Incorporating Relationships Between Tweets for Topic Derivation in Twitter

A Dissertation Presented in Fulfillment of the Requirements for the Degree of Doctor of Philosophy

Robertus Setiawan Aji Nugroho

Department of Computing
Faculty of Science and Engineering
Macquarie University, NSW 2109, Australia

Submitted October 2017

Declaration

I certify that the work in this thesis entitled INCORPORATING RELATIONSHIPS BETWEEN TWEETS FOR TOPIC DERIVATION IN TWITTER has not previously been submitted for a degree nor has it been submitted as part of the requirements for a degree to any other university or institution other than Macquarie University. I also certify that the thesis is an original piece of research and it has been written by me. Any help and assistance that I have received in my research work and the preparation of the thesis itself have been appropriately acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Signed: .....

Date: .....

"*Pursue excellence and success will follow...*"

Rancho in 3 Idiots

Acknowledgements

I would like to express my deepest appreciation to The Indonesian Directorate General of Higher Education (DGHE Indonesia), Macquarie University, CSIRO Data61, The Australian Research Council, and Soegijapranata Catholic University who provide generous support during my PhD.

I am deeply indebted to my supervisor, Prof. Jian Yang for her encouragement, feedback, and inspiring advises. Her excellent supervision is second to none. I am grateful for her full support and guidance, without which I would not be able to finish this thesis. My sincere gratitude to Dr. Cécile Paris, Dr. Surya Nepal, and Dr. Weiliang Zhao for their ultimate supervisions, discussions, help, and time generosity. I really appreciate their support and hard work.

I thank Dr. Youliang Zhong and Dr. Diego Molla-Aliod for their advice, comments, and stimulating discussions. I also thank A/Prof. Yan Wang for his continuous support and encouragement during my PhD. My gratitude also goes to Yan Mei, Lei Han, Pengbo Xiu, Zizhu Zhang, and all other fellow PhD students for their feedback and interesting discussion throughout the years. I am truly grateful to all staffs of Macquarie University, especially the Department of Computing, and CSIRO Data61 for their generous support.

Last but not least, I thank my beloved wife Bernadia, my kids Andra and Bimo, my parents, big family, and friends for their full support and love, especially in the most difficult time of my study. I love you all.

Abstract

Twitter has become one of the most popular social media platforms, widely used for discussions and information dissemination on all kinds of topics. As a result, much research has been concerned with deriving topics from Twitter and applying the outcomes in a variety of real-life applications such as emergency management, business advertisements, and corporate/government communication. However, deriving topics in this short text based and highly dynamic environment remains a huge challenge.

In Twitter, the frequency of term co-occurrences across messages (tweets) is very low due to the limit on the number of characters allowed for posting. In addition, a tweet often includes informal language expressions, such as emoticons, abbreviations, and misspelled terms. This leads to a very sparse relationship between tweets and the terms used in the tweets. It renders methods that exploit only content features ineffective. Deriving topics from tweets is also problematic due to the highly dynamic environment, where topics change quickly over a short period of time.

To address these problems, we propose a novel topic derivation approach that incorporates tweet text similarity and time-sensitive interactions measures. Besides the tweet contents, the approach takes into account several types of interactions amongst tweets: tweets which mention the same user, replies, and retweets. We propose a joint probability model that can effectively integrate the effects of the content similarity, user mentions, and replies-retweets to measure the tweet relationships. Given the dynamic aspect of the environment, we also hypothesize that temporal features could further improve the quality of topic derivation results. We incorporate a time factor, introducing a half-life exponential decay function to deal

with this dynamic environment.

Topic derivation is done through our proposed Non-negative Matrix inter-joint Factorization (NMijF) method, in which we conduct co-factorization jointly over our tweet-to-tweet relationships matrix and tweet-to-term relationship matrix within a single iterative-update process. NMijF effectively clusters the tweets based on their relationships and meanwhile learns the topic-words by using the tweet clusters and content features of the tweets.

We conducted a number of experiments on several Twitter datasets to reveal both the individual and integrated effects of the various features being considered. Experimental results with TREC2014, tweetSanders, and tweetMarch datasets demonstrate that the proposed method is able to consistently outperform other advanced topic derivation methods and results in 10-70% improvements in all evaluation metrics.

Contents

List of Publications

This thesis has resulted in the following publications:

1. **R. Nugroho**, W. Zhao, J. Yang, C. Paris, and S. Nepal. The Joint Effects of Tweet Content Similarity and Tweet Interactions for Topic Derivation. In Proocedings of The 37th IEEE International Conference on Distributed Computing Systems (ICDCS 2017), Atlanta, GA, USA, June 5-8, 2017, IEEE Services Computing. [93]

2. **R. Nugroho**, J. Yang, W. Zhao, C. Paris, and S. Nepal. What and with whom? identifying topics in twitter through both interactions and text. IEEE Transactions on Services Computing, PP(99):1-1, 2017. ISSN 1939-1374. [92].

3. **R. Nugroho**, W. Zhao, J. Yang, C. Paris, and S. Nepal. Using time-sensitive interactions to improve topic derivation in twitter. World Wide Web, pages 1-27, 2016. ISSN 1573-1413. [91].

4. **R. Nugroho**, W. Zhao, J. Yang, C. Paris, S. Nepal, and Y. Mei. Time-sensitive topic derivation in twitter. In Proocedings of the Web Information Systems Engineering - WISE 2015: 16th International Conference, Miami, FL, USA, November 1-3, 2015, Proceedings, Part I, pages 138-152, Cham, 2015. Springer International Publishing. **(Best Paper Award)** [88].

5. **R. Nugroho**, J. Yang, Y. Zhong, C. Paris, and S. Nepal. Deriving topics in twitter by exploiting tweet interactions. In Proocedings of the 2015 IEEE International Congress on Big Data, pages 87-94, New York, USA, June 2015. **(Best Student Paper Award)**. [87].

6. **R. Nugroho**, Y. Zhong, J. Yang, C. Paris, and S. Nepal. Matrix inter-joint factorization - a new approach for topic derivation in twitter. In Proocedings of the 2015 IEEE International Congress on Big Data, pages 79-86, New York, USA, June 2015. [89].

7. **R. Nugroho**, D. Molla-Aliod, J. Yang, C. Paris, and S. Nepal. Incorporating tweet relationships into topic derivation. In Proocedings of the 2015 Conference of the Pacific Association for Computational Linguistics (PACLING 2015), Bali, Indonesia, pages 177-190, May 2015. PACLING. [90].

8. Y. Zhong, J. Yang, and **R. Nugroho**. Incorporating tie strength in robust social recommendation. In Proceedings of the 4th IEEE International Congress on Big Data, pages 63-70, New York, USA, July 2015. IEEE Services Computing Community. [143].

9. Y. Mei, Z. Zhang, W. Zhao, J. Yang, and **R. Nugroho**. A hybrid feature selection method for predicting user influence on Twitter. In Proceedings of the Web Information Systems Engineering - WISE 2015: 16th International Conference, Miami, FL, USA, November 1-3, 2015, Part I, pages 478-492, Cham, 2015. Springer International Publishing. [82].

10. **R. Nugroho**, C. Paris, S. Nepal, J. Yang, and W. Zhao. A Survey of Recent Methods on Deriving Topics from Social Media: Algorithm to Evaluation. ACM Computing Survey. 2017. **[Submitted]**.

List of Figures

List of Tables

Chapter 1

Introduction

Twitter (http://twitter.com) is a phenomenal social media platform for online information dissemination, covering a wide range of topics over the time. With around 350 thousand Twitter messages (tweets) per minute at the time of writing[1], Twitter is one of the social media platforms that generates very large and unstructured big data [7]. With such rapidly-changing information, deriving topics in Twitter is in demand to understand the current events in the world.

This thesis presents a thorough study of topic derivation in Twitter and proposes a novel approach to improve the quality of the derived topics. The ability to effectively derive topics from tweets is critical for navigating through this big data and exploring the information. Topic derivation provides an underlying service for a wide range of research areas and applications, for example, detecting events or marketing in business, sensing circumstances in a specific area or time, making recommendations, and determining hot issues [123].

1.1 Overview

Topic derivation in Twitter is the unsupervised task of clustering tweets based on their main topics and listing the most important keywords to represent the identified topics. In general, it can be addressed by observing the hidden thematic structures of a collection and selecting the representative words for every structure. Popular

[1] http://www.internetlivestats.com/twitter-statistics/, accessed 26 July 2017

topic derivation methods include *Latent Dirichlet Allocation* (LDA) [8], *Probabilistic Latent Semantic Analysis* (PLSA) [41], and *Non-negative Matrix Factorization* (NMF) [62]. These common methods are based on the exploitation of content to find the most important words to represent the topics found in document collections. Each term is observed to find its semantic relationships and similarities with other terms in the collections. These methods work well on lengthy documents (e.g., web content, collection of emails, research papers), where the frequency of term co-occurrences across the documents is high.

Deriving topics in the Twitter environment is a challenging problem. In Twitter, the frequency of term co-occurrences across the tweets is very low. A tweet is limited to a small number of characters[2]. Additionally, a tweet often includes expressions in informal language, such as emoticons, abbreviations, and misspelled terms. This leads to a very sparse relationship between tweets and the terms used in the tweets. It renders the methods mentioned above that exploit only the content features ineffective due to the extremely low occurrences of overlapping terms. Statistical analysis shows that the density of the relationship between terms matrix (i.e., the percentage of non-zero elements of the term-to-term matrix) is only 0.274% on average [91]. Consequently, this extreme sparsity hurts the quality of topic derivation [29].

Topic derivation in Twitter is also problematic due to the highly dynamic environment, where topics change quickly over a short period of time. Our investigation (discussed in Chapter 6) shows that conversations on a particular topic in Twitter typically quickly reach their peak (largest number of tweets involved in discussing the topic) with an average of about 15 minutes and then fade away at different rates. A topic can also evolve to another new topic or merge with other topics after some time. In the following, we discuss relevant works that address two key issues identified above, sparsity and dynamic, and their shortcomings.

[2]https://support.twitter.com/articles/15367, accessed 25 September 2017

Quite a few studies have been conducted to deal with the specific nature of tweets for topic derivation. Ramage et al. [106] propose a variant of *labeled-LDA* to work in the Twitter environment, with the hashtags and other content features (e.g., word distributions based on specific emoticons and other social signal) as labels for a partially supervised topic learning process. The approaches reported in [136] and [135] exploit the term co-occurrence patterns to improve the topic learning process in short text environments. Hong and Davison [42] use the aggregated content to train the tweets to be processed by the LDA method. Those approaches exploit different types of term relationships based on the tweet content and thus still suffer from the sparsity issue.

The study of [104] evaluates the implementation of the Author-Topic (AT) model [35] and the Author-Recipient-Topic (ART) model [79] in microblogs. Both AT and ART models are based on LDA. The AT method assumes that a document's topic distribution is influenced by its content and authors. The ART model improves on the AT method by incorporating not only the authors but also the recipients of the document. The experiments in [104] show that the LDA is still the best in most cases (number of topics is less than 50). In a higher number of topics, AT and ART are only able to present a limited improvement over the original LDA method. Sparsity remains a problem to be solved.

The work in [2] tackles the sparsity problem in Twitter while filtering tweets in real-time by proposing a query expansion method to enrich the knowledge of the topic by deriving terms that are relevant from users' query and document collections. Lv et al. [70] propose a knowledge-based expansion method using the knowledge terms from Freebase. Likewise, the work in [122] addresses the problem of sparsity when modeling the multifaceted topics in Twitter by augmenting the content with the help of hashtag based semantic enrichment and auxiliary semantic from linked external sources. However, relying on external documents brings an extra burden when dealing with highly dynamic environments like Twitter.

Different from other approaches, the study of [100] takes the context of Twitter users (e.g., following/followers, mentions) into account, but ignores the content of the tweets. The study of [130] incorporates the users following/followers characteristics and LDA-based topic derivation process to identify influential users in Twitter. The study of [48] reports that discussed topics derived from tweets that have social interactions have much higher credibility than if such interactions are not available.

To deal with the dynamic nature of the Twitter environment, several methods propose to include temporal features. Most of the works are aimed at implementing topic derivation in an incremental/online fashion to learn the movement of topics over the time. [107] proposes a time-based regularization in NMF method to learn the topics in social media. [61] presents an online variant of LDA to periodically model the topics from tweets based on time slices. The study in [14] introduces the content aging theory to mine the emerging topics from the Twitter stream. Stilo and Velardi [112] propose *Symbolix Aggregate approximation* (SAX) to discretize the temporal series of terms to discover the events from Twitter content. All these studies still focus on content. As a result, they still suffer from the sparsity issue. Most of them also still view the time aspect as a time slicing window to specify the interval of the serial or incremental learning process over time. The effect of the time aspect on the interactions happening in Twitter is still overlooked.

The nature of the Twitter environment makes topic derivation a challenging task. Current methods that rely only on the semantic features of tweet content mostly fail to provide high quality topics because of the sparsity and dynamic issues. In this thesis, we analyze various aspects of the Twitter platform to address these problems. We find that taking both tweet interactions and content similarity into account when identifying topics discussed in Twitter results in a significant improvement of the quality of the derived topics. The inclusion of temporal aspect is also important to further improve the quality of topics as the interactions could be time-sensitive. The motivating example discussed in the next section illustrates how the interactions in

Table 1.1: Motivating example

Id	User	Timestamp	Tweets
t_1	a	02/08/2016, 03:40 PM	New senate, exciting times in #Canberra @b
t_2	b	02/08/2016, 03:52 PM	@a true, and what a start with the census in Australia!
t_3	c	02/08/2016, 06:13 PM	RT @a New senate, exciting times in #Canberra @b
t_4	d	02/08/2016, 07:04 PM	#Floriade in #Canberra, biggest celebration of spring in Australia
t_5	e	02/08/2016, 07:10 PM	@d any special event in particular worth coming for?
t_6	d	02/08/2016, 07:12 PM	@e NightFest always has fantastic performers and great tasting pates from #Canberra and surrounding areas

Twitter could help improve the accuracy of topic identification.

1.2 Motivating Example

Twitter supports several important interactions. As shown in Table 1.1, a tweet can be (1) a single and *self-contained* statement [24] (e.g., t_4), (2) a *retweet*, which is an action of forwarding message to other audiences (e.g., t_3, indicated with *RT*), or (3) a part of a conversation, such as mentioning other people ('@username') inside the tweet content (e.g., t_1), or a reply to another tweet (e.g., t_2, t_5, and t_6, usually begins with @username). Tweet content can also contain a *hashtag*, a word started with the hash (#) symbol, for example: *#Canberra* in t_1, t_3, and t_4. Hashtags have been widely adopted by users to bookmark the content of a tweet, or to present users' interest in particular topics [138]. Although it does not necessarily represent a topic, we consider hashtags as important features that indicate the indirect relationship between tweets. So, we maintain the hashtag form and elaborate it with other interaction features.

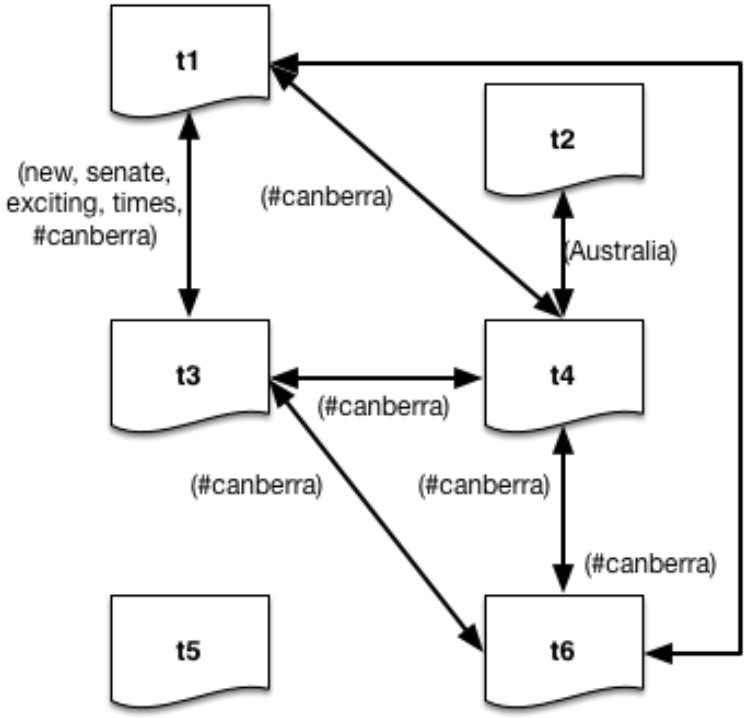


Figure 1.1: Tweet relationships based on the co-occurrence of terms

Taking the above interactions into account, we see that two topics can be identified from the motivating example in Table 1.1: one concerns the politics in Australia, and the one other the Floriade celebration that is being held in Canberra. However, if we only exploit the tweets content, we will most likely get the following topics: *#Canberra* and a special event. Figure 1.1 shows the relationships between the tweets and terms. It shows that t_1 is related to t_3 because all the terms in t_1 are available in t_3. Similarly, t_1, t_3, and t_4 are related to t_6 as they have "#Canberra" as a common term. Also, t_2 and t_4 share one term "Australia". In contrast, t_5 is isolated because it does not contain any terms in common with other tweets.

Nevertheless, if we take the interactions between tweets into account, we can obtain more accurate topics. Figure 1.2 shows the interactions between the tweets. We find that t_1 and t_2 are strongly related because t_1 mentions user b, and user b then replies to the author of t_1 through t_2. We can see that t_1 and t_2 are parts of the same conversation even though they do not share any terms. Moreover,

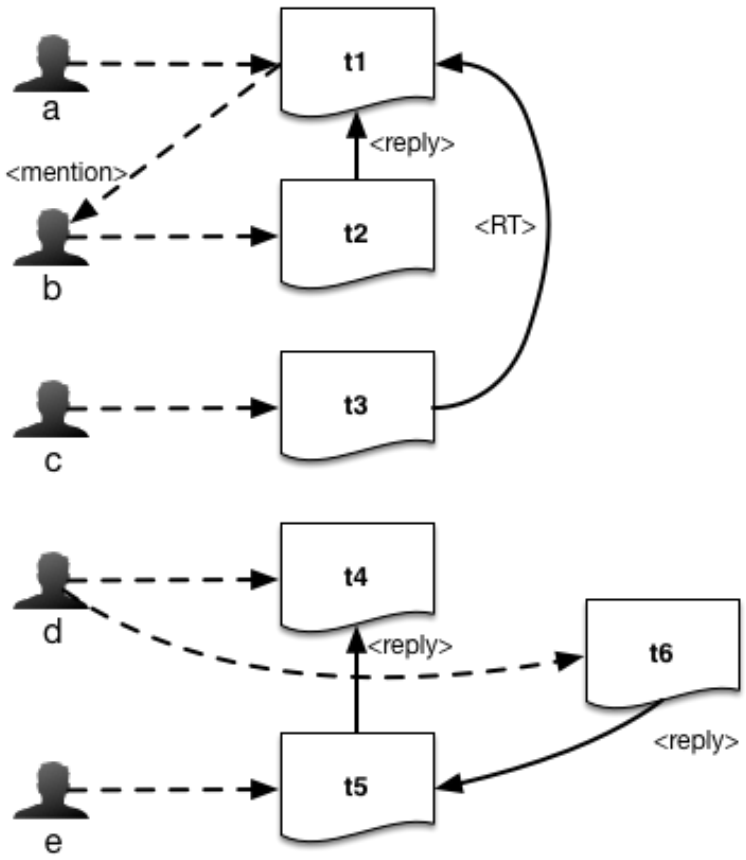


Figure 1.2: Tweet relationships based on tweets interactions

since t_3 is a retweet of t_1, it is obvious that they should belong to the same topic concerning politics. Similarly, t_4, t_5 and t_6 belong to another topic because t_5 is a reply to t_4, and t_6 is a reply to t_5. Based on these interactions, we can conclude that tweets t_4, t_5, and t_6 form a message group talking about the Floriade celebrations in Canberra. Intuitively, if tweets are part of a conversation (shown by interaction features, such as user mentions, replies or retweets), they most likely share a topic. Thus understanding the relationship between tweets based on interactions as well as content should improve the quality of topic derivation.

We also find that conversations in Twitter are typically time-sensitive. When a tweet t_7 by user b mentioning user a posted the day after at $03/08/2016, 10:31\ PM$ ("*just saw an accident near hume highway this morning @a*") is added to the

motivating example in Table 1.1, the topic discussed in this tweet is totally different compared to tweet t_1 or t_2 although they share the same user mention interactions. Our analysis on the impact of time to the topical connectivity shows that tweets with the mentions of the same users nearly at the same time are more likely to be about the same topic than tweets with mentions of same users after a long time interval. Therefore, incorporating the temporal aspect when looking at the interactions may further improve the quality of topic derivation.

1.3 Research Questions

This research addresses the sparsity and dynamic problems when deriving the topics in Twitter. We hypothesize that interactions (user mentions, replies, or retweets) amongst tweets are a strong indication that those tweets are part of a discussion about a particular topic. We believe that incorporating both tweet contents and tweet interactions along with the temporal aspect should enable us to achieve a significant improvement on the topic derivation quality. To support this hypothesis, we need to address the following main research questions:

1. How are tweets associated with topics according to their content similarity and tweet interactions (user mentions, replies, and retweets)?

2. What are the effects of content similarity and each of the interactions on the topical connectivity amongst tweets, and how can those effects be integrated to represent the topical relationships between tweets?

3. How can we effectively incorporate the topical relationships between tweets for topic derivation in Twitter?

4. How does time affect the topical connectivity between tweets and how to incorporate this time sensitivity in the topic derivation process to deal with the dynamic environment of the Twitter platform?

1.4 Thesis Contribution

In this thesis, we propose a novel approach to derive topics on Twitter by incorporating both the tweet content similarity and the interactions among tweets. This approach mainly takes an inter-joint Non-negative Matrix Factorization for (1) clustering the tweets based on their relationships, and (2) using the clustering result to learn the representative words for each cluster. The two factorization actions are jointly executed in a single iterative-update process. In comparison with other topic derivation methods, our proposed approach uses the underlying network in the Twitter environment formed by time-sensitive tweet interactions (i.e., *retweets*, *replies*, and *user mentions*) as well as the tweets' content similarity. As a result, it consistently achieves a high accuracy of topic derivation. The contributions of this thesis can be summarized as follows:

- We define the *topical relationships* between tweets and its representation as a combination of tweets content similarity and their interactions. The interactions between tweets include replies, retweets, and user mentions. We use several Twitter datasets to analyze how tweets are associated with topics according to both content similarity and tweet interactions. We form a *tweet-to-tweet* relationships matrix based on this definition. It presents a much less sparse matrix compared to other types of conventional relationships matrices. Our tweet-to-tweet relationships matrix provides 13.141% of density on average, while the tweet-to-term relationships has only 0.073% matrix density on average, and the term-to-term relationships has only 0.274% matrix density on average.

- We propose a joint probability model to measure the strength of relationships between a pair of tweets to integrate the effects of content similarity, user mentions, and replies-retweets between considered tweets. We first consider

each of these effects as an individual probability (without considering others) and then join them using our proposed joint probability model. Based on our statistical analysis, we find that tweets linked by replies-retweets, user mentions, and content similarity have different probabilities to be about the same topic. Replies and retweets explicitly show a strong signal of topical relationships between a pair of tweets. When two tweets are linked by reply or retweet (i.e., (1) one is a reply or retweet of another one, or (2) these two tweets are a reply or retweet of a particular tweet), it is safe to assume that both tweets share the same topic. A pair of tweets linked by user mention and/or content similarity has a reasonable chance to be about the same topic, but it is not as explicit as if they are linked by reply or retweet. The higher the number of common users and/or common terms shared between two tweets, the stronger the topical connectivity between them. Our proposed probabilistic model integrates the different effects of each individual relationship to accurately represent the strength of the relationships between tweets.

- We develop a Non-negative Matrix inter-joint Factorization method (*NMijF*), which performs a joint factorization over the symmetric tweet-to-tweet relationship matrix and the tweet-to-term relationship matrix in a single iterative-update process. NMijF effectively clusters the tweets based on their relationships and learns the topic-words by using the tweet clusters and content features of the tweets. The proposed NMijF is an extension of the popular Non-Negative Matrix Factorization (NMF) method [62]. NMF is one of the most effective methods to uncover the hidden thematic structure or latent features of a relationship-based matrix by factorizing the matrix into its lower dimensional representation. Our proposed NMijF approach can achieve the same objective as other popular topic derivation methods such as those which are based on LDA methods, and in addition, it is more flexible when incorporating

the strength of the relationships between tweets.

- We incorporate a time factor, introducing a half-life exponential decay function to deal with the dynamic environment in Twitter. We investigate the tweets interactions' behavior to see how time affects the topical connectivity between tweets. We find that the replies or retweets interactions are not affected by time. Tweets linked by replies or retweets are almost always about the same topic regardless of the posting time. In contrast, tweets linked by user mentions interaction are sensitive to time. These tweets tend to be about the same topic only when they are posted within the same period of time. A statistical analysis shows that most of user mentions related to a particular topic reach a peak in about 15 minutes and then gradually fade away. We propose a half-life exponential decay function by modeling the process of fading away to provide a more precise relationship measurement when tweets are linked by user mentions. Experimental results show that the inclusion of this temporal aspect into the process further improves the quality of topic derivation in Twitter.

- We conduct a comprehensive set of experiments on several Twitter datasets, including publicly available datasets (*TREC2014* and *tweetSanders*) and a dataset we collected *tweetMarch*, using various evaluation metrics. Each dataset has different characteristics which represent various situations that could happen in the Twitter environment. Our evaluations demonstrate that the incorporation of both content similarity and time-sensitive tweet interactions alleviates the sparsity problem and helps to produce high quality topic derivation in Twitter. We also perform the evaluation of our method by scrutinizing tweets grouped in a series of time periods to test the performance of the proposed method in an online situation. The results show that our proposed method can cope with the dynamic tweet stream better than the

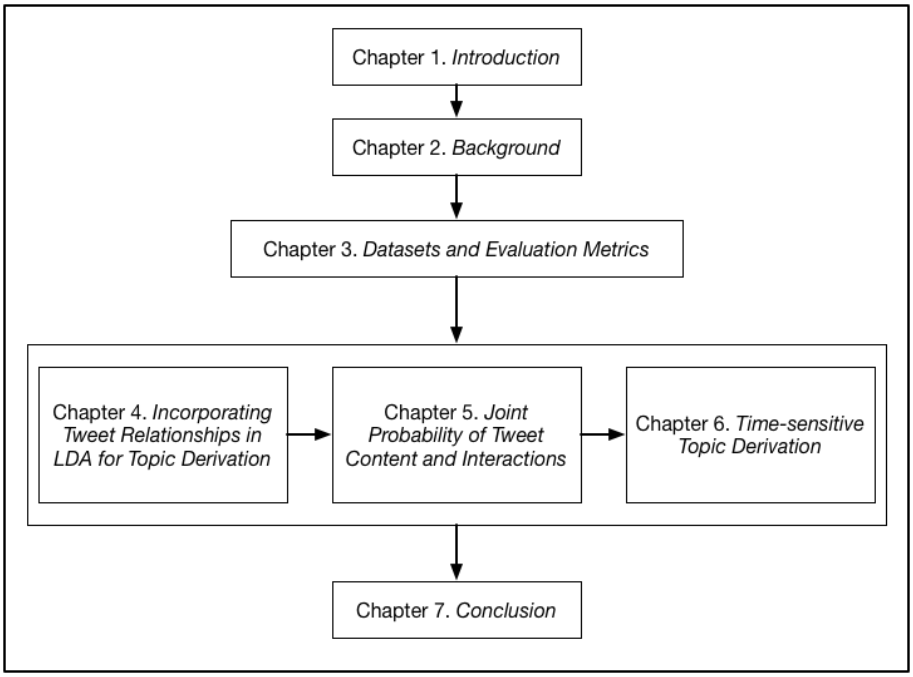


Figure 1.3: Thesis organization and relationships between chapters

baseline methods.

1.5 Thesis Outline

This thesis consists of 7 chapters. Figure 1.3 shows the thesis organization and relationships between chapters. Each chapter is summarized as follows:

- **Chapter 2. Background**

  In this chapter, we present an overview of topic derivation in Twitter. We first provide insights as to why Twitter is an important source for topic derivation and discuss why deriving topics in Twitter is a challenging problem. We then discuss the major techniques used to identify topics in traditional media. Finally, we conduct a comprehensive literature review related to topic derivation in the Twitter environment and discuss the observed research gaps.

- **Chapter 3. Datasets and Evaluation Metrics**

This chapter describes the datasets and evaluation metrics used throughout the thesis. Datasets are an integral part of the analysis and evaluation process. We use three datasets (two publicly available and a dataset we collected) to cover various characteristics of the Twitter environment. In this chapter, we also examine several widely used metrics for evaluating topic derivation algorithms.

- **Chapter 4. Incorporating Tweet Relationships in LDA for Topic Derivation**

  In this chapter, we present our observation of tweet topic prominence and topical connectivity between tweets through their interactions and content similarity. We propose intLDA, a variant of LDA, to incorporate both tweets interactions and content similarity to derive topics from Twitter. We also discuss our implementation of a simple variation to LDA that takes into account the tweets relationships (*eLDA*). It expands the tweets content based on their relationships with other tweets. Experimental results show that intLDA results in a significant improvement when compared to the eLDA in most scenarios, and other baseline methods in all cases. This chapter is related to our published paper [90].

- **Chapter 5. Joint Probability of Tweet Content and Interactions**

  This chapter discusses our proposed model to quantify the strength of relationships between tweets to achieve a high quality topic derivation results. We present our analysis on the accuracy of relationships between tweets when they are associated with topics according to content similarity and interactions between tweets. We then discuss our joint probability model to integrate the effects of each relationship component. An inter-joint Matrix Factorization Approach (*NMijF*) is proposed to incorporate the matrix formed by the joint probability model. We find that the proposed NMF based extension

is much more flexible to incorporate the strength of relationships between tweets, and it results in a significant improvement over our previous intLDA and other baseline methods. This chapter is related to our published paper [87, 89, 92, 93].

- **Chapter 6. Time-sensitive Topic Derivation**

  In this chapter, we discuss our finding of the time sensitivity of tweet interactions based on our observations of the relationships between topics and interactions between tweets. For exploration of time in Twitter, we use a dataset collected from the top-15 Twitter users in Australia. We then describe our proposed half-life exponential decay to model the user mention based time sensitivity. We report the results of the experiments against all datasets discussed in Chapter 3. We also present the evaluation of our proposed method in an online environment by scrutinizing tweets grouped in a series of time periods. The results show that our proposed method can cope with the dynamic Twitter stream better than the baseline methods. This chapter is related to our published papers [88, 91].

- **Chapter 7. Conclusion and Future Work**

  In this final chapter, we conclude the thesis with a summary and outline possible future works.

Chapter 2

Background

Social media in general, and Twitter in particular, are being used by a large community of people worldwide to post short pieces of information on any matters that are directly relevant to them. People might post for a wide range of reasons, such as to state someone's mood in a moment, to advertise one's business, to comment occurring events, or to report an accident or disaster. With the widespread and continuous use of Twitter by such a large community, there is a need to understand what are the topics under discussion. This is the goal of topic derivation.

In this chapter, we outline the task of topic derivation in Twitter. We first present general information about Twitter as one of the most popular social network platforms. We then describe major techniques to derive topics from a document collection and review the existing methods for topic derivation in Twitter. Finally, we highlight the research gaps.

