can usually observe the widely repost of some insignificant microblogs. For example, the microblogs posted by a famous user who has a great number of followers are very easy to get widely reposted due to celebrity worship, despite the less meaningful content. However, without enough follow-ups, these posts can hardly develop into topics. This motivates us to investigate the problem of microblog tracking. Given a collection of microblogs (*e.g.*, the viral posts predicted in the previous stage), we would

like to track their related posts in the incoming data stream. If a microblog is detected to receive enough subsequent posts, we will report this microblog as well as the subsequent ones as a viral topic. In addition to the utilization in topic detection, the proposed microblog tracking task can also be of great help in many other scenarios. For example, it can also facilitate the process of information filtering and tracking, as well as successive microblog discovery.

The tracking of topics can be traced back to the TDT (topic detection and tracking) task [5]. Given a topic described by a few positive samples, the task of topic tracking is to monitor a stream of news stories and find out the samples discussing the given topic. Superficially, the task of microblog tracking is quite similar to topic tracking. However, compared with topic tracking, microblog tracking has the following challenges. Firstly, in order to perform topic tracking, the topic description (*e.g.*, several keywords) or several topic samples need to be provided. Therefore, the topic tracking system has a clear understanding of what to track, or can learn signals from the positive samples. While in our case, the tracking target is only one microblog, which makes the task even more challenging. Secondly, in traditional topic tracking task, the number of topics being tracked at the same time is limited. When there are several target topics, they are generally tracked individually. On the other hand, our tracking targets are massive and highly dynamic. Since the microblogs to be tracked are from the previous viral microblog prediction stage, with continuous candidate posts being predicted, the number of targets will keep increasing. In this circumstance, tracking each target individually will result in too much cost. Therefore, it would be much better if a method can be proposed to deal with all tracking microblogs simultaneously, merging them into a single task. Finally, since a topic in microblog streams may evolve with time, we are faced

with the transition of topic content. How to incrementally update the topic information is also a major concern.

Considering the above challenges, we model the microblog tracking problem using an evolutional dictionary learning method which jointly tracks all microblog targets in a unified framework. Based on sparse coding, each microblog representation is modeled as a sparse linear combination of basis elements from a dictionary. We expect to learn a dictionary of atoms as a compact representation of all the tracking targets, such that each microblog can be approximately represented by a linear combination of a few atoms. If a new microblog can be represented as a sparse linear combination of these atoms with low error, it is a good indicator that this microblog is similar enough to a tracking target. Since the contents of the tracking targets may evolve with time, the tracked results are then used to update the dictionary for tracking new microblogs. Assuming a smooth evolution of the contents of the tracking targets, we introduce a dictionary-transition matrix to capture the relationship between the dictionaries of two adjacent time slots. By doing this, the information about the current batch of data and the accumulated history are both taken into consideration.

4.2 Task Description

The problem is formalized as follows: a stream of microblog posts arrive continuously, and each post has an attached timestamp that indicates its arriving time. Instead of using the exact time as the timestamp, we adopt a coarser granularity such as a few minutes or hours. Given a collection of target posts, we aim at tracking these targets in the microblog stream to identify the posts that discuss the same topics with the targets. We assume a fixed collection of

target posts first, *i.e.*, no new target post will be provided during the tracking procedure. Later in Section 4.3.3, we extend the setting to dynamic target collection, where new tracking targets can be provided to the tracking system at any time, as in the real scenario.

Let C_0 be the collection of target microblogs initially provided to the system. Let C_t denote the set of all the tracked posts with timestamp t, thus all the tracking results until time t are represented by $C_{\leq t} = \bigcup C_i$, $0 \leq i \leq t$. We adopt the conventional vector space model with TF-IDF term-weighting to represent the posts in our data stream. One problem of this representation method is that when new terms are identified, the vocabulary size will increase. For simplicity, we assume a stable global vocabulary of size m which is independent of t. We can easily extend to the case where the vocabulary size increases with t by applying zero-padding to the representation matrix. With this assumption, we can represent C_t with a matrix P_t of size $m \times |C_t|$, where $|C_t|$ is the number of posts in C_t and each column of P_t is the feature vector of a microblog post.

We expect to learn a dictionary matrix $A \in \mathbb{R}^{m \times k}$ as a compact summary representation for all the tracking targets. Each column of A is called a *basis vector*. Ideally, we expect the dictionary to have a set of representative basis vectors for each of the tracking targets. With such a representative dictionary, the target posts can be represented as a linear combination of these basis vectors. Here, we are faced with two problems. The first one is how to identify posts that are related to the tracking targets using A, *i.e.*, how to obtain C_t at each time t. Secondly, since the contents of the tracking targets may evolve with time, it is inappropriate to use a fixed dictionary A_0 which is learned from the original tracking targets to track all later posts. Therefore, the dictionary A needs to be updated when new posts are identified. In other words, we need to

derive A_t from A_{t-1} using the new tracked posts in C_t. Next, we will introduce our methods for addressing these two problems.

4.3 Problem Formulation

4.3.1 Identifying Related Microblogs

Suppose after time t, our dictionary matrix is A_t. Given a new post represented by feature vector y with timestamp $t+1$, we see whether y could be represented as a sparse linear combination of the basis vectors (columns of A_t). The sparsest representation is the solution of

$$\min_x \|x\|_0 , s.t.\ y = A_t x, x \geq 0 \tag{4.1}$$

where $\|x\|_0$ is the ℓ_0-norm, counting the number of non-zero entries of a vector. Generally, solving Eq. (4.1) is NP-hard. Recent research has shown that one could obtain the solution to Eq. (4.1) by solving the following convex relaxation problem:

$$\min_x \|x\|_1 , s.t.\ y = A_t x, x \geq 0 \tag{4.2}$$

In most practical situations, Eq. (4.2) is not applicable because y cannot be represented exactly by $A_t x$. In such cases, we could instead represent $y = A_t x + e$, where e is a sparse noise vector. With constrain on e to handle the noise, Eq. (4.2) is then relaxed to the following problem:

$$f(y) = \min_x \|y - A_t x\|_2^2 + \lambda \|x\|_1 , s.t.\ x \geq 0 \tag{4.3}$$

which is known as the Lasso [122]. This formulation naturally takes into account

both the reconstruction error and the complexity of the sparse decomposition.

Given a new post y with timestamp t and dictionary A_t, we solve Eq. (4.3) to determine whether y is related to the tracking targets or not. Since the contents of all the tracking targets are embedded into the basis vectors of A_t, if $f(y)$ is larger than a threshold δ, then y cannot be well reconstructed by the basis vectors of the tracking contents and should be viewed as irrelevant to the targets. However, if $f(y)$ is smaller than δ, then y has good reconstruction by A_t, which is a good indication that this microblog is related to the targets. Therefore, this post should be added to the result set C_{t+1}. Note that all the vector features are normalized to unit ℓ_1 length, and hence the objective values are in the same scale.

After a microblog post is reported as a related data sample, the following step is to assign it to the corresponding tracking target. Suppose all the previous content information about the i-th target is represented with a vector $c_i \in \mathbb{R}^k$, which can be viewed as the reconstruction coefficient using dictionary A_t. Then a tracked microblog with reconstruction coefficient x can be assigned to the target whose representation c_i is closest to x, *i.e.*,

$$Target\ ID = \arg\min_i \|c_i - x\|_1 \tag{4.4}$$

We now describe how to get c_i. c_i is initialized with the reconstruction coefficient of the i-th target microblog. After each time slot, the representation c_i is updated according to the following equation:

$$c_i^{t+1} = \frac{n_i^{\leq t} \cdot c_i^t + \sum_{j=1}^{n_i^{t+1}} x_i^j}{n_i^{\leq t} + n_i^{t+1}} \tag{4.5}$$

where $n_i^{\leq t}$ is the total number of posts tracked for target i until time slot t, n_i^{t+1} is the number of posts tracked for target i within time slot $t+1$, and x_i^j is the reconstruction coefficient of the j-th post assigned to target i.

The performance of this tracking algorithm relies on the quality of the dictionary A_t. We now describe the dictionary learning method.

4.3.2 Dictionary Learning

Given a set of training samples $S = [s_1, s_2, ..., s_n]$, classic dictionary learning techniques learn the dictionary A from S by optimizing the following formulation:

$$\min_{x_i, A} \sum_{i=1}^{n} (\|s_i - Ax_i\|_2^2 + \alpha \|x_i\|_1) + \beta \|A\|_1, s.t.\ A \geq 0, x_i \geq 0 \tag{4.6}$$

which can be equivalently written as minimizing the following loss function over X and A jointly:

$$g(X, A) = \|S - AX\|_F^2 + \alpha \|X\|_1 + \beta \|A\|_1, s.t.\ A \geq 0, X \geq 0 \tag{4.7}$$

The initial dictionary A_0 can be learned by setting the training samples S in Eq. (4.7) to the original tracking target posts P_0 and solving this loss function. Since the contents of the tracking targets may evolve over time, with an increasing number of related posts being detected, we may observe a shift of the contents from the original tracking targets. In this scenario, it is inappropriate to use the initial dictionary A_0 to track later posts. In order to capture the evolution of the contents, one method is to aggregate all the identified posts to generate a complete data matrix $P_{\leq t}$, and use it as

the training matrix S to learn the dictionary A_t. However, as time passes, the number of detected posts increases, which will result in a very large data matrix $P_{\leq t}$. Learning dictionary using Eq. (4.7) with such a huge matrix is very inefficient. In addition, the complete data matrix $P_{\leq t}$ may contain many out-of-date data. Using these data to estimate current contents may lead to wrong inference. Another typical strategy, consists of directly learning the dictionary from the current batch of data P_t, will be more efficient. The problem about this strategy is that the entire tracking history will be ignored. For a tracking target, if no microblog post is detected to be relevant to it in the current time interval, its information will be lost forever. One is therefore faced with the trade-off between past and present observations. In order to efficiently learn the tracking dictionary from the current batch of data P_t, while also taking the past information into consideration, we propose to learn the dictionary in an incremental manner.

In the case where only the current batch of data is used, the dictionary is learned by solving the following formulation:

$$\min_{A_t, X_t} \|P_t - A_t X_t\|_F^2 + \alpha \|X_t\|_1 + \beta \|A_t\|_1, s.t.\ A_t \geq 0, X_t \geq 0 \tag{4.8}$$

However, we would like to say something about the current data P_t in terms of the accumulated history. Important information about the past is revealed by A_{t-1}, the previous learned dictionary. Although the observed data is dynamic, we may comfortably assume that the contents about the tracking targets evolved smoothly during one time step, and the current data are related to those appeared in the previous time slots. Therefore, we suppose that the new data may also be represented by the previous dictionary A_{t-1}, by solving

the following formulation:

$$\min_{H_t} \|P_t - A_{t-1}H_tX_t\|_F^2 + \lambda \|H_t - I\|_F^2 + \gamma \|H_t\|_1, s.t.\ H_t \geq 0 \tag{4.9}$$

The above formulation directly explains the current data P_t jointly by the present and the past dictionary through a mapping factor H_t. The matrix H_t is a dictionary-transition matrix, which captures how much the current dictionary A_t may be linearly represented by the previous one A_{t-1}. The regulation on the ℓ_1 norm of H_t has the effect of promoting a smooth evolution. The regularization term $\lambda \|H_t - I\|_1$ controls how much to bias A_t towards A_{t-1}. The parameter λ balances the present and past information: it quantifies the extent to which the model past oriented ($\lambda \to \infty$) or present oriented ($\lambda \to 0$). In this way, we jointly learn content evolution given by A_t and the temporal dependencies given by H_t. The joint constraint proposed in this model is soft as it operates indirectly through X_t, common to both sparse representations. Integrating Eq. (4.8) and Eq. (4.9), we derive the following formulation:

$$\begin{aligned} \mathcal{L} = \min_{A_t, X_t, H_t} & \|P_t - A_tX_t\|_F^2 + \|P_t - A_{t-1}H_tX_t\|_F^2 \\ & + \lambda \|H_t - I\|_F^2 + \alpha \|X_t\|_1 + \beta \|A_t\|_1 + \gamma \|H_t\|_1, \\ & s.t.\ A_t \geq 0, X_t \geq 0, H_t \geq 0 \end{aligned} \tag{4.10}$$

Since this model learns a dictionary in an evolutional manner, we name it Evolutional Dictionary Learning.

Once the dictionary is updated, we need to update the representation vector c_i for each tracking target correspondingly. Suppose c_i^{t-1} is the reconstruction coefficient with respect to dictionary A_{t-1}. After A_{t-1} is updated to

A_t, we need to modify c_i^{t-1}, updating it to be the reconstruction coefficient with respect to dictionary A_t. We do this by solving Eq. (4.3), where A_{t-1} is replaced with A_t and y is replaced with $A_{t-1}c_i^{t-1}$, which is:

$$c_i^t = \arg\min_x \|A_{t-1}c_i^{t-1} - A_t x\|_2^2 + \lambda\|x\|_1, \ s.t.\ x \geq 0 \tag{4.11}$$

4.3.3 Dynamic Target Collection

In the previous part, we learn the dictionary based on a fixed target collection. After each time slot, the dictionary is updated to capture the dynamic content change. In the real case, the tracking target collection is not always fixed, and new targets can be added to the system at any time. Considering this, the dictionary needs to be updated to capture the new tracking contents. Assuming after time slot t, we collected two sets of microblog posts: P_t, which is the tracking results as described in the previous sections, and NT_t, which is a set of new tracking targets. Our goal is to update the old dictionary A_{t-1} to express both the two sets of data. We adopt a two-step approach. In the first step, we update A_{t-1} to A_t by solving the problem in Eq. (4.10) as described above. In the second step, we do this update procedure one more time by replacing P_t in Eq. (4.10) by NT_t, and A_{t-1} by the new dictionary just achieved in the previous step. The reason for adopting this two-step approach instead of merging P_t and NT_t into one input set is that, the number of new tracking targets in NT_t is generally much smaller than the number of tracking results in P_t. With this two-step update strategy, we can control the regularization parameters to make the dictionary emphasize more on the tracking targets, thus avoiding the learning process to be dominated by the large number of tracking results.

4.4 Algorithm

The problem in Eq. (4.10) is not convex for all parameters A_t, H_t, X_t simultaneously. However, we can find a local minimum for the objective function using a multiplicative updates, similar to the strategy proposed for NMF [72]. Considering the Karush-Kuhn-Tucker (KKT) first-order conditions applied to our problem, we derive:

$$\begin{aligned} &W_t \geq 0,\ A_t \geq 0,\ H_t \geq 0, \\ &\nabla_{X_t}\mathcal{L} \geq 0,\ \nabla_{A_t}\mathcal{L} \geq 0,\ \nabla_{H_t}\mathcal{L} \geq 0, \\ &X_t \odot \nabla_{X_t}\mathcal{L} = 0,\ A_t \odot \nabla_{A_t}\mathcal{L} = 0,\ H_t \odot \nabla_{H_t}\mathcal{L} = 0. \end{aligned} \tag{4.12}$$

where $\odot$ is the element-wise product.