2.1 Twitter

Twitter was founded in 2006. It was based on the idea of Jack Dorsey (Twitter co-founder) to broadcast users' status update to friends utilizing an SMS-based messaging platform [12]. The first tweet, shown in Figure 2.1, was posted by Jack Dorsey (@jack) on 21 March 2006.

In March 2007, Twitter won the *Web Award* from the South by Southwest (SXSW) Interactive conference [119]. It is a prestigious award given to honor the best and

Figure 2.1: The first posted tweet in Twitter by Jack Dorsey (Twitter founder) on 21 March 2006. *Source: https://twitter.com/jack/status/20*

Table 2.1: Twitter facts

Items	Facts/Number
Monthly active users	313M
Tweets per day	500M
Unique monthly visits to sites with embedded tweets	1B
Active users on mobile	82%
Accounts outside the US	79%
Number of employees	3860
Number of offices around the world	35+
Number of supported languages	40+

Note: The facts are compiled from *https://about.twitter.com/company* (accessed 30 March 2017, numbers are approximate as 30 June 2016) except the tweets per day are from *http://www.internetlivestats.com/twitter-statistics/* (accessed 30 March 2017).

most exciting technology development in the digital era. In just around 3 years after the first tweet, the number of tweets posted in Twitter had reached a billion [120].

Table 2.1 shows some statistics about Twitter in early 2017. There were around 313 million active Twitter users per month, who posted roughly 500 million tweets per day (±6000 tweets per second). 82% of the active users posted their tweets from mobile devices. Twitter has 35 offices around the world with more than 3000 employees. With 79% of the accounts outside the US, the Twitter platform now supports more than 40 languages. These facts make Twitter one of the most active

social network platforms worldwide.

With its large number of users and its ability to deliver real-time updates, Twitter has been used as an important source by journalists and government organizations to obtain the latest information about unfolding event. The photo of the US Airways plane crashed in the Hudson River was first posted and seen on Twitter before being reported by news media [119]. In 2011, Twitter proved to be one of the mass communication media that reported on the unfolding events in Arab spring movement, attracting the interests of journalist to source the news from this platform [38]. Other types of events or topics were also frequently and widely spread through Twitter. They include, for example, the 2009 earthquake in Japan [94], the floods in Australia [18] and Obama's presidential election [119].

Nowadays, following the high level of user activities on this platform, most of the news channels have accounts on Twitter and post their current headlines to the platform [84]. More brands and public figures, including actresses/actors, athletes, and politicians, are taking advantages of the exponential rise of Twitter users' to maximize their influence. More than 80% of world leaders are active on Twitter [21][1].

Twitter has attracted the interests of businesses and researchers to perform analysis on this type of social networks for various areas of study or applications, including sentiment analysis [1, 96, 54], influence maximization [115, 82], business engagement [141, 49, 97], community detection [137, 127], emergency and outbreak detection [59, 13, 23], social sciences [47, 39], and topic identification [42, 106, 91]. Twitter provides a comprehensive Application Programming Interfaces (APIs)[2] for developers and researchers to access tweets or data. It includes the Streaming API[3] to retrieve a sample of public tweets in real time, and a REST

[1]According to the Digital Policy Council (DPC) annual report on 2015 World Leader Ranking on Twitter [21], a total of 139 world leaders from 167 countries have an account in Twitter.

[2]https://dev.twitter.com/overview/documentation, accessed 3 August 2017

[3]https://dev.twitter.com/streaming/overview, accessed 3 August 2017

API[4] for accessing historical tweets. These specific APIs are important to provide access to data for deriving topics in the Twitter environment.

2.2 Deriving Topics from a Collection of Documents

In general, a topic can be defined as a set of stories which are linked by some seminal real-world event [3]. A topic of a specific music festival in the town could include, for example, reviews of the musicians that will perform on the stage, prices of the ticket, or even the security issues at the event. For a document collection, a topic is formally defined as a distribution over a fixed set of terms (vocabulary), where each document in the collection itself is a mixture of a set of topics [9]. Thus, topic derivation from a collection of documents can be defined as the unsupervised task of characterizing the main topic of each document in the collection (cluster documents based on their main topics), and listing the most important keywords to represent each discovered topic.

The task of topic derivation from a collection of documents has long been studied. One of the earliest approaches to reveal the latent topics from a document collection is the *Latent Semantic Analysis (LSA)* [25]. LSA takes the advantage of the relationship between the documents and terms represented in the term-document matrix by decomposing the matrix into its lower representation using the singular-value decomposition (SVD) method.

In 1999, Hofmann presented the extension of LSA called *Probabilistic Latent Semantic Analysis (PLSA)* [41] to deal with the different meaning and types of words. In 2000, the study of [62] investigated the properties of a method for matrix decomposition called *Non-Negative Matrix Factorization (NMF)*. The method is now widely adopted for various matrix dimensional reduction problems, including document clustering and system recommendations. Later in 2003, the study of [8]

[4] https://dev.twitter.com/rest/public, accessed 3 August 2017

introduced the *Latent Dirichlet Allocation (LDA)*, which is currently considered as the state of the art method in the topic modeling area. LDA is a fully generative method, like PLSA, in which a document is a mixture of topics. These four major methods in topic derivation share common property to be able to find the k number of latent features (topics) through a dimensional reduction process. Each method is discussed in turn in the next subsections.

2.2.1 Latent Semantic Analysis (LSA)

LSA [25] is a text mining approach proposed to derive the latent semantic structure of a document collection. It was designed to deal with the inability of existing techniques to retrieve information to take account of conceptual content rather than just matching words to queries. In this work, Deerwester et al. [25] highlight two issues pertaining to words matching that penalize the precision of the result: *synonymy* and *polysemy*. Synonymy is described as the use of various words to refer to the same object. Polysemy is the fact that a word can have more than one meaning, or can refer to more than one object.

LSA uses the relationship between documents and all unique terms (vocabulary) from the document collection to take the conceptual content into account. It constructs a term-document matrix V and performs matrix decomposition on this matrix to derive the k number of latent structures. LSA utilizes Singular Value Decomposition (SVD) [31, Chapter 9] to decompose the term-document matrix into its lower dimensional representation.

In LSA, SVD is viewed as a method for inferring a set of indexing variables to determine the latent structures. LSA simplifies the SVD method by taking only the first k largest singular values so that the matrices produced by the decomposition process are of rank k. The term-document matrix decomposition in LSA is formulated as follows:

$$V = TSD^T \tag{2.1}$$

where V is the term-document matrix with the size of $t \times d$ (t is the number of unique terms in the document collection, and d is the number of documents). Matrices T and D^T represent the rank k lower dimensional matrix V. $k \leq \min(t, d)$ is the number of expected latent structures. T and D^T have the size of $t \times k$ and $k \times d$, respectively. S is the diagonal matrix of singular values with the size of $k \times k$.

In a document collection, the latent structures derived by LSA can be referred to as topics. Since matrix V is the representation of the relationship between documents and the unique terms available in the document collection, the matrix T can be viewed as the representation of term-topic relationships. The matrix D^T then can be viewed as the representation of relationships between the topic and document. LSA approach has been successfully implemented for various applications, including document clustering [76, 6] and language modeling [34, 5].

2.2.2 Probabilistic Latent Semantic Analysis (PLSA)

PLSA [41] was introduced in 1999 to improve the performance of LSA. PLSA is claimed to have a more solid statistical foundation than LSA and is defined as a generative data model. The LSA model employs the Frobenius norm approximation in its objective function to get the most optimal decomposition result, which allows for negative values on the main matrix. In contrast, PLSA employs the likelihood principle for its objective function, and the model only allows positive entries to optimize the 'bag-of-words' based data modeling approach.

PLSA derives the statistical latent class model as a mixture decomposition model. For a document collection, the latent variables of the model can be considered as the topics. In PLSA, the probability of the co-occurrences between documents and words ($P(d, w)$) is generated independently as a mixture of conditionally

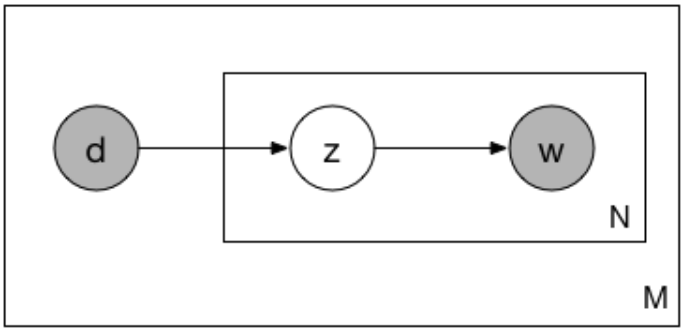


Figure 2.2: PLSA model on a plate notation

multinomial distributions:

$$P(d,w) = P(d)P(w|d) \tag{2.2}$$

$$where \quad P(w|d) = \sum_{z \in Z} P(w|z)P(z|d) \tag{2.3}$$

In the above equations, w is a word in the vocabulary $W = \{w_1, ..., w_N\}$, and d is a document in a document collection $D = \{d_1, .., d_M\}$. z is the unobserved variable in the latent class $Z = \{z_1, .., z_K\}$. Figure 2.2 shows the plate notation representation of the PLSA model. From this plate notation, we can see the process of generating z as the latent variable from the multinomial topic distribution in document $P(z|d)$, and w which is drawn from the word-topic distribution $P(w|z)$.

2.2.3 Non-negative Matrix Factorization (NMF)

NMF is a method to decompose a matrix into its lower dimensional matrix representations. The method only allows positive values for all involved matrices, including the decomposed matrix and the resulted matrices. NMF becomes popular after Lee and Seung [62] investigated two different multiplicative algorithms (Least Square Error and Kullback-Leibler divergence) for NMF implementation. NMF has been applied in numerous domains, including unsupervised clustering [56, 37, 51, 109], recommendation system [143, 71, 45, 72, 144], topic derivation [136, 87, 89],

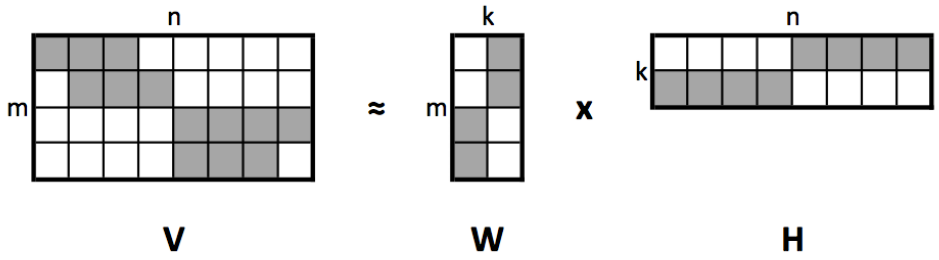


Figure 2.3: Non-negative matrix factorization process on document-term matrix $V \in \mathbb{R}^{m \times n}$ to derive the latent structures on factor matrices $W \in \mathbb{R}^{m \times k}$ and $H \in \mathbb{R}^{k \times n}$.

image processing [63, 43, 139, 20], and bioinformatics [26, 50, 116].

For a document collection, NMF is able to uncover the hidden thematic structures of the collection by finding the factor matrices approximation for a document-term matrix. The document-term matrix represents the relationship of each document to every unique term in the document collection. The factorization process can be formulated as follows:

$$V \approx WH \tag{2.4}$$

Figure 2.3 illustrates the NMF method. Let the $V \in \mathbb{R}^{m \times n}$ be a document-term matrix with the size of $m \times n$ (m is the number of documents and n is the number of unique terms), the product of matrices $W \in \mathbb{R}^{m \times k}$ and $H \in \mathbb{R}^{k \times n}$ is the approximation to the matrix V. In this process, rank $k < \min(m, n)$ can be considered as the number of expected latent topics. The main topic of each document can then be determined by choosing the maximum value on each vector in matrix W, and the x number of keywords to represent each topic can be chosen by taking the $top\ x$ values from each vector in matrix H.

In NMF, the value of elements in three matrices V, W, H are all positive. This non-negativity feature is a useful constraint in NMF to allow only additive combination in the factorization process [63]. NMF is considered to be equivalent to PLSA method

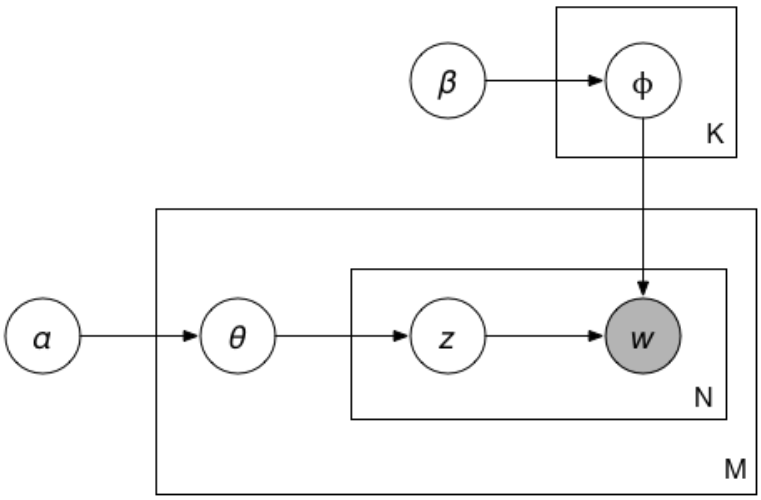


Figure 2.4: LDA model on a plate notation

when the *Kullback-Leibler (KL) divergence* [58] is used as its objective function [32]. More detail of Non-negative matrix factorization will be discussed in Chapter 5.

2.2.4 Latent Dirichlet Allocation (LDA)

In 2003, the study of Blei et al. introduced the LDA method [8]: a generative probabilistic model for a document collection. Like PLSA, in LDA, each document in the collection is modeled as a mixture over a set of latent topics. However, Blei et al. criticize the PLSA model to be not fully generative, as there is no generative probabilistic model for mixing the proportion of the latent variables, and thus it becomes problematic for unseen documents (those which are outside the training set). Different from PLSA, LDA uses Dirichlet prior for both the distribution of topic in the document collection and the distribution of word in every topic, making it fully generative to infer topics from unseen documents. Since then, LDA becomes very popular, and it is currently considered as the *state of the art* method in the area of topic modeling.

Figure 2.4 illustrates the generative process of LDA on a plate notation. Referring to the figure, we have M number of documents in a collection, α is the dirichlet

prior for the distribution of topics in document θ, and β is the dirichlet prior for the distribution of words in topic ϕ with K being the number of the latent topics and N the number of words in the document. z is the topic assigned to a word w in the current iteration. The generative process of LDA can be described as follows:

1. For each document in the collection, choose $\theta \sim Dir(\alpha)$.
2. For each topic, choose $\phi \sim Dir(\beta)$
3. For every word w_i in the current document:
   (a) Choose a topic $z_i \sim Multinomial(\theta)$
   (b) Choose a word $w_i \sim Multinomial(\phi_{z_i})$

Mathematically, the probability of the LDA is formulated as follows:

$$P(W, Z, \theta, \phi; \alpha, \beta) = \prod_{i=1}^{K} P(\phi_i; \beta) \prod_{j=1}^{D} P(\theta_j; \alpha) \prod_{t=1}^{N} P(z_{j,t}|\theta_j) P(w_{j,t}|\phi_{z_{j,t}}) \quad (2.5)$$

The original LDA model was based on the variational method and Expectation-Maximization (EM) algorithm for the Bayes parameter approximation. Later in 2004, Griffiths and Steyvers introduced the use of *Gibbs sampling* inference strategy as an alternative to the variational Bayes estimation [35]. The work shows that Gibbs sampling implementation on LDA model is simple and more efficient in memory in comparison with the previous approach.

Further studies show that, in particular situations, LDA is equivalent to PLSA. The study in [33] presents the relationship between PLSA and LDA. It shows that PLSA is in fact a maximum likelihood (ML) estimated LDA model under a uniform Dirichlet prior. The work in [78] compares LDA and PLSA as a dimensionality reduction methods for the task of document clustering. It finds that both LDA and PLSA are far superior to a random projection. However, it does not find any meaningful

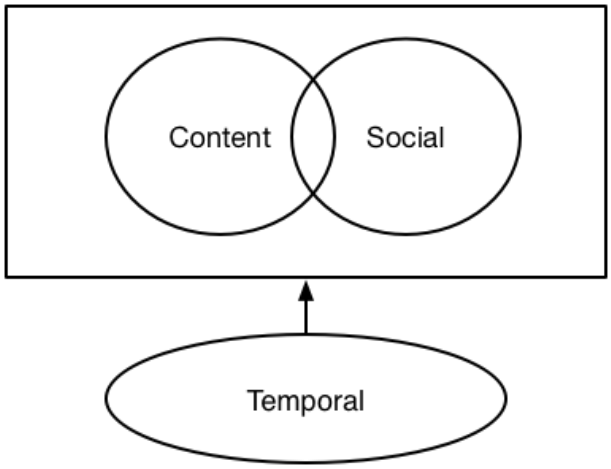


Figure 2.5: Features used for topic derivation in social media, especially Twitter

difference between LDA and PLSA for a dimensionality reduction problem. More detail about the LDA model will be discussed in Chapter 4.

2.3 Deriving Topics from Twitter

The major techniques for topic derivation discussed in the previous section were mainly focused on various semantic relationships of words in documents. The methods have been applied and extended for many types of (lengthy) documents such as email [79, 125, 80], academic papers [27, 10], and web pages [69, 105]. However, social media poses other challenging problems. First is the severe sparsity of content. Posts are often very short and include many irrelevant characters or terms such as emoticons and misspelled words. They could lead to an extremely low number of overlapping terms within a collection of tweets. Defining text-based semantic relationships for topic identification thus becomes more problematic. The next challenge is the dynamics of the social media platform. With the speed of information propagation and the large number of incoming tweets, identifying topics on Twitter is a non-trivial task. A topic can quickly grow, decay, or even merge with another topic.

In this section, we review key studies that focus on deriving topics from a social

media platform. Most of the works are still based on the major methods discussed in the previous sections. Many extensions have been proposed to take advantage of the unique features offered by social media to derive high quality topics. As shown in Figure 2.5, we classify the features that are often incorporated for topic derivation into three categories: *content, social, and temporal*. We discuss each feature and related existing methods in the subsections below.

2.3.1 Focus on Content Exploitation

Despite the extreme sparsity of the posted messages, a lot of studies still focus solely on the exploitation of content. Some have simply applied well defined methods such as PLSA, LDA, and NMF. Others have built on them to include the content merging (merge several or all tweets into a single entity), content expansion (expand every tweet with external resources), or the incorporation of various semantic relationships between terms in the collection of content. In this section, we review studies that primarily focus on content for topic derivation.

The direct application of the major topic derivation methods has shown a relatively good performance on specific Twitter datasets. The work in [102] successfully applies the LDA method to uncover topics in a public-health related Twitter data. It uses a dataset based on tobacco-related terms such as '*smooking*', '*tobacco*', '*cigarette*', '*cigar*', '*hookah*', and '*hooka*'. The study of [53] proposes a modified LDA method to identify topics from disaster-related tweet collection. Instead of using an equal weight for the distribution of the topics in a document (θ), Kireyev et al. used a word's *specificity* weighting scheme, where more specific words will have a higher weight in the topic assignment process, to deal with the sparsity problem. The original LDA method is also used in the work of [126] to extract events from Twitter for an automatic crime prediction, focusing on hit-and-run cases.

Other works find that merging the content from each tweet could bring a positive impact when dealing with the short content in Twitter. In the work of Weng et al.

[130], all tweets' content are aggregated into a single big document to be processed with the original LDA method. The derived topics are then used as a factor for identifying influential Twitter users. The study of [42] conducts an empirical study of topic modeling in Twitter using the LDA model and LDA extension author-topic (AT) model [111]. Based on the experiment against the dataset, the study finds that aggregating the content of tweets could improve the effectiveness of the trained topic models. It concludes that the performance of the standard LDA approach in the Twitter environment is better than its extension author-topic model.

Quite a few approaches employ various techniques of using external resources to tackle the low level of terms co-occurrences in a sparse text environment. The work in [98] and [99] propose a method to alleviate the sparsity by converting an external knowledge base as an additional "universal dataset" to enhance the short content. Hu et al. propose a method to employ a hierarchical three-level structure for short text clustering by integrating multiple semantic knowledge bases such as *Wikipedia* and *WordNet* [44]. Jin et al. propose the *Dual Latent Dirichlet Allocation (DLDA)* model to infer the topics from Twitter data, with the help of auxiliary lengthy datasets like *Wikipedia* content through a joint transfer learning process [46].

The work in [70] proposes a method to identify the topics from Twitter data by expanding the query using terms generated from *Freebase* as the knowledge base. It chooses Freebase as the main external resource as it consists of data harvested from various other sources like the Semantic Web and Wikipedia. Furthermore, the structure of Freebase generally represents human knowledge. The study of Nguyen et al. in [85] incorporates latent feature vector representations of word into two different dirichlet multinomial topic models (LDA and Dirichlet Multinomial Mixture (DMM) [86]) for topic modeling in short documents. The vector representation used in the approach is trained on a very large corpora. The method proposed in [140] utilizes the title of articles in Wikipedia to represent the topic for every post in Twitter. In that work, topic identification relies mainly on *TF-IDF* and cosine

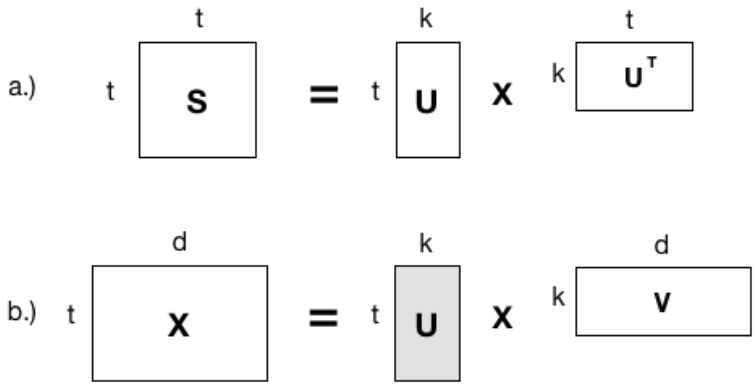


Figure 2.6: Learning topics using two steps matrix factorization process [136]. In the first step, the term-topic matrix U is inferred from the factorization of the term correlation matrix S. The observer term-topic matrix U is then used to learn the topic document matrix V in the process of factorizing the term-document matrix X in the second process.

similarity computations of both the Twitter datasets and Wikipedia title collection.

A method proposed in [136] explores the correlation between terms in the dataset for learning the topics from a sparse environment like Twitter. It reports that the term correlation matrix is much denser if compared to the generally used term-document relationship matrix. The term correlation matrix is considered to be more capable to capture the latent structure for topic identification. Figure 2.6 shows the topic learning process proposed in [136]. It employs a two steps matrix factorization process. The first step is the factorization of the term correlation symmetric matrix S to infer the term-topic matrix U. The second factorization is used to solve the topic-document matrix V by using the observed term-topic matrix U when factorizing the term-document matrix X. Experiments with the TREC2011[5] Twitter dataset and several other short-text type datasets show that the proposed method is able to outperform the state of the art LDA model and other baseline methods.

Similar to the work in [136], the study of Ma et al. performs the factorization of

[5]http://trec.nist.gov/data/tweets/

the term correlation matrix to obtain the term-topic matrix as the first step of the topic derivation in the microblog environment [73]. However, for the second step, instead of using the NMF method, they employ the PLSA model on the term-topic matrix to infer the relationships between document and topic.

The works in [135, 17] propose the *biterm topic model (BTM)* for modeling topics in short text environments. BTM directly models the co-occurrence of words patterns in the whole document collection to enhance the process of topic derivation. It uses the aggregated word patterns to deal with the sparsity problem.

Xu et al. employ the BTM method to get the word co-occurrence pattern model as a parameter in the proposed semantically similar hashing method (SSHash) [132]. The hashing method provides fast and efficient matching techniques for mining semantically similar topics in the short text environments. The work in [64] integrates *K-means algorithm* into the BTM approach to derive topics from the dataset. First, BTM is applied to infer potential topics from the dataset, and next, K-means is used to get topic-based clusters.

The semantic relationship between words for topic modeling is also explored in [95, 146]. The study of Ozdikis et al. implements semantic expansion techniques based on the statistics of co-occurrence words in a tweet collection [95]. Recently, the work in [146] proposes a word co-occurrence network-based model to deal with the sparsity problem. The method uses the sliding window technique to build the network of words where any two distinct words from a document occurring in the same window will be considered as connected to each other. The resulted network of words is then turned into a pseudo-document set and processed with the Gibbs sampling for LDA [35] to observe the latent topics.

2.3.2 Incorporating Social Features

Unlike the other types of short text (e.g., collections of titles, *RSS*, instant messages, image captions), social media platforms provide features to interact with other

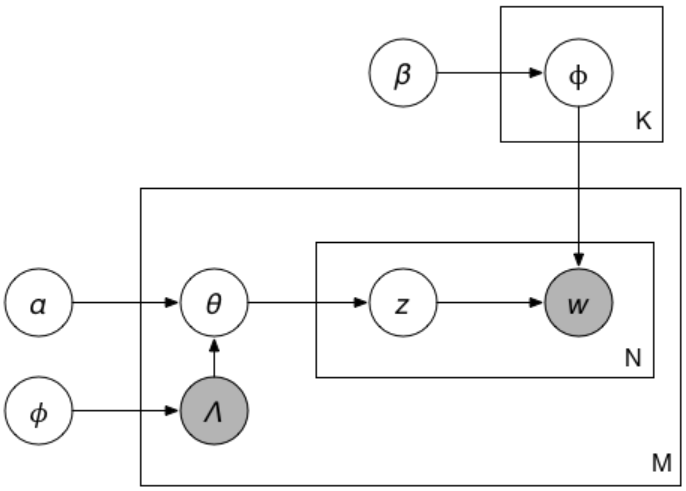


Figure 2.7: Plate notation for the Labeled LDA model [106, 105]

users or explicitly refer to events. *Mentions*, *replies*, *retweets*, and *hashtags* are some examples of social features that are popular amongst Twitter users. A *mention* is generally used to initialize a conversation with other users, or to involve other users into the current discussion about a particular topic. Other users can reply to or re-share someone's post. A hashtag is a specific term starting with '#' to tag the tweet. A hashtag in a tweet generally refers to a particular discussion, location, or event. Researchers find that exploiting such features along with the content can improve the quality of the derived topics in social media environment.

Ramage et al. address the problem of characterizing the information in microblog with the help of a topic modeling approach [106]. The work implements the *Labeled LDA* method [105] to analyze the content of Twitter posts. Labeled LDA is an extension of the LDA method that incorporates labels to partially supervise the learning process. *Hashtags*, *replies*, *@users*, and *emoticons* are used as predefined labels. Figure 2.7 shows the plate notation of the Labeled LDA model. The model assumes that each tweet will use only some labels from a set of labels Λ with hyperparameter Θ. It allows the modeling of a collection of tweets as a mixture of some labels and also as a combination of latent topics as in the original LDA method.

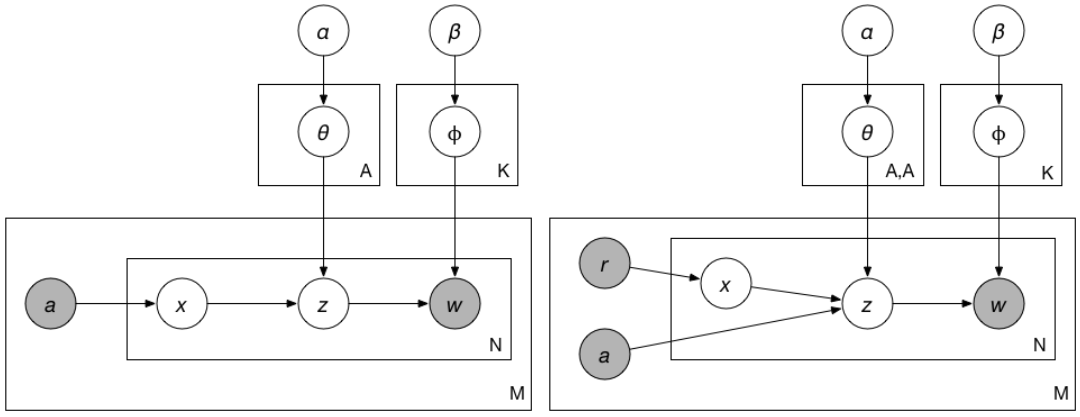
The work in [81] investigates the methods to improve the original LDA when applied to Twitter. The paper proposes a combination of pooling scheme (to get more coherent input for the LDA learning process) and automatic topic labeling (to further improve the results of identified topics). Hashtags are used for both pooling the tweets to build the aggregated text and for labeling the derived topics automatically. The work in [101] incorporates another method for pooling to improve the input of LDA process, based on a community detection approach by aggregating content from groups of users who have common interests and interactions.

The proposed method in [36] incorporates hashtags as a specific feature of tweets along with external news entities to help extracting the text-to-text correlation to enrich the short text data. The study of Wang et al. proposes a hashtag graph based topic model to discover more distinct and coherent topics in Twitter [129]. In that work, a hashtag is used as a weekly-supervised information point to model the topic.

The work in [118] clusters the short messages into general domains relying on social tagging features such as hashtags. The clustering process is broken down into two steps. The first step is to use a collection of hashtagged tweets to achieve stable clusters based on the hashtags. The clusters are then incorporated into the second step to do the clustering of tweets which mostly are not tagged.

Ma et al. propose *Tag-Latent Dirichlet Allocation* (TLDA), an extension of LDA that incorporates the observed hashtags as a mixture of topics into the process [74]. The study of [122] proposes a unified framework that integrates social aspect and external resource as additional information to model the multifaceted topics in Twitter. The framework extracts all hashtags from the tweet collection as social semantics, and retrieve the top k terms from the web documents included as URL in tweets as the auxiliary semantics. Both social and auxiliary semantics are then used to enrich the content for the topic identification process.

Chierichetti et al. investigate the behavior of tweets and retweets when a particular event is happening [19]. The tweets and retweets form a "heartbeat" pattern



(a) (b)

Figure 2.8: Plate notation of (a). Author-Topic (AT) Model [111] and (b). Author-Recipient-Topic (ART) Model [79]

.

that can be observed for event detection. This work finds that looking only at the volume of tweets and retweets, an event being discussed on Twitter can be more accurately detected than their baseline methods.

The study of Rajani et al. in [104] reports on the comparison of the application of original LDA, the Author-Topic (AT) model [111], and the Author-Recipient-Topic (ART) model [79] to extract topics from Twitter data. Both the AT and ART models are built on the original LDA model. Figure 2.8a and 2.8b show the plate notation of the AT and ART models respectively. In the AT model, each document is assumed to have a set of observed authors a. For every document, author x is sampled from the set of authors a, and the topic z is sampled from the distribution of the authors over topics θ. In the ART model, each document is assumed to have both set of observed authors a and set of observed recipients r, and the process of topic sampling is influenced by both a and r. In Twitter, the author is the user who posts the tweet itself, and the recipient is the mentioned user in the tweet. The research finds that deriving topics with ART model presents the best performance, followed by the original LDA and AT respectively. However, ART model can only be applied to tweets that involve the mention feature as the recipient of the tweets.

The work in [103] proposes a behavior-topic-model (B-LDA) to obtain topics from social media environments like Twitter. The proposed approach jointly models the users' social-based behavioral pattern and their interests in topics. Xia et al. propose another LDA extension (Plink-LDA) to incorporate link or similarity between documents to improve the quality of topic model [131]. In Twitter, the link between posts can be derived from hashtags or URLs. The link information is then used to control the topic sampling process along with the document collection itself.

2.3.3 Incorporate Temporal Aspect

In social media environments, users' posts continuously arrive as they are posted, and topics change rapidly. In Twitter for example, a tweet posted by a user might not be about the same topic as the tweet posted by the same user several hours earlier. When a specific event happens, users tweets could be about the same topic during the time of the event, but the discussion can move quickly to various topics in periods when there are no major events. The fact that topics can rapidly change makes this environment very dynamic. Thus, the topic derivation methods applicable online (in real time) need to take the temporal aspect into account.

The study of Cataldi et al. proposes a real-time topic detection method, aimed especially to observe the most emergent topics in Twitter [14]. The approach includes the process of modeling the term life cycle according to the novel aging theory to automatically identify coherent topics across the different time intervals.

The work in [61] proposes a variant of the Online-LDA method [4, 40]. In this model, new tweets are partitioned based on discretized time slices. The key difference between the approach and the Online-LDA method is that, in Online-LDA, the vocabulary is assumed to be fixed. In the Lau et al.'s approach, the vocabulary is regenerated at each update by adding new incoming words and remove existing words with a frequency below a particular threshold.

Saha and Sindhwani introduce a variant of Non-Negative Matrix Factorization

method that works in an online environment like Twitter [107]. Temporal regularization is used in the matrix factorization process to capture topics from the stream of incoming posts. The study in [128] presents Temporal-LDA (TM-LDA), an extension of LDA to mine the text streams in social media. Specifically, TM-LDA learns the parameters for topic transitions dynamically when new messages arrive.

Finally, the work in [16] develops an incremental clustering framework to derive topics and to characterize the emerging topics from the Twitter online environment. Starting with the crawling strategy to obtain more organized data, the proposed method employs temporal sequence features to detect the emerging topics in a semi-supervised way. The work in [28] proposes a non-parametric Topics over Time (npTOT) method to model the time-varying topics from a corpus that spans a long time period. The proposed method employs Gibbs sampler based on the Chinese restaurant franchise approach [117]. The evaluation is conducted against a dataset of tweets obtained by the authors between January to March 2011, originating from Egypt. The study in [112] proposes the SAX* algorithm for discretizing the temporal series of terms to get the patterns of collective attention to discover events in Twitter.

2.4 Discussion

In this chapter, we looked at the task of topic derivation in Twitter and presented the review of its key techniques and features used to improve the quality of the derived topics. We first provided the insight into why Twitter is an important source of data for topic derivation work and why deriving topics in Twitter is challenging. We then reviewed the popular and state of the art methods to derive topics in a document collection, followed by a review of key studies focusing on deriving topics from the Twitter environment.

LSA, PLSA, NMF, and LDA are the major techniques to derive topics from a

document collection in an unsupervised way. PLSA is proposed to improve the performance of LSA, providing a more solid statistical foundation than the LSA method. NMF is a method to decompose a matrix into its lower dimensional representations. For a document collection, NMF is able to reveal the latent structure of the documents by finding the factor matrices for the document-term relationship matrix. A study of [32] show that NMF is equivalent to PLSA when KL-divergence is used as an objective function. LDA is a fully generative method to uncover the hidden topics from document collection, and thus makes it more flexible when dealing with unobserved documents.

The above mentioned methods work solely on the document content. A lot of extensions have been proposed to address the sparsity that arises in social media environments like Twitter. Extensions include: incorporating more text, adding social features, and taking the temporal aspect into account. Based on our review of the current key studies on topic derivation in Twitter, we observe the following:

- Methods that rely entirely on the tweet content still suffer from the sparsity issue. The density of the co-occurrence of terms matrix in a tweet collection can be as low as 0.274% on average [91]. With these very low rates of overlapping terms, exploiting various semantic relationships to derive topics solely from internal content will less likely be effective for providing significant improvements over the state-of-the-art methods.

- Augmenting the short text data with auxiliary content from external resources seems to be a promising solution. However, the newly added terms inferred from the resources often include noise and are often unrelated to the context. It thus can be harmful to the learning process [11]. The informal language used in tweets with a lot of misspelled words and abbreviations, can itself be very challenging for the matching process with the auxiliary content. Furthermore, relying on external resources faces scalability issues, as it could bring an extra

burden when dealing with a highly dynamic environment like Twitter.

- Most methods that incorporate social features still focus on content based interactions such as hashtags. Hashtags are important and often used by users to participate in discussions for a particular topic. However, they are still part of the tweet content, and most of the tweets do not include hashtags. The methods thus still suffer from the sparsity issue. Some methods try to include the tweets author and/or recipients. However, unlike in a specific type of document such as academic papers or news articles, where authors have a strong relationship with topics, in Twitter, a tweet is authored by only one user, and a user can post tweets in various topics. Furthermore, if a method requires recipients information to be available for the learning process, the method will not be suitable for the majority of the tweets, as most of them do not contain users' mention.

- In a highly dynamic environment like Twitter, time is an important feature to deal with varying topics, especially in real time. Most methods that incorporate temporal feature still view the time aspect as a time slicing window to specify the interval of the serial or incremental learning process over time. Time aspect is not yet seen as a factor that can improve the quality of topic derivation for a static document collection.

Combined with semantic relationships of tweets content, more complex social interaction features need to be incorporated to deal with the sparsity issue. Moreover, the temporal aspect in Twitter should be considered as an important factor even in an offline situation. The relationships between time and the interaction features should be investigated to make sure that the proposed method can also handle the dynamic environment, both for static collections of tweets or for analysis in real time.

In the next chapter, we will describe the datasets and evaluation metrics used throughout the thesis. Datasets are an integral part of the thesis, as we use them for both investigating the characteristics of the relationships between the three aspects (content, social, and temporal) and evaluating our proposed methods.

Chapter 3

Datasets and Evaluation Metrics

In this chapter, we discuss the details of the datasets and metrics used for evaluation throughout the thesis. In the first section, we describe the Twitter Application Programming Interface (API) used to obtain the datasets. The second section discusses Twitter datasets that will be used for analysis and evaluation of our proposed methods. The evaluation metrics are discussed in the following section, where we look at how to evaluate the quality of the derived topics.

3.1 Twitter API

Twitter provides several *Application Programming Interfaces (APIs)* to interact with it programmatically. In this section, we discuss two APIs we used to obtain datasets: the *REST API* and the *Streaming API*.

3.1.1 REST API

To maintain the integrity of its product, Twitter has a specific policy for the developers. Section 6 of the developer agreement and policy[1] presents a guide for Twitter dataset providers, which limits the distribution of datasets to only list the tweet IDs and/or user IDs, without any content or metadata. This policy is aimed to respect a Twitter users' control and privacy on their data, so that if a user edits or deletes

[1] Section I.6.b of Twitter Developer Agreement & Policy, "*If you provide Content to third parties, including downloadable datasets of Content or an API that returns Content, you will only distribute or allow download of Tweet IDs andor User IDs*", https://dev.twitter.com/overview/terms/agreement-and-policy, accessed 15 July 2017

his/her tweets, the developer will have the latest update of the tweets or will not be able to access a tweet that was deleted. It means all existing Twitter datasets available online for research purposes only list the tweet IDs. To retrieve the content of the posts from their IDs, we need to use the *REST API*.

The *REST API* gives a read/write access to the Twitter data. Some examples of the methods are: searching for a particular tweet, retrieving a Twitter user's profile, follower data or users timeline, and publishing a tweet. Although this API is able to read Twitter data, it is not intended to be used in real time. Unlike the *Streaming API*, the *REST API* does not need a continuous connection to the Twitter server, and all queries are submitted and addressed individually.

To obtain a specific tweet based on its ID, we use the "*GET statuses/show/:id*" method. This method requires the tweet ID as the main parameter, and it returns the corresponding tweet in *JSON-encoded* object[2]. The attributes of this JSON-encoded object follow the standard tweet payloads field guide as described in [121].

3.1.2 Streaming API

With thousands of tweets sent every seconds, the real-time access to the Twitter data for analysis has gathered interest from many people, including academics, business, and governments. A Streaming API is provided by Twitter to accommodate this need. This API gives a free and low latency access to samples of near real-time data flowing through the Twitter server.

The Streaming API requires a continuous connection to Twitter from the user's server. Once a connection is opened and accepted by Twitter, tweets are streamed in near real time until the connection is closed. During the streaming, streamed tweets can be saved and/or processed by the user's server. Figure 3.1 illustrates the minimal streaming mechanism.

[2]JavaScript Object Notation, a lightweight data-interchange format which is built on two structures: a collection of name/value pairs and an ordered list of values. http://json.org/

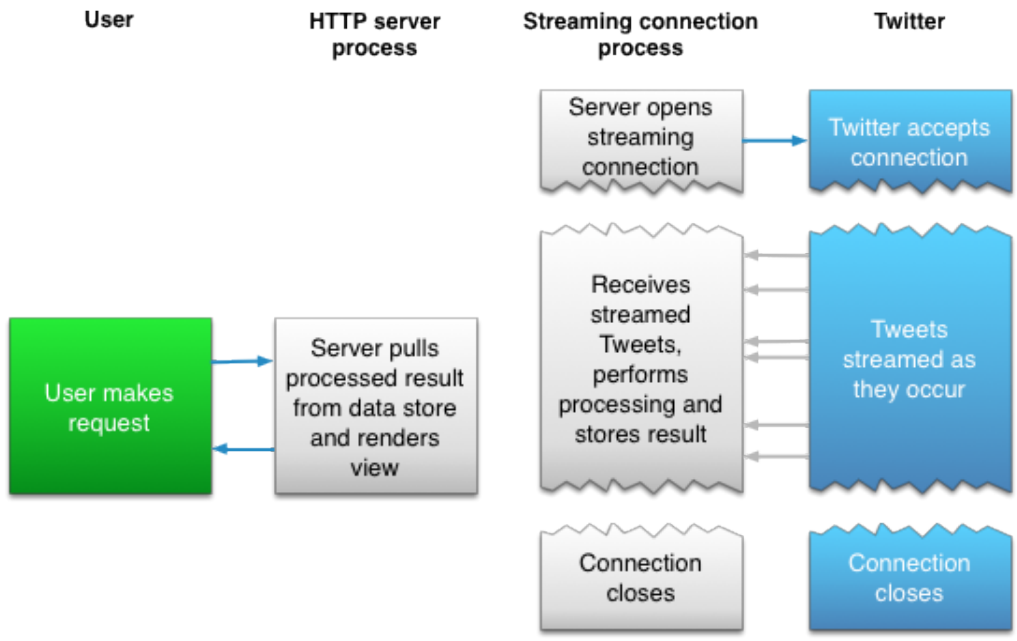


Figure 3.1: Streaming API process [121]

Twitter offers several streaming endpoints [121]: *Public Stream*, *User Stream*, and *Site Stream*. The *Public stream* is the Streaming API endpoints for public data. It is used to stream all the tweets that are public based on user request parameters. Different from the *Public Stream*, the *User Stream* only provides a stream of tweets specific to a single and authenticated user rather than the public data. The *Site Stream* is similar to User Stream but allows more than one user.

To access the Streaming API, HTTP requests must be authorized securely according to the *OAuth* specification. *OAuth* framework[3] is a standard open protocol to enable third-party applications to get a limited access to services like *Streaming API*.

Each individual tweet streamed by the API is in the form of *JSON encoding*, similar to the output of "*GET statuses/show/:id*" in the *REST API*. The Streaming API is not meant to provide full access to all Twitter data. Currently, only Twitter *firehose* providers can give access to the 100% Twitter data in real time and it is not free. Consequently, the volume of streamed sample data when using the streaming

[3]http://oauth.net/

API is low and not constant. The tweets are not delivered in sorted order, but within a few seconds of a total ordering. There is also a possibility to get duplicated tweets, usually if there is a backfilling process in the streaming API connections.

3.2 Twitter Datasets

To evaluate the proposed methods, we require labeled datasets as ground truth. For this purpose, we use several datasets: *TREC microblog datasets*, *tweetSanders* dataset, and our own *tweetMarch* dataset. TREC microblog datasets and tweetSanders are available online, and widely used by researchers in the area of social network analysis to evaluate their proposed methods. Each of these datasets has different characteristics, especially related to the availability of interaction features, number of topics involved, and the density of term co-occurrences.

3.2.1 TREC Microblog Datasets

This dataset is provided by *The Text REtrieval Conference* (TREC), a community co-sponsored by the National Institute of Standards and Technology (NIST) and U.S. Department of Defense. The TREC community regularly releases labeled datasets for research purposes, including a Twitter dataset with topics labels for the microblog track. For analysis and evaluation purposes, we use the TREC 2014 microblog track version, which was the latest microblog dataset at the time experiments were conducted. This dataset is denoted as TREC2014. It is available online at http://trec.nist.gov/data/microblog2014.html.

TREC2014 consists of more than 50,000 tweets posted between 31 January 2013 and 31 March 2013. This dataset is built based on the tweet timeline generation (TTG) [65], which is to cluster relevant tweets ordered chronologically. It offers varying topics to represent the dynamics of the Twitter environment, where the number of tweets for each topic is changing over time. Each of the tweets in this

Table 3.1: Example of topics and their related tweets in the *TREC2014* dataset

Code	**Topic**	**Related Tweets**
MB171	*Ron Weasley birthday*	• Happy birthday to my boy Ron Weasley. • :-* "@Potteristic: Happy Birthday to our King Ron Weasley! http://twitpic.com/c7nouh". • It's Ron Weasley's birthday! The ginger who vomited slugs out from his mouth' happy birthday Ron! #RonWeasleyBirthday.
MB178	*Tiger Woods regains title*	• Woods moves in front of McDowell: Tiger Woods moves into a two-shot lead over Northern Ireland's Graeme McDowell at the World Golf Ch... • Tiger woods is number one in the world again! I don't like golf but I love the sound of this! • Woods back on top in golf: ORLANDO, Fla. (AP) - The moment was vintage Tiger Woods, and so was his reaction.Se... http://nbcnews.to/YuD3rn
MB192	*Whooping cough epidemic*	• Whooping cough cases 'falling': The largest outbreak of whooping cough for 20 years shows signs of slowing as cases fall for two mont... • #WhoopingCough news: Whooping cough rates spike: Rates of the deadly disease whooping cough have ... http://bit.ly/11RFpka #pathogenposse • Health News Daily: Whooping cough vaccine protection wanes http://u.robinspost.com/33tgMc

dataset has been annotated to one of 55 available topics. Some examples of the topics and their related tweets are shown in Table 3.1.

To download the data, we use *Twitter REST API* discussed in Section 3.1.1. Unlike the *streaming API* that allows a persistence connection to the Twitter server, *REST API* has a rate limit. The user is only allowed to send 180 queries per 15 minutes. To deal with this limit, we put 10 seconds delay before requesting data for the next

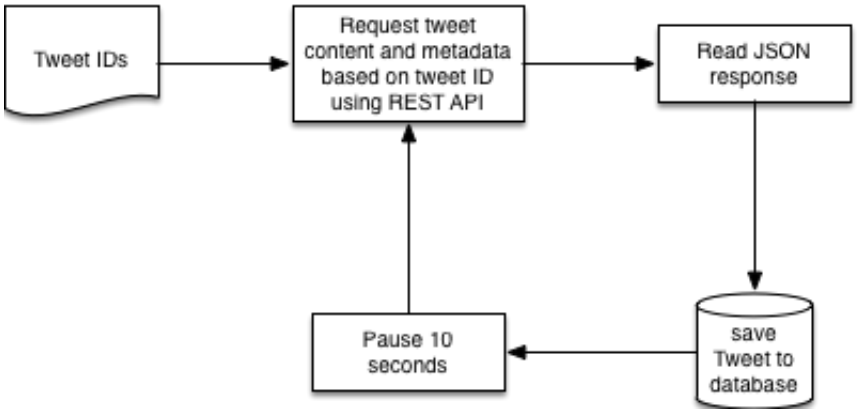


Figure 3.2: Process of reading Twitter data based on Tweet IDs using *REST API*

tweet ID. The whole process of reading the Twitter data based on a set of tweet IDs can be seen in Figure 3.2.

From the list of the ID available in TREC2014, only 46,572 tweets can be downloaded. This could be due to many reasons: the tweet might have been deleted or the status of the tweet might have been changed to protected. The 46,572 downloaded tweets were authored by 35,670 users. This dataset has no *retweet* and only 3,463 of them are *reply* tweets.

3.2.2 tweetSanders

This dataset is available online and free to download from http://www.sananalytics.com/[4]. It includes over 5,500 tweets, each manually classified as belonging to one of four different topics (Apple, Microsoft, Google, Twitter). Only 4,572 tweets can be downloaded. We denote this collection of tweets as *tweetSanders*. It has 297 *reply* tweets and 269 *retweets*. The tweets are from 3,711 different users. Table 3.2 shows the four topics and samples of tweets for each of them.

[4]accessed January 20, 2014.

Table 3.2: Examples of tweets for each topic in tweetSanders

Topic	Tweet Code	**Tweet**
Apple	*a1*	Houston we have a problem!! My iPad has been restoring for 12+ hours after installing @apple IOS5. This can't be right....
	a2	hmmmm a lot of #siri feature don t work in canada location and direction seriously come on
	a3	#ios5 is nice and a it had to be thanks
Microsoft	*m1*	#Microsoft shows 'touch screen' for any surface | Nanotech - The Circuits Blog - CNET News http://cnet.co/oQKvoG via @cnet
	m2	Jus updated my computer to Windows 7 .....I'm on thanks to #microsoft
	m3	#Microsoft CEO Steve Ballmer on Not Buying #Yahoo: âĂIJSometimes, YouâĂŹre LuckyâĂİ http://goo.gl/fb/KIrVu #uncategorized
Google	*g1*	#Android #Google Samsung and Google introduce GALAXY Nexus http://bit.ly/qfXlSU #DhilipSiva
	g2	The Samsung Galaxy Nexus and Ice Cream Sandwich are sick! #android #icecreamsandwich #google
	g3	Google is gonna need to do better than this to beat #iOS #Android #icecreamsandwich #Google http://youtu.be/android
Twiter	*t1*	62 Ways to Use #Twitter for Business: http://bit.ly/smbiz60 #tweets #socialmedia
	t2	My Facebook messed up and I had to make a new one so... add me! Haha at least twitter is reliable
	t3	my cute friend finally got a #twitter

3.2.3 tweetMarch

The *TREC2014* and *tweetSanders* datasets have been widely used by many researchers to evaluate their methods. These datasets have been annotated, and they can thus be used as gold standards in evaluating topic-based tweet clustering. However, it seems that only important tweets related to the assigned topic were included in the datasets.

To get more varied tweets, including both well-structured and tweets with misspelled words or with full of emoticons, URL and other noise, we collected data using the Twitter *Streaming API*. As discussed in Section 3.1.2, this API offers a free access to a sample of the global stream of tweet data flowing through the Twitter server.

Data collection was done between 03 March 2014 and 07 March 2014. Since *track* or *locations* parameter cannot be empty, we supply several keywords on the *track* parameters to determine the tweets that will be delivered in the stream. Since major topics have been made available by the other two datasets, keywords used to retrieve the *tweetMarch* dataset are more related to day-to-day activities and communications, such as: *day, school, uni, book, bus, train, car, bike, traffic, accident, coffee, tea, cake, government, politic*. Only English tweets are included in the dataset. We filter the tweets by using the *language* parameter. The total tweets in this dataset are 729,334, involving 509,713 users around the world. This dataset has 12,221 *reply* tweets and 101,272 *retweets*.

For evaluation purpose, we asked two annotators to label each tweet from a subset of *tweetMarch* dataset, the first 10,000 posts, ordered by the posting timestamp. Each tweet was labeled by both annotators with a topic from 6 available topics: *food, day activities, life expressions, people communications, politics,* and *travel and transport*.

The annotators agreed on 83% of the classifications for the 10,000 tweets. Since

Table 3.3: Examples of tweets for each topic in the tweetMarch dataset

Topic	**Tweet**
food	• @KavadaKedavra coffee with milk and 2 sugars is incredible. Cappucino is nice • I just want coffee and bacon. Is that really too much to ask?
day activities	• Had a fun weekend visiting lilkdm at her uni • If you missed SBS2 coverage of #sydneymardigras, here is the 90-minute show. Brilliant, informative, fun television. http://www.sbs.com.au/ondemand/video/155613763809/Sydney-Gay-And-Lesbian-Mardi-Gras-2014 ...
life expressions	• YOU THINK JUST A SECOND IF LARRY IS FAKE THE EVIL THAT YOU HAVE DONE DURING THESE YEARS? LOUIS IS HAPPY WITH ELEANOR. #EleanorWeStayWithYou • Puberty hit them like a Bus
people communcations	• yes my cousin life there • Are you doing them with miss Abbott cause i think my session are on Tuesday
politic	• @PMO_W @RenzSiniscalchi Kerry says 'Russia is going to lose' if Putin's troops continue to advance in Ukraine • Labour set up National Apprenticeship Week (3-7 March). Join me @ChukaUmunna in supporting it blog: http://www.manufacturingconference.co.uk/blog/post/Manufacturing-talent-what-were-doing-to-back-apprenticeships.aspx ... | #NAW2014
travel and transport	• #boston road construction roadway reduced to one lane on harvard st • @ottawacity lower the bus prices I only have 4 quarters .....

Table 3.4: *Kappa* interpretation based on Landis and Koch [60]

Kappa value	**Strength of Agreement**
< 0	Poor
$0.01 - 0.20$	Slight
$0.21 - 0.40$	Fair
$0.41 - 0.60$	Moderate
$0.61 - 0.80$	Substantial
$0.81 - 1.00$	Almost perfect

this observed agreement is not reliable due to the fact that agreement by chance is not taken into account, we also calculate the *kappa* (κ) value [30] to get annotators' agreement. The *Fleiss' kappa* measures the consistency of rating when several people assign a label to a number of items. The *kappa* is defined as:

$$\kappa = \frac{\overline{P} - \overline{P}_e}{1 - \overline{P}_e} \tag{3.1}$$

$$\begin{aligned} with, \quad \overline{P} &= \frac{1}{Nn(n-1)}\left(\sum_{i=1}^{N}\sum_{j=1}^{k} n_{ij}^2 - Nn\right) \\ and, \quad \overline{P}_e &= \sum_{j=1}^{k} p_j^2 \end{aligned} \tag{3.2}$$

where the quantity $1 - \overline{P}_e$ measures the degree of agreement attainable over and above what would be predicted by chance, and $\overline{P} - \overline{P}_e$ is the degree of agreement actually attained in excess of chance. In equation 3.2, $\overline{P}$ is the observed agreement, N is the total number of tweets, n is the number of annotators, k is the number of topics assigned to each tweet, and $\overline{P}_e$ is the mean proportion of agreement for agreement by chance.

In our case, the observed agreement value ($\overline{P}$) is 0.83, and the mean proportion of agreement for random assignment ($\overline{P}_e$) is 0.24. Thus, the *kappa* value is 0.77, which based on Landis and Koch interpretation [60] shown in Table 3.4, is a *substantial*

Table 3.5: Examples of tweets clustering based on Gold Standard (C) and derived (W)

Tweet Code	Gold Standard (C)	Derived clusters (W)
a1	c_1	w_1
a2	c_1	w_3
a3	c_1	w_1
m1	c_2	w_2
m2	c_2	w_2
m3	c_2	w_3
g1	c_3	w_3
g2	c_3	w_2
g3	c_3	w_4
t1	c_4	w_4
t2	c_4	w_1
t3	c_4	w_1

agreement. Only tweets agreed by both annotators are used for the evaluation process.

3.3 Evaluation Metrics

Topics are obtained from clusters of posts. To evaluate the derived topics, we use metrics appropriate to measure the quality of the clusters, with the labeled tweets as gold data. The metrics are *Purity*, *Normalized Mutual Information (NMI)* and *Pairwise F-measure* [77]. We illustrate each metric with the example shown in Table 3.5. Due to the extreme sparsity of correlation between terms, a statistical analysis of the coherency between words in the topic representation cannot give reliable results for different runs and methods. So, the most important words in each topic will be evaluated qualitatively to see the readability of the words representation.

3.3.1 Purity

Purity [142] evaluates the extent to which tweets are clustered in the same way as in our labeled datasets. The accuracy of the topic assignment is measured by

Table 3.6: *Matching matrix* of the gold standard and the output clusters

	w_1	w_2	w_3	w_4
c_1	2	0	1	0
c_2	0	2	1	0
c_3	0	1	1	1
c_4	2	0	0	1

the number of correctly assigned tweets divided by the total number of the labeled tweets in the dataset.

Let N be the number of labeled tweets in the gold standard, k the number of derived clusters, j the number of clusters in the gold standard. w_i is a cluster in the set of cluster W, and c_i is a cluster in the gold standard set C. The *purity* of cluster W is defined to be:

$$purity(W,C) = \frac{1}{N}\sum_{k}\max_{j}|w_i \cap c_j| \,. \tag{3.3}$$

The result of this metric ranges between 0 to 1. Low quality clustering has a purity value of 0, and a perfect clustering has a purity value of 1.

We illustrate this metric with the tweets in Table 3.5. We first create a *matching matrix*[5] of the gold standard and the output clusters. Each element in the matrix is calculated by looping through each cluster in C and counting how many tweets are correctly clustered in the equivalent derived cluster W.

From the *matching matrix* in Table 3.6, we can calculate the purity by selecting the maximum value from each column in w_i, summing them together and dividing by the total number of tweets. So, the purity of the sample from Table 3.5 is 0.5.

$$purity(W,C) = (2+2+1+1)/12 = 0.5$$

Using this metric, the perfect clustering value of 1 can be achieved regardless of the number of clusters. It means that when the number of clusters (k) is the same

[5]A matrix/table to visualize the performance of the method compared to the gold standard.

as the number of the tweets (N) and each tweet gets its own cluster ($k = N$), the value of *purity* will be 1. A high purity value is easily achieved when the number of clusters is large [77]. Thus, the best way to evaluate the purity is using the same number of topics for both gold standard and the output of the algorithm.

3.3.2 Normalized Mutual Information (NMI)

Purity is a simple measure, but as explained above, a larger number of clusters tends to increase the purity value itself. To measure the trade-off between the quality of the clusters against the number of clusters, we employ *Normalized Mutual Information (NMI)* [113].

NMI measures the mutual information $I(W, C)$ shared between clusters W and the gold standard set C, normalized by the mean of the entropy of clusters $H(W)$ and classes $H(C)$. Similar to *Purity*, the values of *NMI* range from 0 to 1, with larger the values of NMI meaning better clustering accuracy.

$$NMI(W, C) = \frac{I(W; C)}{[H(W) + H(C)]/2} \,. \tag{3.4}$$

In this metric, mutual information $I(W, C)$ quantifies the statistical information shared by the pair of clusters W and C [22], defined in Equation 3.5 below.

$$I(W, C) = \sum_{k} \sum_{j} P(w_k \cap c_j) \log \frac{P(w_k \cap c_j)}{P(w_k)P(c_j)} \tag{3.5}$$

where k and j are the numbers of clusters in W and C respectively. $P(w_k)$ is the probability of a tweet being in cluster w_k, $P(c_j)$ is the probability of a tweet being in cluster c_j, and $P(w_k \cap c_j)$ is the probability of a tweet being in both the cluster w_k and in the gold standard c_j. So, Equation 3.5 is equivalent to the Equation 3.6 below for maximum likelihood of the probabilities as the corresponding relative frequencies [77].

$$I(W,C) = \sum_k \sum_j \frac{|w_k \cap c_j|}{N} \log \frac{N|w_k \cap c_j|}{|w_k||c_j|} \tag{3.6}$$

where N is the total number of tweets in the gold standard, $|w_k|$ is the number of tweets in the cluster w_k, $|c_j|$ is the number of tweets in the cluster c_j, and $|w_k \cap c_j|$ is the number of tweets occurring in both the cluster w_k and the gold cluster c_j.

The minimum value of the mutual information $I(W,C)$ is 0, and the maximum is 1. This maximum value happens if clusters in W exactly recreate the gold standard C. However, a value of 1 is also reached if the clusters in W, while recreating the gold, are further subdivided into smaller clusters. Thus, similar to *Purity*, mutual information still faces a problem about the trade-off between the quality of the clusters and the number of clusters. To eliminate this bias, mutual information is normalized with the mean of the entropy of the clusters $H(W)$ and gold standards $H(C)$. Following [77], we use the arithmetic mean of $H(W)$ and $H(C)$ since $[H(W) + H(C)]/2$ is a tight upper bound on $I(W,C)$.

Entropy is a measure of uncertainty for a probability distribution [22]. Entropy $H(C)$ of a gold standard C is defined by:

$$H(C) = -\sum_j P(c_j) \log P(c_j) \tag{3.7}$$

Based on maximum likelihood estimates of the probabilities, Equation 3.7 is equivalent to:

$$H(C) = -\sum_j \frac{|c_j|}{N} \log \frac{|c_j|}{N} \tag{3.8}$$

Similar to $H(C)$, the entropy $H(W)$ of a set of clusters W is defined by:

$$\begin{aligned} H(W) &= -\sum_{k} P(w_k) \log P(w_k) \\ &= -\sum_{k} \frac{|w_k|}{N} \log \frac{|w_k|}{N} \end{aligned} \tag{3.9}$$

Based on the examples in Table 3.5, the mutual information $I(W,C)$ value is 0.87, the entropy $H(C)$ of the gold standard C is 2.0, and the entropy of the clusters W is 1.96. So, the NMI value of the clusters W from Table 3.5 is 0.44.

3.3.3 F-Measure

As a final measure of the quality of clustering result, we include the pairwise *F-Measure* metric [77] to compute the harmonic mean of precision P and recall R.

$$F = 2 \times \frac{P \times R}{P + R} \,. \tag{3.10}$$

where precision p is the fraction of pairs of tweets correctly put in the same cluster, and recall r is the fraction of actual pairs of tweets that were identified. Definition of both precision and recall are shown in Equation 3.11 and 3.12 below.

$$P = \frac{TP}{TP + FP} \tag{3.11}$$

$$R = \frac{TP}{TP + FN} \tag{3.12}$$

In this metric, TP (True Positive) is the number of pairs of tweets from clusters in the gold standard which are correctly assigned to the same cluster in the output. TN (True Negative) is the number of pairs of tweets from different clusters in the gold standard that are assigned to different clusters. The False Positive (FP) is the number of pairs of tweets that should not be in the same cluster, but are assigned to the same cluster. False Negative (FN) is the number of pairs of tweets that should

be in the same cluster, but are assigned to different clusters.

To compute the precision and recall of the examples in Table 3.5, we need to compute the TP, TN, FP and FN of all possible pairs of available tweets. Since the number of tweets (N) in Table 3.5 is 12, the total number of possible pairs of tweets is $N(N-1)/2$, i.e., is 66.

To calculate the four variables TP, TN, FP and FN, we break down the process into two different steps: obtain the total number of positive ($TP+FP$) and the total number of negative ($TN+FN$). To calculate the total number of positive, we compute all pairs of tweets that exist in each cluster from the set of clusters W.

$$TP+FP=\binom{4}{2}+\binom{3}{2}+\binom{3}{2}+\binom{2}{2}=13 \tag{3.13}$$

Since *True Positive* (TP) consists only the pairs of tweets from the gold standard that assigned the same clusters in the set of cluster W, we can calculate *TP* by counting the pairs of tweets that have correct clusters. In our example, those include: $a1$ and $a3$, $m1$ and $m2$, and $t2$ and $t3$.

$$TP=\binom{2}{2}+\binom{2}{2}+\binom{2}{2}=3 \tag{3.14}$$

Thus, the *False Positive* (FP) of the examples in Table 3.5 is $(TP+FP)-TP$, which is 10.