From the loss function in Eq. (4.10), we derive the gradients according to each parameter:

$$\nabla_{A_t}\mathcal{L} = A_t X_t X_t^T - (P_t X_t^T - \beta) \tag{4.13}$$

$$\nabla_{X_t}\mathcal{L} = (A_t^T A_t + H_t^T A_{t-1}^T A_{t-1} H_t) X_t - (A_t^T P_t + H_t^T A_{t-1}^T P_t - \alpha) \tag{4.14}$$

$$\nabla_{H_t}\mathcal{L} = (A_{t-1}^T A_{t-1}) H (X_t X_t^T) + \lambda H_t - (A_{t-1}^T P_t X_t^T + \lambda I - \gamma) \tag{4.15}$$

Theorem 1. The loss function $\mathcal{L}$ in Eq. (4.10) is non-increasing under the following update rules:

$$A_t = A_t \odot \frac{P_t X_t^T - \beta}{A_t X_t X_t^T} \tag{4.16}$$

$$X_t = X_t \odot \frac{A_t^T P_t + H_t^T A_{t-1}^T P_t - \alpha}{(A_t^T A_t + H_t^T A_{t-1}^T A_{t-1} H_t) X_t} \tag{4.17}$$

$$H_t = H_t \odot \frac{A_{t-1}^T P_t X_t^T + \lambda I - \gamma}{(A_{t-1}^T A_{t-1}) H (X_t X_t^T) + \lambda H_t} \tag{4.18}$$

Proof. The cost function can be reformulated as follows:

$$\begin{aligned}\mathcal{L} = &\min_{A_t, X_t, H_t} \sum_i \|P_t(:,i) - A_t X_t(:,i)\|_2^2 \\ &+ \sum_i \|P_t(:,i) - A_{t-1} H_t X_t(:,i)\|_2^2 \\ &+ \lambda \|H_t - I\|_F^2 + \alpha \sum_i \|X_t(:,i)\|_1 + \beta \|A_t\|_1 + \gamma \|H_t\|_1 , \\ &s.t.\ A_t \geq 0, X_t \geq 0, H_t \geq 0\end{aligned} \tag{4.19}$$

where $P_t(:,i)$ and $X_t(:,i)$ is the i-th column of P_t and X_t, respectively.

Keeping A_t and H_t fixed, we can minimize $\mathcal{L}$ with respect to each column p of P_t and x of X_t separately:

$$\ell(x) = \|p - A_t x\|_2^2 + \|p - A_{t-1} H_t x\|_2^2 + \alpha \|x\|_1 \tag{4.20}$$

Consider a current approximation $\hat{x}$ of the solution and formulate the following problem:

$$\begin{aligned}\hat{\ell}(x) = &\|p - A_t x\|_2^2 + \|p - A_{t-1} H_t x\|_2^2 \\ &+ \alpha \|x\|_1 + (x - \hat{x})^T S_x (x - \hat{x})\end{aligned} \tag{4.21}$$

where $S_x = Diag(q) - M$ with $Diag$ the diagonal operator creating a diagonal matrix from an input vector, $M = A_t^T A_t + H_t^T A_{t-1}^T A_{t-1} H_t$ and $q = [M\hat{x}]/[\hat{x}]$.

Since S_x is semidefinite positive, we can get $\hat{\ell}(x) \geq \ell(x)$ for all x with $\hat{\ell}(\hat{x}) = \ell(\hat{x})$. Let x^* be the value of x that minimize $\hat{\ell}(x)$, then $\hat{\ell}(x^*) \leq \hat{\ell}(\hat{x})$. This implies that $\ell(x^*) \leq \hat{\ell}(x^*) \leq \hat{\ell}(\hat{x}) = \ell(\hat{x})$. Therefore, suppose we can obtain an update rule of x^* in terms of $\hat{x}$, we have proved that the cost function ℓ will be non-increasing under this update rule.

In order to get x^* that minimize $\hat{\ell}(x)$, we set $\nabla_x \hat{\ell}(x)$ to zero, and obtain:

$$\nabla_x \hat{\ell}(x) = \nabla_x \ell(x) + S_x(x - \hat{x}) = 0 \tag{4.22}$$

According to Eq. (4.14), let $b = A_t^T p + H_t^T A_{t-1}^T p - \alpha$, then we can obtain $\nabla_x \ell = Mx - b$. Therefore,

$$\begin{aligned}
&\nabla_x \ell(x) + S_x(x^* - \hat{x}) \\
&= Mx^* - b + S_x(x^* - \hat{x}) \\
&= (M + S_x)x^* - (S_x\hat{x} + b) \\
&= Diag(q)x^* - (Diag(q)\hat{x} - M\hat{x} + b) \\
&= Diag(q)x^* - (Diag(M\hat{x})Diag^{-1}(\hat{x})\hat{x} - M\hat{x} + b) \\
&= Diag(q)x^* - (M\hat{x} - M\hat{x} + b) \\
&= Diag(q)x^* - b
\end{aligned} \tag{4.23}$$

From Eq. (4.22) and Eq. (4.23), we can get:

$$Diag(q)x^* - b = 0 \tag{4.24}$$

Therefore, we can derive the following update rule for x:

$$x^* = bDiag(\hat{x})Diag^{-1}(M\hat{x}) = \hat{x} \odot \frac{b}{M\hat{x}} \tag{4.25}$$

According to the above formulation, we can obtain the update rule for the matrix X_t, as listed in Eq. (4.17). According to previous discussion, the cost function ℓ is non-increasing under the update rule in Eq. (4.25). Hence, we proved that the cost function $\mathcal{L}$ is non-increasing under the update rule in

Eq. (4.17). Analogously, we can also prove that $\mathcal{L}$ is non-increasing under the update rules in Eq. (4.16) and Eq. (4.18). □

4.5 Experiments

In this section, we will evaluate the performance of our proposed method for two tasks. The first task is microblog tracking: given a set of microblogs as targets, the system need to identify all related microblogs in the subsequent microblog stream. The second task is the early-stage detection of viral topics in microblog stream, where the method proposed in this chapter will be integrated with the framework introduced in Chapter 3 to generate a two-stage detection procedure. The detailed task description will be described below.

4.5.1 Tasks and Evaluation Methodology

Task 1: microblog tracking. The purpose of this task is to test the ability of our method in identifying related posts from the evolving microblog stream. The input to the testing process is a set of pre-selected microblog posts $\{T_i\}$. The following experimental settings will be tested, including different number of tracking targets, different arrival pattern, and different window size or update rate.

- *Setting 1.* In this setting, we will provide to the system different number tracking targets with the same timestamp as input, to test how the system will perform with increasing number of targets.

- *Setting 2.* In this setting, we will keep a fixed number of targets, but test various arrival pattern. For example, all the targets can arrive at the

same time, or their arrival times can scatter to different extend.

- *Setting 3.* With this setting, we test the case where the system is provided with a large number of tracking targets whose arrival time span a relative long time period. This setting is to simulate the real-world scenario where tracking targets arrive to the system continuously. Generally, the life cycle of a topic-related microblog (from the time it is created to the time its related topic vanishes) will be very short. Under this circumstance, it is not economic for the system to remember all the targets, especially those very old ones. Therefore, we will test the performance of various update rates for our method, *i.e.*, the parameter λ in Eq. (4.9), which measures how fast our system forgets past information. Analogously, for those baseline methods that deal with different targets independently, we will test their performance under a small window sizes, *i.e.*, the number of targets kept in the system.

For the simplicity of evaluation, we assume that each microblog post arriving later than $\{T_i\}$ is labeled to only one target in $\{T_i\}$, or not labeled to any target. The tracking results are evaluated against the labeled ground-truth using standard performance metrics commonly used in informational retrieval: precision, recall and F1 calculated by averaging results from all targets.

Task 2: early-stage viral topic detection. In this task, we will test the overall performance of the proposed framework for early-stage viral topic detection, by running the whole pipeline integrating the works described in Chapter 3 and this chapter. The evaluation for early-stage viral topic detection is to test the ability of the proposed two-step approach for viral topic prediction: whether our framework can successfully detected the viral topics with satisfactory accuracy

in the very early stage of the viral topics. Recall that in Chapter 3, we propose a method for vial microblog prediction: finding the microblogs that have the potential to become viral in the near future from a stream of microblog posts. Based on this, the whole pipeline for early-stage viral topic detection takes the following two steps: we first adopt the viral microblog prediction method to predict whether each microblog will become viral, those microblogs with positive prediction are then taking as input to our microblog tracking framework. During the tracking procedure, for any tracking target, if the number of its related microblogs exceeds the threshold, we will report it as a new topic.

Similarly as previous, we assume that each microblog in the corpus has been labeled with a single, most dominant ground-truth topic. Assume that the system generates n topics, and the number of ground-truth topics is m. In general, $n \neq m$, since the system is allowed to generate any number of topics. In order to evaluate the performance of the topic detection system, we adopt the following metrics.

F1: this metric is generally adopted to intuitively measure the ability of a topic detection system in discovering viral topics. Specifically, we would like to calculate how many ground-truth topics have been successfully identified by the system. Denote this number as s. Here, a ground-truth topic is successfully identified means there exist a system-generated topic whose topic description is quite similar to this ground-truth topic (we adopt a threshold for similarity to determine whether two topics are the same topic or different topics). The precision and recall is therefore defined as s/n and s/m, respectively. Following the standard definition, F1 is the harmonic mean of precision and recall, *i.e.*, $F1 = 2(precision \cdot recall)/(precision + recall)$.

In order to calculate the similarity between two topics, we use the centroid of all posts corresponding to a certain topic as the expression for this topic. Considering that it is common to define a topic using the list of the top words [124], we consider the top 10 words appearing in the word distribution of each topic. Given the word distribution q of a system-generated topic, its similarity to a ground-truth topic with word distribution p is calculated with the KL-divergence $KL(q\|p)$.

MicroF1: this metric has been commonly reported in topic detection and tracking literature as a fine-grained measurement for the quality of the topic detection performance. We first construct the $m \times n$ confusion matrix between ground-truth topics and system-generated topics, where the $(i,j) - th$ element is the number of post that is tagged as ground-truth topic i, and tagged as topic j by the system. From this matrix, for each ground-truth topic, we identify the set of most frequently co-occurring system topics $top(i)$. The micro precision and recall can then be computed as follows:

$$\begin{aligned} MicroPrecision &= \frac{\sum_{i=1}^{m} |P_{true}(i) \bigcap P_{system}(i)|}{\sum_{i=1}^{m} |P_{system}(i)|} \\ MicroRecall &= \frac{\sum_{i=1}^{m} |P_{true}(i) \bigcap P_{system}(i)|}{\sum_{i=1}^{m} |P_{true}(i)|} \end{aligned} \tag{4.26}$$

where $P_{true}(i)$ is the set of posts with topic i as ground-truth label, $P_{system}(i)$ is the set of posts tagged by the system with a topic in $top(i)$, $|\cdot|$ and $\bigcap$ denote the cardinality and set intersection respectively.

MissRate: This measure attempts to intuitively capture the following notion: how much of a topic has been missed before the system is able to report its existence to the user. It is a direct measure of the reaction speed of an online topic detection model to serve as an early warning system for emerging

themes. A lower miss rate reflects the ability to detect emerging topics earlier. Suppose $time_{detect}$ is the timestamp that a topic is first detected by the system, the miss rate is then computed as follows:

$$MissRate(t) = \frac{|p : p \in P_{true}(t) \wedge timestamp(p) < time_{detect}(t)|}{|P_{true}(t)|} \tag{4.27}$$

4.5.2 Dataset and Experimental Settings

In order to test the performance of the proposed method, we need a stream of microblogs, a part of which are tagged with ground-truth topic label. The microblog stream can be easily crawled with the public stream API, or using other crawling strategy. However, the large volume of the crawled microblogs makes it nearly impossible to manually discover ground-truth viral topics, and label all the microblogs related to these topics. Therefore, we adopted an inverse crawling strategy: achieving the ground-truth viral topics and the corresponding microblogs first, then getting the remaining microblogs (denoted as background microblog set in the following) which are not related to the ground-truth topics. Using this strategy, we crawled a real-world dataset from Tencent Weibo. The crawling process is as follows: first, we got a list of all hot topics from July 1, 2014 to August 31, 2014 using the topic API. Some topics may span a few days and returned by the API as different topics. After merging these topics into a single one, we got 176 viral topics in total. Since some topics were actually not very hot and widely spread, and some were listed for advertising purpose, we then filtered out those advertising topics and ranked the remaining topics according to their hotness, and the top 60 topics as well as the corresponding microblogs were selected as our ground-truth topic set. From

the 60 topics, we randomly select 10 as the training set for parameter tuning, and the remaining 50 topics were used as the testing set. In the next step, these topic-related microblogs were merged into our previous dataset (dataset adopted for evaluation in Chapter 3), in which all the microblogs were posted during the time duration from July 1, 2013 to August 31, 2013. To do this, we modified the year from 2014 to 2013 for the timestamps of all the topic-related microblogs, and kept the date and time unchanged. The reason for generating this semi-synthetic dataset instead of directly crawling background microblog set from the same time period is to guarantee that none of the background microblog is related to the ground-truth topics.

The dataset generated with this method directly provides the required ground-truth for our Task 2. In order to evaluate Task 1, we selected the first microblog (the one with the earliest timestamp) from the microblog set of each topic. These microblogs served as the tracking targets. For each tracking target, we labeled those microblogs with the same ground-truth topic to this target as positive, and the remaining ones as negative. For different settings in Task 1, we adopt the following strategies to construct the required dataset:

- *Setting 1.* To construct test case for this setting, we kept the relative time of the targets and the related microblogs fixed, while aligning the timestamp of all the targets to the same time. Then we increased the target number from 1 to 50 to test the performance. For each target number (except for 50 where all targets need to be selected), we randomly selected 10 sets of targets. The final performance was reported by averaging the results of the 10 sets.

- *Setting 2.* In this setting, we kept the number of tracking targets to

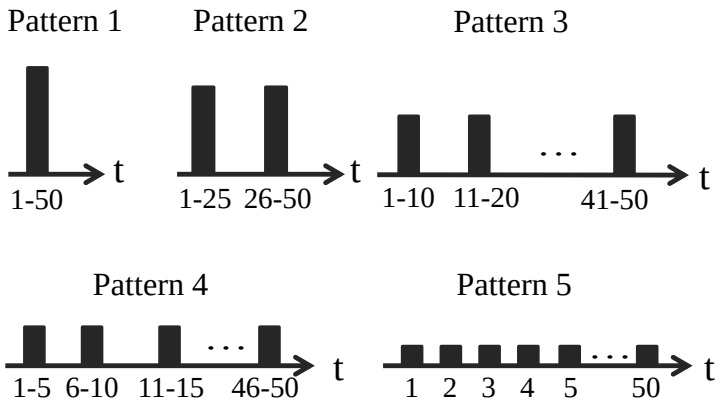


Figure 4.1: Distribution pattern for the 50 tracking targets. In each pattern, the number indicates the target ID (*e.g.*, 1-10 means a group of targets consisting of the 1st to 10th tracking target). The arrival time of targets within the same group are aligned to the same timestamp, and the time interval between two groups is fixed to 1 day.

50, and aligned their timestamp according to the patterns described in Figure 4.1.

- *Setting 3.* In order to test this case, we made the timestamp of the 50 targets spanned evenly through 10 days, and fixed the window size to 5 for the comparing methods. Then we tried different λ value to test how our method behave under different trade-off parameter.

We adopt a granularity of 10 minutes for our algorithm to update the dictionary, and also for other comparing methods which process the data in a batch manner. We tuned the $\ell 1$ regularization parameters on the training set, and fixed them to 0.01 for all the experiments. The number of atoms in the dictionary was set to 10 times the number of targets for the first two settings and 10 times the window size for the last setting, respectively. The trade-off parameter λ was selected to be 0.5 for updating the dictionary with tracking results, and 0.05 for updating the dictionary with new tracking targets.

4.5.3 Evaluation for Microblog Tracking

Since the problem of microblog tracking has never been investigated previously, we borrow the ideas and methods proposed for topic tracking, and apply these methods to our scenario. We compare our method (denoted as EDL) with the following baseline methods:

- Static method (SM). We adopt this method as the most basic baseline. In this method, each tracking target is expressed by the initial microblog, and is kept fixed during the whole tracking procedure. A new microblog will be assigned to a target if their similarity exceeds the threshold.

- Centroid-based method (CB). Different from the static method, the expression of the tracking target will be updated dynamically with the centroid of the tracking results during the tracking procedure. The tracking procedure is similar to the static method.

- k Nearest Neighbor [137] (kNN). For this method, a k-nearest-neighbor classification model is constructed for each tracking target to track relevant documents.

- Relevance model [71] (RM). This method uses relevance modeling to enhance the language model estimate associate with a topic. Similar to query expansion, relevance modeling expands the set of words associated with a topic to include other strongly related words.

The results for *Setting 1* and *Setting 2* are shown in Figure 4.2 and Figure 4.3, respectively. From the result, we can observe that our method consistently outperforms the comparing baseline methods for different number of tracking targets. Generally speaking, with an increasing number of targets to be

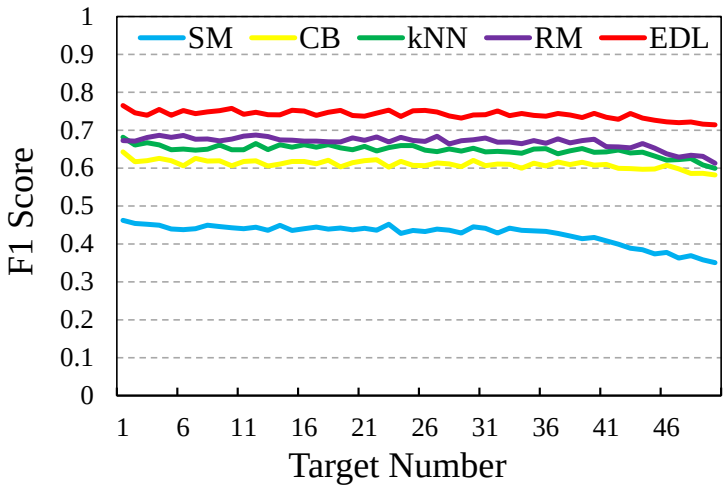


Figure 4.2: F1 score of our method (EDL) and the comparing methods for microblog tracking under *Setting 1*.

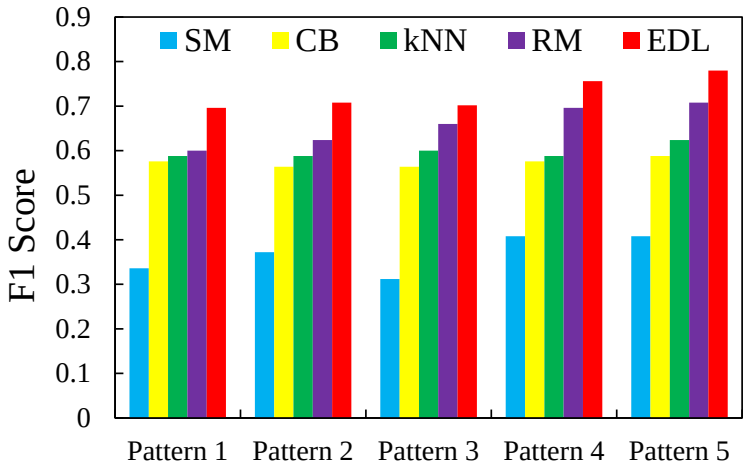


Figure 4.3: F1 score of our method (EDL) and the comparing methods for microblog tracking under *Setting 2*.

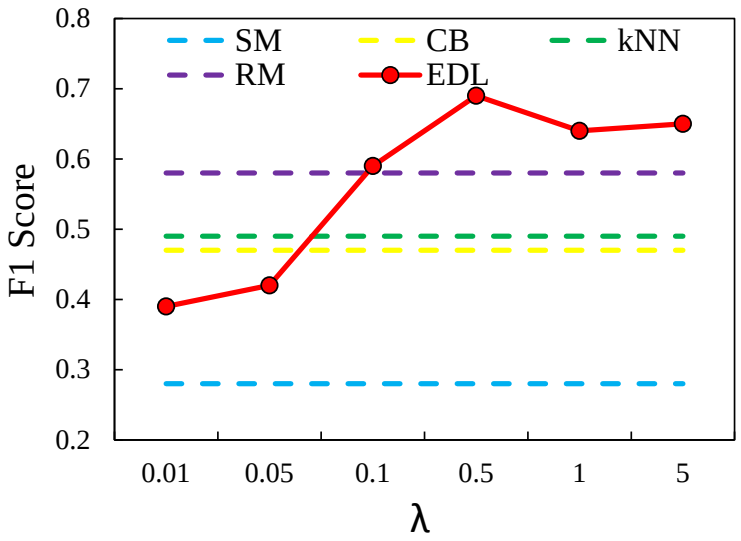


Figure 4.4: F1 score of our method (EDL) with various λ value for microblog tracking under *Setting 3*. The dashed lines represents the results of the comparing methods when the window size is fixed to 5.

tracked at the same time (*Setting 1*), or if the tracking targets are more centralized in the time dimension, the tracking performance will degrade. The comparing methods are target-specific: each target will have an individual model for tracking. Therefore, the degradation is mainly caused by the miss classification. Which means, for two similar targets, if their timestamps are very close, then the relevant results may be well-separated. However, if they are far from each other in the time dimension, the previous target will have more time to learn enough discriminative representation, then the tracking method will not get confused between these two targets. While for our method, the decrease of the accuracy is mainly due to the representation ability of the dictionary. Compared with the centralized pattern that introduces many targets at the same time, the decentralized pattern leaves more time for the dictionary to fully learn the representation for every target, and the dictionary can also evolve more smoothly.

In Figure 4.4, the results of our method with various λ values are presented. Beside, this figure also shows the results of the comparing methods with window size fixed to 5. Recall that *lambda* is used as a trade-off parameter to balances the present and past information: it quantifies the extent to which the model past oriented ($\lambda \to 1$) or present oriented ($\lambda \to 0$). From the results, we can observe that under a fixed small window size, the performance of the comparing methods drops a lot. However, by choosing a suitable value for λ, our method can outperforms the baseline methods, thus addressing the problems introduced by fix window size in the real scenario.

4.5.4 Evaluation for Early-Stage Detection of Viral Topics

We compare our method with the following topic detection algorithms:

- Twitter Monitor [82] (TM). This system is originally designed for trending topic detection over the Twitter stream. It first identifies the keywords that suddenly appear in tweets at an unusually high rate. These keywords are subsequently grouped into trends based on their co-occurrences.

- Dictionary learning [64] (DL). This method is also based on dictionary learning. The overall framework is divided into two stages: determining novel documents from the stream first, then identifying cluster structure among the novel documents subsequently. The difference between this method and our method is that the dictionary learning is used to identify novel documents (documents that are dissimilar to previous documents) in this method, while in our framework, the dictionary learning is used for

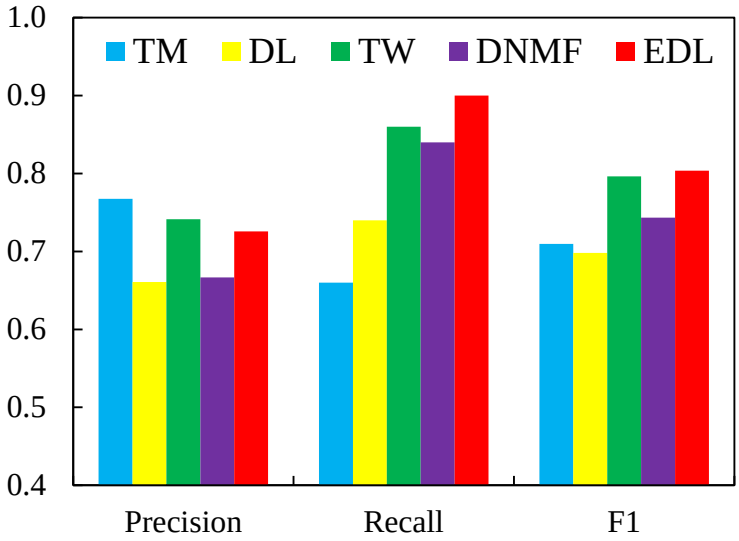


Figure 4.5: Precision, recall and F1 score of all methods for topic detection.

tracking existing results. Besides, our method also adopts an evolutional manner to manage the tradeoff between history and current data.

- Term weighting schemes [101] (TW). This method proposes a new term weighting scheme which models the sparse aspect, global weight and local weight of each topic. Based on this new weighting scheme, the incremental clustering algorithm is applied to detection emerging events.

- Dynamic NMF [106] (DNMF). This is an online nonnegative matrix factorization framework under a temporal regularization. The temporal regularization is formulated by chaining together trend extraction with a margin-based loss function to penalize static or decaying topics.

The results of all the methods in terms of Precision, Recall and F1 score are shown in Figure 4.5. The results in this figure represents the ability of a topic detection system in discovering trending topics. While Figure 4.6 shows the MicroPrecision, MicroRecall, and MicroF1 score for all the methods. Compar-

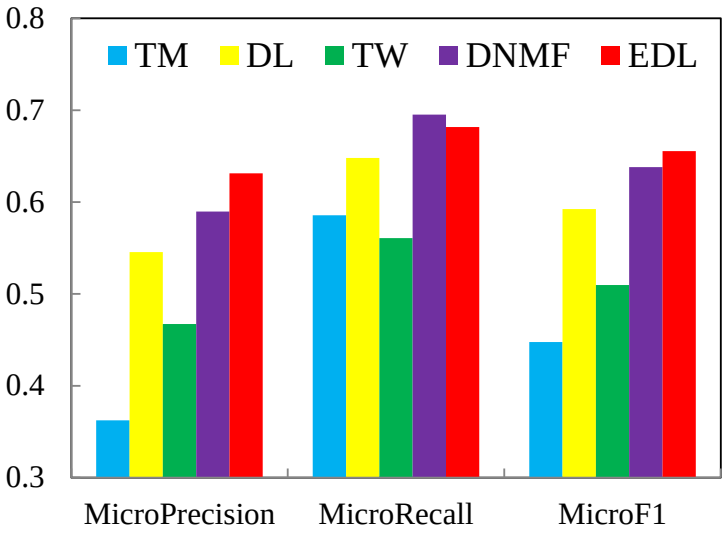


Figure 4.6: MicroPrecision, MicroRecall, and MicroF1 score of all methods for topic detection.

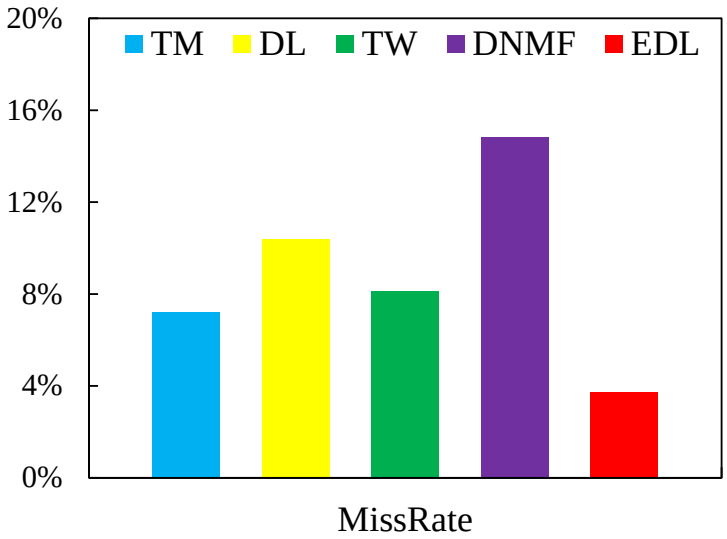


Figure 4.7: MissRate of all methods for topic detection.

ing to F1, Micro F1 is a fine-grained criterion to evaluate the performance of a topic detection system. It reflects how many of the microblogs corresponding to the trending topics can be identified. From the results, we can observe that our method outperforms the comparing methods in terms of both F1 and MicroF1. Considering F1, the term weighting schemes (TW) performs nearly as good as our method. However, the results are not comparable in terms of MicroF1, which means although TW can discover the topics, it fails to find relevant microblogs. The reason is that as a keyword-based method, it discovers relevant microblogs by searching keywords. Therefore, although this method performs well for finding informative keywords, the MicroPrecision of this keyword-based method will be very low.

The focus of our system is prediction: we would like to detect a trending topic in the very early stage. Therefore, how fast the topic can be detected is a crucial measurement for a prediction system. Since the outbreak speed of different topics may be quite different from each other, directly evaluate through time is not a good choice. The MissRate, however, can overcome this problem. The MissRate of all methods are listed in Figure 4.7, where a lower MissRate indicates that this system can detect the topics earlier. From the results, we can observe that our framework performs significantly better than the comparing methods. The keyword-based methods, TM and TW, are much faster than the rest two methods. However, all of these methods need to observe enough data to increase the confidence to report a new topic (early report will generates too many topics, resulting in very low precision), which will inevitably slow down the detection procedure. While our framework consists a stage to predict viral microblogs in the first step. This step provides valuable prediction information, therefore the following topic detection stage no longer needs a high threshold

to ensure the detection precision. In conclusion, our framework can provide the topic detection results in a very early stage, while ensuring a satisfactory detection performance.

4.6 Summary

In this Chapter, we introduced the problem of microblog tracking. Similar to the need for topic tracking in traditional media, there also exists the demand for tracking individual microblog in microblog streams. Given a set of microblog posts as targets, we designed a method to jointly tracking all of the targets in a unified framework. Based on sparse coding, we proposed an evolutional dictionary learning framework to learn a dictionary of atoms as a compact summary representation for all the tracking targets, with which the corresponding microblogs can be approximately represented by a linear combination of a few atoms. Taking the evolution of the tracking contents into consideration, a dictionary-transition matrix is introduced to capture the relationship between the dictionaries of two adjacent time. Therefore, the proposed method is capable of balancing the current batch of data and the accumulated history. Experiments on real-world dataset were conducted to demonstrate the effectiveness of the proposed framework for the following two tasks: 1) directly applying the method for tracking microblogs with various experiment settings to simulate various scenarios, and 2) combining this component with the viral microblog prediction method proposed in Chapter 3, integrating into a unified framework for early-stage detection of viral topics in microblog stream.

CHAPTER 5

Multimedia Topic Summarization

5.1 Introduction

After the previous two stages, the existence of viral topics can be predicted. At this moment, however, a viral topic exists in the form of a collection of many related microblog posts. Although the related posts collection can provide cues of the existence of this viral topic, this form of information is not user-friendly, as it leaves too many details for the readers to browse. Without effective summarization mechanism, the users are often confronted with incomplete, irrelevant and duplicate information, which makes it difficult to capture the essence of the topic and possible to miss information of a valuable direction. Therefore, it would be of great benefit if an effective mechanism can be provided for summarizing the detected viral topics. In this chapter, we focus on the step after viral topic detection: given the microblog posts related to a predicted viral topic, we

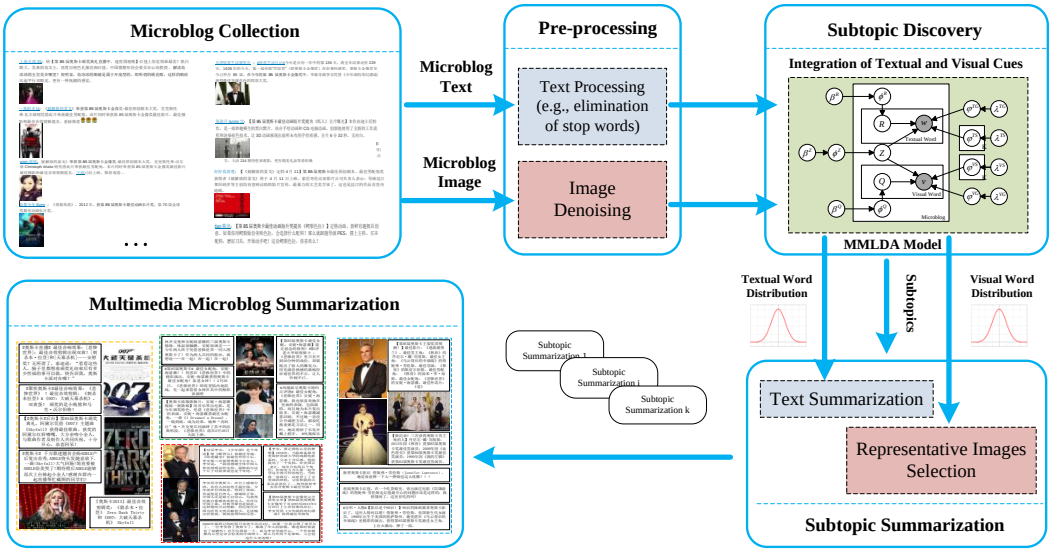


Figure 5.1: Flowchart of the proposed multimedia topic summarization framework.

target at mining the different divisions under this topic (denoted as subtopics), as well as summarizing the multimedia contents under these subtopics precisely and concisely.