Once we get the total number of positive ($TP+FP$), we can calculate the total number of negative. Total positive plus total negative must equal to the total possible pairs of the tweets. Hence, the total negative is $66-13=53$.

False Negative (FN) can be calculated by counting the tweets that should be in the same cluster, but are not. Going back to our example, consider c_1. Cluster w_1 has two tweets that match it ($a1, a2$) and one other tweet ($a3$) that is mismatched in cluster w_3. We count this as ($2\times 1=2$). Similarly, for tweets that should be in c_2, cluster w_2 has two tweets ($m1, m2$) that match, and one other tweet ($m3$) is

Table 3.7: *Matching matrix* of the TP, TN, FP, and FN

	Same cluster in W	**Different cluster in** W
Same cluster in C	$TP = 3$	$FN = 9$
Different cluster in C	$FP = 10$	$TN = 44$

mismatched in cluster w_3 $(2 \times 1 = 2)$. Then, for tweets that should be in c_3, one tweet is in cluster w_3 and two others are in different clusters $(1 \times 2 = 2)$. Since the two other tweets $(g2, g3)$ are also separated ($g2$ in cluster w_2 and $g3$ in cluster w_4), we count this as $(1 \times 1 = 1)$. Finally, two of tweets that should be in c_4 match cluster w_1, but the other one is mismatched in cluster w_4 $(2 \times 1 = 2)$. FN is the sum of all those counts, which is:

$$FN = (2 \times 1) + (2 \times 1) + (1 \times 2) + (1 \times 1) + (2 \times 1) = 9 \tag{3.15}$$

Since we know the value of FN, we can compute *True Negative* (TN) by subtracting the total negative with FN $(53 - 9 = 44)$. The matching matrix of the TP, TN, FP, and FN is shown in Table 3.7.

$$P = \frac{TP}{TP + FP} = \frac{3}{3 + 10} = 0.23 \tag{3.16}$$

$$R = \frac{TP}{TP + FN} = \frac{3}{3 + 9} = 0.25 \tag{3.17}$$

$$F = 2 \times \frac{P \times R}{P + R} = 2 \times \frac{0.23 \times 0.25}{0.23 + 0.25} = 0.24 \tag{3.18}$$

Having all the required parameters, we can calculate the precision, recall, and F-Measure of our example from Table 3.5. The calculations are shown in Equation 3.16, 3.17, and 3.18 respectively.

3.4 Discussion

In this chapter, we have presented three labeled datasets and several evaluation metrics used for evaluation throughout the thesis. Two publicly available datasets (TREC2014 and tweetSanders) and one dataset we collected on March 2014 (tweetMarch) will be used in all experiments to test the proposed methods in more varied tweets and different scenarios.

Purity, NMI, and Pairwise F-Measure are metrics used to measure the performance of our proposed methods with the labeled tweets as gold data. Those three evaluation metrics, along with labeled datasets, are widely used and considered as the best methods to evaluate the performance of topic derivation in Twitter environment. Due to the extreme sparsity of the Twitter content, automatic internal analysis of the topic model cannot give reliable results for different runs and methods. Furthermore, other way of evaluating the probabilistic model like the log-likelihood or perplexity might not be able to capture the coherency of topics and often negatively correlated with human judgment [15].

Chapter 4

Incorporating Tweet Relationships in LDA for Topic Derivation

4.1 Introduction

Unlike traditional documents with lengthy and structured content, a tweet is short and could include expressions in informal language, such as emoticons, abbreviations, and misspelled terms. Given their short and informal content environment, deriving topics from tweets is a challenging problem. The very low co-occurrences between terms heavily penalizes the topic derivation process. Because of this sparsity problem, existing methods for topic derivation discussed in Chapter 2, still do not work well in the Twitter environment.

The limitations of those methods have inspired us to go beyond content to address the sparsity problem. We investigate the possibility of incorporating the social interaction features in Twitter. Studies by [100] and [48] show that these features play an important role in both topic quality and credibility in the Twitter environment.

We propose a new method, *intLDA*, that uses the contents of tweets *and* specific relationships between tweets to perform a topic derivation. In this chapter, we define the relationships between tweets as the interactions based on *user mentions*, *replies-retweets* and content similarity. Our analysis and experimental results show that our proposed method can significantly outperform other advanced methods

and configurations in terms of topic derivation quality. The main contribution of the chapter can be summarized as follows:

- We observe that tweets are topically related to each other through both interactions and content features. Our analysis reveals that a matrix of tweet relationships has a higher density than one that based on term-to-term or tweet-to-term relationships.
- We develop a novel extension of the LDA method, *intLDA*, to incorporate the tweet relationships into topic derivation. Our proposed *intLDA* method can effectively determine and characterize the main topic of each tweet.
- We conduct comprehensive experiments on three Twitter datasets, using the widely accepted topic derivation metrics. Both the Twitter datasets and the evaluation metrics have been introduced in Chapter 3. The experimental results demonstrate significant improvements over popular methods such as LDA, Plink-LDA [131] and NMF. We also discuss an implementation of a simple variation to LDA that takes into account tweet relationships (*eLDA*) and show that *intLDA* is still far better in comparison to this simpler method.

This chapter is organized as follows. Section 4.2 describes our observations on the topic prominence in Tweets when using LDA method for topic derivation. Section 4.3 introduces the relationships that exist between tweets based on their interactions and content. Section 4.4 describes how to incorporate these relationships into LDA. Details of the experiments and evaluation are presented in Section 4.5. We provide a discussion of our work in Section 4.6.

4.2 Topic Prominence in Tweets

In general, topic modeling methods such as LDA model a document as a bag of words drawn from a mixture of topics. LDA has been used to determine the most

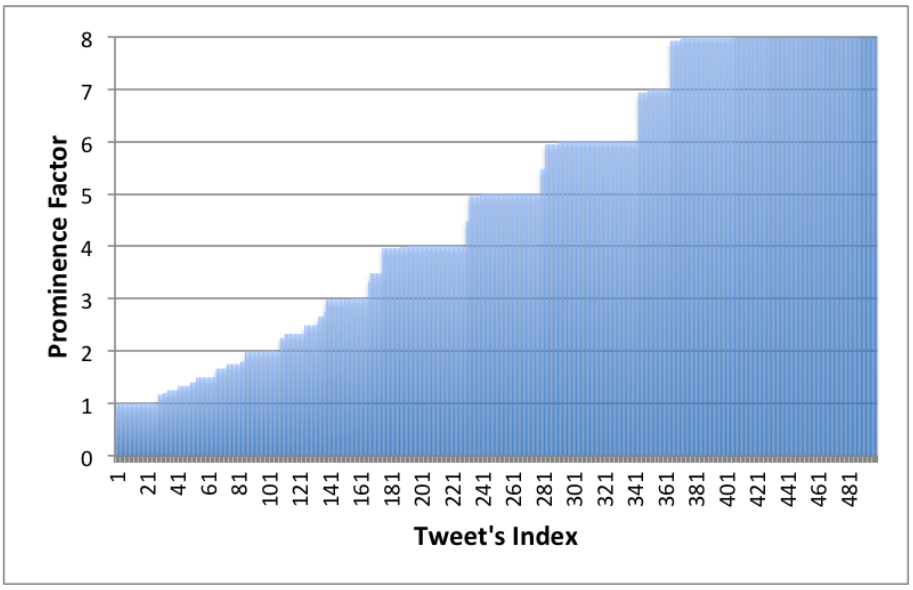


Figure 4.1: Topic prominence in the tweets of a collection of 500 tweets, sorted by prominence factor (ratio between the highest and the second highest topic probability for each tweet). The values are clipped at a factor of 8.

likely distribution of words per topic and the most likely distribution of topics in documents. After performing LDA, it is straightforward to determine the most salient topics in a document and the most salient words in a topic. Since a document is considered as a mixture of topics, it is not trivial to determine the most important topics in the collection. However, a tweet is much shorter than a general document. An analysis of the topic prominence in tweets is required to see if it is sensible to characterize a tweet by its most salient topic.

We have performed LDA on the first 500 tweets of the tweetMarch dataset and observed a marked predominance of one topic per tweet, as we describe below. For any tweet, let t_1 be the topic with the highest probability (p_1) and t_2 the next ranking topic (with probability p_2), as determined by LDA. We call the ratio of $p1/p2$ the "Prominent Factor" or PF. If t_1 is much more prominent than t_2, PF will be high. Figure 4.1 shows the prominent factor for each tweet, in ascending order, after performing LDA with 20 topics. The values are clipped at a factor of 8, but we observed a maximum factor of 2000. Furthermore, around 400 tweets (80%) have

a prominent factor over 2 (e.g., 0.8 for the highest probability and 0.4 for the next ranking). The figure shows that more than 95% of the tweets have a prominent factor of 1.3 or higher. A factor of 1.3 (e.g., 0.69 for the highest probability and 0.51 for the next ranking) or higher means that one topic is relatively predominant for this tweet. The larger the factor, the more predominant the topic.

Given the marked preference of one topic in each tweet for most tweets, it is sensible to characterize a tweet by its most salient topic. By establishing this one-to-one mapping from tweets to topics, we can determine the importance of a topic in the collection of tweets by counting how many tweets are mapped to the topic. We, therefore, perform *topic derivation* of a collection of tweets by determining the main topic of every tweet and grouping tweets on the same topic, then by characterizing the most important topics of the collection of tweets by listing their most important words.

4.3 Observing the relationships between tweets

Topic derivation by straight LDA suffers from the fact that tweets are very short, and there is thus a sparse relationship between the tweets and the terms [29]. In the approach presented here, we use the interactions between tweets as means to address the sparsity problem to achieve higher quality of topics.

Owing to the social networking nature of Twitter, there are various relationships on the Twitter platform. Twitter provides a *following-follower* mechanism to connect users, so that all followed users' tweets will be shown on a user's home page. In addition, Twitter offers several interactive features enabling users to interact with each other through tweets, such as *mention*, *reply*, *retweet*, and *hashtag*. These features have made Twitter a network of not only people but also information. However, for the task of topic derivation, incorporating the following-follower information is difficult because of the scalability issues, as the detail of each involved users would

need to be queried from Twitter independently from the tweets themselves. Thus, we define the relationships between tweets for topic derivation as the interactions based on *user mentions*, *replies-retweets*, and *content similarity*.

User mentions and *replies* are helpful mechanisms for initiating or joining a conversation in Twitter. Intuitively, all tweets belonging to the same conversation have a high probability of sharing the same or similar topic even if no terms co-occur in their content. A *mention*, denoted as '@' followed by a username, directly refers to another user. In contrast, a *reply* is used to send out a message in reply to a specific tweet. In a *reply* tweet, the username of the original tweet's author is automatically included in the message.

Different from the *mention* and *reply* relations, a *retweet* is a re-posting of someone else's tweet. This can be used to further disseminate a tweet, for example to ensure one's followers see it. Since a *retweet* has many words in common with the original tweet, the term co-occurrence between the two tweets (original and retweet) will be high, and both tweets are likely to share a topic.

Last, a *hashtag* is another important feature in Twitter, popularly used to bookmark the content of a tweet, or to present the users' interest on particular topics [138]. A *hashtag* starts with a hash character ("#") followed by one word or more, for example: *#Canberra, #Floriade, #Monday, #CarsTheMovie*. However, the same *hashtag* in several tweets does not necessarily indicate that these tweets are about the same topic. For example, a *hashtag #Canberra* in two tweets may involve two different topics for each author.

Looking back at the example of tweet collection presented in Table 1.1, we can see how the interactions could help connecting tweets to a particular topic discussion. Figure 4.2 shows an illustration of possible interactions between tweets from Table 1.1. We see that t_1 (*'New senate, exciting times in #Canberra @b'*) and t_2 ('@a true, and what a start with the census in Australia') are related from the fact that tweet t_2 is a reply to tweet t_1, even though they do not share any common

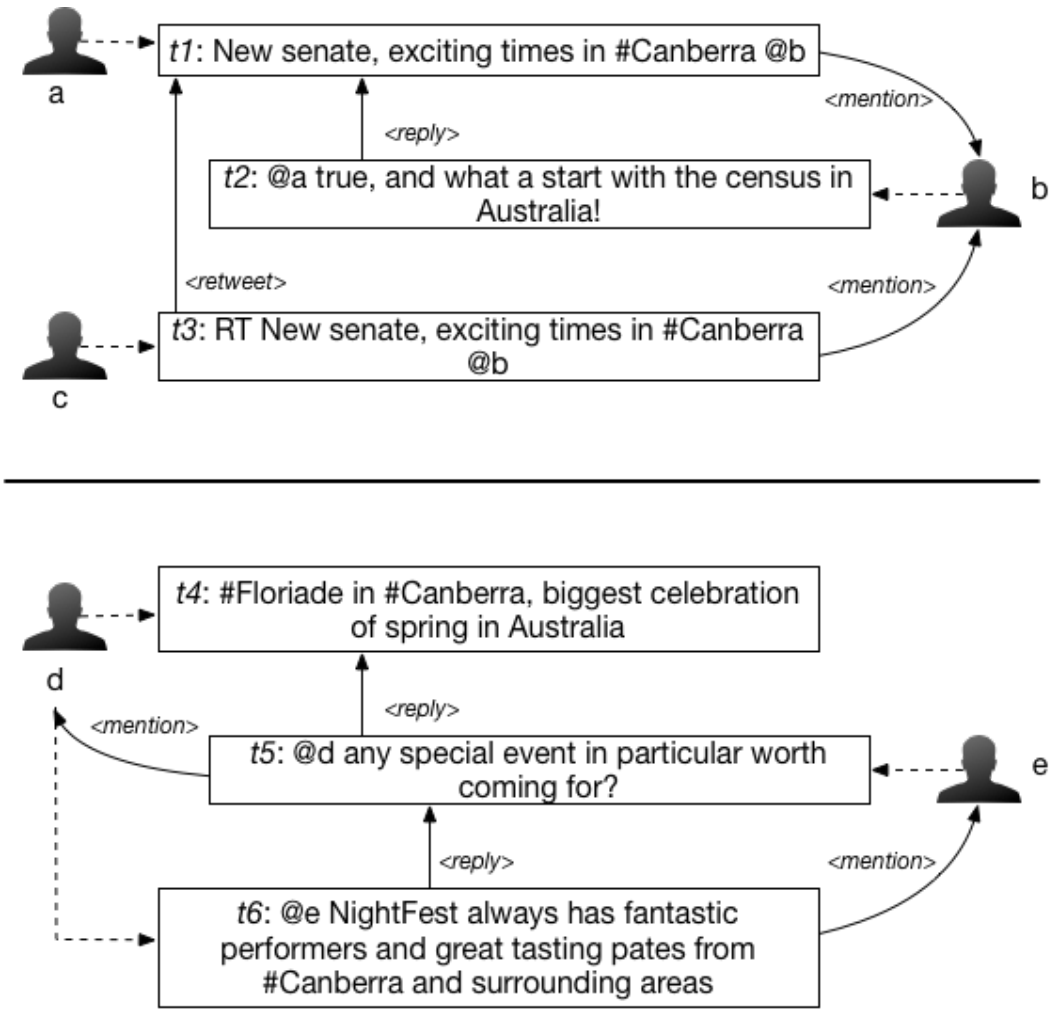


Figure 4.2: An illustration of possible interactions between tweets

terms. t_3 and t_1 are explicitly related to each other as t_3 is a retweet of t_1, and it shares almost all of t_1's content. The interactions in these tweets are able to show the topical connectivity between each other that they are under the same topic concerning about politic.

Using a similar approach, we can determine the main topic of tweets t_4, t_5 and t_6 by looking at the involved interactions. t_5 ('*@d any special event in particular worth coming for?*') is a reply to t_4 ('*#Floriade in #Canberra, biggest celebration of spring in Australia*'), and t_6 ('@e NightFest always has fantastic performers and great tasting pates from #Canberra and surrounding areas') is a reply to t_5. These

tweets are discussing the Floriade celebrations in Canberra.

The above discussion shows that interactions can provide information about the topical relationships between tweets. They often represent conversational activities between users through the tweets. Incorporating these interactions into the topic derivation process should result in improvements in the derived topics. We classify the interactions based on user mentions and replies-retweets. Let U_{t_i} be the set of users mentioned in tweet i_i (including the author of tweet t_i), and U_{t_j} be the set of users mentioned in tweet t_j (including the author of tweet t_j). Then, $m(t_i, t_j)$ uses the *user mention* relationship and is defined as the number of common mentioned Twitter users in tweets t_i and t_j.

$$m(t_i, t_j) = |U_{t_i} \cap U_{t_j}| \,. \tag{4.1}$$

The replies-retweets based interaction $act(t_i, t_j)$ is defined as follows. If a tweet t_i is a *retweet* or a *reply* of another tweet t_j or vice-versa, or if both tweets are replying or retweeting the same tweet, the interaction value $act(t_i, t_j)$ is 1, otherwise 0. Generally speaking, an $act(t_i, t_j)$ value of 1 means that two tweets have a strong relationship with each other, and most likely they share the same topic.

$$act(t_i, t_j) = \begin{cases} 1, (rtp_{t_i} = j)\ or\ (i = rtp_{t_j})\ or\ (rtp_{t_i} = rtp_{t_j}) \\ 0,\ otherwise \end{cases} \tag{4.2}$$

where rtp_{t_i} stands for the *retweet* or *reply* information of a tweet t_i.

There are also a large number of *self-contained* tweets, i.e., tweet with no references (*mention*, *reply* or *retweet* relation) to another tweet [24]. We thus also need to include content based interactions in the relationship between tweets for the purposes of topic derivation. We use a *content similarity* ($sim(t_i, t_j)$) between two tweets t_i and t_j to measure the content based interaction. In this chapter, we

Table 4.1: Density comparison of the non-zero elements between the tweet-to-tweet (A) matrix, tweet-to-term (V) matrix, and term-to-term (T)

Dataset	A	V	T
TREC2014	2.695%	0.056%	0.249%
tweetSanders	23.887%	0.089%	0.301%
tweetMarch	12.842%	0.075%	0.271%

will simply use the word overlap between t_i and t_j. Thus, if C_{t_i} denotes the set of words of tweet t_i and if C_{t_j} denotes the set of words of tweet t_j, then:

$$sim(t_i, t_j) = |C_{t_i} \cap C_{t_i}|. \tag{4.3}$$

Before measuring the content similarity, a preprocessing step was performed for all tweets in the datasets to remove characters that are not relevant with topic derivation (e.g., emoticons and punctuations). Stop words and words with less than 3 characters are also removed. The remaining words are then stemmed using the stemmer from NLTK python packages[1]. Our analysis shows that the average length of each tweet after the preprocessing steps for all datasets discussed in the previous chapter is only 4 words. With this short average length, a single overlapping word is considered important for content similarity. Thus, we do not put a higher threshold for this relationship. The topical accuracy of relationships based on content similarity and/or tweet interactions will further be discussed in the next chapter.

We can now formalize the relationship between tweets t_i and t_j ($R(t_i, t_j)$) based on user mentions, replies-retweets, and content similarity, as shown in Equation 4.4 below:

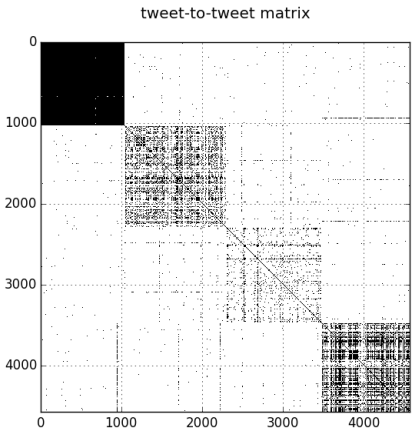
$$R(t_i, t_j) = \begin{cases} 1 & \text{if } m(t_i, t_j) > 0 \text{ or } act(t_i, t_j) > 0 \\ & \text{or } sim(t_i, t_j) > 0 \\ 0 & \text{otherwise .} \end{cases} \tag{4.4}$$

[1]http://www.nltk.org/. Accessed 20 July 2017.

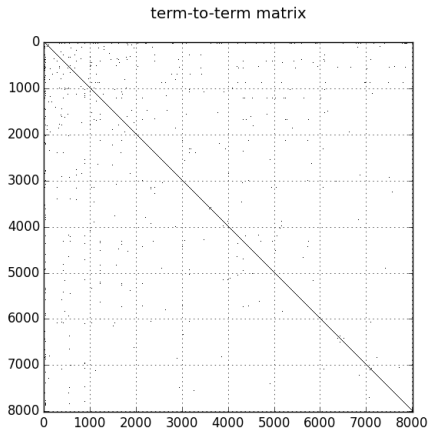
Values of $R(t_i, t_j)$ from all the possible relationships between the tweets in the collection form the *tweet-to-tweet* matrix *A*. This matrix is much denser than matrices based on other types of relationships, such as tweet-term relationship, or term-term relationship. Table 4.1 shows the comparison of the density between between different matrices: the *tweet-to-tweet* (A) matrix, the *tweet-to-term* (V) matrix, and the *term-to-term* (T) matrix. This was obtained from three datasets presented in Chapter 3. The tweet-to-term matrix (V) is calculated using *tf-idf* function [108]. For the term-to-term matrix (T), we use the positive point mutual information (PPMI) function described in [136]. From Table 4.1, we can see that our definition of the tweet-to-tweet relationships provides the densest non-zero element in comparison with other types of relationships, even if the number of tweets that have interactions is low (e.g., TREC2014 with only around 8% tweets that are replies and no retweets at all). The tweetSanders dataset is the least sparse on all relationships compared with the other datasets due to its apparent involved topics.

The visualization of the each type of relationship density for the tweetSanders dataset in Figure 4.3 illustrates how the tweet-to-tweet relationships are not only significantly much denser than the others, but also show how the relationships could represent the topical connectivity between tweets. For a dataset that has relatively dense relationships like tweetSanders, the tweet-to-tweet visualization (Figure 4.3a) shows that most of the tweets are clustered into four groups. According to the labeled tweets, the tweetSanders dataset has four topics: *Apple*, *Google*, *Microsoft*, and *Twitter*.

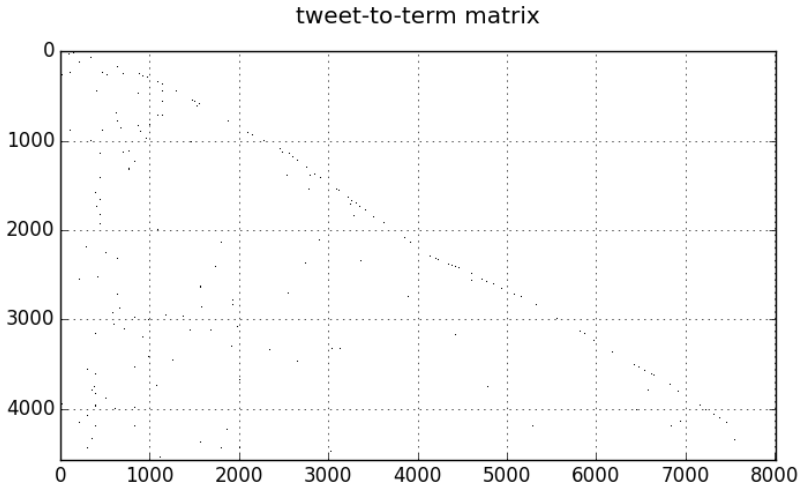
TREC2014 and tweetMarch datasets are extremely sparse, with only 0.249% and 0.271% of all unique terms correlated with each other respectively. Our definition of relationships between tweets results in a matrix 10 times denser than a matrix obtained with the term-to-term relationship. Figure 4.4 and Figure 4.5 show the visualization of each type of relationships density for TREC2014 and tweetMarch, respectively. With this density improvement, we believe that the tweet-to-tweet



(a) *tweet-to-tweet matrix (A)*

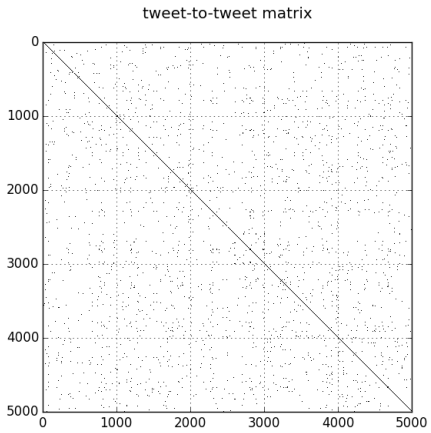


(b) *term-to-term matrix (T)*

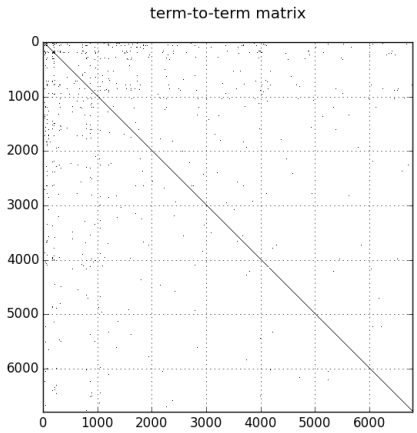


(c) *tweet-to-term matrix (V)*

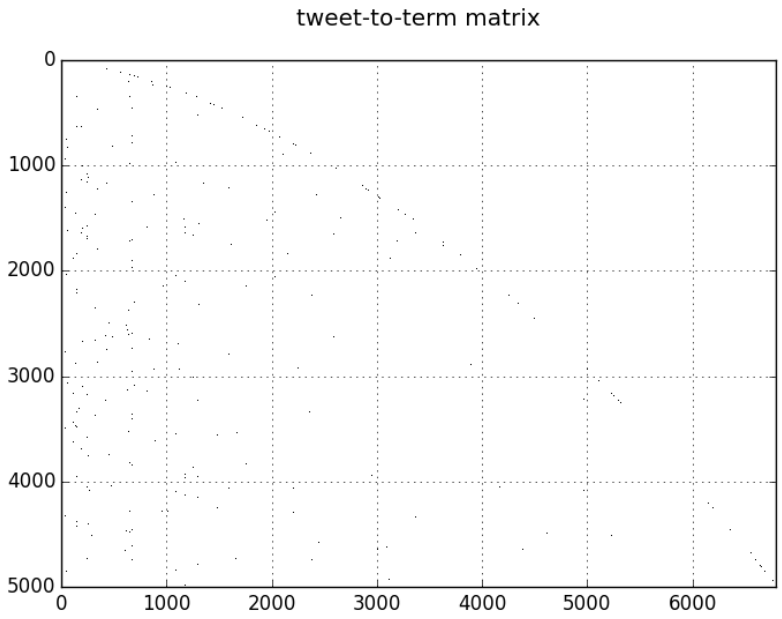
Figure 4.3: Density visualization of the tweetSanders dataset for (a) tweet-to-tweet matrix A, (b) term-to-term matrix T and (c) tweet-to-term matrix V



(a) *tweet-to-tweet matrix (A)*

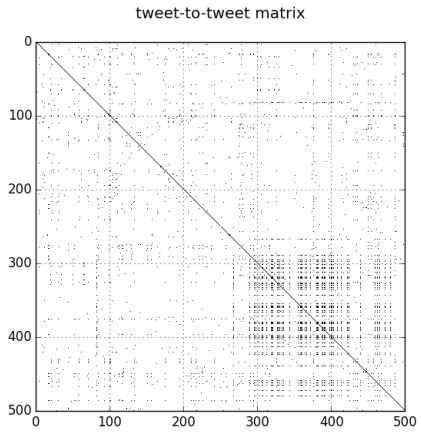


(b) *term-to-term matrix (T)*

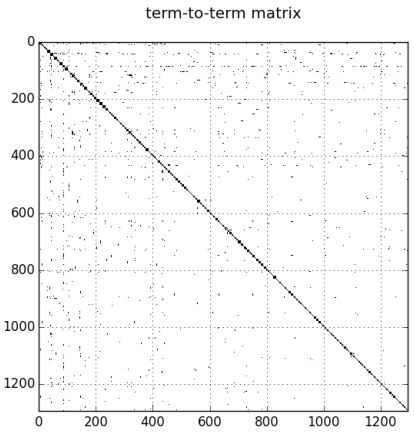


(c) *tweet-to-term matrix (V)*

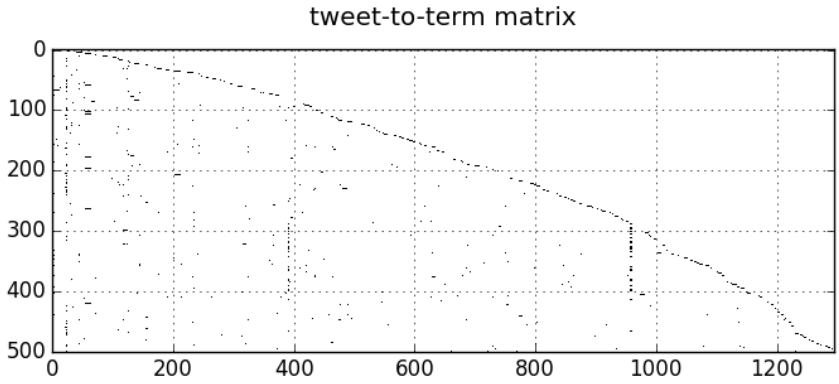
Figure 4.4: Density visualization of the tweetTREC2014 dataset for (a) tweet-to-tweet matrix A, (b) term-to-term matrix T and (c) tweet-to-term matrix V



(a) *tweet-to-tweet matrix (A)*



(b) *term-to-term matrix (T)*



(c) *tweet-to-term matrix (V)*

Figure 4.5: Density visualization of the tweetMarch dataset for (a) tweet-to-tweet matrix A, (b) term-to-term matrix T and (c) tweet-to-term matrix V

relationships will be important to help achieve a high quality of topic derivation in the Twitter environment.

4.4 Incorporating tweet relationships into LDA

In this section, we discuss our method of incorporating the tweet relationships into the LDA process. We first discuss the basic LDA method, then a simple method we call *eLDA*, our naïve way of expanding the tweet content by adding the new content from the related tweets. We then present our proposed method *intLDA*, another variant of LDA that directly incorporates the relationships between tweets.