It is non-trivial to integrate textual and visual information to generate comprehensive summaries in the circumstance of microblogs due to the following critical challenges. 1) As observed in the crawled microblogs from Sina Weibo, the inconsistency of textual part and visual part in the same microblog is very common. Therefore, we are faced with irrelevant microblog images in the input data stream. For instance, according to our statistics, the percentage of relevant images in our *Social Trends* datasets is only 67.1% (refer to Section 5.4 for details). Directly utilizing such noisy images may severely degrade the performance of subtopic discovery and summarization. 2) If the intrinsic correlations between textual and visual information are not well explored under the circumstance of microblogs, they may exert negative influence on each other.

Based on the above analysis, we propose a novel multimedia topic summarization framework to generate a holistic visualized summary from the microblogs with multiple media types. The flowchart of the proposed multimedia topic summarization framework is illustrated in Figure 5.1. Specifically, the proposed framework comprises three stages: removal of irrelevant data, cross-media subtopic discovery and multimedia summary generation. First, we devise a data cleansing approach to automatically eliminate those irrelevant/noisy images. In the second stage, we propose a novel cross-media probabilistic model, termed *Cross-Media-LDA* (CMLDA), to jointly exploit the microblogs of multiple media types for discovering subtopics. The CMLDA model not merely well explores and exploits the intrinsic correlations among different media types, but also simultaneously characterizes both the general distribution and the subtopic specific distribution from the microblog data of various media types for reinforcing the subtopic discovery process. Besides, this step could also handle the noise of the input data, and remove those microblog examples from the next summarization step. Finally, based on the cross-media distribution knowledge of all the discovered subtopics, we generate a holistic visualized summary for the topics by pinpointing both the representative textual and visual samples in a joint fashion. In particular, by utilizing the cross-media distributions of microblog text, we specify three criteria, namely coverage, significance and diversity to measure the summarization capability of individual textual samples. We then devise a greedy algorithm for identifying the representative microblog texts based on the combination of the three criteria. For visual summarization, we employ the cross-media knowledge of the subtopics as the prior knowledge for ranking the visual samples and selecting the most representative ones. In order to improve the descriptive power and the diversity of viewpoints, we

first partition the images within a subtopic into groups via spectral clustering. Then, for each group we apply a manifold algorithm with the cross-media prior knowledge as initial ranking scores to identify the top-ranked image as representative. It is remarkable that both the textual and visual summarization processes utilize the cross-media knowledge of the discovered subtopics and thus are intrinsically connected to reinforce each other.

5.2 Problem Definition

Suppose we are given a microblog stream $\mathcal{M} = \{M_1, ..., M_{|\mathcal{M}|}\}$ related to the same topic $\mathcal{T}$, which can be either provided by any topic service of online microblog platform or detected by any topic detection method. Each microblog $M_i = \{T_i, I_i\}$ consists of two components: textual component T_i and visual component I_i. Note that I_i may be empty, which means M_i contains no visual sample. $|\cdot|$ denotes the cardinality of a set. The objective of our framework is to automatically generate a multimedia summary (i.e., both textual and visual) from the microblog collection $\mathcal{M}$ for revealing multiple subtopics of the topic $\mathcal{T}$. For topic $\mathcal{T}$, we define its topic-level summary as the union of all its subtopics' summaries. For each subtopic, a subtopic-level summary comprises both textual and visual exemplars selected from $\mathcal{M}$.

5.3 Multimedia Topic Summarization

In this section, we elaborate the details of the proposed multimedia topic summarization framework, including the removal of irrelevant data, the cross-media subtopic discovery and the multimedia summary generation.

5.3.1 Removal of Irrelevant Data

As a kind of user-generated content, the quality of microblogs cannot be guaranteed. It has been observed that many microblog images are irrelevant to their corresponding texts (e.g., spam images). Directly applying our framework on such noisy image set may severely degrade the performance of the summarization. Since the input microblog collection is gathered with text-based methods, the problem of noisy images is more severe than that of noisy texts. Therefore, it is necessary to first pre-filter microblog images to eliminate those noisy images. For the problem of noisy texts, it will be addressed in the following subtopic discovery procedure.

Specifically, we develop the noise removal procedure by exploiting a spectral filtering model [78]. Without loss of generality, suppose we have n microblog images $X = \{x_1, x_2, ..., x_n\}$ corresponding to all the non-empty images I_i of the given topic $\mathcal{T}$, where $x_i \in \mathbb{R}^d$ and d is the dimension of visual space. We first build a neighborhood graph $G = (V, E, W)$, where V is a vertex set composed of n vertices representing our n images in X, $E \subseteq V \times V$ is an edge set connecting neighboring vertices, and $W \in \mathbb{R}^{n \times n}$ is a weighting matrix measuring the strength of the edges, i.e., the similarity between two data points. There are various methods to compute W. In this work, we adopt the widely used k-Nearest-Neighbors similarity graph:

$$W_{ij} = \begin{cases} \exp(-\dfrac{d(x_i, x_j)^2}{\sigma^2}), & \text{if } x_i \in \mathcal{N}_k(x_j) \text{ or } x_j \in \mathcal{N}_k(x_i) \\ 0 \qquad\qquad , & \text{otherwise,} \end{cases}$$

where $d(\cdot, \cdot)$ is a distance measure such as Euclidean distance, and σ is the bandwidth parameter. $\mathcal{N}_k(x_i)$ denotes the set of k nearest neighbors of x_i in

X. We further define D as a degree matrix whose diagonal elements $D_{ii} = \sum_{j=1}^{n} W_{ij}$. With D and W, the normalized graph Laplacian is defined as

$$L = I - D^{-1/2} W D^{-1/2}.$$

As previously discussed, an intuition is that images depicting the same subtopic should be visually similar to each other. Therefore, it is reasonable to assume that the truely relevant images should reside in multiple high-density regions, while the irrelevant images will present a more random distribution. It has been demonstrated [114] that when data points have formed clusters, each high density region implicitly corresponds to certain low-frequency (smooth) eigenvector. The data points which belong to the region will take relatively large absolute values corresponding to the eigenvector, while for data points elsewhere, the values are close to zero. With this assumption, we can exploit the spectrum of the kNN similarity graph G, which is a set of eigenvalue/eigenvector pairs $\{\lambda_i, \mathbf{u_i}\}_{\mathbf{i=1}}^{\mathbf{n}}$ of the normalized graph Laplacian L to find the high density regions. For simplicity, we assume that the eigenvalues are sorted in a nondecreasing order, thus the top eigenvectors have the lowest frequency.

Let $\mathbf{y} \in \mathbb{R}^{n \times 1}$ be a label vector indicating the relevance of each image to the given topic. Ideally, $\mathbf{y}$ takes the value of 1 for all relevant images and 0 for noisy ones. Consider the top m smoothest eigenvectors $\mathbf{u_2}, .., \mathbf{u_{m+1}}$ as eigenbases ($\mathbf{u_1}$ is eliminated because it is nearly constant and $\lambda_1 = 0$ when the graph is connected, thus does not form any region). According to the multi-region assumption, $\mathbf{y}$ should lie in the subspace spanned by these eigenbases. Let $U = [\mathbf{u_2}, ..., \mathbf{u_{m+1}}] \in \mathbb{R}^{\mathbf{n \times m}}$, $\Lambda = diag(\lambda_2, \lambda_3, ...\lambda_{m+1})$ and $\mathbf{y} = \mathbf{1}$ initially. The spectral filter reconstructs the noisy label vector $\mathbf{y}$ with the sparse eigenbases

U by solving the following problem:

$$\min_{\mathbf{z}\in\mathbb{R}^{\mathbf{m}}} \|U\mathbf{z}-\mathbf{y}\|^2 + \alpha_1\|\mathbf{z}\|_1 + \alpha_2\mathbf{z}^{\mathbf{T}}\Lambda\mathbf{z} \tag{5.1}$$

where $\mathbf{z}$ is the sparse coefficient vector, $\|\mathbf{z}\|_1 = \sum_{j=1}^{m}|z_j|$ is the ℓ_1-norm. α_1 and α_2 are two regularization parameters. Note that the last term $\mathbf{z}^T\Lambda\mathbf{z} = \sum_{i=1}^{m}\lambda_{i+1}{z_i}^2$, which is actually a weighted ℓ_2-norm, imposes that smoother eigenbases with smaller eigenvalues are preferred in the reconstruction of $\mathbf{y}$.

Once the solution $\mathbf{z}$ to Eq. (5.1) is obtained, the truely relevant label vector $\hat{\mathbf{y}}$ is set to $round(U\mathbf{z})$, where the function $round(\cdot)$ is defined as follows:

$$(round(\mathbf{x}))_{\mathbf{i}} = \begin{cases} 1, & x_i \geq \theta\cdot\max\{\mathbf{x}\} \\ 0, & \text{otherwise} \end{cases} \tag{5.2}$$

where θ is the threshold indicating the confidence level of a data to be regarded as relevant. With the final label vector $\hat{\mathbf{y}}$, we eliminate those image samples with $\hat{y}_i = 0$, and retain a more reliable image set.

5.3.2 Cross-media Subtopic Discovery

In this subsection, we propose a novel cross-media probabilistic model, termed *Cross-Media-LDA* (CMLDA), to discover subtopics by jointly exploring the intrinsic correlation between the textual and visual aspects of microblogs. The CMLDA model substantially characterizes the multiple facets of the topic by exploring two underlying properties of various media types, i.e., *inter-media consistency* and *intra-media discrimination*. Besides, this model is also capable of eliminating possible noisy microblog posts from the data collection gathered

for the following summarization process.

Inter-media consistency

It is observed that the microblogs associated with the same topic contain various inter-correlated media types, such as texts and images. If we can properly capture and model the intrinsic correlations among these media types, we may achieve a better understanding of the topic. Intuitively, different media types of the same topic should be related to certain common topics or share some common high-level semantics. In other words, the semantics should be consistent across different media types. Based on this analysis, we model the common semantics shared among different media types via a subtopic indicator Z, which is able to jointly generate both the textual words and visual words in the microblogs. With the cross-media subtopic indicator, we manage to capture the inter-media correlations for effective subtopic discovery. It is worth noting that while the traditional latent dirichlet allocation (LDA) [15] model assigns multiple topics to each individual document and one topic for each word, the proposed CMLDA model is designed to associate only one topic (subtopic) with each individual microblog. The underlying reason is that microblog content is usually short and focused, and thus it is reasonable to assume that each microblog is related to only one subtopic.

Intra-media discrimination

Within each individual media type, it is non-trivial to directly employ traditional topic modeling approach (e.g., LDA) to discover the subtopics within the same topic because the semantics of different subtopics may be heavily overlapped, while we target at discovering the discriminative knowledge of each

subtopic.

Normally, we may assume that all subtopics of the same topic share certain general words indicating common semantics related to the topic; while each individual subtopic uniquely possesses certain specific semantics, which distinguishes itself from other subtopics. Take "Lushan Earthquake" as an example, "earthquake", "Lushan" and "death" are more likely to be general words; while words like "hypocenter", "collapse" and "Premier" are more probable to appear in different subtopics. If the proportion of general contents is large, then they may dominate the result. In order to exclude the influence of general contents and discover discriminative cues for each subtopic, two new latent variables R and Q are introduced to guarantee *intra-media discrimination* in the generation of textual and visual words, respectively. For each textual (visual) word, R (Q) indicates whether it is generated from the general distribution or from the specific distribution corresponding to its subtopic.

CMLDA modelling and inference

In this part, we elaborate the details of the modelling and inference processes of the CMLDA model. Figure 5.2 illustrates the graphical model representation, and the generation process is as follows:

1. For the topic $\mathcal{T}$, draw $\varphi^{TG} \sim Dir(\lambda^{TG})$ and $\varphi^{VG} \sim Dir(\lambda^{VG})$, indicating the general textual and visual distribution, respectively. Then draw $\phi^{Z} \sim Dir(\beta^{Z})$, which indicates the distribution of subtopics over the microblog collection corresponding to $\mathcal{T}$.

2. For each subtopic, draw $\varphi_k^{TS} \sim Dir(\lambda^{TS})$ and $\varphi_k^{VS} \sim Dir(\lambda^{VS})$, $k \in \{1, 2, ..., K\}$, corresponding to the specific textual and visual distribution.

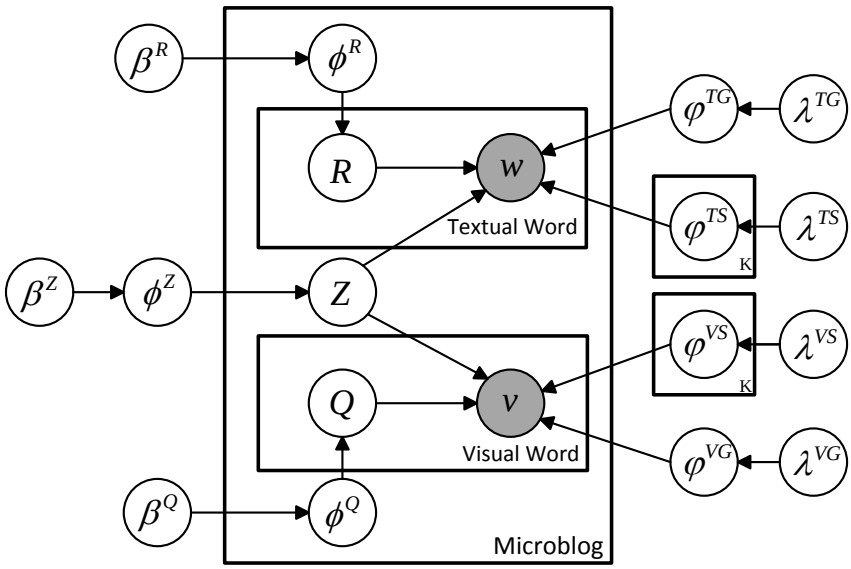


Figure 5.2: Graphical model representation of the CMLDA model.

3. For each microblog M_i, draw $Z_i \sim Multi(\phi^Z)$, corresponding to the subtopic assignment for M_i. Then draw $\phi_i^R \sim Dir(\beta^R)$, indicating the general/specific textual word distribution of M_i. Similarly, draw $\phi_i^Q \sim Dir(\beta^Q)$.

4. For each textual word position of M_i, draw a variable $R_{ij} \sim Multi(\phi_i^R)$:

   - If R_{ij} indicates *General (G for short)*, then draw a word $w_{ij} \sim Multi(\varphi^{TG})$.
   - If R_{ij} indicates *Specific (S for short)*, then draw a word w_{ij} from the Z_i-th specific distribution $w_{ij} \sim Multi(\varphi_{Z_i}^{TS})$.

5. The generation of visual words is similar to step 4.

In the CMLDA model, the subtopic indicator Z as well as the general/specific indicator R and Q are latent variables to be inferred from observations, i.e., textual and visual words. We use Gibbs sampling to achieve the inference due to

its efficiency and effectiveness in handling high-dimensional data. The update rules for latent variables are shown as follows:

$$P(R_{ij} = S|Z_i, w_{ij}, \cdots) \propto \frac{N^R_{i,\neg j}(S) + \beta^R_S}{N^R_{i,\neg j} + \beta^R_G + \beta^R_S} \times \frac{N^{TS}_{Z_i,\neg j}(w_{ij}) + \lambda^{TS}}{\sum_{t\in V^t}\left(N^{TS}_{Z_i,\neg j}(t) + \lambda^{TS}\right)}$$

$$P(R_{ij} = G|w_{ij}, \cdots) \propto \frac{N^R_{i,\neg j}(G) + \beta^R_G}{N^R_{i,\neg j} + \beta^R_G + \beta^R_S} \times \frac{N^{TG}_{\neg j}(w_{ij}) + \lambda^{TG}}{\sum_{t\in V^t}\left(N^{TG}_{\neg j}(t) + \lambda^{TG}\right)}$$

$$P(Q_{ij} = S|Z_i, v_{ij}, \cdots) \propto \frac{N^Q_{i,\neg j}(S) + \beta^Q_S}{N^Q_{i,\neg j} + \beta^Q_G + \beta^Q_S} \times \frac{N^{VS}_{Z_i,\neg j}(v_{ij}) + \lambda^{VS}}{\sum_{u\in V^v}\left(N^{VS}_{Z_i,\neg j}(u) + \lambda^{VS}\right)}$$

$$P(Q_{ij} = G|v_{ij}, \cdots) \propto \frac{N^Q_{i,\neg j}(G) + \beta^Q_S}{N^Q_{i,\neg j} + \beta^Q_G + \beta^Q_S} \times \frac{N^{VG}_{Z_i,\neg j}(v_{ij}) + \lambda^{VG}}{\sum_{u\in V^v}\left(N^{VG}_{Z_i,\neg j}(u) + \lambda^{VG}\right)}$$

$$P(Z_i = k|R_{i\cdot}, Q_{i\cdot}, w_{i\cdot}, v_{i\cdot}, \cdots) \propto \frac{N^Z_{\neg i}(k) + \beta^Z}{\sum_{l=1}^{K}\left(N^Z_{\neg i}(l) + \beta^Z\right)} \times \prod_{Q_{ij}=S} \frac{N^V_{k,\neg i}(v_{ij}) + \lambda^V}{\sum_{u\in V^v}\left(N^V_{k,\neg i}(u) + \lambda^V\right)} \times \prod_{R_{ij}=S} \frac{N^S_{k,\neg i}(w_{ij}) + \lambda^S}{\sum_{t\in V^t}\left(N^S_{k,\neg i}(t) + \lambda^S\right)}$$

where V^t and V^v denote the textual and visual vocabulary, respectively. The variables with subscript i are corresponding to the i-th microblog M_i, while subscript j correspond to the j-th textual/visual word. $N(\cdot)$ stores the number of samples satisfying certain requirements during the iterative sampling process. For example, $N^{TS}_{k,\neg i}(t)$ represents the number of word t (excluding the words in M_i) in the k-th specific textual distribution.

After Gibbs sampling, we obtain the latent variables. Besides, K specific distributions φ^{TS} and φ^{VS} can also be easily computed. For a textual word

$w \in V^t$, $\varphi_k^{TS}(w)$ measures the probability of w appearing in the k-th specific distribution. It is similar for visual distribution $\varphi_k^{VS}(u)$. Therefore, they can be evaluated as follows:

$$\varphi_k^{TS}(w) = \frac{N^w(Z = k, R = S) + \lambda^{TS}}{\sum_{t \in V^t} (N^t(Z = k, R = S) + \lambda^{TS})} \tag{5.3}$$

$$\varphi_k^{VS}(u) = \frac{N^u(Z = k, Q = S) + \lambda^{VS}}{\sum_{u \in V^v} (N^u(Z = k, Q = S) + \lambda^{VS})} \tag{5.4}$$

With the CMLDA model, textual and visual components will facilitate each other to discovery the cross-media knowledge of the subtopics hidden in the topic. The obtained textual/visual distribution pair (φ^{TS},φ^{VS}) depicts the discriminative multimedia cues for each subtopic. According to the subtopic indicator Z for each microblog, the CMLDA model partitions the microblog collection $\mathcal{M}$ into K subsets $\{\mathcal{S}_k\}|_{k=1}^K$ corresponding to K subtopics where each subset contains both textual part $\mathcal{S}_k^t$ and visual part $\mathcal{S}_k^v$. Intuitively, if a subtopic contains a small number of textual or visual samples, the topic of this subtopic may not be important or related to the topic. We argue that such subtopics are probably composed of those noisy microblogs and should be removed. In our work, we remove all subsets whose sizes are smaller than $\epsilon \times |\mathcal{M}|$, where ϵ is the threshold. In the following subsection, we will employ the cross-media knowledge achieved with CMLDA for the summarization.

5.3.3 Multimedia Summary Generation

In this subsection, we explore how to utilize the cross-media distribution knowledge of all the discovered subtopics to facilitate the generation of the holistic

visualized summary with various media types for topics.

Cross-media summarization for microblog texts

In this part, we propose a method for text summarization based on the cross-media distribution information inferred from both the textual and visual aspects of the microblogs in the subtopic discovery procedure. Specifically, a greedy algorithm is developed to sequentially select representative samples based on a novel selection criterion, which takes three fundamental requirements into consideration:

Coverage. Intuitively, if a summary is able to well "cover" the information of its corresponding subtopic, then the word distributions over both of them should be close to each other. We use the similarity of word distributions over a summary and its corresponding subtopic for measuring coverage. Denote $\mathcal{G}_k$ as the current summary set consisting of the selected samples, then the word distribution over $\mathcal{G}_k$, denoted as $\Theta_{\mathcal{G}_k}$, can be estimated as:

$$p(w|\Theta_{\mathcal{G}_k}) = \frac{tf(w, \Theta_{\mathcal{G}_k})}{\sum_{t \in V^t} tf(t, \Theta_{\mathcal{G}_k})}, \ \forall w \in V^t, \tag{5.5}$$

where $tf(w, \Theta_{\mathcal{G}_k})$ denotes the term frequency of word w in $\mathcal{G}_k$. We use φ_k^{TS} as the word distribution over the corresponding subtopic, which is the distribution estimated in the learning process of CMLDA model (Eq. (5.3)). We employ Kullback-Leibler (KL) divergence to measure the distance of two distributions D_1 and D_2:

$$D_{KL}(D_1 \parallel D_2) = \sum\nolimits_w p(w|D_1) log \frac{p(w|D_1)}{p(w|D_2)} \tag{5.6}$$

Given the current summary set $\mathcal{G}_k$, the new sample T_i to be selected should be the one which makes the new summary (i.e., $\mathcal{G}_k \cup \{T_i\}$) achieve the best

coverage (i.e., minimize the distance between $\Theta_{\mathcal{G}_k \cup \{T_i\}}$ and φ_k^S). Therefore, the coverage of each candidate T_i could be measured by the following criterion:

$$\mathcal{U}_C(T_i) = D_{KL}(\Theta_{\mathcal{G}_k \cup \{T_i\}} \parallel \varphi_k^{TS}) \tag{5.7}$$

Significance. In the circumstance of microblogging, each microblog can propagate between users by the repost action. In general, the popularity of a microblog can be revealed from the repost number. A large repost number implies that the microblog attracts a lot of attention and interest from other users, and hence indirectly represents the importance of this microblog. The users will be more satisfied if more of these hot microblogs are shown in the summary. Therefore, we use a smooth function over the repost number to measure the significance of a candidate:

$$\mathcal{U}_S(T_i) = log(RepostNum(T_i) + 1) \tag{5.8}$$

Diversity. The information diversification is favored in the final summary. We take the information redundancy into consideration for sample selection. Consider a candidate T_i, the redundancy it brings to the summary set can be measured by the similarity between this candidate and the previously generated summary, which is:

$$\mathcal{U}_D(T_i) = D_{KL}(\Theta_{T_i} \parallel \Theta_{\mathcal{G}_k}) \tag{5.9}$$

Overall Selection Score. The overall selection score is defined as a weighted linear combination of the scores of coverage, significance and diversity. Since small distance $\mathcal{U}_C(T_i)$ indicates high coverage, we compute the overall selection

score as:

$$\mathcal{U}(T_i) = \omega_1 (1 - \mathcal{F}(\mathcal{U}_C(T_i))) + \omega_2 \mathcal{F}(\mathcal{U}_S(T_i)) + \omega_3 \mathcal{F}(\mathcal{U}_D(T_i))$$

where $\omega_1, \omega_2, \omega_3$ are trade-off parameters with $\sum_i \omega_i = 1$. $\mathcal{F}(x) = 1/(1 + exp(-x))$ is a logistic increasing function for normalizing all the scores to the interval $[0, 1]$.

With the above selection score for all the microblog samples, we may derive a greedy algorithm for representative sample selection. In each iteration, we select the one with the largest score from all the remaining samples.

Cross-media summarization for microblog images

Consider the visual subset $\mathcal{S}_k^v$, which contains all images related to the k-th subtopic. The objective of this step is to employ the cross-media knowledge of the discovered subtopics to reinforce the selection of the most representative image samples. The selected images should provide enough visually descriptive power as well as diverse viewpoints. We develop a two-step approach to automatically select representative images satisfying the above two criteria. We first partition the images within a subtopic into groups via spectral clustering. Then, for each group we apply a manifold algorithm with the cross-media prior knowledge as initial ranking scores to identify the top-ranked image as representative.

Clustering step. With the similarity matrix W previously constructed in the step of the noise removal, the similarity matrix for $\mathcal{S}_k^v$ can be directly obtained by extracting the columns and rows corresponding to images in $\mathcal{S}_k^v$, i.e., $W^k = [W_{ij} | I_i, I_j \in \mathcal{S}_k^v]$. Then normalized cut is applied to the image set,

and visual diversity is achieved across clusters.

Ranking step. In order to discover images with best representative ability within each cluster, we adopt manifold ranking algorithm to rank the images. Let $\mathbf{r}$ denote the vector of ranking score, manifold ranking defines an iterative update process as follows:

$$\mathbf{r^{t+1}} = \gamma \mathbf{D^{-1/2}} \mathbf{W} \mathbf{D^{-1/2}} \mathbf{r^t} + (\mathbf{1} - \gamma)\mathbf{h} \tag{5.10}$$

where $\mathbf{h}$ represents the vector of initial ranking scores, which is an all-one vector in standard manifold ranking setting. However, in our scenario, we expect $\mathbf{h}$ to possess the prior knowledge of the importance of each image. Recall that with CMLDA model, we have achieved the discriminative visual information for this subtopic, which is φ_k^{VS}. Intuitively, if an image is more consistent with φ_k^{VS}, it would have better descriptive ability for the whole subtopic image set, and should gain more emphasis. Therefore, instead of all-one vector which takes equal weighting for all images, we express $\mathbf{h}$ as prior knowledge measured by the KL divergency of an image I_i and φ_k^{VS}, i.e., $h_i = 1 - \mathcal{F}(D_{KL}(I_i \parallel \varphi_k^{VS}))$. By integrating the prior knowledge in the ranking scheme, the descriptive ability for the cluster as well as for the subtopic image set are both taken into consideration. Note that $\mathbf{r}$ has a closed form when the update process converges:

$$\mathbf{r} = (\mathbf{1} - \gamma)(\mathbf{I} - \gamma \mathbf{D^{-1/2}} \mathbf{W} \mathbf{D^{-1/2}})^{-1} \mathbf{h} \tag{5.11}$$

Finally, the image with the largest ranking value in $\mathbf{r}$ is selected from each cluster to construct the visual summarization set.

5.4 Experiments

5.4.1 Dataset and Experimental Settings

We conducted the evaluation of our framework on two datasets that were collected by ourselves: (1) ***Social Trends***, which include 20 trending topics that were listed as hot trends in February 2013 by *Sina Weibo*. For each trending topic, we crawled the related microblogs in the life cycle of this topic using the trending topic API provided by Sina Weibo. The total number of microblogs is 310,097, of which 114,426 contain image; (2) ***Product Topics***, which was collected by Gao *et al.* [45]. It includes 20 product-related topics and 13,932 microblogs, and 11,736 of them contain image. Due to limited information appended to repost action, only the original microblogs are included in our datasets. In order to evaluate the quality of the generated summaries, five volunteers were invited to manually generate a textual summary for each topic as golden standard individually. All volunteers are Chinese-speaking Weibo users, who are also very familiar with the collected topics. Each manually generated summary consists of 50 microblogs selected from the microblog datasets. In the following evaluation, we run all experiments five times, with each generated summary as the ground-truth. The final reported results are the average of five experiments.

In text pre-processing procedure, we first segmented Chinese words using IKAnalyzer[1], and then removed the stop words, low-frequency words with document frequency of less than 5 and mentions (@somebody) from textual vocabulary. Texts containing less than 3 words were also eliminated. For visual feature extraction, scale-invariant feature transform (SIFT) descriptors were

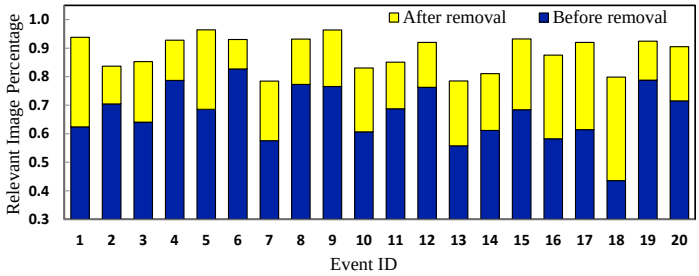
[1]http://code.google.com/p/ik-analyzer/

first extracted from each image. Then we trained a codebook of $1,000$ visual words with descriptors sampled from images of all topics. With the trained codebook, each descriptor was quantized into a visual word. Each image was further represented as a $1,000$-dimensional ℓ_2-normalized bag-of-visual-words feature vector.

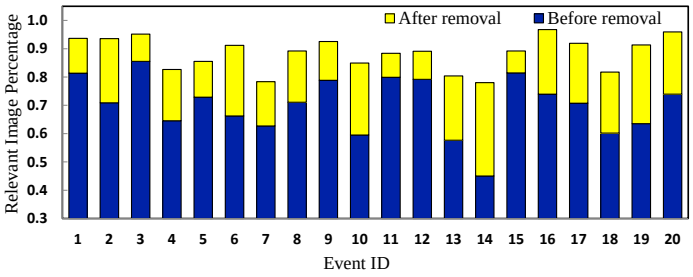
When constructing the image similarity graph, we set the number of nearest neighbors k to 20 and bandwidth parameter σ to 0.1. For the spectral filtering model in Eq. (5.1), we use $m = 40$ eigenbases for label vector reconstruction, and set $\alpha_1 = 0.1$, $\alpha_2 = 0.1$. For concentration parameters in CMLDA model, as stated in [55], the more specific a distribution is meant to be, the smaller its parameter. Accordingly, we set $\lambda^{TG} = 0.1$, $\lambda^{TS} = 0.01$, $\lambda^{VG} = 1$, $\lambda^{VS} = 0.1$, $\beta^R = 0.1$, $\beta^Q = 0.1$, and $\beta^Z = 1$. For the final representation image selection procedure, the parameter γ is set to 0.85. The threshold ϵ is set to $0.3 \times 1/K$ and all subtopics with size smaller than $\epsilon \times |\mathcal{M}|$ are removed. The total number of the selected microblogs is chosen to be 50, which is the same as the number of microblogs in the gold standards. The 50 microblogs quota are assigned to the remaining subtopics according to the proportion of microblog number in each subtopic.