4.4.1 Topic derivation in Twitter using LDA

Latent Dirichlet Allocation (LDA) was presented by Blei et al. in [8]. This method is used to automatically discover the topics from a collection of documents, with the intuition that every document exhibits multiple topics. LDA models the words of a document as generated randomly from a mixture of topics where each topic has a latent distribution of word probabilities. It has a *bag of words* assumption where the order of the words in a document does not affect the process. The generative process of LDA is illustrated in Figure 2.4 of Chapter 2. The documents and their words are generated according to the following generative process:

1. For each document m ($m \in \{1, ..., M\}$), draw a topic distribution θ_m, which is randomly sampled from a Dirichlet distribution with hyperparameter α. ($\theta_m \sim Dir(\alpha)$)

2. For each topic z ($z \in \{1, ..., K\}$), draw a word distribution ϕ_z, which is randomly sampled from a Dirichlet distribution with hyperparameter β. ($\phi_z \sim Dir(\beta)$)

3. For each word n in document m:

Table 4.2: Summary of the LDA variables definition

Variable	Description
α	Dirichlet prior parameter for document-topic distributions θ
β	Dirichlet prior parameter for topic-word distributions ϕ
θ	Distribution of topic probability in a document
ϕ	Distribution of word probability in a topic
Z	Topics assignment for all words in all documents
W	Unique words in all documents
z	Topic assignment for a word in a document
w	Word assignment for a topic
M	Number of documents in the collection
N	Number of words in all documents
K	Number of topics

(a) Choose a topic z_n sampled from the topic distribution θ_m. ($z_n \sim Cat(\theta_m)$)

(b) Choose a word w_n from $p(w_n|z_n, \beta)$, a multinomial probability conditioned on the topic z_n. ($w_n \sim Cat(\phi_{z_n})$)

The original LDA model proposed by Blei et al. was based on the variational method and the expectation-maximization (EM) algorithm. In 2004, the use of *Collapsed Gibbs sampling* inference strategy was introduced by Griffiths and Steyvers [35] as an alternative to the variational estimation for the posterior distribution. The LDA approach with Gibbs sampling is now widely adopted due to its simpler implementation and memory efficiency. The total probability of LDA can be formulated as follows:

$$P(W, Z, \theta, \phi; \alpha, \beta) = \prod_{i=1}^{K} P(\phi_i; \beta) \prod_{j=1}^{M} P(\theta_j; \alpha) \prod_{t=1}^{N} P(z_{j,t}|\theta_j) P(w_{j,t}|\phi_{z_{j,t}}) \quad (4.5)$$

The summary of the LDA variables is available in Table 4.2. For the collapsed Gibbs sampling approach, integrating over θ and ϕ from the total probability in Equation 4.5 we get:

$$\begin{aligned} P(Z,W;\alpha,\beta) &= \int_{\theta}\int_{\phi} P(W,Z,\theta,\phi;\alpha,\beta)d\phi\, d\theta \\ &= \int_{\phi}\prod_{i=1}^{K} P(\phi_i;\beta)\prod_{j=1}^{M}\prod_{t=1}^{N} P(W_{j,t}|\phi z_{j,t})d\phi \\ &\quad \int_{\theta}\prod_{j=1}^{M} P(\theta_j;\alpha)\prod_{t=1}^{N} P(Z_{j,t}|\theta_j)d\theta \\ &= \prod_{i=1}^{K}\int_{\phi i} P(\phi_i;\beta)\prod_{j=1}^{M}\prod_{t=1}^{N} P(W_{j,t}|\phi z_{j,t})d\phi_i \\ &\quad \prod_{j=1}^{M}\int_{\theta j} P(\theta_j;\alpha)\prod_{t=1}^{N} P(Z_{j,t}|\theta_j)d\theta_j \end{aligned} \tag{4.6}$$

Focusing on θ, we can rewrite $P(\theta_j;\alpha)$ for $\theta_j = 1...D\ Dirichlet_K(\alpha)$ as:

$$P(\theta_j;\alpha) = \frac{\Gamma(\Sigma_{i=1}^{K}\alpha_i)}{\prod_{i=1}^{K}\Gamma(\alpha_i)}\prod_{i=1}^{K}\theta_{j,i}^{\alpha_i-1} \tag{4.7}$$

and $\prod_{t=1}^{N} P(Z_{j,t}|\theta_j)$ into:

$$\prod_{t=1}^{N} P(Z_{j,t}|\theta_j) = \prod_{i=1}^{K}\theta_{j,i}^{n_{j,(.)}^{i}} \tag{4.8}$$

$n_{j,(.)}^{i}$ in above equation denotes the number of words token in j^{th} document assigned to the i^{th} topic. Thus, the θ part of the Equation 4.6 can be rewritten as:

$$\begin{aligned} \int_{\theta j} P(\theta_j;\alpha)\prod_{t=1}^{N} P(Z_{j,t}|\theta_j)d\theta_j &= \int_{\theta j}\frac{\Gamma(\Sigma_{i=1}^{K}\alpha_i)}{\prod_{i=1}^{K}\Gamma(\alpha_i)}\prod_{i=1}^{K}\theta_{j,i}^{\alpha_i-1}\prod_{i=1}^{K}\theta_{j,i}^{n_{j,(.)}^{i}}d\theta_j \\ &= \int_{\theta j}\frac{\Gamma(\Sigma_{i=1}^{K}\alpha_i)}{\prod_{i=1}^{K}\Gamma(\alpha_i)}\prod_{i=1}^{K}\theta_{j,i}^{n_{j,(.)}^{i}\ +\ \alpha_i-1}d\theta_j \end{aligned} \tag{4.9}$$

Having the similar property to the Dirichlet distribution,

$$\int_{\theta j} \frac{\Gamma(\Sigma_{i=1}^{K} n_{j,(.)}^{i} + \alpha_i)}{\prod_{i=1}^{K} \Gamma(n_{j,(.)}^{i} + \alpha_i)} \prod_{i=1}^{K} \theta_{j,i}^{n_{j,(.)}^{i} + \alpha_i - 1} d\theta_j = 1 \tag{4.10}$$

we can simplify the formula in Equation 4.9 into:

$$\begin{aligned}
\int_{\theta j} P(\theta_j; \alpha) \prod_{t=1}^{N} P(Z_{j,t}|\theta_j) d\theta_j &= \int_{\theta j} \frac{\Gamma(\Sigma_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \prod_{i=1}^{K} \theta_{j,i}^{n_{j,(.)}^{i} + \alpha_i - 1} d\theta_j \\
&= \frac{\Gamma(\Sigma_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \frac{\prod_{i=1}^{K} \Gamma(n_{j,(.)}^{i} + \alpha_i)}{\Gamma(\sigma_{i=1}^{K} n_{j,(.)}^{i} + \alpha_i)} \int_{\theta j} \frac{\Gamma(\Sigma_{i=1}^{K} n_{j,(.)}^{i} + \alpha_i)}{\prod_{i=1}^{K} \Gamma(n_{j,(.)}^{i} + \alpha_i)} \prod_{i=1}^{K} \theta_{j,i}^{n_{j,(.)}^{i} + \alpha_i - 1} d\theta_j \\
&= \frac{\Gamma(\Sigma_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \frac{\prod_{i=1}^{K} \Gamma(n_{j,(.)}^{i} + \alpha_i)}{\Gamma(\sigma_{i=1}^{K} n_{j,(.)}^{i} + \alpha_i)}
\end{aligned} \tag{4.11}$$

The process of integrating out ϕ is similar to the θ part:

$$\begin{aligned}
\int_{\phi i} P(\phi_i; \beta) \prod_{j=1}^{M} \prod_{t=1}^{N} P(W_{j,t}|\phi z_{j,t}) d\phi_i &= \int_{\phi i} \frac{\Gamma(\Sigma_{r=1}^{V} \beta_r)}{\prod_{r=1}^{V} \Gamma(\beta_i)} \prod_{r=1}^{V} \phi_{i,r}^{\beta_r - 1} \prod_{r=1}^{V} \phi_{i,r}^{n_{(.),r}^{i}} d\phi_i \\
&= \int_{\phi i} \frac{\Gamma(\Sigma_{r=1}^{V} \beta_r)}{\prod_{r=1}^{V} \Gamma(\beta_r)} \prod_{r=1}^{V} \phi_{i,r}^{n_{(.),r}^{i} + \beta_r - 1} d\phi_i \\
&= \frac{\Gamma(\Sigma_{r=1}^{V} \beta_r)}{\prod_{r=1}^{V} \Gamma(\beta_r)} \frac{\prod_{r=1}^{V} \Gamma(n_{(.),r}^{i} + \beta_r)}{\Gamma(\sigma_{r=1}^{V} n_{(.),r}^{i} + \beta_r)}
\end{aligned} \tag{4.12}$$

Thus, the final target probability can be formulated as follows:

$$\begin{aligned}
&P(Z, W; \alpha, \beta) = \\
&\prod_{j=1}^{D} \frac{\Gamma(\Sigma_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \frac{\prod_{i=1}^{K} \Gamma(n_{j,(.)}^{i} + \alpha_i)}{\Gamma(\sigma_{i=1}^{K} n_{j,(.)}^{i} + \alpha_i)} \times \prod_{i=1}^{K} \frac{\Gamma(\Sigma_{r=1}^{V} \beta_r)}{\prod_{r=1}^{V} \Gamma(\beta_r)} \frac{\prod_{r=1}^{V} \Gamma(n_{(.),r}^{i} + \beta_r)}{\Gamma(\sigma_{r=1}^{V} n_{(.),r}^{i} + \beta_r)}
\end{aligned} \tag{4.13}$$

The general goal of the LDA process is to approximate $P(Z|W; \alpha, \beta)$, where it can be directly derived from the final probability in equation 4.13.

4.4.2 *eLDA:* expanding tweet content based on relationships between tweets

LDA works solely on the tweet content, without incorporating the relationships that may exist between tweets. It has a "bag of words" assumption where the order of the words in the documents does not have any effect on the topic derivation process. When dealing with short texts such as tweets, term co-occurrences amongst tweets can be extremely low, which hurts the topic derivation process. A naïve way of improving the LDA method is to augment the tweet content to increase the term co-occurrences. While expanding the content of the tweets using external documents seems ideal [2], the method would not be able to deal with Twitter's highly dynamic environment, as already mentioned. Furthermore, the language used in tweets is mostly informal, and therefore the words occurring in a tweet may not match terms in external corpora.

A simple, intuitive use of the tweet-to-tweet relationship matrix A consists in expanding the tweet content by adding the words from the related tweets (tweets with the observed tweet relationships discussed in Section 4.3). In this approach, we add only words that are not already occurring in the original tweet, after stop-words have been removed. For example, from the illustration of the interactions between tweets in Figure 4.2, tweet t_2 (with the words *true, start, census, Australia*) is a reply to tweet t_1 (which contains the words *new, senate, exciting, times, #Canberra*). Based on this interaction, the terms from t_1 that are not available in t_2 are added into tweet t_2. The content of t_2 becomes *(true, start, census, Australia, new, senate, exciting, times, #Canberra)*.

Our implementation of this content expansion is denoted as *eLDA*. A possible drawback of this method is that the added words might not be related to the tweet's topic, therefore introducing noise, which could lead to the lower topic derivation quality.

4.4.3 *intLDA:* incorporating the tweet relationship to improve the tweet-topic distributions

In LDA, each tweet i defines a multinomial distribution θ_i of topics. The global tweet-topic distribution θ can be learned based on the observed words present in each tweet through a Markov Chain Monte-Carlo algorithm such as Gibbs sampling [35]. As discussed in Section 4.4.1, the general goal of the LDA process is to approximate $P(Z|W;\alpha,\beta)$, which can be derived from the final probability $P(Z,W;\alpha,\beta)$ shown in equation 4.13. In the LDA based Gibbs sampling strategy, the approximation is obtained by deriving the conditional probability shown in equation 4.14 below.

$$P(z_{(m,n)}|z_{-(m,n)},W;\alpha,\beta) = \frac{P(z_{(m,n)},z_{-(m,n)},W;\alpha,\beta)}{P(Z_{-(m,n)},W;\alpha,\beta)} \tag{4.14}$$

where, as shown in Table 4.2, $z(m,n)$ denotes the z hidden topic of the n^{th} word token in the m^{th} tweet, $z_{-m,n}$ denotes all $z \in Z$ except the $z_{m,n}$, and W is the vocabulary. Since we only need to get a sample value of $Z_{m,n}$, the equation can be simplified as:

$$\begin{aligned} P(Z_{m,n}=k|Z_{-m,n},W;\alpha,\beta) &\propto P(Z_{m,n}=k,Z_{-m,n},W;\alpha,\beta) \qquad (4.15)\\ &= \left(\frac{\Gamma(\sum_{i=1}^{K}\alpha_i)}{\prod_{i=1}^{K}\Gamma(\alpha_i)}\right)^{D}\prod_{j\neq m}\frac{\prod_{i=1}^{K}\Gamma(n^i_{j,(,)}+\alpha_i)}{\Gamma(\sum_{i=1}^{K}n^i_{j,(.)}+\alpha_i)}\left(\frac{\Gamma(\sum_{r=1}^{V}\beta_r)}{\prod_{r=1}^{V}\Gamma(\beta_r)}\right)^{K}\prod_{i=1}^{K}\prod_{r\neq v}\Gamma(n^i_{(.),r}+\beta_r)\\ &\quad\times\frac{\prod_{i=1}^{K}\Gamma(n^i_{m,(,)}+\alpha_i)}{\Gamma(\sum_{i=1}^{K}n^i_{m,(.)}+\alpha_i)}\prod_{i=1}^{K}\frac{\Gamma(n^i_{(.),v}+\beta_v)}{\Gamma(\sum_{r=1}^{V}n^i_{(.),r}+\beta_r)}\\ &\propto \frac{\prod_{i=1}^{K}\Gamma(n^i_{m,(,)}+\alpha_i)}{\Gamma(\sum_{i=1}^{K}n^i_{m,(.)}+\alpha_i)}\prod_{i=1}^{K}\frac{\Gamma(n^i_{(.),v}+\beta_v)}{\Gamma(\sum_{r=1}^{V}n^i_{(.),r}+\beta_r)} \end{aligned}$$

With the property of Γ, where Γ is an extension of the factorial to real numbers, the above equations can be further simplified as:

$$\propto \prod_{i\neq k} \frac{\Gamma(n_{m,(.)}^{i,-(m,n)}+\alpha_i)}{\Gamma(\sum_{i=1}^{K} n_{m,(.)}^{i,-(m,n)}+\alpha_i)} \prod_{i\neq k} \frac{\Gamma(n_{(.),v}^{i,-(m,n)}+\beta_v)}{\Gamma(\sum_{r=1}^{V} n_{(.),r}^{i,-(m,n)}+\beta_r)} \times \frac{\Gamma(n_{m,(.)}^{k,-(m,n)}+\alpha_k+1)}{\Gamma\left((\sum_{i=1}^{K} n_{m,(.)}^{i,-(m,n)}+\alpha_i)+1\right)} \frac{\Gamma(n_{(.),v}^{k,-(m,n)}+\beta_v+1)}{\Gamma\left((\sum_{r=1}^{V} n_{(.),r}^{k,-(m,n)}+\beta_r)+1\right)} \tag{4.16}$$

$$= \prod_{i\neq k} \frac{\Gamma(n_{m,(.)}^{i,-(m,n)}+\alpha_i)}{\Gamma(\sum_{i=1}^{K} n_{m,(.)}^{i,-(m,n)}+\alpha_i)} \prod_{i\neq k} \frac{\Gamma(n_{(.),v}^{i,-(m,n)}+\beta_v)}{\Gamma(\sum_{r=1}^{V} n_{(.),r}^{i,-(m,n)}+\beta_r)} \times \frac{\Gamma(n_{m,(.)}^{k,-(m,n)}+\alpha_k)}{\Gamma(\sum_{i=1}^{K} n_{m,(.)}^{i,-(m,n)}+\alpha_i)} \frac{\Gamma(n_{(.),v}^{k,-(m,n)}+\beta_v)}{\Gamma(\sum_{r=1}^{V} n_{(.),r}^{k,-(m,n)}+\beta_r)} \times \frac{n_{m,(.)}^{k,-(m,n)}+\alpha_k}{\sum_{i=1}^{K} n_{m,(.)}^{i,-(m,n)}+\alpha_i} \frac{n_{(.),v}^{k,-(m,n)}+\beta_v}{\sum_{r=1}^{V} n_{(.),r}^{k,-(m,n)}+\beta_r}$$

$$= \prod_{i} \frac{\Gamma(n_{m,(.)}^{i,-(m,n)}+\alpha_i)}{\Gamma(\sum_{i=1}^{K} n_{m,(.)}^{i,-(m,n)}+\alpha_i)} \prod_{i} \frac{\Gamma(n_{(.),v}^{i,-(m,n)}+\beta_v)}{\Gamma(\sum_{r=1}^{V} n_{(.),r}^{i,-(m,n)}+\beta_r)} \times \frac{n_{m,(.)}^{k,-(m,n)}+\alpha_k}{\sum_{i=1}^{K} n_{m,(.)}^{i,-(m,n)}+\alpha_i} \frac{n_{(.),v}^{k,-(m,n)}+\beta_v}{\sum_{r=1}^{V} n_{(.),r}^{k,-(m,n)}+\beta_r}$$

$$\propto \frac{n_{m,(.)}^{k,-(m,n)}+\alpha_k}{\sum_{i=1}^{K} n_{m,(.)}^{i,-(m,n)}+\alpha_i} \frac{n_{(.),v}^{k,-(m,n)}+\beta_v}{\sum_{r=1}^{V} n_{(.),r}^{k,-(m,n)}+\beta_r}$$

Finally, the derivation of conditional probability can be summarized as follows:

$$P(Z_{m,n}=k|Z_{-m,n},W;\alpha,\beta) \propto \frac{n_{m,(.)}^{k,-(m,n)}+\alpha_k}{\sum_{i=1}^{K} n_{m,(.)}^{i,-(m,n)}+\alpha_i} \frac{n_{(.),v}^{k,-(m,n)}+\beta_v}{\sum_{r=1}^{V} n_{(.),r}^{k,-(m,n)}+\beta_r} \tag{4.17}$$

Since working only on content makes LDA suffer from the sparsity problem, we extend the model algorithmically to directly incorporate the observed relationships

between tweets R in the process of learning θ. We use R as an additional constraint to the θ distributions, so that if two tweets are related, then the θ of those two tweets will be simultaneously adjusted based on the sampled topic. This proposed method is denoted as *intLDA*.

The difference between LDA and intLDA is in the process of sampling the tweet-topic distribution using Gibbs sampling. In each iteration of Gibbs sampling, LDA updates the document-topic counts of each tweet i independent of each other. In contrast, intLDA updates the document-topic counts of tweet i, as in LDA, but in addition, it updates the document-topic counts for the sampled topic z of all tweets j that are related to i as defined by $R_{i,j}$. In other words, the estimation of the document-topic distribution θ_i for tweet i is affected by information from related tweets. In Algorithm 3 below, the difference between LDA and *intLDA* is the addition of lines 14 to 16.

Algorithm 1 intLDA Gibbs Sampling

INPUT: tweets t, number of tweets D, number of topics K
OUTPUT: topic assignments z and counts cdt, cwt and ct

1: randomly initialize z and increment counters
2: **for** $i = 1 \rightarrow D$ **do**
3: **for** $l = 1 \rightarrow N_i$ **do**
4: $w \leftarrow t_{i,l}$
5: $topic \leftarrow z_{i,l}$
6: $cdt_{i,topic} -= 1; cwt_{w,topic} -= 1; ct_{topic} -= 1$
7: **for** $k = 1 \rightarrow K$ **do**
8: $p_k = (cdt_{i,k} + \alpha_k) \frac{cwt_{k,w} + \beta_w}{ct_k + \beta \times W}$
9: $n\_topic \leftarrow$ sample from p
10: $z_{i,l} \leftarrow n\_topic$
11: $cdt_{i,n\_topic} += 1;$
12: $cwt_{w,n\_topic} += 1;$
13: $ct_{n\_topic} += 1$
14: **foreach** j such that $R_{ij} == 1$ **do**
15: $cdt_{j,topic} -= 1$
16: $cdt_{j,n\_topic} += 1$
17: **return** z, cdt, cwt, ct

4.5 Experiments

In this section, we discuss the details of our experiments, including the baseline methods and the results. For the experiments, we use the labeled datasets discussed in Chapter 3: TREC2014, tweetSanders, and tweetMarch datasets. The Purity, NMI, and F-Measure evaluation metrics described in Section 3.3 of Chapter 3 are used to measure the performance of the proposed methods.

4.5.1 Baseline Methods

We evaluate the proposed eLDA and intLDA against the following baseline methods:

- *LDA* [8]: a straight use of LDA. The implementation is based on the Gibbs sampling inference strategy discussed in [35].
- *Plink-LDA* [131]: a variant of LDA that uses relationships between documents as prior information for topic derivation. This variant of LDA is thus closest to our approach. In the original paper, citation based linked datasets are used to assess the model. For the purpose of this evaluation, we use our observed tweet relationships as the link information between tweets.
- *NMF* [62]: a popular algorithm of Non-Negative Matrix Factorization, which directly factorizes the tweet-to-term matrix into the tweet-topic matrix and the topic-term matrix.
- *TNMF* [136]. This method is an extension of NMF which incorporates the correlation between terms (term-term) matrix to derive the topics.

4.5.2 Results

We have conducted experiments on several possible setups for all the methods. For the purity evaluation, the number of topics is set to the number of labels available

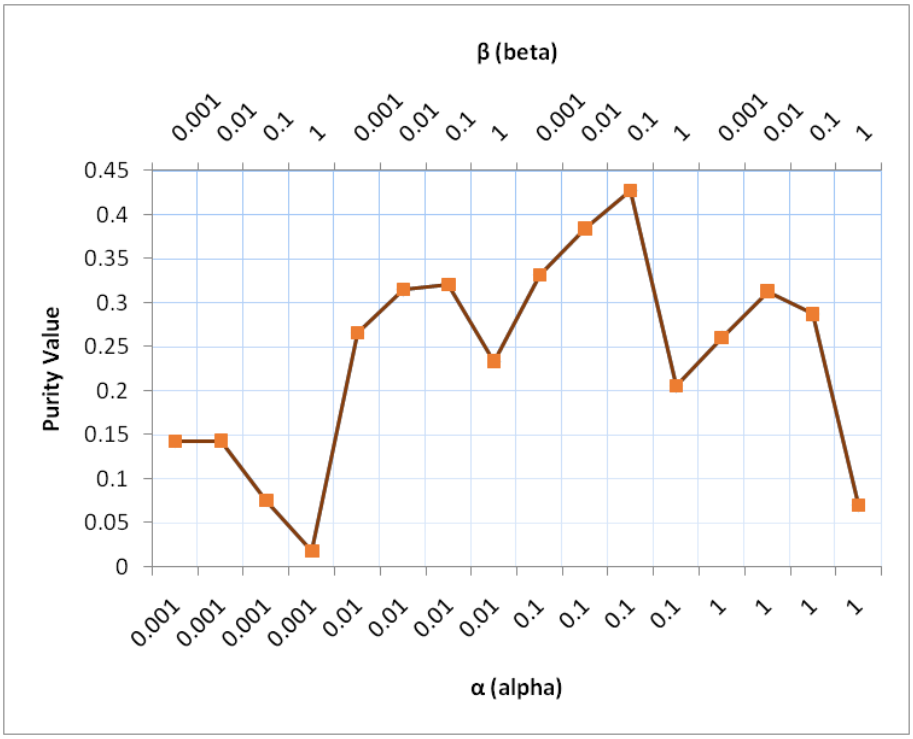


Figure 4.6: Purity results of LDA method with various combinations of α and β on TREC2014 dataset.

in each labeled datasets. We use the NMI measure to evaluate the trade-off between topic quality and the number of topics. For this evaluation metric, we assess all methods for each labeled dataset with a different number of topics. For every experiment, we ran the algorithms over the datasets 30 times and noted the mean of each evaluation metric. α and β hyperparameters in all LDA-based methods are tuned to achieve the best performance for topic derivation in a Twitter environment. Figure 4.6 shows the purity results of LDA method against several combinations of α and β on TREC2014 dataset. Based on these experiments, the best performance is achieved when α and β are set to 0.1 with the purity value of 0.43.

Figure 4.7, 4.8, and 4.9 show the experimental results of all methods for the TREC2014 dataset. We see that our proposed intLDA method outperforms all other methods. For all evaluation metrics, intLDA results in more than 15% improvement over Plink-LDA, and 30-60% improvements over other baseline methods.

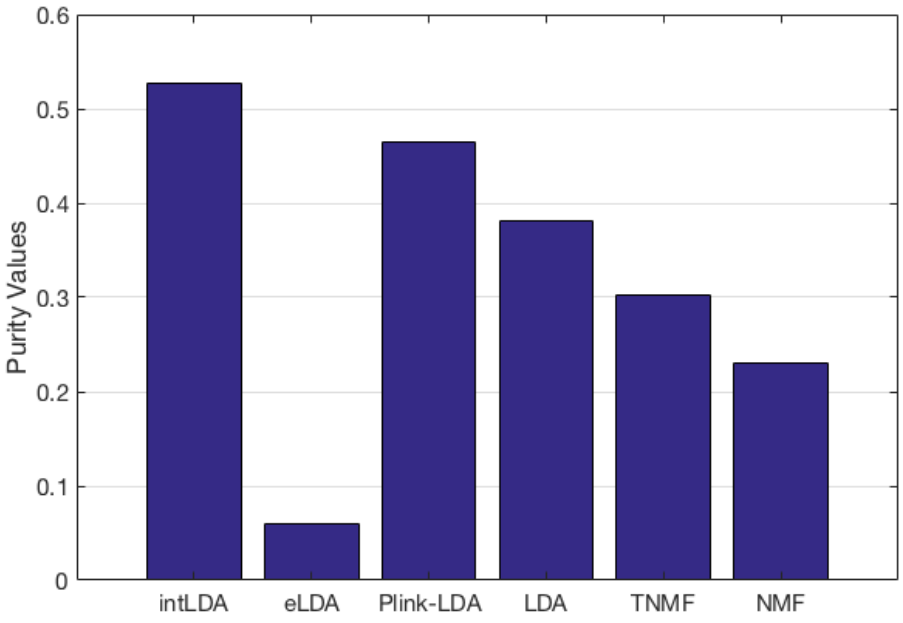


Figure 4.7: Experimental results for TREC2014 dataset using the purity metric

The purity metric evaluates the accuracy of clustering results against the labeled dataset. In Figure 4.7, we see that intLDA achieves the highest accuracy with a purity value of 0.528. Plink-LDA is the next best method with the purity value 0.464, followed by 0.381 for the straight LDA, 0.302 for TNMF, and 0.231 for the original NMF method. Note that intLDA, eLDA and Plink-LDA incorporate the relationships between tweets defined in section 4.3 of this chapter. However, eLDA, the method that augments the content of tweet by the content of another tweet that has a relationship with it, does not perform very well in our experiments, with only a 0.061 purity value in average. It suggests that the added words might not be related to the original tweet's topic, and they eventually become noise, reducing the quality of the derived topics.

We use NMI to find out the trade-off between the given number of topics and the accuracy of the clustering. Figure 4.8 shows the NMI results of all methods for the TREC2014 dataset. We use a different set of k numbers of topics in the

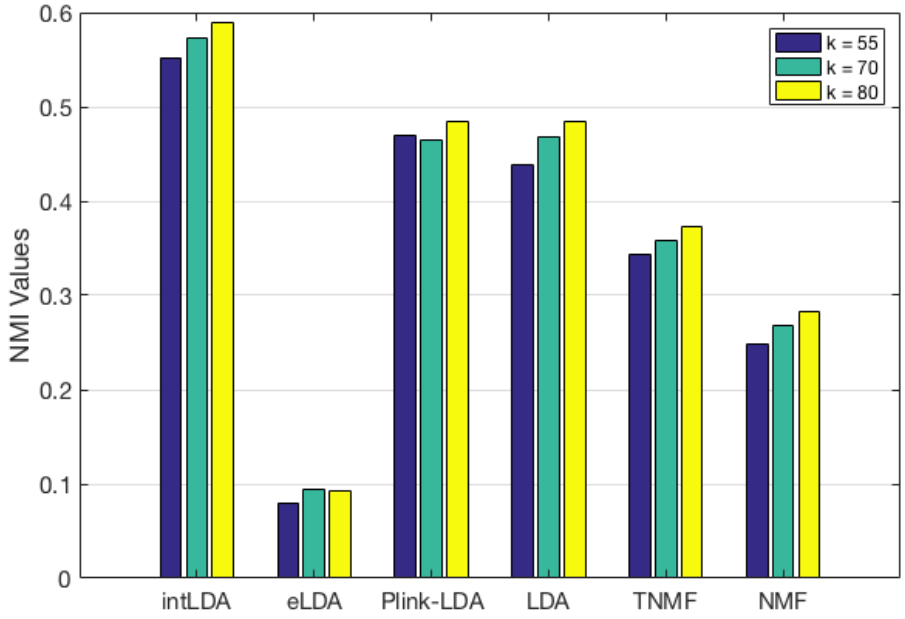


Figure 4.8: Experimental results for TREC2014 dataset using the NMI metric

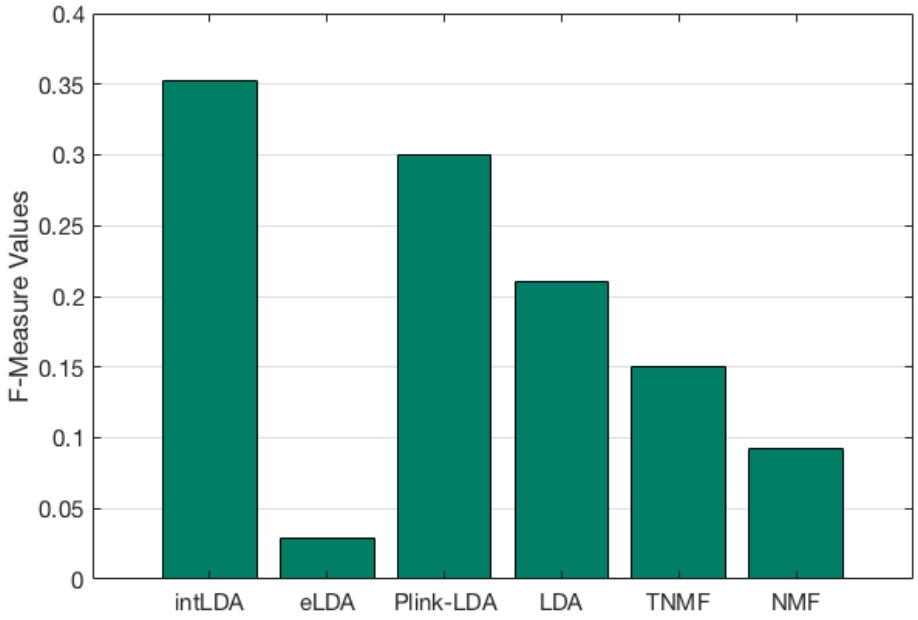


Figure 4.9: Experimental results for TREC2014 dataset using the F-Measure metric

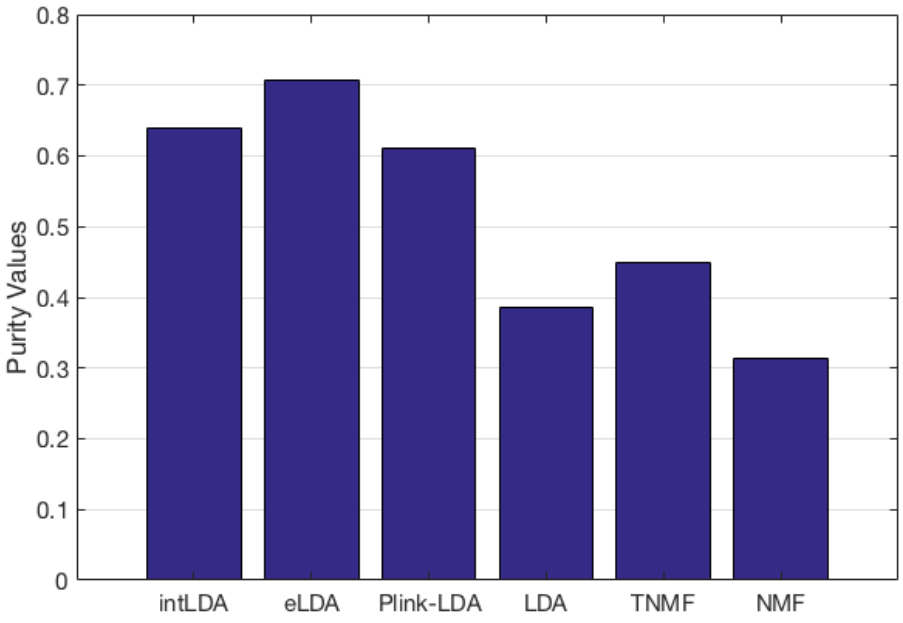


Figure 4.10: Experiment results using the purity metric for tweetSanders dataset

experiments ($k = 55$ - same as labeled dataset, $k = 70$, and $k = 80$). The figure shows that intLDA provides the best NMI results for any k number of topic. Most methods show an improvement of the cluster quality with the higher number of topics, except Plink-LDA with the NMI declined when $k = 70$, which gets a lower value compared to the straight LDA for the same number of topics.

intLDA also presents the best result in the F-Measure evaluation. F-Measure evaluates the harmonic mean between precision and recall of the clustering results against the labeled datasets. Figure 4.9 shows the F-Measure results for all methods for TREC2014, with a similar trend to the purity evaluations where intLDA outperforms other methods with around 15-60% improvements over Plink-LDA, LDA, TNMF, NMF, and eLDA.