5.4.2 Capability of Irrelevant Image Removal

We demonstrate the capability of our developed irrelevant image removal component in Figure 5.3. The percentage of relevant images before and after the removal procedure for each topic is listed. As aforementioned, the original image collections of all the topics contain many images that are irrelevant to the corresponding topic. The average percentage of relevant images is 67.1% and



(a) *Social Trends*



(b) *Product Topics*

Figure 5.3: Effects of irrelevant image removal.

Figure 5.4: Illustrative examples of removed images and those remained after irrelevant image removal.

69.9%, respectively, across all topics for the two datasets. We apply spectral filtering on the image collection of each individual topic separately. As shown in Eq. (5.2), one important factor which controls the performance of spectral filtering is the parameter θ of the $round(\cdot)$ function. There is a tradeoff between the performance of irrelevant image removal and the number of remaining images: in general, the higher the relevance percentage, the smaller the number of remaining images. In our framework, the quality of image collections is very crucial for the cross-media subtopic discovery and summarization. In our experiments, the controlling parameter θ is set to 0.5 for *Social Trends* dataset, which results in a relatively high relevance percentage (88.4%), as well as a reasonable number of images ($54,800$ images, or 51% of the original collection size). Similarly, we set θ to 0.6 for *Product Topics*, and achieved $6,570$ remaining images with 91.5% of them being relevant. On average, our proposed method improves the percentage of relevant images by around 21%. Figure 5.4 shows several examples of removed and remaining images for Topic #1 of *Social Trends* and Topic #13 of *Product Topics*. As can be clearly seen, for both topics, the exemplars with high rank orders are truely relevant to the corresponding topics while those images with low ranks are really noisy.

5.4.3 Summarization Performance

In this subsection, we evaluate the effectiveness of our proposed framework as compared to several summarization approaches. For fairness of evaluation, we select 50 microblogs for all the comparing approaches to form the summaries. For evaluation metric, we employ ROUGE evaluation toolkit [76] which automatically determines the quality of a summary as compared to human generated

golden standards. In particular, F-measure scores of ROUGE-1, ROUGE-2, ROUGE-W (with W set to 1.2) and ROUGE-SU are reported. Take ROUGE-N as an example. Denote the golden standards as GS, and the generated summary as S, ROUGE-N-Recall is an N-gram recall metric computed as follows:

$$\text{ROUGE-N-Recall} = \frac{\sum_{I \in GS} \sum_{\text{N-gram} \in I} Count_{match}(\text{N-gram})}{\sum_{I \in GS} \sum_{\text{N-gram} \in I} Count(\text{N-gram})}$$

and ROUGE-N-Precision is an N-gram precision matric as follows:

$$\text{ROUGE-N-Precision} = \frac{\sum_{I \in S} \sum_{\text{N-gram} \in I} Count_{match}(\text{N-gram})}{\sum_{I \in S} \sum_{\text{N-gram} \in I} Count(\text{N-gram})}$$

For the ROUGE-N value reported in our experimental results, we adopt the F score of the above recalled based and precision based metrics:

$$\text{ROUGE-N} = \frac{2 \times \text{ROUGE-N-Precision} \times \text{ROUGE-N-Recall}}{\text{ROUGE-N-Precision} + \text{ROUGE-N-Recall}}$$

We compare our proposal with the following multi-document summarization approaches:

- RANDOM: which selects all samples randomly.
- LSA [48]: which conducts SVD on sample by term matrix first and starting from most significant left eigenvector, and select samples with highest entry value.
- NMF [94]: which performs NMF on sample by term matrix and select samples best represent the discovered bases.
- SNMF [127]: which constructs the sample-sample similarity matrix first,

clusters all samples with Symmetric Non-negative Matrix Factorization (SNMF) and extracts centering sentences from the clusters.

- KMEANS [99]: which performs K-means clustering over the dataset, and samples nearest to cluster centers are selected.

- NCUT [113]: which is similar to KMEANS, while use normalized cut as clustering method.

Besides, the following text-based microblog summarization approaches are also compared:

- PR [112]: the Phrase Reinforcement algorithm, which generates summaries by looking for the most commonly occurring phrases.

- HTF-IDF [58]: which selects summary posts by their Hybrid TF-IDF weights, and filters redundant posts with similarity threshold.

- CLUSTER [58]: another method proposed by Inouye *et al.* [58]. Similar to the traditional clustering-based multi-document summarization approach, this method first conducts kmeans++ to cluster the data samples. When selecting summary posts from each cluster, the above HTF-IDF is utilized to assign weights to the samples.

For our proposed approach, two specific methods are evaluated for comparison:

- MMTS: the proposed multimedia topic summarization (MMTS) framework that uses both text and visual contents in building CMLDA model.

Table 5.1: Comparison among different summarization approaches on the *Social Trends* dataset. Average results of the 20 topics are reported for all evaluation measurement.

System	ROUGE-1	ROUGE-2	ROUGE-W	ROUGE-SU
RANDOM	0.2453	0.0759	0.0217	0.0693
LSA	0.4039	0.1562	0.0367	0.1400
NMF	0.3119	0.1010	0.0251	0.0860
SNMF	0.3243	0.1598	0.0294	0.1377
KMEANS	0.3279	0.0985	0.0285	0.0951
NCUT	0.3761	0.1385	0.0357	0.1336
PR	0.3333	0.1564	0.0287	0.1366
HTF-IDF	0.3478	0.1306	0.0347	0.1558
CLUSTER	0.3586	0.1807	0.0357	0.1225
MMTS-I	**0.4503**	**0.2419**	**0.0529**	**0.1761**
MMTS-R	**0.4793**	**0.2631**	**0.0589**	**0.1997**
MMTS	**0.5049**	**0.3076**	**0.0696**	**0.2356**

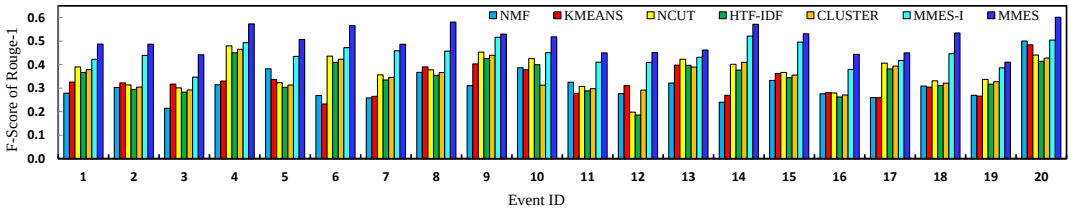
- MMTS-I: MMTS without utilizing the visual information. In the subtopic discovery stage, when applying CMLDA model, all microblog samples are assumed to be comprised of texts only.

- MMTS-R: MMTS without the process of irrelevant image removal, where the whole noisy image collections are used.

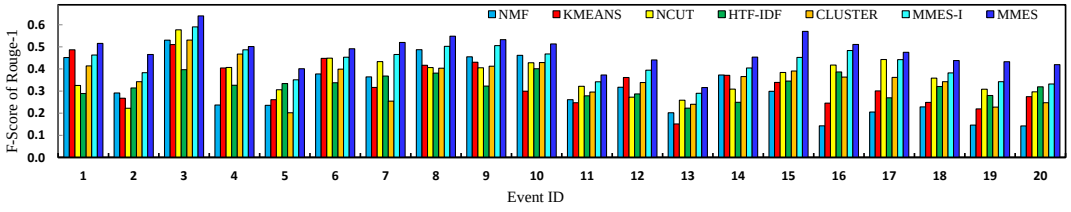
The overall comparison of proposed MMTS, MMTS-I and MMTS-R with other approaches are shown in Table 5.1 and Table 5.2. In addition, detailed ROUGE-1 performance for each topic is shown in Figure 5.5. For conciseness, only seven selected comparing methods are shown in the figure. As can be seen from the results, the proposed MMTS outperforms other methods for all topics as well as all evaluation measurements. The good performance of MMTS benefits from the following three aspects:

Table 5.2: Comparison among different summarization approaches on the *Product Topics* dataset. Average results of the 20 topics are reported for all evaluation measurement.

System	ROUGE-1	ROUGE-2	ROUGE-W	ROUGE-SU
RANDOM	0.2534	0.1210	0.0734	0.0741
LSA	0.3681	0.1886	0.1084	0.1265
NMF	0.3105	0.1546	0.0882	0.0923
SNMF	0.3079	0.1507	0.0888	0.0945
KMEANS	0.3336	0.1750	0.0977	0.0987
NCUT	0.3691	0.1747	0.1044	0.1271
PR	0.3150	0.1598	0.0917	0.0950
HTF-IDF	0.3215	0.1456	0.0956	0.1056
CLUSTER	0.3565	0.1621	0.1002	0.1121
MMTS-I	**0.4223**	**0.2271**	**0.1196**	**0.1653**
MMTS-R	**0.4533**	**0.2421**	**0.1256**	**0.1751**
MMTS	**0.4780**	**0.2797**	**0.1492**	**0.1877**



(a) *Social Trends*



(b) *Product Topics*

Figure 5.5: Detailed performance (ROUGE-1) of MMTS, MMTS-I and five selected comparing approaches over all topics.

Table 5.3: Effects of coverage, significance and diversity criteria in subtopic discovery on the *Social Trends* dataset.

	ROUGE-1	**ROUGE-2**	**ROUGE-W**	**ROUGE-SU**
MMTS-C	0.3602	0.2108	0.0335	0.1340
MMTS-S	0.3791	0.2225	0.0327	0.1660
MMTS-D	0.4207	0.2469	0.0322	0.1843
MMTS	**0.5049**	**0.3076**	**0.0696**	**0.2356**

First of all, MMTS explores the joint correlation between the textual and visual aspects of microblogs. The impact of multimedia knowledge can be demonstrated by comparing the results of MMTS and MMTS-I. The latter approach differs from MMTS only with the lack of visual component. The performance illustrates the degradation of summarization ability when only a single media type is used. In addition, by comparing the results of MMTS and MMTS-R (which uses the noisy image collections), it clearly demonstrates the necessity for removing irrelevant images from the original datasets.

Secondly, MMTS and MMTS-I discover subtopics before the summarization procedure. As a result, all important branches for a topic are covered in the final summarization. Although some comparing methods also consider the coverage of the summarization for the dataset, the coverage is only considered at the topic-level rather than the subtopic-level. In case a subtopic contains a small number of microblogs, there is a high probability that the microblogs related to this subtopic will be ignored with comparing methods. The high performance of MMTS-I as compared to all the baseline methods demonstrates the effectiveness of subtopic discovery for enhancing summarization performance.

Thirdly, three criteria are specified in MMTS for generating the summary of each subtopic, namely coverage, significance and diversity. These three cri-

Table 5.4: Effects of coverage, significance and diversity criteria in subtopic discovery on the *Product Topics* dataset.

	ROUGE-1	**ROUGE-2**	**ROUGE-W**	**ROUGE-SU**
MMTS-C	0.3502	0.2034	0.0506	0.1023
MMTS-S	0.3701	0.1872	0.0823	0.1205
MMTS-D	0.4213	0.2356	0.0956	0.1203
MMTS	**0.4780**	**0.2797**	**0.1492**	**0.1877**

teria are able to further facilitate the summary generation. We conduct further experiment to evaluate the effectiveness of each individual component by removing each of the three criteria from our framework. The result is shown in Table 5.3 and Table 5.4. MMTS-C denotes the method of using only significance and diversity, without taking coverage into consideration. Similarly, MMTS-S is the method without considering significance, and MMTS-D represents our method without considering diversity. For each comparing methods, the parameter ω corresponding to the removed criterion was set to 0, while the parameters for other two factors were kept unchanged (The parameter value is described in the next subsection). As can be seen, the performance of removing any criterion becomes worse, which illustrates that all components are necessary for our framework.

An example of our summarization result is shown in Figure 5.6. This is a summary on Topic #1 of *Social Trends* dataset. As shown, five subtopics are discovered. Due to space limitation, only the top 3 images and top 5 texts for each subtopic are listed. This example demonstrates the ability of our proposed framework in 1) well organizing the messy microblogs into structured subtopics; 2) generating high-quality textual summary at subtopic level; and 3) selecting the most representative images for summarizing the topic.

Figure 5.6: An illustrative example of multimedia topic summarization on Topic #1 in *Social Trends* dataset.

5.4.4 Parameter Tuning

The overall selection score is a weighted linear combination of the three criteria coverage, significance and diversity. In this part, we examine the effects of

the corresponding weighting parameters ω_1, ω_2 and ω_3 to achieve the optimal parameter setting. Keeping other two parameters fixed to 1, we vary the remaining ω from 0 to 10 to examine its influence on the final results, and select the value which achieves the best F-score for the ROUGE values. After achieving the corresponding values for ω_1, ω_2 and ω_3, we adjust $\omega_i = \omega_i/(\omega_1+\omega_2+\omega_3)$ to make the sum of the three weighting parameters to 1. With this procedure, the parameters are selected as $\omega_1 = 0.4$, $\omega_2 = 0.4$ and $\omega_3 = 0.2$ for the *Social Trends* dataset, and $\omega_1 = 0.2$, $\omega_2 = 0.5$, and $\omega_3 = 0.3$ for the *Product Topics* dataset. In order to prove the above results are the optimized combination, we further fix two of the ω values fixed as the achieved value, and vary the third one. According to the results shown in Figure 5.7, all parameters perform the best when they are at the achieve optimized value, e.g., the best performance for ω_1 in *Social Trends* dataset is 0.4, which is consistent with our result, thus proves the optimization of the tuned parameter values.

Another important parameter is the number of subtopics K. Figure 5.8 shows the performance of MMTS with various subtopic number K in terms of ROUGE-1 result. Very small K fails to achieve satisfactory performance, as the ability to discover subtopics is not fully utilized in this situation. However, large K does not lead to significant growth for the summarization performance, and may exert negative influence. By taking a detailed observation of our dataset, we can see that the microblog discussion for the same topic is usually limited to a few directions, which means the number of subtopics will not be too large in our specific scenario. If we set the subtopic number to an improper large number, less important topic branches will be extracted, and corresponding microblogs will be included in the final summary, which will hurt the summarization performance. Furthermore, too many subtopics will

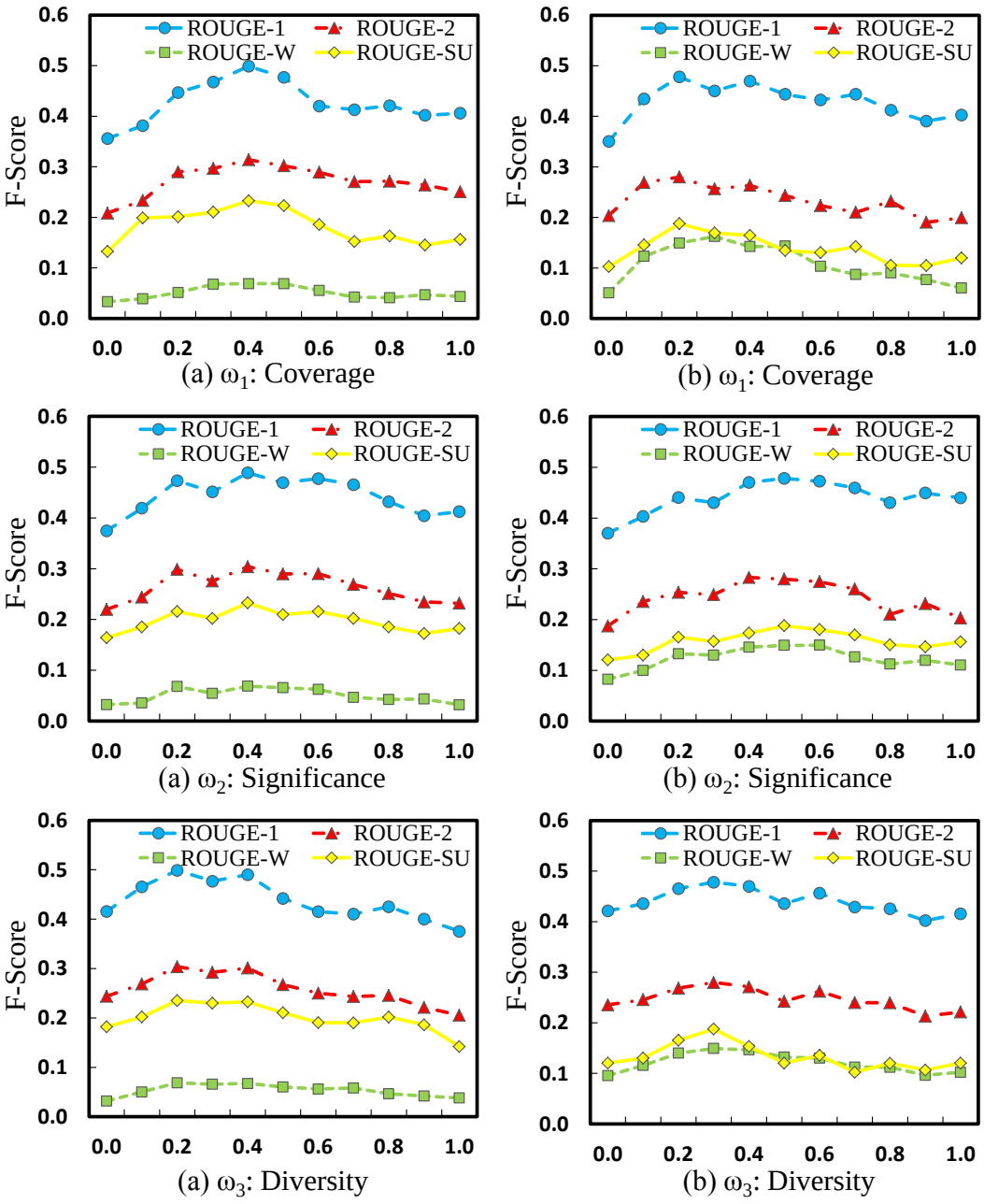


Figure 5.7: Performance of parameter ω_1 ω_2 and ω_3 on the two datasets: (a) *Social Trends* and (b) *Product Topics*.

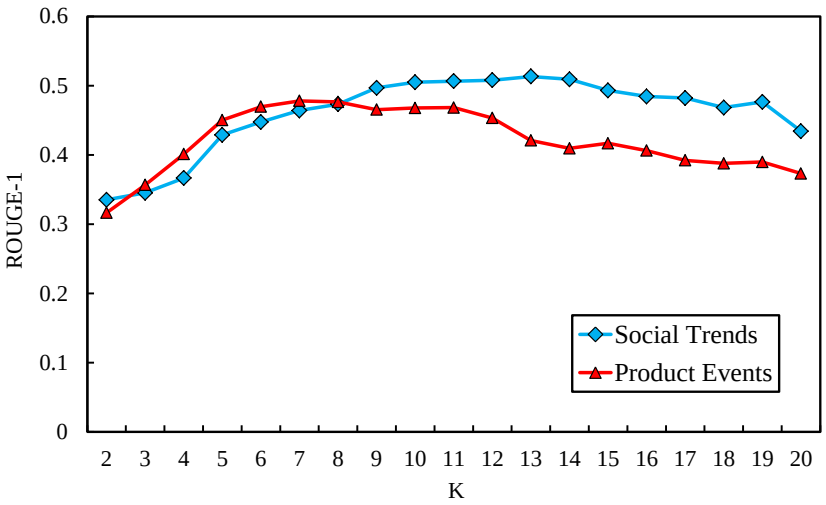


Figure 5.8: Summarization performance of MMTS with various subtopic number K.

hurt the "concise" principle of summarization. Taken the above points into consideration, we set the subtopic number K to 10 for *Social Trends* and 7 for *Product Topics*.

5.5 Summary

In this chapter, we presented a multimedia topic summarization framework which automatically generates a holistic visualized summary from the microblogs of various media types. The proposed framework features the exploration of the intrinsic correlations among different media types for enhancing the summarization performance. In particular, we developed three major stages to accomplish the summarization. First, we devised an effective approach for eliminating the potentially noisy images from raw microblog image collection. Then, we proposed a novel *Cross-Media-LDA* (CMLDA) model, to discover subtopics from microblogs of different media types. Finally, we generated multimedia summary for topics utilizing the cross-media distribution knowledge of all the discovered

subtopics. We conducted extensive experiments on two real-world microblog datasets collected by ourselves to show the superiority of our proposed method as compared to the state-of-the-art approaches.

CHAPTER 6

Conclusion and Future Work

6.1 Conclusion

This thesis thoroughly explored the viral topic monitoring system for microblog streams. Microblog, with its nature of sufficiency in timely information and extensiveness in information propagation, is a valuable source to keep users updated about the real world. Specifically, we targeted at the study of viral topics to inform users of the occurrence of outbreaking topics. The ability of the state-of-the-art methods for viral topic detection is restricted in the way that the occurrence of viral topics cannot be detected until a large amount of discussion have engaged, which fails to meet the requirement for timeliness in modern systems. Therefore, this work proposed a framework to push the monitoring of viral topics one step ahead, aiming at predicting viral topics in the early stage. Specifically, we investigated the following research problems.

First, how to predict the occurrence of a viral topic in the early stage? We proposed a two-step method targeting at this problem. The first step focuses on the prediction of viral microblogs, the results of which are feed into the second microblog tracking component to filter out non-topic microblogs and identify viral topics.

The first step, viral microblog prediction, was investigated through the angle of analyzing the diffusion of information in the microblog network. Specifically, we examined the social nature of microblog networks and proposed an influence model to learn the social influence factors among microblog users. The learnt social influence factors were found to be able to affect users' diffusion actions, which were then adopted in our information diffusion model to predict whether an incoming microblog is going to become viral in the near future. Empirical experiments on real-world dataset demonstrated the effectiveness of our proposed framework. It is worth mentioning that the elaborated model not only can provide binary prediction results (becoming viral or not) for incoming microblog posts, but is also able to offer detailed future diffusion paths and stages, which is valuable for applications like opinion tracking or designing advertisement strategies.

In the second step, we investigated the problem of microblog tracking, which was addressed by the proposed Evolutional Dictionary Learning (EDL) algorithm. Given a set of microblogs as targets, EDL tackles the tracking of these targets with a dictionary learning based method. A novel dictionary-transition matrix was introduced, which aimed at capturing the evolving contents of the tracking targets. We conducted sufficient evaluations to demonstrate the effectiveness of EDL in the task of microblog tracking. Various experimental settings was introduced to simulate different scenarios. We fur-

ther conducted another set of empirical experiments to validate the two-step approach for viral topic prediction. From the experimental results, we showed that this two-step approach not only can achieve satisfactory detection performance, but also fulfil the requirement of detecting the viral topics in the very early stage.

Second, after the system indicating the occurrence of a viral topic, how can it be presented to users? We proposed a novel multimedia summarization framework intended for describing microblog topics. Given a collection of microblogs related to the same topic, the proposed method can summarize the contents of the collection, resulting in a concise and precise summary which captures both textual and visual information. Specifically, we investigated how to remove the large portion of irrelevant images, and also examined how to jointly exploit the information of multiple types. Our work well compensates the previous literatures, which focused primarily on textual summarization. We demonstrated that visual contents can enhance the summarization performance by improving the quality of the generated summary. In addition, the incorporation of concrete multimedia exemplars can also assist users to gain a more visualized understanding of the topic.

To conclude, the elaborated system possesses the ability of predicting and summarizing viral topics in microblog networks, which makes it a practical tool to monitor the events in the real world.

6.2 Future Work

One promising extension for our system is to provide personalized topic prediction and summarization results. The current system focuses on discovering

universal topics. It would be more useful and meaningful if user preferences could be integrated into the whole process. The pursuing of personalization demands the ability to construct user profiles, and the capacity to match the original results towards intended user's requirements. In particular, the following challenges need to be addressed. First, the profile content provided by the users themselves can only reveal general information, such as age gender and profession. Therefore, an automatic mechanism need to be designed to extract more detailed information from the users' behavior, focusing on the features and characteristics that could reveal user preferences. Second, we should also investigate how specific the profile information should be. If the profile is too general, it will fails to convey sufficient details, resulting in unrelated information. On the other hand, too specific contents will leads to narrow understanding about the users, causing many new and diverse topics to be missed.

In addition to the above extension, we can also investigate how to better utilize the prediction results. We have shown that our system is able to predict the possible future diffusion range of a topic. This inspires us to think about the following problem: what could we do if we would like to promote the spread of a topic? We may take advantage of the proposed social influence model to discover users with the highest influence for promoting the diffusion of a certain type of topic.

Bibliography

[1] http://www.statista.com/statistics/282087/number-of-monthly-active-twitter-users/. vii, 4

[2] http://www.statista.com/statistics/273172/twitter-accounts-with-the-most-followers-worldwide/. vii, 5

[3] http://www.statista.com/statistics/274709/worldwide-audience-reach-of-online-retail-and-auction-sites/. vii, 6

[4] E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne. Finding high-quality content in social media. In *WSDM*, pages 183–194, 2008. 2

[5] J. Allan. Introduction to topic detection and tracking. In *Topic detection and tracking*, pages 1–16. 2002. 76

[6] J. Allan. *Topic detection and tracking: event-based information organization*, volume 12. Springer, 2002. 18

[7] E. Amitay and C. Paris. Automatically summarising web sites: is there a way around it? In *CIKM*, pages 173–179, 2000. 28

[8] A. Anagnostopoulos, R. Kumar, and M. Mahdian. Influence and correlation in social networks. In *SIGKDD*, pages 7–15, 2008. 48

[9] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts. Everyone's an influencer: quantifying influence on twitter. In *WSDM*, pages 65–74, 2011. 48

[10] R. Barzilay and N. Elhadad. Inferring strategies for sentence ordering in multidocument news summarization. *Journal Of Artificial Intelligence Research*, 17:35–55, 2002. 28

[11] R. Barzilay and N. Elhadad. Sentence alignment for monolingual comparable corpora. In *EMNLP*, pages 25–32, 2003. 32

[12] R. Barzilay and L. Lee. Catching the drift: Probabilistic content models, with applications to generation and summarization. In *HLT-NAACL*, pages 113–120, 2004. 30

[13] H. Becker, F. Chen, D. Iter, M. Naaman, and L. Gravano. Automatic identification and presentation of twitter content for planned events. In *ICWSM*, 2011. 24

[14] J. Bian, Y. Yang, and T.-S. Chua. Multimedia summarization for trending topics in microblogs. In *CIKM*, pages 1807–1812, 2013. 46

[15] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *JMLR*, 3:993–1022, 2003. 110

[16] C. Bouras and V. Tsogkas. Assigning web news to clusters. In *ICIW*, pages 1–6, 2010. 19

[17] D. Cai, X. He, Z. Li, W.-Y. Ma, and J.-R. Wen. Hierarchical clustering of www image search results using visual, textual and link information. In *MM*, pages 952–959, 2004. 35

[18] J. Carbonell and J. Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *SIGIR*, pages 335–336, 1998. 28, 33

[19] C. Castillo, M. Mendoza, and B. Poblete. Information credibility on twitter. In *WWW*, pages 675–684, 2011. 20

[20] A. Celikyilmaz and D. Hakkani-Tur. A hybrid hierarchical model for multi-document summarization. In *ACL*, pages 815–824, 2010. 30

[21] M. Cha, H. Haddadi, F. Benevenuto, and P. K. Gummadi. Measuring user influence in twitter: The million follower fallacy. *ICWSM*, 10:10–17, 2010. 48

[22] D. Chakrabarti and K. Punera. Event summarization using tweets. In *ICWSM*, pages 66–73, 2011. 40

[23] Y. Chang, X. Wang, Q. Mei, and Y. Liu. Towards twitter context summarization with user influence models. In *WSDM*, pages 527–536, 2013. 39

[24] W. Chen, C. Wang, and Y. Wang. Scalable influence maximization for prevalent viral marketing in large-scale social networks. In *SIGKDD*, pages 1029–1038, 2010. 48

[25] W. Chen, Y. Wang, and S. Yang. Efficient influence maximization in social networks. In *SIGKDD*, pages 199–208, 2009. 48

[26] Y. Chen, H. Amiri, Z. Li, and T.-S. Chua. Emerging topic detection for organizations from microblogs. In *SIGIR*, pages 43–52, 2013. 7, 68

[27] Y. Chen, J. Z. Wang, and R. Krovetz. Content-based image retrieval by clustering. In *IMIR*, pages 193–200, 2003. 34

[28] T.-S. Chua, H. Luan, M. Sun, and S. Yang. Next: Nus-tsinghua center for extreme search of user-generated content. *IEEE MultiMedia*, 19(3):81–87, 2012. 46

[29] P. Clough, H. Joho, and M. Sanderson. Automatically organising images using concept hierarchies. In *Multimedia Workshop at SIGIR*, 2005. 37

[30] M. Cooper, J. Foote, A. Girgensohn, and L. Wilcox. Temporal event clustering for digital photo collections. *TOMCCAP*, 1(3):269–288, 2005. 38

[31] A. Crooks, A. Croitoru, A. Stefanidis, and J. Radzikowski. # earthquake: Twitter as a distributed sensor system. *Transactions in GIS*, 17(1):124–147, 2013. 7

[32] P. Cui, S. Jin, L. Yu, F. Wang, W. Zhu, and S. Yang. Cascading outbreak prediction in networks: a data-driven approach. In *SIGKDD*, pages 901–909, 2013. 49, 63, 68

[33] H. Daumé III and D. Marcu. Bayesian query-focused summarization. In *ACL*, pages 305–312, 2006. 30

[34] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by latent semantic analysis. *JASIS*, 41(6):391–407, 1990. 29

[35] J.-Y. Delort, B. Bouchon-Meunier, and M. Rifqi. Enhanced web document summarization using hyperlinks. In *Hypertext and hypermedia*, pages 208–215, 2003. 28

[36] H. P. Edmundson. New methods in automatic extracting. *Journal of the ACM (JACM)*, 16(2):264–285, 1969. 32

[37] G. Erkan and D. R. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *JAIR*, 22(1):457–479, 2004. 29, 31

[38] E. Filatova and V. Hatzivassiloglou. A formal model for information selection in multi-sentence text extraction. In *Computational Linguistics*, page 397, 2004. 29

[39] M. Fuentes, E. Alfonseca, and H. Rodríguez. Support vector machines for query-focused summarization trained and evaluated on pyramid data. In *ACL*, pages 57–60, 2007. 32

[40] G. P. C. Fung, J. X. Yu, P. S. Yu, and H. Lu. Parameter free bursty events detection in text streams. In *VLDB*, pages 181–192, 2005. 20

[41] P. Fung and G. Ngai. One story, one flow: Hidden markov story models for multilingual multidocument summarization. *TSLP*, 3(2):1–16, 2006. 30

[42] M. Galley. A skip-chain conditional random field for ranking meeting utterances by importance. In *EMNLP*, pages 364–372, 2006. 29

[43] B. Gao, T.-Y. Liu, T. Qin, X. Zheng, Q.-S. Cheng, and W.-Y. Ma. Web image clustering by consistent utilization of visual features and surrounding texts. In *MM*, pages 112–121, 2005. 35

[44] W. Gao, P. Li, and K. Darwish. Joint topic modeling for event summarization across news and social media streams. In *CIKM*, pages 1173–1182, 2012. 41

[45] Y. Gao, F. Wang, H. Luan, and T.-S. Chua. Brand data gathering from live social media streams. In *ICMR*, page 169, 2014. 119