The evaluation results for the tweetSanders dataset are shown in Figure 4.10, 4.11, and 4.12. tweetSanders has the highest density for different types of relationships compared with the TREC2014 and tweetMarch datasets (see Table 4.1).

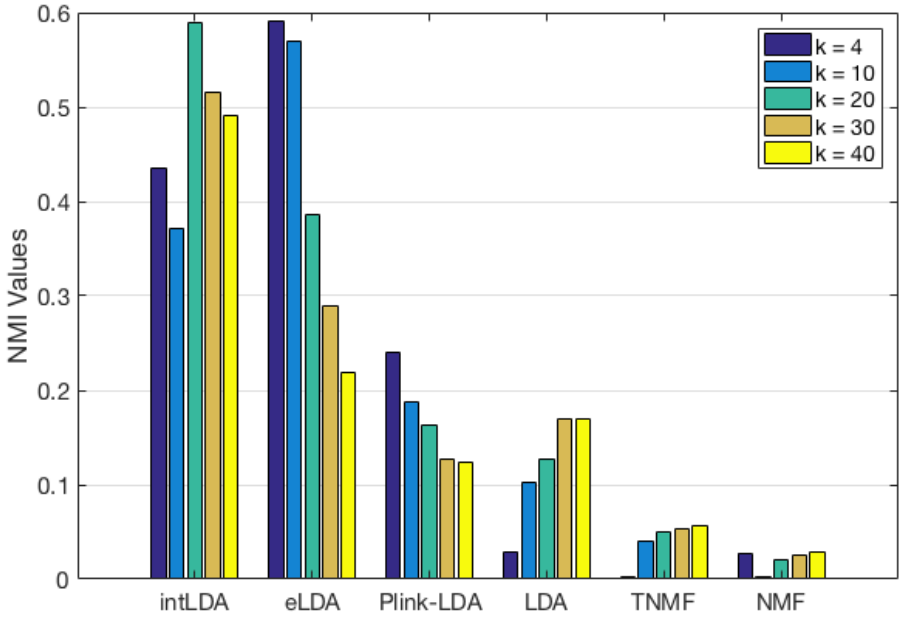


Figure 4.11: Experiment results using the NMI metric for tweetSanders dataset

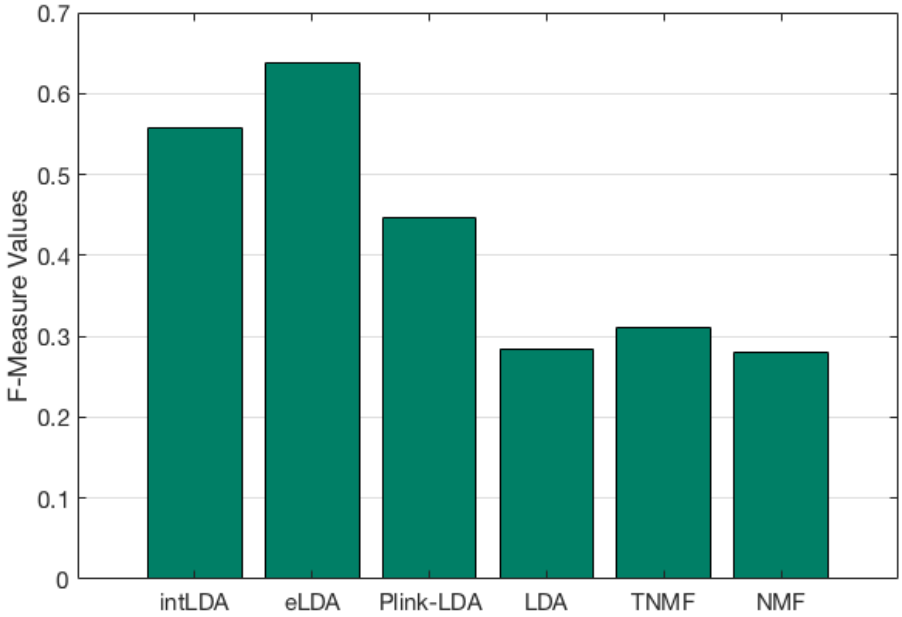


Figure 4.12: Experiment results using the F-Measure metric for tweetSanders dataset

The purity evaluation results for the tweetSanders dataset are shown in Figure 4.10. We see that methods that incorporate the much denser tweet-to-tweet relationships, such as intLDA, eLDA, and Plink-LDA, are able to result in a high performance in purity evaluation. Interestingly, for this specific dataset, eLDA outperforms the purity results of other methods. eLDA gets 0.706 purity value, followed by intLDA (0.639) and Plink-LDA (0.611). The good performance of eLDA on the tweetSanders dataset is explained by the fact that there are much more correlated terms within the connected tweets, as illustrated in Figure 4.3a. The tweet-to-tweet matrix for tweetSanders can accurately capture the 4 main topics available for this dataset. The density of the term-to-term relationships for tweetSanders also expresses the higher correlation between terms over the other datasets. This dramatically improves the performance of methods that incorporate the term correlations like TNMF over the LDA and original NMF.

Figures 4.11 and 4.12 show the results of the NMI and F-Measure evaluations for the tweetSanders dataset. Similar to the purity evaluation, the NMI and F-Measure results show that eLDA outperforms other methods, with around 10 - 20% improvement over the intLDA as the second best, and 20 - 50% improvement over other baseline methods. However, eLDA is more sensitive to the number of topics. From Figure 4.11, we see that the NMI values for eLDA fall when the number of topics is increased.

The results of our evaluations for the tweetMarch dataset are available in Figures 4.13, 4.14, and 4.15. In the purity evaluation, shown in Figure 4.13, all methods are able to result in a quite high performance, with intLDA achieving the highest purity value. The superiority of intLDA is confirmed by the results of the NMI and F-Measure evaluations shown in Figures 4.14 and 4.15 respectively. NMI results show that intLDA is able to maintain the quality of the topic derivation over a different number of topics. The Plink-LDA and TNMF purity values are almost tied. However, in the NMI and F-Measure evaluations, Plink-LDA (which incorporates the

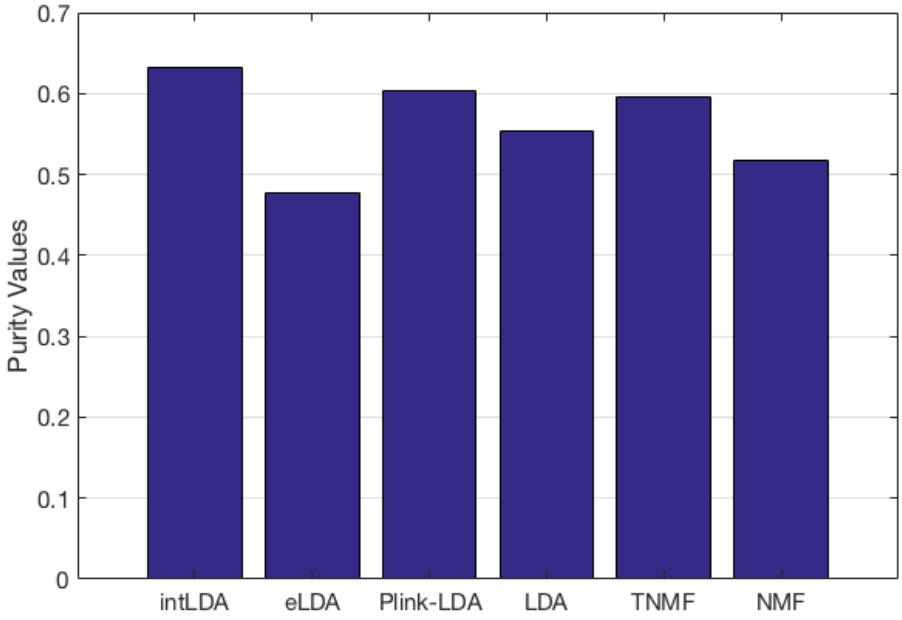


Figure 4.13: Experiment results using the purity metric for tweetMarch dataset

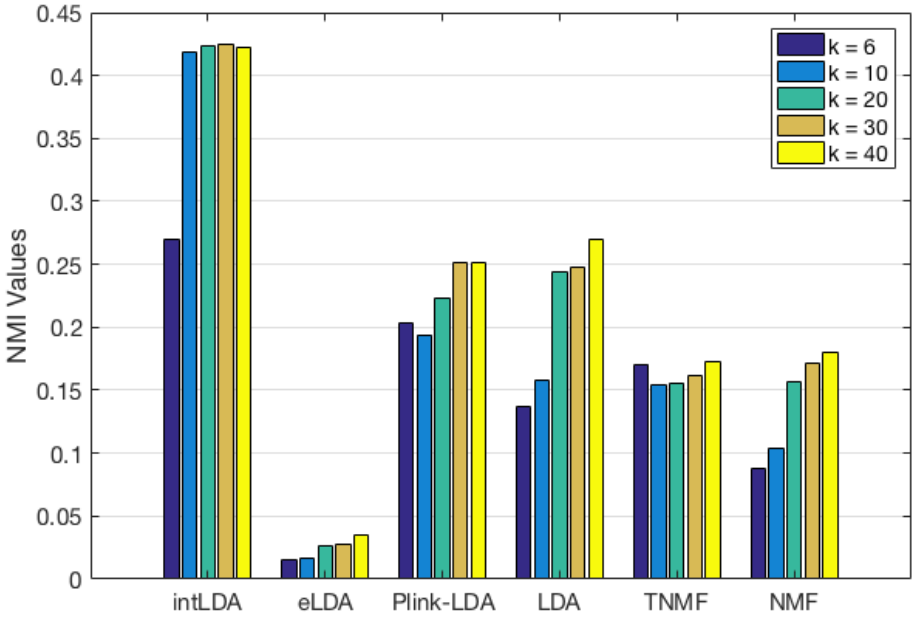


Figure 4.14: Experiment results using the NMI metric for tweetMarch dataset

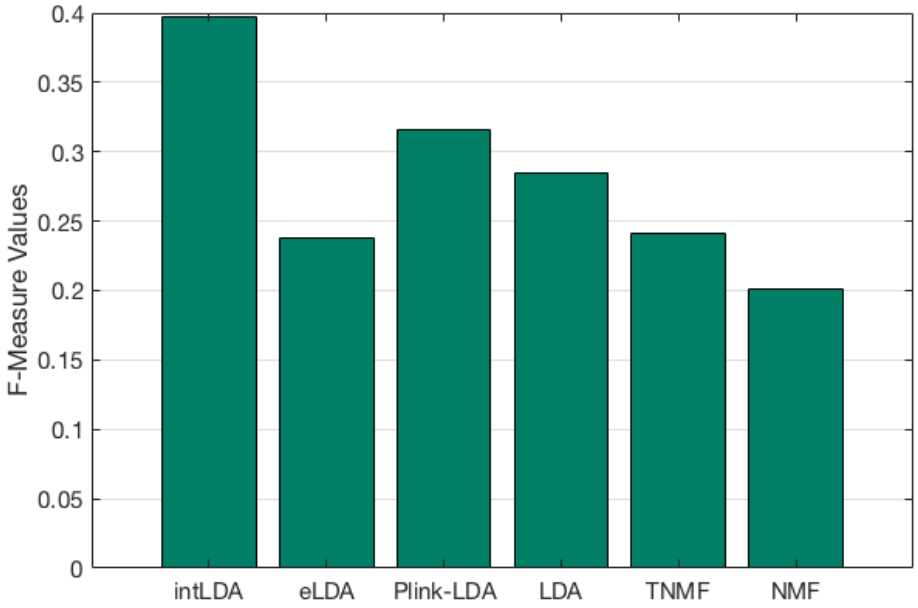


Figure 4.15: Experiment results using the F-Measure metric for tweetMarch dataset

relationship between tweets) outperform NMF by 10 - 20%. Unlike the performance in tweetSanders, eLDA has the lowest results for all evaluation metrics for the tweetMarch dataset.

4.6 Discussion

In this chapter, we have presented the incorporation of relationships between tweets to improve the quality of topic derivation. The relationships between tweets are defined by both social interactions and content similarity. A pair of tweets can be connected by social interactions, such as user mentions, replies, and retweets, and/or content similarity, including hashtags. The availability of mentions, replies, or retweets based social interactions is a sign of the users' involvement in discussions about a particular topic.

Our definition of the tweet relationships results in a much higher density of tweet-to-tweet matrix as compared with another type of content-only based relationships,

including tweet-to-term and term-to-term. This can improve the quality of topic derivation in Twitter.

We have proposed an extension of the well-known LDA method, intLDA, to take the relationships between tweets into account when deriving topics in Twitter. intLDA modifies the LDA Gibbs Sampling model algorithmically to incorporate the observed tweet-to-tweet matrix. The observed relationships between tweets are used as an additional constraint to the θ distribution in the Gibbs sampling process. In this chapter, we have also discussed an implementation of another LDA extension, eLDA, a simple and intuitive use of the tweet-to-tweet relationships to expand the content of the tweet by adding new words to tweets from their related tweets.

Experimental results for three different datasets show that incorporating the relationships between tweets is useful to improve the quality of topic derivation in Twitter. Methods that incorporate the tweet-to-tweet relationships are able to outperform other baseline methods that employ other types of relationships or are based solely on content exploitation. The proposed intLDA method has consistently produced high quality topics for datasets, although they have different characteristics. intLDA also maintains its superiority as compared to Plink-LDA, which also incorporates our tweet-to-tweet matrix, for all datasets and evaluation metrics. The performance of eLDA, however, strongly depends on the quality of the relationships between tweets. The noise from expanded content has a big impact on the accuracy of the clustering. The experiments for TREC2014 and tweetMarch have shown that the performance of eLDA is penalized by the potentially unrelated words during the derivation process. This suggests that, in various situations, incorporating the observed relationships between tweets directly in the sampling process is more robust to noise than introducing words from the related tweets.

Our definition of the tweet-to-tweet relationships has not yet considered the strength of the relationships. In these LDA-based extensions, we use the tweet-to-tweet relationships as additional information to the derivation process, without

considering the accuracy of topical connectivity between pairs of tweets. Based on the experimental results, we find that accuracy of the relationship between tweets is important to further improve the quality of topic derivation. When the relationships are able to capture the topical connectivity between pair of tweets, the accuracy of the clustering is also improved. For example, in the evaluation for the tweetSanders dataset, which has the highest density of tweet-to-tweet relationships, the improvements of the methods that incorporate relationships between tweets over other baseline methods are quite high. In the next chapter, we analyze the effect of each type of relationship on the topical connectivity between tweets. We discuss a probabilistic model to join the effect of both social interactions and tweet content, and propose a new method to incorporate the model for a better quality of topic derivation in Twitter.

Chapter 5

Joint Probability of Tweet Content and Interactions

5.1 Introduction

The experimental results discussed in the previous chapter show that the incorporation of tweet interactions and content similarity through the relationships between tweets can help improve the quality of topic derivation. We now conduct an analysis of the topical connectivity between tweets. We find that tweets linked by interactions and content similarity have different probabilities to be about the same topic. Our previous model of relationships between tweets did not accurately reflect these different probabilities.

In this chapter, we propose a joint probability model for the topical relationships between tweets to integrate the effects of the replies-retweets, user mentions, and the content similarity more accurately. The joint probability model provides a new foundation to build the tweet-to-tweet relationship matrix. However, LDA-based methods discussed in the previous chapter are less flexible to incorporate the weight of the relationships as additional information for topic derivation. The bag-of-words model in LDA relies on the frequency or count of word co-occurrences rather than the weight of their relationships to sample the topics. We thus propose a new approach to topic derivation which utilizes an inter-joint of Non-Negative Matrix Factorization (NMF) technique to process the tweet-to-tweet relationship matrix for

topic derivation. The work in this chapter is summarized as follows:

- We analyze how tweets are associated with topics according to content similarity and interactions between tweets. Tweets linked by replies and retweets are almost always about same topics; tweets linked by user mentions and content similarity have a reasonable chance to be about the same topics. We also note that the number of tweets linked by user mentions and content similarity is much larger than the number of tweets linked by replies and retweets.

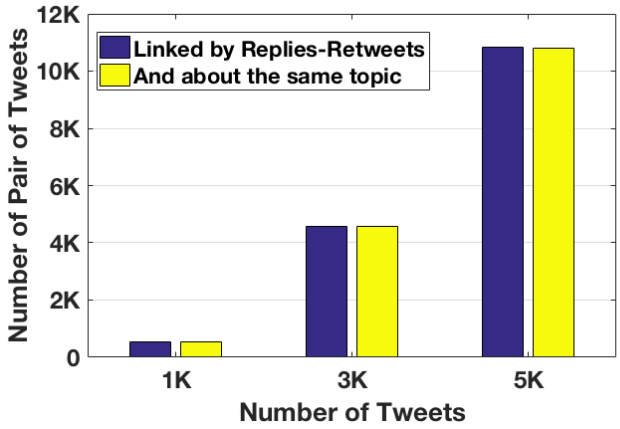
- We develop a joint probability model for the tweet-to-tweet relationships to integrate the effects of content similarity, user mentions, and replies-retweets between tweets. Matrix factorization techniques are used to derive topics based on the proposed joint probability model.

- We again conduct a set of experiments using the evaluation metrics and datasets introduced in Chapter 3. The results show that the proposed model can significantly improve the quality of topic derivation compared to our LDA-based methods described in the previous chapter.

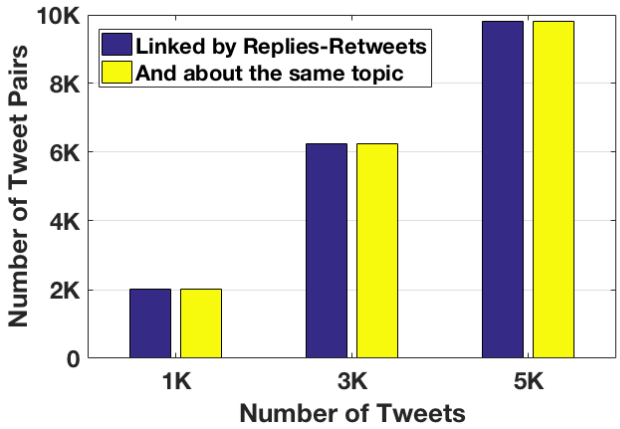
The rest of the chapter is organized as follows. We discuss the model of the tweet-to-tweet relationships in Section 5.2, followed by the implementation of the model into a matrix factorization process in Section 5.3. The experiments and evaluation results are described in Section 5.4. Section 5.5 presents a summary and discussion of the chapter.

5.2 Modeling Relationships between Tweets

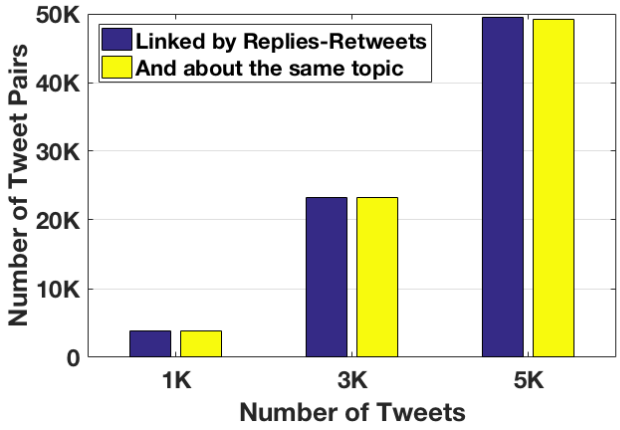
In this section, we present our analysis of the topical connectivity between tweets according to the tweet interactions and content similarity. We first discuss the topical connectivity between the tweets, then propose a joint probability model for measuring the relationships between tweets.



(a) TREC2014



(b) tweetSanders



(c) tweetMarch

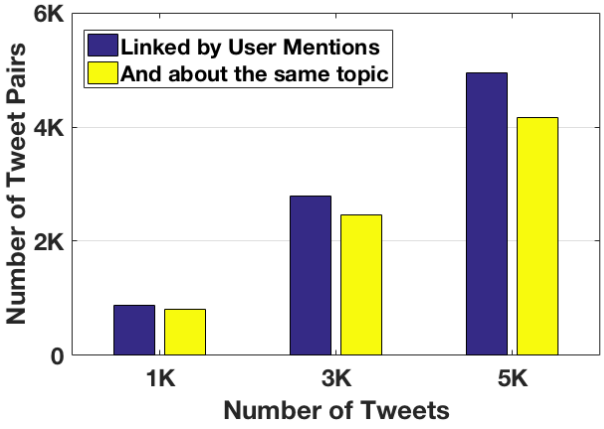
Figure 5.1: The total number of tweet pairs linked by replies-retweets vs the number of tweet pairs linked by replies-retweets and about the same topic

5.2.1 Topical connectivity between tweets

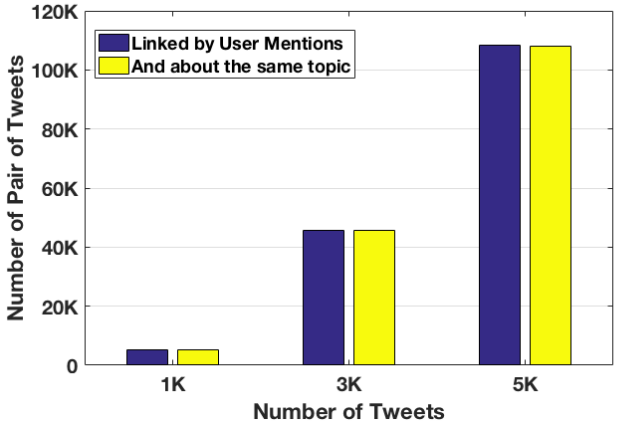
In this subsection, we discuss some observations of the topical connectivity for tweets with respect to the content similarity and other types of interactions. We use the TREC2014, tweetSanders, and tweetMarch datasets described in Chapter 3. The first 1K, 3K, and 5K tweets (ordered by the time they were posted) in those datasets are examined to see if tweet pairs linked by replies-retweets, user mentions, and content similarity are about the same topic.

Figure 5.1 shows the statistics of tweet pairs linked by replies or retweets. The blue (dark) bar in Figure 5.1a, 5.1b, and 5.1c shows the total number of tweets linked by reply or retweet interactions, and the yellow (light) bar shows the number of tweets linked by reply or retweet about the same topic. We find that, for all datasets, more than 99% of tweets linked by replies or retweets are about the same topic. It is safe to conclude that tweets linked by replies and retweets are about the same topic.

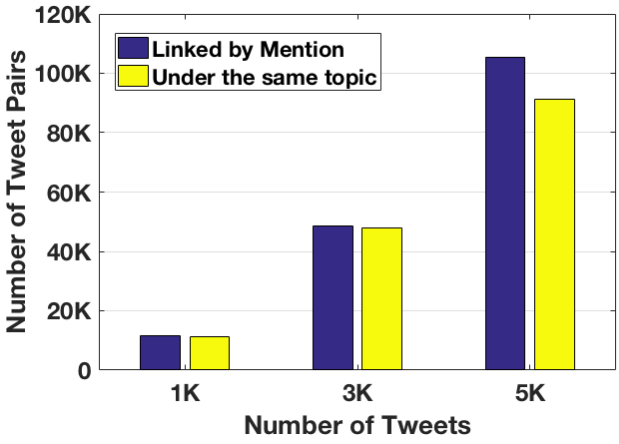
Figure 5.2 shows the statistics of tweet pairs linked by user mentions. We again compare the total number of tweet pairs linked by mentions with the number of tweet pairs linked by mentions about the same topic. The results show tweets linked by user mentions have a high probability to be about the same topic. The TREC2014 and tweetMarch datasets have quite similar statistical results for the 1K, 3K, and 5K tweets. For both datasets, about 80% of the tweet pairs linked by user mentions are about the same topics. tweetSanders shows an even stronger correlation between the mention feature and topics. Figure 5.2b shows that almost 99% of tweets connected through user mentions are about the same topics. This dataset only has 4 topics: Apple, Google, Microsoft, and Twitter, and most of the tweets are likely to mention specific usernames like *@apple*, *@google*, *@microsoft*, and *@twitter*, which explains the bias between topics and user mention features in this dataset.



(a) TREC2014

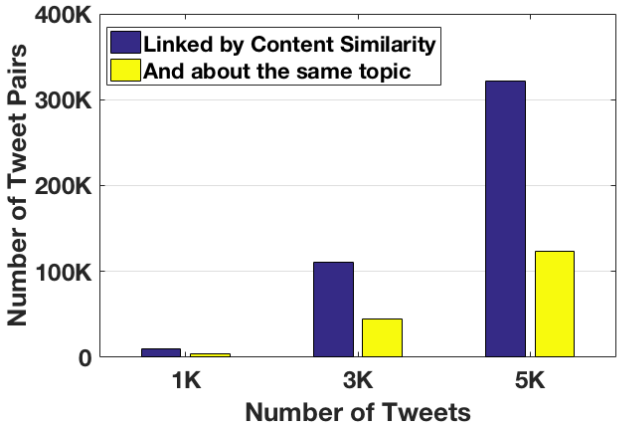


(b) tweetSanders

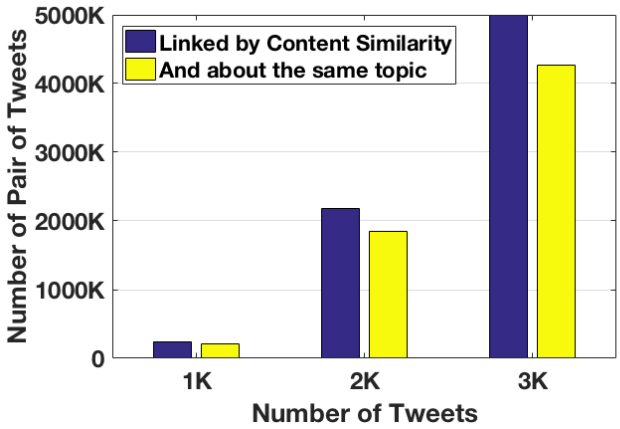


(c) tweetMarch

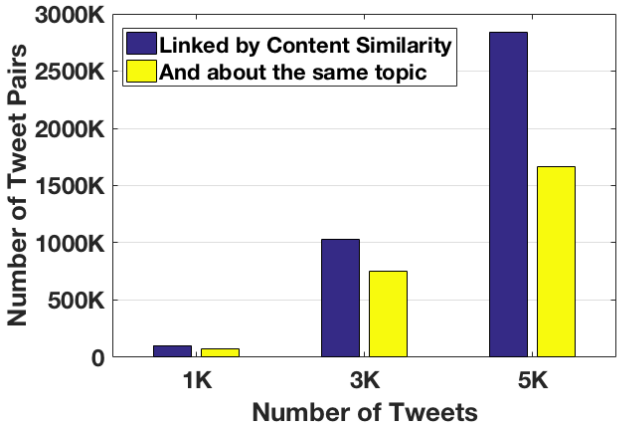
Figure 5.2: The total number of tweet pairs linked by user mentions vs the number of tweet pairs linked by user mentions and about the same topic



(a) TREC2014



(b) tweetSanders



(c) tweetMarch

Figure 5.3: The total number of tweet pairs linked by content similarity vs the number of tweet pairs linked by content similarity and about the same topic

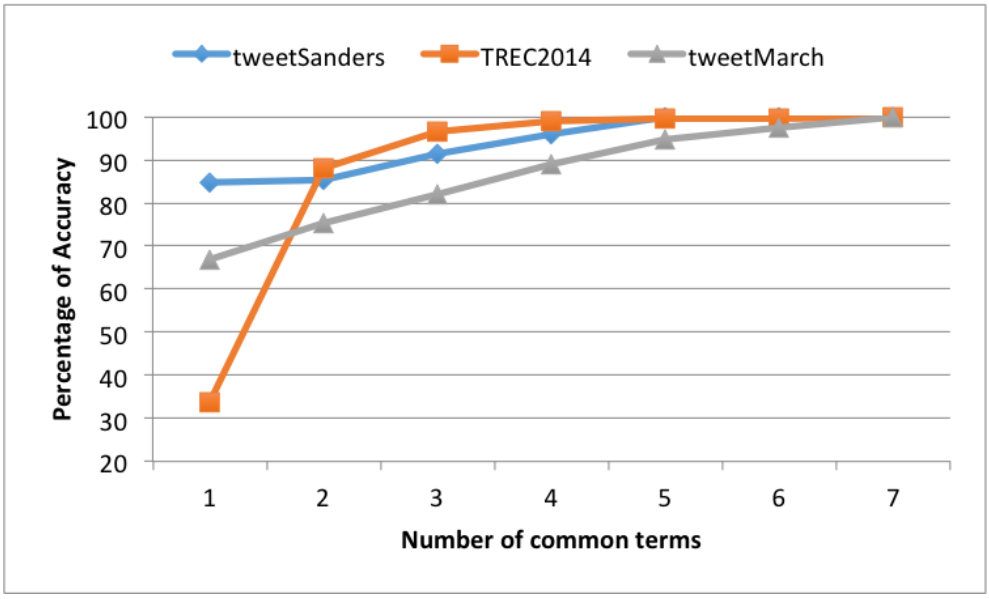


Figure 5.4: Topical connectivity of tweet pairs linked by content similarity with different numbers of common terms

Figure 5.3 shows the statistics of tweet pairs linked by their content similarity. Tweets linked by content similarity have less chance to be about the same topic compared to tweets linked by replies, retweets, and user mentions. About 51% of tweet pairs linked by the content similarity are about the same topic. Furthermore, as shown in Figure 5.4, the probability of two tweets being on the same topic increases as the number of terms they have in common increases. However, tweet pairs with more than one common term are rare, with about 90% of tweet pairs linked by the content similarity having only one term in common.

To summarize, we find that tweets linked by reply-retweets, mentions, and content similarity have different probabilities to be about the same topic. Two tweets linked by reply or retweet are always about the same topic; two tweets linked by user mentions have a high probability to be about the same topic; two tweets linked by content similarity have some chance to be about the same topic.

5.2.2 Joint Probability Model

Based on the analysis presented above, we define a joint probability model for the topical relationship between tweets integrating the effects of replies-retweets, user mentions, and content similarity. We at first consider each of these with individual probabilities (without considering others), then join these probabilities together.