[46] D. Gillick, K. Riedhammer, B. Favre, and D. Hakkani-Tur. A global optimization framework for meeting summarization. In *Acoustics, Speech and Signal Processing*, pages 4769–4772, 2009. 33

[47] J. Goldenberg, B. Libai, and E. Muller. Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing letters*, 12(3):211–223, 2001. 45

[48] Y. Gong and X. Liu. Generic text summarization using relevance measure and latent semantic analysis. In *SIGIR*, pages 19–25, 2001. 29, 123

[49] S. Goorha and L. Ungar. Discovery of significant emerging trends. In *SIGKDD*, pages 57–64, 2010. 20

[50] A. Goyal, F. Bonchi, and L. V. Lakshmanan. Learning influence probabilities in social networks. In *WSDM*, pages 241–250, 2010. 48, 69

[51] A. Goyal, F. Bonchi, and L. V. Lakshmanan. A data-based approach to social influence maximization. *PVLDB*, 5(1):73–84, 2011. 48

[52] A. Graham, H. Garcia-Molina, A. Paepcke, and T. Winograd. Time as essence for photo browsing through personal digital libraries. In *Digital libraries*, pages 326–335, 2002. 37

[53] D. Gruhl, R. Guha, D. Liben-Nowell, and A. Tomkins. Information diffusion through blogspace. In *WWW*, pages 491–501, 2004. 48

[54] A. Guille, H. Hacid, C. Favre, and D. A. Zighed. Information diffusion in online social networks: A survey. *SIGMOD Record*, 42(2):17, 2013. 45

[55] A. Haghighi and L. Vanderwende. Exploring content models for multi-document summarization. In *NAACL HLT*, pages 362–370, 2009. 30, 120

[56] Q. He, K. Chang, and E.-P. Lim. Analyzing feature trajectories for event detection. In *SIGIR*, pages 207–214, 2007. 20

[57] L. Hong, O. Dan, and B. D. Davison. Predicting popular messages in twitter. In *WWW*, pages 57–58, 2011. 68

[58] D. Inouye and J. K. Kalita. Comparing twitter summarization algorithms for multiple post summaries. In *SocialCom*, pages 298–306, 2011. 39, 124

[59] A. Jaffe, M. Naaman, T. Tassa, and M. Davis. Generating summaries and visualization for large collections of geo-referenced photographs. In *MIR*, pages 89–98, 2006. 38

[60] A. Java, X. Song, T. Finin, and B. Tseng. Why we twitter: understanding microblogging usage and communities. 2

[61] F. Jing, C. Wang, Y. Yao, K. Deng, L. Zhang, and W.-Y. Ma. Igroup: web image search results clustering. In *MM*, pages 377–384, 2006. 36

[62] H. Jing. Using hidden markov modeling to decompose human-written summaries. *Computational linguistics*, 28(4):527–543, 2002. 32

[63] A. M. Kaplan and M. Haenlein. Users of the world, unite! the challenges and opportunities of social media. *Business horizons*, 53(1):59–68, 2010. 2

[64] S. P. Kasiviswanathan, P. Melville, A. Banerjee, and V. Sindhwani. Emerging topic detection using dictionary learning. In *CIKM*, pages 745–754, 2011. 23, 26, 98

[65] D. Kempe, J. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *SIGKDD*, pages 137–146, 2003. 45, 48, 63

[66] J. Kleinberg. Bursty and hierarchical structure in streams. *Data Mining and Knowledge Discovery*, 7(4):373–397, 2003. 20

[67] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. *TIT*, 47(2):498–519, 2001. 60

[68] R. Kumar, J. Novak, P. Raghavan, and A. Tomkins. On the bursty evolution of blogspace. *WWW*, 8(2):159–178, 2005. 49

[69] J. Kupiec, J. Pedersen, and F. Chen. A trainable document summarizer. In *SIGIR*, pages 68–73, 1995. 32

[70] T. La Fond and J. Neville. Randomization tests for distinguishing social influence and homophily effects. In *WWW*, pages 601–610, 2010. 48

[71] V. Lavrenko, J. Allan, E. DeGuzman, D. LaFlamme, V. Pollard, and S. Thomas. Relevance models for topic detection and tracking. In *ICHLT*, pages 115–121, 2002. 95

[72] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999. 85

[73] R. Lee and K. Sumiya. Measuring geographical regularities of crowd behaviors for twitter-based geo-social event detection. In *SIGSPATIAL*, pages 1–10, 2010. 24, 25

[74] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance. Cost-effective outbreak detection in networks. In *SIGKDD*, pages 420–429, 2007. 45, 49

[75] C. Lin, C. Lin, J. Li, D. Wang, Y. Chen, and T. Li. Generating event storylines from microblogs. In *CIKM*, pages 175–184, 2012. 40

[76] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: ACL-04 Workshop*, pages 74–81, 2004. 122

[77] H. Liu, X. Xie, X. Tang, Z.-W. Li, and W.-Y. Ma. Effective browsing of web image search results. In *IMIR*, pages 84–90, 2004. 34

[78] W. Liu, Y.-G. Jiang, J. Luo, and S.-F. Chang. Noise resistant graph ranking for improved web image search. In *CVPR*, pages 849–856, 2011. 107

[79] 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*, pages 652–663. 2011. 23

[80] G. Luo, C. Tang, and P. S. Yu. Resource-adaptive real-time new event detection. In *SIGMOD*, pages 497–508, 2007. 19

[81] D. Marcu. The automatic construction of large-scale corpora for summarization research. In *SIGIR*, pages 137–144, 1999. 32

[82] M. Mathioudakis and N. Koudas. Twittermonitor: trend detection over the twitter stream. In *SIGMOD*, pages 1155–1158, 2010. 22, 26, 98

[83] R. McDonald. *A study of global inference algorithms in multi-document summarization*. Springer, 2007. 33

[84] D. Metzler, C. Cai, and E. Hovy. Structured event retrieval over microblog archives. In *NAACL HLT*, pages 646–655, 2012. 25

[85] R. Mihalcea. Graph-based ranking algorithms for sentence extraction, applied to text summarization. In *ACL*, page 20, 2004. 31

[86] L. P. Morales, A. D. Esteban, and P. Gervás. Concept-graph based biomedical automatic summarization using ontologies. In *3rd Textgraphs Workshop on Graph-Based Algorithms for Natural Language Processing*, pages 53–56, 2008. 32

[87] M. Naaman, Y. J. Song, A. Paepcke, and H. Garcia-Molina. Automatic organization for digital photographs with geographic coordinates. pages 53–62, 2004. 37

[88] A. Nenkova and A. Bagga. Facilitating email thread access by extractive summary generation. *AMSTERDAM STUDIES IN THE THEORY AND HISTORY OF LINGUISTIC SCIENCE*, page 287, 2004. 28

[89] A. Nenkova and K. McKeown. A survey of text summarization techniques. In *Mining Text Data*, pages 43–76. 2012. 27

[90] A. Nenkova and L. Vanderwende. The impact of frequency on summarization. *Microsoft Research, Redmond, Washington, Tech. Rep. MSR-TR-2005-101*, 2005. 29

[91] J. Nichols, J. Mahmud, and C. Drews. Summarizing sporting events using twitter. In *IUI*, pages 189–198, 2012. 40

[92] M. Osborne. Using maximum entropy for sentence extraction. In *ACL*, pages 1–8, 2002. 32

[93] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. *TNN*, 22(2):199–210, 2011. 53

[94] S. Park, J.-H. Lee, D.-H. Kim, and C.-M. Ahn. Multi-document summarization based on cluster using non-negative matrix factorization. In *SOFSEM*, pages 761–770. 2007. 123

[95] S. Petrović, M. Osborne, and V. Lavrenko. Streaming first story detection with application to twitter. In *NAACL HLT*, pages 181–189, 2010. 22

[96] S. Phuvipadawat and T. Murata. Breaking news detection and tracking in twitter. In *WI-IAT*, volume 3, pages 120–123, 2010. 23

[97] A. Pigeau and M. Gelgon. Organizing a personal image collection with statistical model-based icl clustering on spatio-temporal camera phone meta-data. *Journal of Visual Communication and Image Representation*, 15(3):425–445, 2004. 37

[98] A.-M. Popescu and M. Pennacchiotti. Detecting controversial events from twitter. In *CIKM*, pages 1873–1876, 2010. 24, 25

[99] D. R. Radev, H. Jing, M. Styś, and D. Tam. Centroid-based summarization of multiple documents. *Information Processing & Management*, 40(6):919–938, 2004. 29, 124

[100] O. Rambow, L. Shrestha, J. Chen, and C. Lauridsen. Summarizing email threads. In *HLT-NAACL*, pages 105–108, 2004. 28

[101] Y. Rao and Q. Li. Term weighting schemes for emerging event detection. In *WI-IAT*, volume 1, pages 105–112, 2012. 23, 26, 99

[102] S. Rastkar, G. C. Murphy, and G. Murray. Summarizing software artifacts: a case study of bug reports. In *Software Engineering*, pages 505–514, 2010. 32

[103] K. Riedhammer, D. Gillick, B. Favre, and D. Hakkani-Tür. Packing the meeting summarization knapsack. In *INTERSPEECH*, pages 2434–2437, 2008. 33

[104] K. Rodden, W. Basalaj, D. Sinclair, and K. Wood. Does organisation by similarity assist image browsing? In *SIGCHI*, pages 190–197, 2001. 34

[105] C. Rother, S. Kumar, V. Kolmogorov, and A. Blake. Digital tapestry. In *CVPR*, volume 1, pages 589–596, 2005. 34

[106] A. Saha and V. Sindhwani. Learning evolving and emerging topics in social media: a dynamic nmf approach with temporal regularization. In *WSDM*, pages 693–702, 2012. 23, 26, 99

[107] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *WWW*, pages 851–860, 2010. 24, 25

[108] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988. 29

[109] J. Sankaranarayanan, H. Samet, B. E. Teitler, M. D. Lieberman, and J. Sperling. Twitterstand: news in tweets. In *SIGSPATIAL*, pages 42–51, 2009. 22

[110] P. Schmitz. Inducing ontology from flickr tags. In *Collaborative Web Tagging Workshop at WWW*, volume 50, 2006. 37

[111] B. Sharifi, M.-A. Hutton, and J. Kalita. Summarizing microblogs automatically. In *NAACL HLT*, pages 685–688, 2010. 39

[112] B. Sharifi, M.-A. Hutton, and J. K. Kalita. Experiments in microblog summarization. In *SocialCom*, pages 49–56, 2010. 39, 124

[113] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE TPAMI*, 22(8):888–905, 2000. 124

[114] T. Shi, M. Belkin, and B. Yu. Data spectroscopy: Eigenspaces of convolution operators and clustering. *The Annals of Statistics*, pages 3960–3984, 2009. 108

[115] I. Simon, N. Snavely, and S. M. Seitz. Scene summarization for online image collections. In *ICCV*, pages 1–8, 2007. 36

[116] T. Snowsill, F. Nicart, M. Stefani, T. De Bie, and N. Cristianini. Finding surprising patterns in textual data streams. In *CIP*, pages 405–410, 2010. 21

[117] J. Steinberger and K. Ježek. Text summarization and singular value decomposition. In *Advances in Information Systems*, pages 245–254. 2005. 30

[118] J. Steinberger, M. Poesio, M. A. Kabadjov, and K. Ježek. Two uses of anaphora resolution in summarization. *Information Processing & Management*, 43(6):1663–1680, 2007. 30

[119] C. Tan, J. Tang, J. Sun, Q. Lin, and F. Wang. Social action tracking via noise tolerant time-varying factor graphs. In *SIGKDD*, pages 1049–1058, 2010. 49, 71

[120] J. Tang, W. Sen, and J. Sun. Confluence: conformity influence in large social networks. In *SIGKDD*, pages 347–355, 2013. 49

[121] J. Tang, J. Sun, C. Wang, and Z. Yang. Social influence analysis in large-scale networks. In *SIGKDD*, pages 807–816, 2009. 48, 69

[122] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society.*, pages 267–288, 1996. 79

[123] K. Toyama, R. Logan, and A. Roseway. Geographic location tags on digital images. In *MM*, pages 156–166, 2003. 37

[124] C. K. Vaca, A. Mantrach, A. Jaimes, and M. Saerens. A time-based collective factorization for topic discovery and monitoring in news. In *WWW*, pages 527–538, 2014. 91

[125] R. H. van Leuken, L. Garcia, X. Olivares, and R. van Zwol. Visual diversification of image search results. In *WWW*, pages 341–350, 2009. 35

[126] X. Wan and J. Xiao. Towards a unified approach based on affinity graph to various multi-document summarizations. In *Research and Advanced Technology for Digital Libraries*, pages 297–308. 2007. 31

[127] D. Wang, T. Li, S. Zhu, and C. Ding. Multi-document summarization via sentence-level semantic analysis and symmetric matrix factorization. In *SIGIR*, pages 307–314, 2008. 123

[128] D. Wang, S. Zhu, T. Li, and Y. Gong. Multi-document summarization using sentence-based topic models. In *ACL-IJCNLP*, pages 297–300, 2009. 30

[129] J. Wang, L. Quan, J. Sun, X. Tang, and H.-Y. Shum. Picture collage. In *CVPR*, volume 1, pages 347–354, 2006. 34

[130] X. Wang, C. Zhai, X. Hu, and R. Sproat. Mining correlated bursty topic patterns from coordinated text streams. In *SIGKDD*, pages 784–793, 2007. 20

[131] X.-J. Wang, W.-Y. Ma, Q.-C. He, and X. Li. Grouping web image search result. In *MM*, pages 436–439, 2004. 35

[132] X.-J. Wang, W.-Y. Ma, L. Zhang, and X. Li. Iteratively clustering web images based on link and attribute reinforcements. In *MM*, pages 122–131, 2005. 35

[133] Z. Wen and W. Yin. A feasible method for optimization with orthogonality constraints. *Mathematical Programming*, pages 1–38, 2013. 56

[134] J. Weng, E.-P. Lim, J. Jiang, and Q. He. Twitterrank: finding topic-sensitive influential twitterers. In *WSDM*, pages 261–270, 2010. 48

[135] R. Xiang, J. Neville, and M. Rogati. Modeling relationship strength in online social networks. In *WWW*, pages 981–990, 2010. 48

[136] C. Yang, J. Shen, and J. Fan. Effective summarization of large-scale web images. In *MM*, pages 1145–1148, 2011. 36

[137] Y. Yang, T. Ault, T. Pierce, and C. W. Lattimer. Improving text categorization methods for event tracking. In *SIGIR*, pages 65–72, 2000. 95

[138] Y. Yang, T. Pierce, and J. Carbonell. A study of retrospective and on-line event detection. In *SIGIR*, pages 28–36, 1998. 19

[139] Y. Yang, Y. Yang, Z. Huang, H. Shen, and F. Nie. Tag localization with spatial correlations and joint group sparsity. In *CVPR*, pages 881–888, 2011. 54

[140] Y. Yang, Y. Yang, and H. Shen. Effective transfer tagging from image to video. *TOMCCAP*, 9(2), 2013. 52

[141] Z. Yang, K. Cai, J. Tang, L. Zhang, Z. Su, and J. Li. Social context summarization. In *SIGIR*, pages 255–264, 2011. 41

[142] S. Ye, T.-S. Chua, M.-Y. Kan, and L. Qiu. Document concept lattice for text understanding and summarization. *Information Processing & Management*, 43(6):1643–1662, 2007. 33

[143] W.-t. Yih, J. Goodman, L. Vanderwende, and H. Suzuki. Multi-document summarization by maximizing informative content-words. In *IJCAI*, volume 2007, page 20th, 2007. 33

[144] L. Zhou and E. Hovy. A web-trained extraction summarization system. In *NAACL HLT*, pages 205–211, 2003. 32