A tweet is defined as a tuple $t = \langle U_t, rtp_t, C_t \rangle$, where U_t is all users mentioned in the tweet including its original author, rtp_t is the reply and retweet information, and C_t is the set of terms from the tweet, including hashtags. The relationship between two tweets t_i and t_j is then denoted as $R(t_i, t_j)$. It is a combination of three components: reply-retweet ($act(t_i, t_j)$), user mention ($m(t_i, t_j)$), and content similarity ($sim(t_i, t_j)$).

When two tweets t_i and t_j are linked by reply or retweet, the probability of reply-retweet relationship $P(act(t_i, t_j))$ is defined as:

$$P(act(t_i, t_j)) = \begin{cases} 1, (rtp_{t_i} = j)\ or\ (i = rtp_{t_j}) \\ \qquad or\ (rtp_{t_i} = rtp_{t_j}) \\ 0,\ otherwise \end{cases} \tag{5.1}$$

where rtp_{t_i} and rtp_{t_j} are the IDs of tweets which are replied to or retweeted. The probability of two tweets about the same topic is 1 if the two tweets in the pair refer to each other or refer to one tweet, otherwise, it is 0.

When two tweets t_i and t_j are linked by mention, the user *mention* relationship $P(m(t_i, t_j))$ is defined as the intersection of U_{t_i} and U_{t_j} divided by the total number of all users involved in both tweets. As defined before, U_t is the set of mentioned users in tweet t including the author of the tweet. The probability of user mention relationship is formulated as follows:

$$P(m(t_i, t_j)) = \frac{|U_{t_i} \cap U_{t_j}|}{|U_{t_i} \cup U_{t_j}|} \tag{5.2}$$

In the motivating example shown in Table 1.2 of Chapter 1, we can see that tweet

t_5 ("*@d any special event in particular worth coming for?*") mentions user d in the post, so $U_{t_5} = \{d, e\}$. Since t_4 ("*#Floriade in #Canberra, biggest celebration of spring in Australia*") does not mention other user, U_{t_4} will contain the author's username only ($U_{t_4} = \{d\}$). Thus, $P(m(t_4, t_5))$ will be 0.5, since d is the only common user available in both tweets as a result of the mention activities.

When two tweets t_i and t_j are linked by the content similarity, the probability of content similarity relationship $P(sim(t_i, t_j))$ is defined as:

$$P(sim(t_i, t_j)) = \frac{|C_{t_i} \cap C_{t_j}|}{|C_{t_i} \cup C_{t_j}|} \tag{5.3}$$

where C_t is the set of unique terms available in tweet t. Similar to the user mention relationship, the probability of each pair of tweets is calculated as the ratio of the number of common terms in tweets t_i and t_j to the total number of terms in these two tweets. Note that, in the preprocessing steps, all terms/characters that potentially degrade the performance of topic identification processes (i.e., emoticons, punctuations, and terms with fewer than 3 characters) are removed. We also remove stop words, and are thus left only with the content-full words. Hashtags are included and kept unchanged.

In the previous chapter, we did not consider the strength of the topical connectivity when incorporating the relationships between tweets. In a replies-retweets situation, this is not a problem. Based on the analysis in the last subsection, the tweets linked by reply or retweet are always talking about the same topic. However, if there is another pair of tweets that have the reply-retweet value 0, user mention value 0, and the content similarity value 0.1, it might not be accurate if we say that both tweets are strongly connected to the same topic. It is unavoidable for the inaccurate evaluation of the relationships to bring in a negative impact on the quality of topic derivation. To overcome this drawback, we now propose a joint probability model for the relationships between tweets.

Two tweets could be linked by reply-retweet, user mention, or content similarity, or their combinations. When the value of reply-retweet is not zero, we use the value of this feature to represent the relationship. It explicitly says that the pair is sharing similar topics. However, if the value of a reply-retweet relationship is zero, we need to calculate tweet relationships by joining the value of probabilities from both user mention and content similarity.

We assume that the mention and content similarity are independent probabilities. To test this assumption, we conduct a ranking test for all relationships made by content similarity and user mentions, and see if they satisfy the rule of independent event $P(m(t_i, t_j)|sim(t_i, t_j)) = P(m(t_i, t_j))$. For each dataset, we assign an ordered number to all possible pairs of tweets for the purpose of sorting in the next process. Then, we choose all pairs that have non-zero value on both mention and content similarity. We make two lists of the pairs and their values. The first list contains the pairs' numbers and the mention values, and the second list contains the pairs' numbers and the content similarity values. Both are sorted by the values followed by the sorting of the number of pairs. Here, we find that, for all datasets, there is no pair of tweets that have the same position when they ranked by both mention and content similarity values.

In the tweetSanders dataset, there are 1,082,500 pairs of tweets connected by mention feature and 4,998,700 connected by the content similarity. From those pairs of tweets, only 122,682 pairs are connected by both mentions and content similarity, and none of them have the same rank when sorted by the probability value and the pair's number. When we classify the probability values and ignore the rank (i.e, probability values are classified into a range of $0-0.25, 0.25-0.5, 0.5-0.75$, and $0.75-1$), only 11.75% of the pairs from TREC2014 and 9.4% from tweetMarch are in the same groups.

Based on the above observations, it is safe to process the mention and content similarity as independent variables. The joint probability model for measuring the

relationships between tweets t_i and t_j is defined as follows:

$$R(t_i, t_j) = \begin{cases} 1, P(act(t_i, t_j)) > 0 \\ P(m(t_i, t_j) \cup sim(t_i, t_j)),\ otherwise \end{cases} \tag{5.4}$$

$$\begin{aligned} where \quad & P(m(t_i, t_j) \cup sim(t_i, t_j)) \\ & = P(m(t_i, t_j)) + P(sim(t_i, t_j)) - P(m(t_i, t_j) \cap sim(t_i, t_j)), \\ and \quad & P(m(t_i, t_j) \cap sim(t_i, t_j)) = P(m(t_i, t_j)) x P(sim(t_i, t_j)) \end{aligned} \tag{5.5}$$

The probability value of the relationships between the tweets falls in the range $[0, 1]$. For the case of the pair of tweets with replies-retweets value 1, mentions 0.8, and content similarity 0.4, the value of the relationship is 1. If two tweets with reply-retweet value 0, the mention value 0, and content similarity value 0.1, the value of the relationship is 0.1. The values of relationships between tweets form a new tweet-to-tweet relationship matrix $A \in \mathbb{R}^{m \times m}$, where $a_{ij} = R(t_i, t_j)$. This matrix is the input of the inter-joint matrix factorization process for topic derivation.

5.3 Matrix inter-joint factorization for topic derivation

In the previous chapter, we proposed the eLDA and intLDA methods to deal with the sparsity problem. eLDA expands the tweet content by addingwords from all connected tweets. intLDA directly incorporates the observed tweet relationships in the process of learning the tweet-topic distribution θ. The relationships between tweets are used as an additional variable to adjust the tweet-topic distribution. If two tweets are related, then the tweet-topic distributions of those two tweets are simultaneously adjusted. The evaluation results showed that intLDA outperformed the original LDA and other advanced baseline methods. We believe that results can be further improved by taking account of the relationship weight between

tweets rather than just connected or unconnected like in intLDA. However, LDA-based methods are less flexible to incorporate the weight of the relationships as an additional information for topic derivation.

Incorporating additional information in an existing process has been extensively studied in clustering and recommender systems to deal with data sparsity such as the *Cold-start* problem, e.g., [68, 145]. Cold-start is a condition where the input data of the systems are very sparse. To improve on topic derivation, the information to be added to the process needs to be integrated with content attributes, as the objective of topic derivation on Twitter is to find the best representative words for topics and achieve accuracy in tweet clustering.

Non-negative Matrix Factorization (*NMF*) [62] is an effective method to uncover the hidden thematic structure of a data matrix by factorizing the matrix into its lower dimensional representation. It offers a more flexible way of incorporating strength based side information into the factorization process. NMF is also highly flexible and can be implemented in a distributed [66] or online system [124]. In this section, we discuss our proposed NMF based approach to incorporate our joint probability based relationships between tweets for topic derivation.

5.3.1 Non-negative Matrix Factorization

NMF is a popular dimensional reduction technique, and one of its main application domains is unsupervised clustering [56, 37, 51, 109, 133]. *NMF* methods commonly output both the tweet clusters of potential topics and the topic-words for each topic.

Let $V \in \mathbb{R}^{m \times n}$ be a tweet-to-term matrix, and k the number of the topics to be derived, *NMF* factorizes the matrix V into a tweet-topic matrix $W \in \mathbb{R}^{m \times k}$ and a topic-term matrix $H \in \mathbb{R}^{k \times n}$. Both W and H are often called latent matrices:

$$V \approx WH \ , \tag{5.6}$$

where the tweet-topic matrix W represents the relationships between the tweets and topics in the form of tweet clusters, each cluster stands for a potential topic; and the topic-term matrix H contains all the topic-words for every topic.

In this context, the tweet-to-term matrix V contains the relationships between the tweets and the unique terms appearing in all these tweets. In particular, each element v_{ts}, with regards to a pair of tweet t and term s, is defined by *tf-idf* or "term frequency - inverse document frequency" [108].

$$v_{ts} = tfidf(s,t,T) = tf(s,t) \times idf(s,T)\ , \tag{5.7}$$

where the term frequency $tf(s,t)$ is the number of times the term s occurs in the tweet t, and the inverse document frequency $idf(s,T)$ is a measure of whether the term s is common or rare in the tweet collection T.

In terms of *Kullback-Leibler (KL) Divergence* [58], the factorization process of *NMF* aims at finding the minimum divergence of $V \approx WH$, using the following cost function:

$$D(V\|WH) = \sum_{ij}(v_{ij}\log\frac{v_{ij}}{(wh)_{ij}}) - v_{ij} + (wh)_{ij}, \tag{5.8}$$

$$\tag{5.9}$$

with update rules for every iteration:

$$W = W\frac{H^T(V/(WH))}{H^T I}\ and\ H = H\frac{(V/(WH))W^T}{IW^T}\ . \tag{5.10}$$

NMF is considered equivalent with PLSA method when the KL divergence is used as objective function [32].

As already mentioned, when working on the Twitter platform, the tweet-to-term matrix V is usually extremely sparse. Consequently, directly extracting the hidden thematic structure from this tweet-term matrix often results in low quality of tweet clusters and poor readability of topics. As seen in the experimental results

described in the previous chapter, the original NMF does not perform well in the Twitter environment. To overcome this problem, we need to extend the method to incorporate the joint probability model of the tweet relationships described in the previous section.

5.3.2 Joint-NMF

In order to address the sparsity issue by incorporating the joint probability based relationships between tweets, we perform two consecutive factorization processes, then jointly use a sharing latent matrix across the processes. The first factorization process aims to cluster the tweets based on their topics, and then the resulted clusters are used to infer keywords to represent each topic.

Clustering Tweets

The relationships between tweets described in Section 5.2.2 is modeled as the combination of various interactions and content similarity. The values of the relationships form a tweet-to-tweet relationship matrix $A \in \mathbb{R}^{m \times m}$, which expresses the topical connectivity between tweets. In our approach, matrix $A \in \mathbb{R}^{m \times m}$ is factorized into its lower dimensional tweet-topic matrix $W \in \mathbb{R}^{m \times k}$ and $Y \in \mathbb{R}^{k \times m}$ where k is the given number of clusters/topics. It can be directly used to generate the topical clusters of the tweets. Since A is a symmetric matrix, either W and Y is able to show the potential cluster for every tweet. The objective of this factorization process is similar to the original NMF, which is to minimize the divergence of A and WY so that $A \approx WY$ with the cost function $D(V||WY)$.

Figure 5.5 shows the results of the factorization process of the tweet-to-tweet relationship matrix A from the sample of tweets in Table 1.2 of Chapter 1. In this figure, W and Y are the latent tweet-topic matrices derived from A with the number of topics $k = 2$. These two matrices are the lower dimensional representations of the matrix A. We can see that, in matrix A, the strong connection between tweets

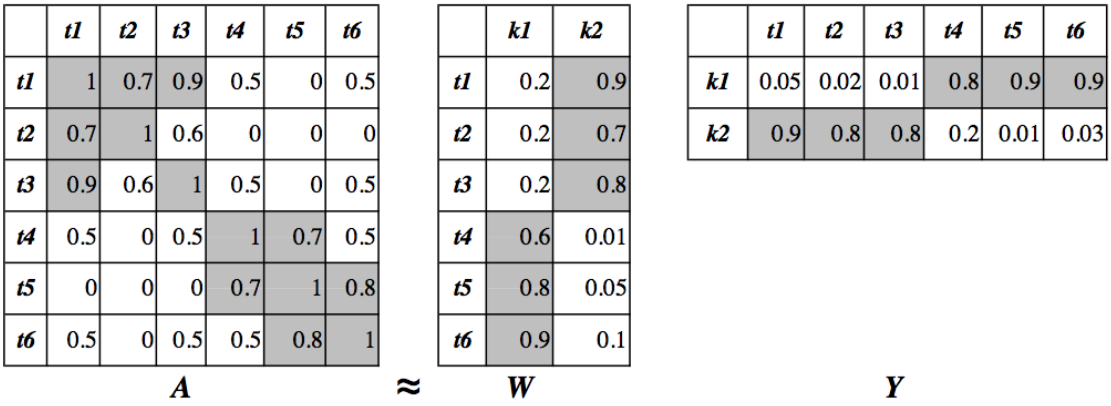


Figure 5.5: Factorization of tweet-to-tweet relationship matrix A into the latent matrix W and Y. The dark areas indicate the potential topical clusters of the tweets.

are marked in the dark areas. The matrix also shows how the tweets are grouped. In both matrices W and Y, the representation of the relationships in k number of topics is consistent. For example, if, for every row in matrix W, we take the highest value to define the cluster membership, t_1, t_2 and t_3 are in cluster k_2, and t_4, t_5, and t_6 are in cluster k_1. In the next step, the tweet-topic matrix $W \in \mathbb{R}^{m \times k}$ is used as an additional information when learning the keywords representation to deal with the sparsity of the tweet-to-term matrix V.

Inferring Keywords Representation for Each Topic

The second step of the joint-NMF process is to infer the best keywords to represent every topic. In a general NMF, the representative keywords are captured by factorizing the tweet-to-term matrix directly into the tweet-topic matrix and the topic-term matrix. Each element in the tweet-to-term matrix is computed using the *term frequency-inverse document frequency* (tf-idf) metric [108]. This metric calculates the weight of every unique term in a tweet. The higher the value of the tf-idf of a term in a tweet, the more important this term is to the tweet.

joint-NMF makes use of the tweet-to-term matrix to infer the representative keywords. In particular, we compute the tweet-to-term matrix V using the *tf-idf*

value for every tweet and all unique terms in each of them. Furthermore, the tweet-to-term matrix $V \in \mathbb{R}^{m \times n}$ is then factorized into tweet-topic matrix $W \in \mathbb{R}^{m \times k}$ and the topic-term matrix $\widetilde{H} \in \mathbb{R}^{k \times n}$. m is the number of tweets in a collection, n is the number of unique terms, and k is the number of potential topics defined by user. The objective function of the second factorization process is $\min D(V \| W\widetilde{H})$ where

$$\min D(V \| W\widetilde{H}) = \sum_{ij} (v_{ij} \log \frac{v_{ij}}{(w\tilde{h})_{ij}}) - v_{ij} + (w\tilde{h})_{ij} \ , \tag{5.11}$$

$$\widetilde{H} = H \frac{(V/(WH))W^T}{IW^T} \ . \tag{5.12}$$

The tweet-to-term matrix V is very sparse. As shown in Tables 4.1 of Chapter 4, the average density of tweet-to-term matrix (the non-zero element in the tweet-to-term matrix which shows the availability of relationship between tweet and term) is less than 0.1%. Thus, to reduce the negative impact of this extreme sparsity, we modify the NMF approach when factorizing the matrix. Firstly, we use the tweet-topic matrix derived from the previous step to initialize the matrix W. Secondly, during the iteration to minimize the divergence between matrix V and WH, we only update the matrix H and retain matrix W in its original value. Matrix W was derived from the tweet-to-tweet relationship matrix A, which is much less sparse if compared to the tweet-to-term matrix V. Our investigation shows that each cluster from the derived matrix W in the first step of the algorithm provides the most accurate topic [87]. The biased update rule for W in the second step will provide additional information for the process of inferring the topic-term matrix H, and, in the same time, reduce the penalty of the extreme sparsity of the tweet-to-term matrix V. In every iteration, the update rule for matrix H is shown in equation 5.12.

The complete two-step process is illustrated in Figure 5.6. From this figure, we can see the connection between the first factorization and the subsequent process. The second factorization process takes the matrix W from the previous step to

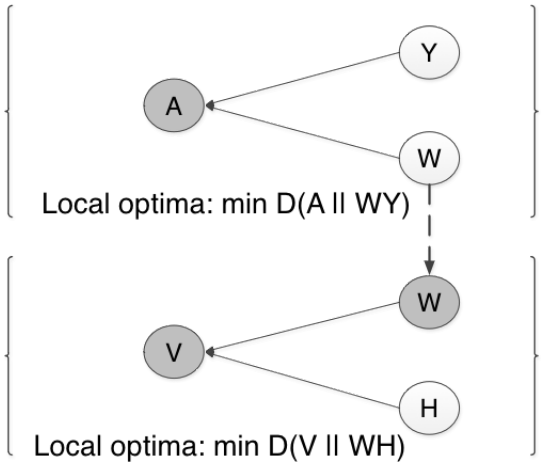


Figure 5.6: *joint-NMF* Model

infer matrix H without updating the matrix W. We call these consecutive steps as *joint-NMF*. This model can also be expressed as follows:

$$A \approx WY \mapsto V \approx W\widetilde{H}, \tag{5.13}$$

In summary, these joint factorization methods can be specified as two independent processes sharing a latent matrix (W). In each step, the factorization aims to find the local optima with the corresponding cost function $\mathscr{T}_{Joint}$:

$$\mathscr{T}_{Joint-1st-process} = D(A\|WY)\,. \tag{5.14}$$

$$\mathscr{T}_{Joint-2nd-process} = D(V\|WH)\,. \tag{5.15}$$

After inferring the topic-term matrix $\widetilde{H}$, a set of top N terms is selected to represent the corresponding topic index. Note that a specific word might occur in several such sets, that is, it might be amongst the representative words for several topics.

Table 5.1 shows the topic-term matrix H after performing joint-NMF on V, the tweet-to-term matrix built using the motivating example from Table 1.2 of Chapter

Table 5.1: Topic-term matrix (H) from Joint-NMF on $V \approx WH$ (Matrix is transposed due to insufficient space)

	k_1	k_2
new	4.47e-10	**0.55**
senate	4.15e-13	**0.51**
exciting	3.54e-15	**0.54**
#canberra	0.17	0.21
census	2.31e-10	**0.43**
#floriade	**0.59**	1.15e-29
celebration	**0.35**	9.45e-30
spring	**0.57**	7.82e-30
event	**0.55**	1.12e-12
nightfest	**0.43**	2.32e-24

1. For readability purposes, words with a very low value for both topics (rows) are removed from the table. Thus, the keywords representation for topic k_1 can be inferred as *#floriade, celebration, spring, event, nightfest*. For cluster k_2, the best topic representation will be: *new, senate, exciting*. In topic derivation, a keyword is listed in several topics. In this case, '#canberra' can be included to represent both k_1 and k_2 as it has a high and almost similar value for both clusters. The whole topic derivation process of joint-NMF is described in the Algorithm 2.

Algorithm 2 Topic derivation using *joint-NMF*

INPUT: number of topics K, tweet-to-term matrix $V \in \mathbb{R}^{m \times n}$
OUTPUT: tweet-topic matrix $W \in \mathbb{R}^{m \times k}$ and topic-term matrix $H \in \mathbb{R}^{k \times n}$

1: get tweet-to-tweet matrix $A \in \mathbb{R}^{m \times m}$
2: initialize W, Y and H
3: NMF on $A \approx W.Y$
4: **repeat**
5: $\quad H \leftarrow f(V, W, H)$
6: **until** $V \approx W.H$
7: **return** W,H

The joint-NMF described above still has limitations caused by the two consecutive NMF processes. Because of this two-processes approach, the tweet-topic latent matrix derived in the first process is not able to take tweet-to-term information into

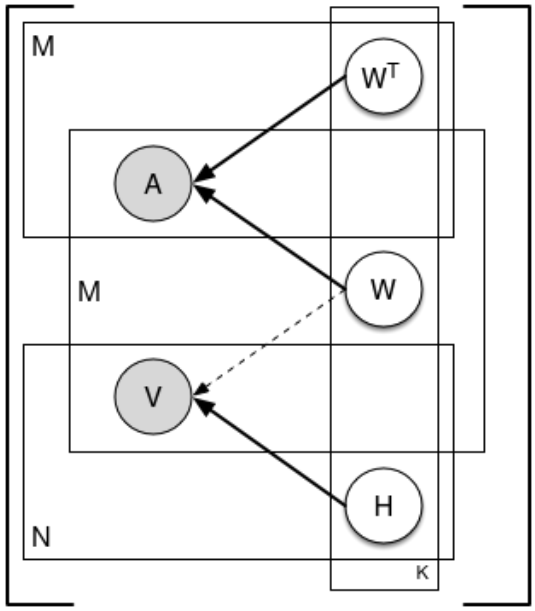


Figure 5.7: Graphical Model of *NMijF*

account. In addition, the two-processes approach generally doubles the computation resources. We thus looked for a way to incorporate the tweet-to-tweet relationships information within a single process, which will be discussed in the next section below.

5.3.3 Non-negative Matrix inter-joint Factorization

To learn topics from two matrices in a single process, Non-negative Matrix Co-factorization or *NMcF* [145, 52, 75] seems to be the natural option. Consider the goal of deriving topics from the tweet-to-tweet matrix A and the tweet-to-term matrix V. NMcF methods are able to factorize these two matrices in a single iterative-update process into two latent matrices: W and H, which represent the tweet-topic and topic-term features respectively. During the process, shared latent matrices W and H are iteratively updated from both A and V.

In contrast to joint-NMF, the NMcF method factorizes two matrices within a single process under one cost function. This enables the interactions of the latent matrices to be learned. In particular, the tweet-topic matrix W will be affected by

both the tweet-to-tweet and tweet-to-term matrices.

Although *NMcF* method has made a great success in recent years in the fields of collaborative filtering, clustering and image/sound processing [52, 75], they do not seem to have had much success with topic derivation on Twitter. Our analysis finds that this is because the extreme sparsity of the tweet-to-term matrix V still heavily penalizes the latent matrix W [89].

To take the advantages of both joint-NMF and NMcF while avoiding their disadvantages, we design a Non-negative Matrix inter-joint Factorization, denoted as *NMijF*. The graphical model of NMijF is shown in Figure 5.7. Similar to NMcF, our proposed NMijF approach factorizes two non-negative matrices, the symmetric tweet-to-tweet matrix $A \in \mathbb{R}^{m \times m}$ and the tweet-to-term matrix $V \in \mathbb{R}^{m \times n}$, in a unified iterative update process. NMijF takes a joint-approach in each iteration, that is, it updates the sharing latent matrix W only according to the tweet-to-tweet matrix A.

The idea behind the NMijF is that the tweet-to-tweet matrix incorporates both interaction and content attributes of the tweets and can provide more information regarding the clustering characteristics of the tweets. Technically, because the tweet-to-tweet matrix A is much less sparse than the tweet-to-term matrix V, a biased update rule for W will reduce the penalty due to the sparsity of V. Furthermore, the formation of the sharing matrix W will be affected by the progress of other latent matrices as the result of having the joint factorization in a single process.

In particular, we construct a cost function $\mathscr{T}_{NMijF}$ that combines the minimum divergences of $A \approx WW^T$ and $V \approx WH$. Such a combination will optimize the latent matrices within a single iterative-update process.

$$\begin{aligned} \mathscr{T}_{NMijF} &= \mathscr{D}(A \| WW^T) \; + \; \alpha \mathscr{D}(V \| WH) \\ &= \sum_{im} d(a_{im} | (ww^T)_{im}) + \alpha \sum_{mn} d(v_{mn} | (wh)_{mn}) \end{aligned} \tag{5.16}$$

where there exists at least one element w and h in each of the matrices W and H such that $w \geq 0$ and $h \geq 0$, and the scaling parameter α satisfies $0 \leq \alpha \leq 1$. α controls the negative effect of the sparse tweet-to-term matrix V.

For each element-wise divergence, we employ generalized *Kullback-Leibler divergence*:

$$d(a_{im}|(ww^T)_{im}) = a_{im}\log\frac{a_{im}}{(ww^T)_{im}} - a_{im} + (ww^T)_{im}, and \tag{5.17}$$
$$d(v_{mn}|(wh)_{mn}) = v_{mn}\log\frac{v_{mn}}{(wh)_{mn}} - v_{mn} + (wh)_{mn}$$

To derive the multiplicative update rules for every element in each iteration, we follow the parameter estimation procedure from [114] by introducing auxiliary variables $r_{i,m,k}$ and $s_{m,n,k}$ ($\sum_k r_{i,m,k} = 1$, $\sum_k s_{m,n,k} = 1$), and use the Jensen's inequality [57] to derive the upper bound $\mathscr{F}$ of $\mathscr{T}_{NMijF}$

$$\mathscr{T}_{NMijF} = \mathscr{D}(A\|WW^T) \ + \ \alpha\mathscr{D}(V\|WH) \tag{5.18}$$
$$\leq \sum_{im}((ww^T)_{im} - a_{im}\sum_k r_{i,m,k}\log\frac{w_{i,k}w^T_{k,m}}{r_{i,m,k}})$$
$$+ \alpha\sum_{mn}((wh)_{mn} - v_{mm}\sum_k s_{m,n,k}\log\frac{w_{m,k}h_{k,n}}{s_{m,n,k}})$$
$$\cong \mathscr{F} \tag{5.19}$$

Equality is achieved if and only if:

$$r_{i,m,k} = \frac{w_{i,k}w^T_{k,m}}{\sum_k w_{i,k}w^T_{k,m}}, s_{m,n,k} = \frac{w_{m,k}h_{k,n}}{\sum_k w_{m,k}h_{k,n}} \tag{5.20}$$

For w_{ik}, the partial differentiation of $\mathscr{F}$ is:

$$\frac{\partial\mathscr{F}}{\partial w_{ik}} = \sum_{m=1}^{M}(w^T_{k,m} - a_{i,m}\frac{r_{i,m,k}}{w_{i,k}}) + \alpha\sum_{n=1}^{N}(h_{k,n} - v_{i,n}\frac{s_{m,n,k}}{w_{i,k}}) \tag{5.21}$$

and by setting the $\frac{\partial \mathcal{F}}{\partial w_{ik}} = 0$, the above equation can be written as follows:

$$w_{i,k} = \frac{\sum_{m=1}^{M} a_{i,m} r_{i,m,k} + \alpha \sum_{n=1}^{N} v_{i,n} s_{m,n,k}}{\sum_{m=1}^{M} w_{k,m}^{T} + \alpha \sum_{n=1}^{N} h_{k,n}} \tag{5.22}$$

Thus, for each iteration, the multiplicative update rule for every element in latent matrix W is:

$$\hat{w}_{i,k} = w_{i,k} \frac{(\sum_{m=1}^{M} \frac{a_{i,m}}{(ww^T)_{i,m}} w_{k,m}^{T} + \alpha \sum_{n=1}^{N} \frac{v_{i,n}}{(wh)_{i,n}} h_{k,n})}{\sum_{m=1}^{M} w_{k,m}^{T} + \alpha \sum_{n=1}^{N} h_{k,n}} \tag{5.23}$$

where $\hat{w}_{i,k}$ is the new value for the element matrix $w_{i,k}$ after each iteration process.

Using a similar procedure, the update rule for the latent matrix H to minimize $\mathcal{T}_{NMijF}$ can be found in the equation below.

$$\hat{h}_{k,n} = h_{k,n} \frac{(\sum_{m=1}^{M} \frac{w_{k,m}}{(wh)_{k,m}} w_{k,m})}{\sum_{m=1}^{M} w_{k,m}} \tag{5.24}$$

The process of *NMijF* for topic derivation is described in Algorithm 3 below.

Algorithm 3 Topic derivation using *NMijF*

INPUT: number of topics k
OUTPUT: tweet-topic matrix $W \in \mathbb{R}^{m \times k}$ and term-topic matrix $H \in \mathbb{R}^{k \times n}$

1: get tweet-to-tweet matrix $A \in \mathbb{R}^{m \times m}$
2: get tweet-to-term matrix $V \in \mathbb{R}^{m \times n}$
3: initialize W and H
4: **repeat**
5: $W \leftarrow f(A, W, W^T)$
6: $H \leftarrow f(V, W, H)$
7: **until** min($\mathcal{T}_{NMijF}$)
8: **return** W,H

The whole process of topic derivation in Twitter using NMijF approach is illustrated in Figure 5.8. Firstly, the content from the collection of tweets are prepro-

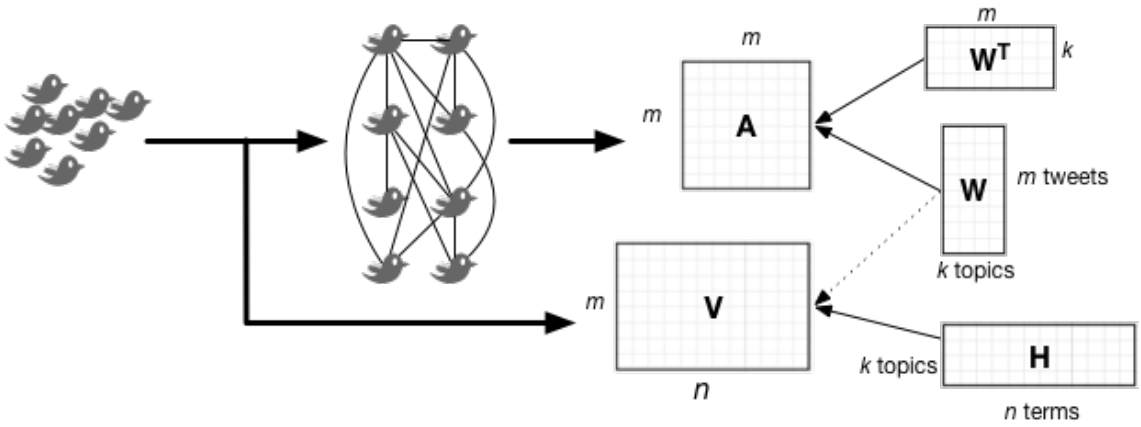


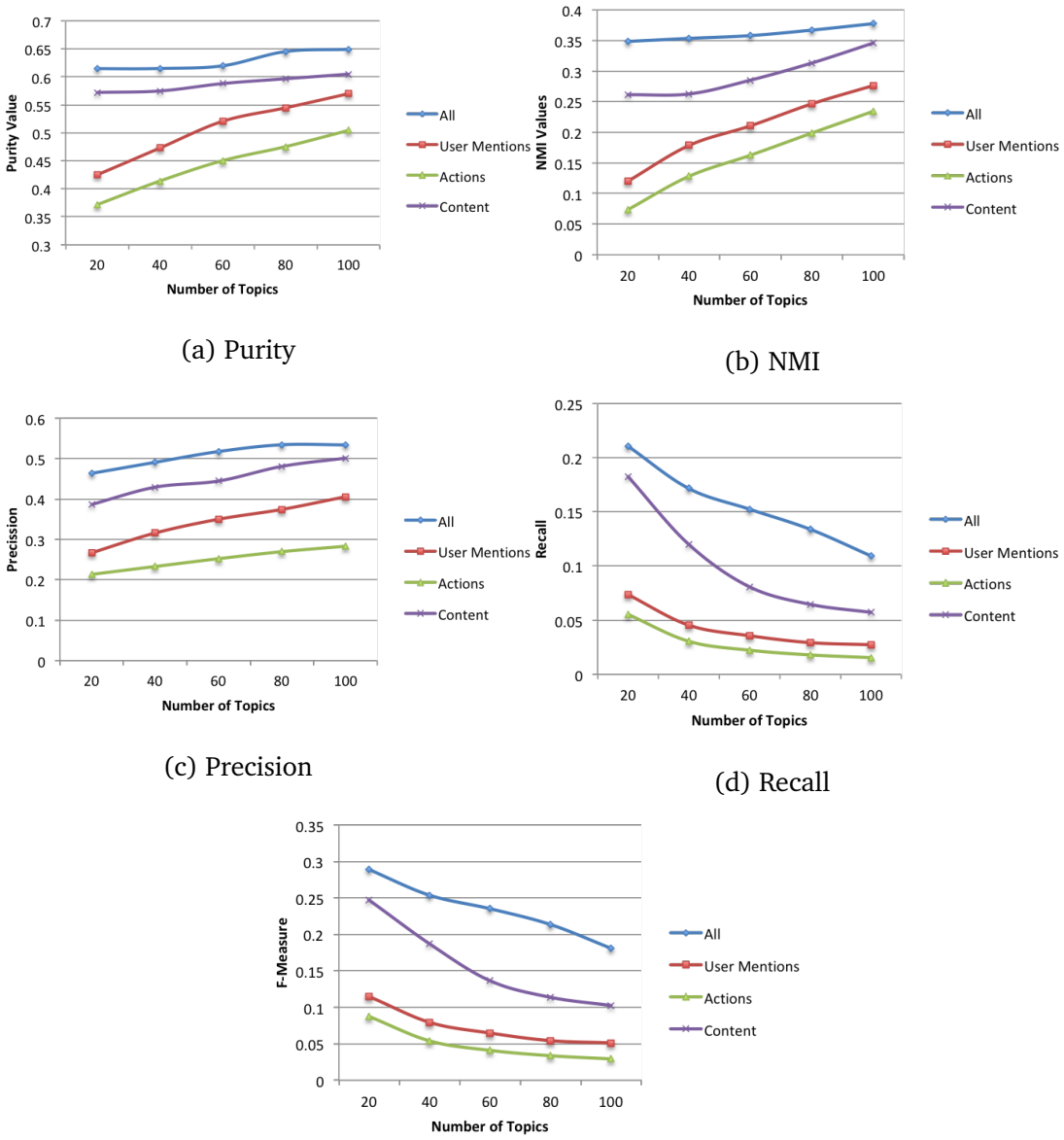
Figure 5.8: Topic derivation process

cessed. Secondly, we compute two important matrices for the topic derivation: the tweet-to-tweet square matrix (A) and the tweet-to-term matrix (V). And finally, we apply the NMijF process to learn the tweet-topic matrix $W \in \mathbb{R}^{m \times k}$ and topic-term matrix $Y \in \mathbb{R}^{k \times n}$.

The tweet-topic matrix W represents the relationship between each tweet in the collection and every topic in k topics. To identify the most important topic of a tweet, we choose the topic with the highest value. To find the keywords representation for each topic, we choose the top-n terms from the topic-term matrix Y.

5.4 Evaluation

In this section, we provide the details of our experiments to see the performance of our proposed method. We again conducted the evaluation of all methods with the labeled datasets discussed in Chapter 3. LDA-based approaches described in the previous chapter are used as baseline methods. Experimental results presented in Chapter 4 have shown that intLDA and Plink-LDA outperform the other methods, including the straight LDA, TNMF and the original NMF. We also compare our method against eLDA on the tweetSanders dataset. eLDA has the highest purity value with tweetSanders, but fails to perform on the other datasets.



(a) Purity

(b) NMI

(c) Precision

(d) Recall

(e) F-Measure

Figure 5.9: Evaluation of the impact of each relationship feature in the tweetMarch dataset

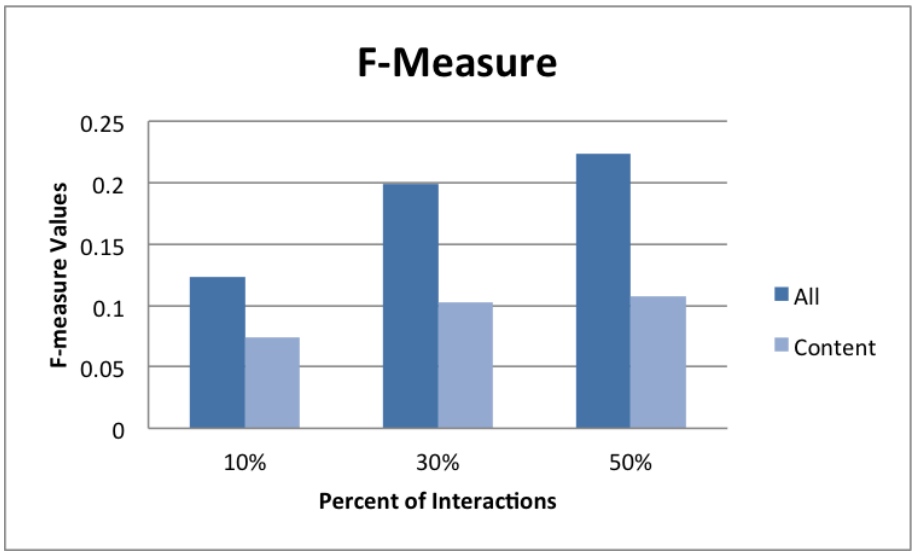


Figure 5.10: Impact of interactions availability on three different subsets of tweetMarch evaluation set

Each experiment executes the topic derivation methods for a particular number of expected topics based on the labeled datasets. For every k and every method, we run the algorithms over both datasets 30 times, and take the average value of each evaluation metric for comparison. We use scaling parameter $\alpha = 0.1$ when executing the inter-joint factorization. It ensures that the sparse matrix V does not heavily penalize both the shared tweet-topic matrix W and topic-term matrix Y.

Impact of interaction features

To see the impact of each individual component of the relationship between tweets in topic derivation, we discuss various configurations and evaluation metrics using the tweetMarch dataset in Figure 5.9. From each subfigure, we can see that the combination of all components provides the best results for all evaluations. All metrics show a similar trend, with content based similarity as the second best, followed by the user mentions and replies-retweets based interactions. This trend matches with the number of connections between tweets from each component as described in Section 5.2.1. As there are very high percentages of content based relationship

amongst the tweets, it is not surprising that this component produces the highest tweet clusters accuracy in comparison with other individual components. However, when all three components are combined, there are significant improvements in all evaluation metrics.

We use several subsets of the tweetMarch dataset to further see the impact of social interactions on the quality of derived topics. Each subset has different proportions of replies and retweets. The first subset has 10% replies and retweets, the second subset 30%, and the third 50%. In this experiment, we use our proposed NMijF method once with all the components and once with only the content similarity. The results are shown in Figure 5.10. We see that, in all cases, incorporating all components outperforms using only the content.

Comparison with baseline methods

Figure 5.11, 5.13, and 5.11 show the purity evaluation results for all methods on the TREC2014, tweetSanders and twetMarch datasets, respectively. In this evaluation, the numbers of topics specified in each algorithm are matched to the numbers of available topics in the labeled evaluation sets. For all the datasets, our proposed NMijF method is able to outperform intLDA and Plink-LDA with about 5 - 20% improvements. NMijF with the joint probability based tweet relationships is able to present around 30 - 70% improvements over the original NMF and other methods that based solely on content exploitation.

The density of the tweet-to-tweet relationships for the TREC2014 dataset is only 2.695%, far less than the tweetSanders and tweetMarch dataset with 23.887% and 12.842%, respectively. The sparse tweet-to-tweet relationships matrix makes it harder for all methods to capture the topics from the collection of tweets. As shown in Figure 5.11, our NMijF method still able to result in more than 5% improvement over our previous intLDA. In the evaluation with the tweetMarch dataset (Figure 5.12), NMiJF gets more than 10% improvement.

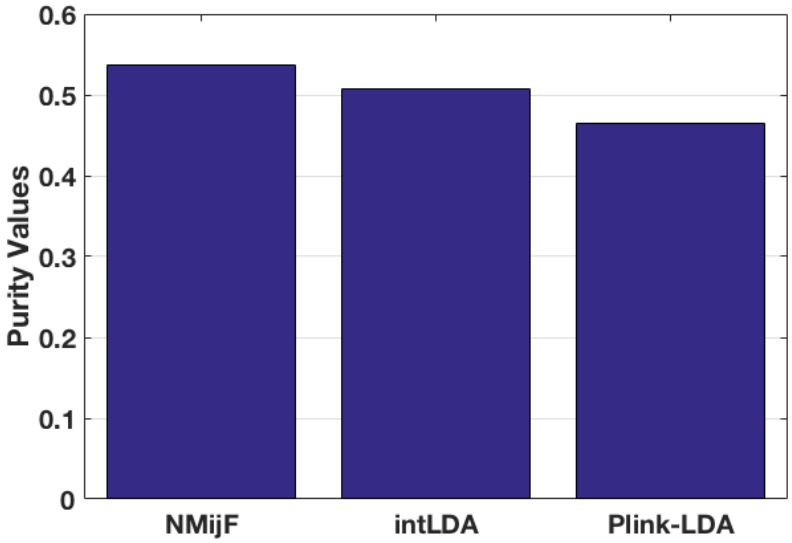


Figure 5.11: Purity results on TREC2014, $k = 55$

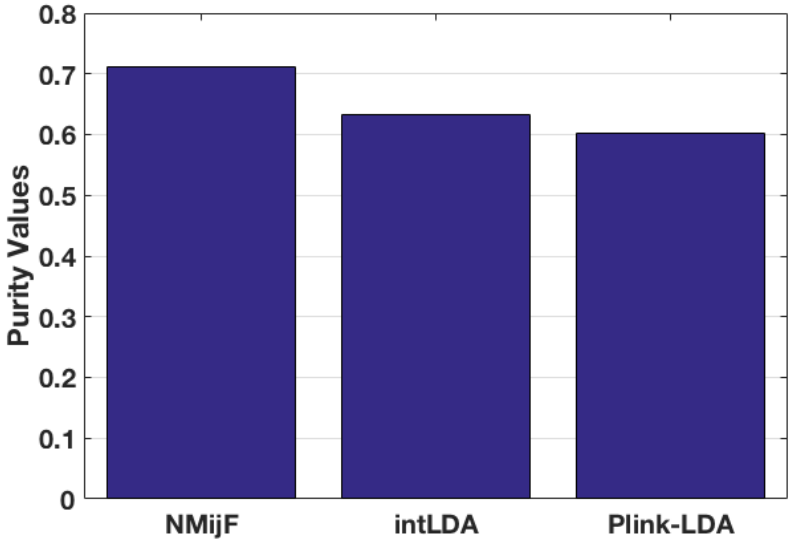


Figure 5.12: Purity results on tweetMarch, $k = 6$

The purity results with tweetSanders are shown in Figure 5.13. The NMijF is again superior to other methods, with a 10% improvement over eLDA and around 20% over intLDA.

The improvement over our previous LDA-based method falls in the range of

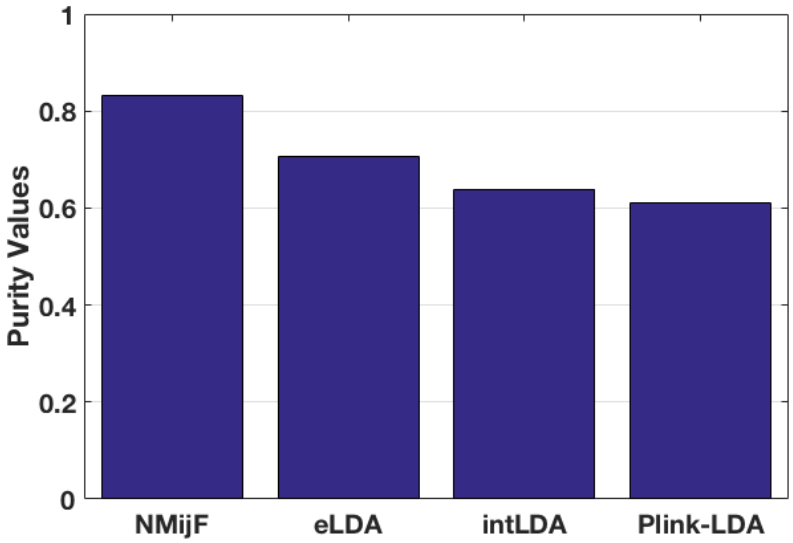


Figure 5.13: Purity results on tweetSanders, $k = 4$

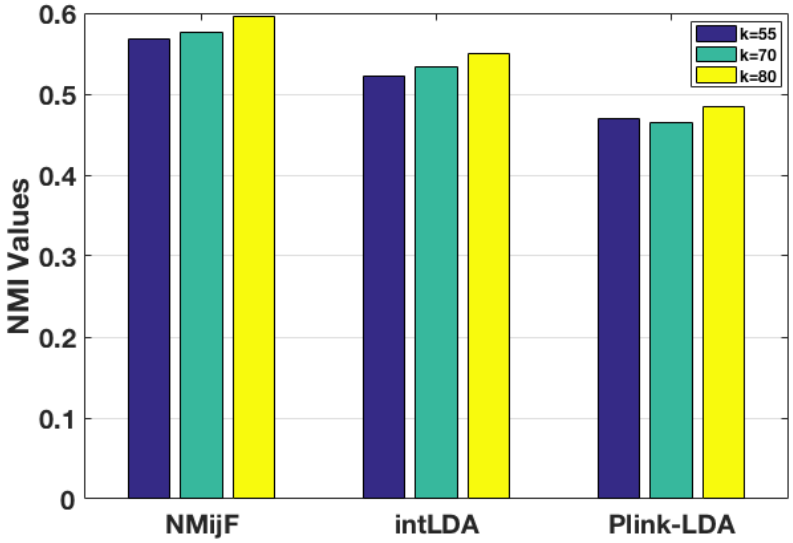


Figure 5.14: NMI results on TREC2014

5-20%. We perform a paired-sample *t-test* method to find out the confidence value, and the results indicate that the improvement is statistically significant at the level of $p < 0.01$.

The evaluations using the NMI metric are presented in Figure 5.14, 5.15, and

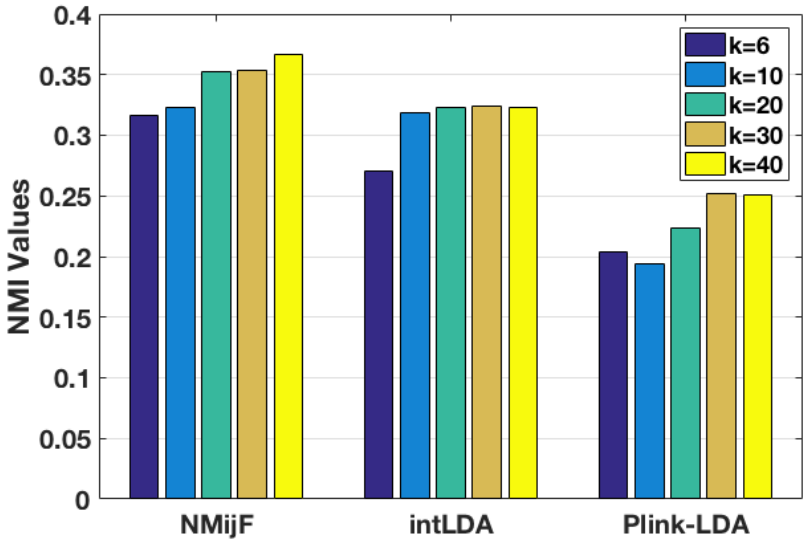


Figure 5.15: NMI results on tweetMarch

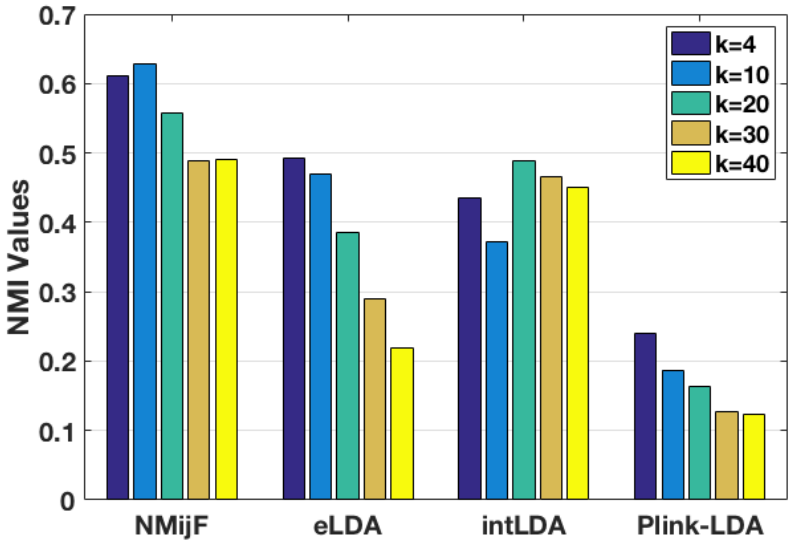


Figure 5.16: NMI results on tweetSanders

5.16. With this metric, our proposed NMijF method again has the best performance for all different setups. Overall, the NMijF brings in around 10-30% improvement over the other methods. The NMI results for different k also shows that NMijF is better at handling the trade-off between the number of topics and the quality of the

Table 5.2: *Precision (p), Recall (r) and F-Measure (F-M)* for three datasets

Method	p	r	F-M
NMijF	**0.370**	**0.344**	**0.356**
intLDA	0.319	0.333	0.326
Plink-LDA	0.305	0.295	0.0.300

(a) TREC2014, $k = 55$

Method	p	r	F-M
NMijF	**0.581**	0.329	**0.420**
intLDA	0.476	**0.341**	0.396
Plink-LDA	0.433	0.248	0.316

(b) tweetMarch, $k = 6$

Method	p	r	F-M
NMijF	**0.767**	**0.876**	**0.818**
eLDA	0.580	0.707	0.637
intLDA	0.438	0.764	0.557
Plink-LDA	0.420	0.475	0.446

(c) tweetSanders, $k = 4$

topic derivation.

The F-Measure results are shown in Table 5.2. NMijF again outperforms other methods. NMijF consistently has the best F-Measure values for all the datasets. Our proposed method not only brings in high precision and recall values in comparison with other methods, but also has the best harmonic mean between these two metrics.

When the density of tweet relationships is low, as in the TREC2014 dataset, our NMijF is still able to result in more than 10% improvement over the intLDA method. For the tweetSanders dataset, the improvement is much more significant. NMijF has 0.767 for precision and 0.876 for recall. The F-Measure value is 0.818. This is much higher than the eLDA as the second best with 0.580, 070 and 0.637 for precision, recall and F-Measure, respectively. By giving much more accuracy in measuring the relationships between tweets, the NMijF is not as sensitive to the density of the tweet relationship matrix compared to our previous LDA-based methods.

The computational complexity of the multiplicative update rules expressed in equation 5.18 for every iteration is $\mathcal{O}(mnk)$ where m is the number of tweets, n is the number of unique terms in the collection of tweets, and k is the number of expected topics. To achieve the results presented above, our proposed method requires only 30 iterations, while the LDA-based methods need at least 50 iterations with similar complexity.

The result of the topic derivation includes not only the identified topics but also the keywords that represent these topics. Table 5.3, 5.4, and 5.5 show the top-5 keywords for several examples of identified topics in TREC2014, tweetMarch, and tweetSanders respectively. In NMijF, the keywords for every topic are retrieved from the topic-term latent matrix (H) learned from the inter-joint factorization process. The topic-term values for every k are sorted to get the top-5 keywords to represent the topic. In Table 5.3, 5.4, and 5.5, keywords tightly correlated with topics are shown in italic font. A topic is more readable if there are more correlated keywords as its representation [83]. In the experiments, NMijF gives more correlated keywords to represent topics than other methods, in particular when the density of the term-to-term matrix is very low. The proposed method NMijF results in better performance for both the tweet clustering based on topics and the keywords identification of topics over baseline methods.

5.5 Discussion

The incorporation of the relationships between tweets has shown to effectively improve the quality of topic derivation in Twitter. However, each component might have a different effect on the accuracy of the topical connectivity between tweets. Our analysis shows that tweets linked by tweet interactions and content similarity have different probabilities of being about the same topic. Tweets linked by replies and retweets are almost always about the same topics; tweets linked by user men-

Table 5.3: Top-5 topic-term for some topics discovered on the *TREC2014* dataset. Words in italic have high connectivity with the topics, stroked words has low connectivity with the topics

Cluster/ Topic Number	**Topic Labels**	**Representative words**		
		NMijF	***intLDA***	***Plink-LDA***
MB171	Ron Weasley birthday	*ron* *weasley* *birthday* harry potter	*ron* ~~book~~ *weasley* ~~watch~~ ~~new~~	*ron* *weasley* potter ~~effect~~ harry
MB172	Merging of US Air and American	*american* *air* *airline* *merger* *airways*	*american* *airways* world *air* *merger*	*airways* *american* *deal* world *air*
MB173	Muscle pain from statins	*pain* *muscle* *arms* fat head	*pain* effect *care* ~~book~~ *statins*	*statins* *pain* ~~winter~~ fat head
MB174	Hubble oldest star	*telescope* *oldest* *star* *hubble* ~~weather~~	*hubble* *telescope* ~~weather~~ ~~storm~~ *oldest*	*hubble* *star* ~~big~~ ~~open~~ *oldest*
MB175	Commentary on naming storm Nemo	*#nemo* *nemo* *snow* *storm* *winter*	*storm* *winter* *nemo* *name* ~~world~~	*name* *winter* *storm* ~~watch~~ ~~bad~~

tions and content similarity have a reasonable chance to be about the same topics, and their number is much larger than the number of tweets linked by reply and retweet. To integrate the effect of these components, we develop a joint probability model for the tweet-to-tweet relationships.

To incorporate the joint probability model, we have designed a Non-negative

Table 5.4: Top-5 topic-term for some topics discovered on the *tweetMarch* dataset.

Topic Labels	**Representative words**		
	NMijF	***intLDA***	***Plink-LDA***
Travel/transport	*train*	*accident*	*#traffic*
	#traffic	*road*	*road*
	accident	*#traffic*	~~time~~
	driver	*train*	*train*
	road	closed	closed
Politics	*policy*	*liberal*	*polict*
	obama	*obama*	*liberal*
	government	people	*obama*
	politic	~~chance~~	~~big~~
	~~process~~	*policy*	~~process~~
Food/Beverages	*tea*	order	*tea*
	coffee	*tea*	~~sleep~~
	drink	cold	*coffee*
	order	~~talk~~	stop
	sweet	*brown*	~~talk~~

Matrix inter-joint Factorization (NMijF) approach. NMijF factorizes both the tweet-to-tweet A and tweet-to-term V matrices into tweet-topic W and topic-term H latent matrices in a unified iterative update process and under one cost function. We applied a biased update rule for the tweet-topic matrix W to take the advantage of the higher density tweet-to-tweet matrix A, but reduce the penalty due to the sparsity of the tweet-to-term matrix V.

Our evaluation results demonstrate that the joint probability model for the tweet relationships has a positive impact on the quality of topic derivation. The proposed NMijF approach consistently outperforms our previous LDA-based methods on all evaluation metrics. The experimental results reveal that, the more accurate integration of content similarity, mentions, and replies-retweets when measuring the relationship between tweets provides a significant improvement of the topic derivation quality in different situations.

Table 5.5: Top-5 topic-term for some topics discovered from the *tweetSanders* dataset.

Topic Labels	**Representative words**			
	NMijF	***eLDA***	***intLDA***	***Plink-LDA***
Google	*#google*	*#google*	*#google*	*#google*
	#android	*#android*	*#android*	*#android*
	google	*sandwich*	*android*	*nexus*
	nexus	*nexus*	*sandwich*	*#cream*
	sandwich	*cream*	*nexus*	apps
Apple	*#apple*	*iphone*	*iphone*	*#apple*
	store	*#apple*	apps	*iphone*
	siri	store	*siri*	~~android~~
	iphone	*siri*	~~#twitter~~	*iphone*
	apps	apps	new	apps

Chapter 6

Time-sensitive Topic Derivation

6.1 Introduction

In the previous chapter, we proposed a topic derivation approach that exploits both interaction features and content similarity using the joint probability model. Deriving topics from Twitter is also problematic due to its highly dynamic environment, where topics rapidly change over time. In this chapter, we propose a method that takes time into account in the topic derivation process and see if it further improves the results.

To address the dynamic aspect of Twitter, some approaches have exploited temporal aspect of the tweet content or associated hashtags, e.g., [107], [14], and [112]. To the best of our knowledge, the temporal aspect of the posts' *interactions* has not been explored for topic derivation in a collection of tweets.

While taking conversations into account as discussed in Chapter 4 and 5 can improve topic derivation quality, conversations typically are time-sensitive. For example, two tweets with the mentions of the same users nearly at the same time are more likely to be about the same topic than two posts with mentions of the same users after a long time interval. Therefore, incorporating the temporal aspect when looking at the interactions may further improve the quality of topic derivation. This chapter is summarized as follows:

- We discuss the relationships between topics, interaction features (*user men-*

tions, *replies*, and *retweets*) and time using a dataset obtained by collecting tweets over a month. We found that the *mention* is time-sensitive with respect to the topic assignation.

- We model the time sensitivity of *mentions* as an exponential decay according to the time difference of two tweets with the same mentions. The decay parameter is based on the performed empirical study. This time sensitivity model is then incorporated in the tweet relationship model in order to influence the matrix inter-joint factorization for topic derivation.

- We conducted a comprehensive set of experiments to evaluate the proposed new model with our Twitter datasets, using the same metrics as in previous chapters. The results show that the new time-sensitive method results in a significant improvement of the quality of topic derivation comparing with our previously proposed approach in Chapter 5.

- We also performed the evaluation of our method by scrutinizing tweets grouped in a series of time periods. The results show that our proposed method can cope with the dynamic Twitter stream better than all baseline methods.

This chapter is organized as follows. Section 6.2 provides a motivating example and discusses the role of time in the topical connectivity. Section 6.3 analyses the different temporal sensitivities of *mentions*, *replies*, and *retweets*. Section 6.4 explains a method to measure the relationships between tweets by incorporating the temporal aspect. Section 6.5 reports on a series of experiments. We conclude in Section 6.6.

Table 6.1: Tweet examples

Id.	User	Timestamp	Tweets
t_1	*user1*	12/01/2015, 5:45 PM	I am having a pizza for dinner as I went to Dominos to go pick one up on my way home.
t_2	*user2*	12/01/2015, 5:50 PM	*@user1* Favorite topping?
t_3	*user3*	12/01/2015, 6:32 PM	*RT @user1:* I am having a pizza for dinner as I went to Dominos to go pick one up on my way home.
t_4	*user4*	12/01/2015, 6:39 PM	Have you started your own label *@user5*? Just noticed this on my #polo shirt #giddyup #youcantpolosolo
t_5	*user6*	13/01/2015, 11:39 AM	More pics from the Portarlington Mussel Festival. *@user5*
t_6	*user7*	13/01/2015, 11:58 AM	Hi *@user5*, the event was a great success. Congratulations

6.2 Motivating Example

Table 6.1 shows some tweet examples that illustrate typical interactions between users on a time. Seven users are involved within these 6 tweets. Figure 6.1 provides a graphical illustration of the relationships between the tweets shown in Table 6.1, with all tweets in the collection grouped into three-time windows based on their timestamps. The first time window is the tweets that were posted between 5.30 PM to 6.00 PM on 12 January 2015. t_1 and t_2 are in this time window. The second time window is between 6.30 PM to 7.00 PM on 12 January 2015. t_3 and t_4 are in this time window. The last time window is between 11.30 AM and 12.00 PM on 13 January 2015. t_5 and t_6 are in this time window.

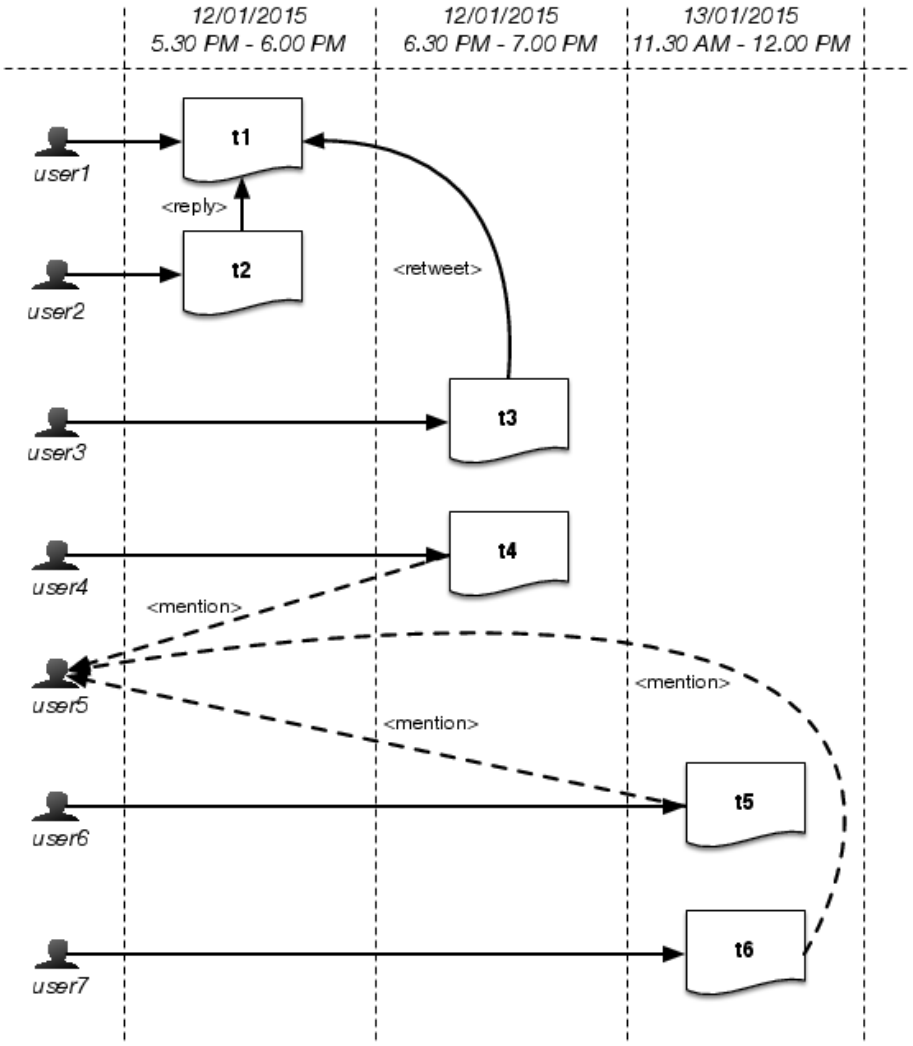


Figure 6.1: Relationships between tweets based on interactions

In Figure 6.1, tweets t_1 and t_2 are related to each other since t_2 is a reply of tweet t_1. t_3 is related to t_1 as its retweet, although they are in different time windows. t_1, t_2, and t_3 are talking about the same topic: '*pizza*'. We can see that both replies and retweets are likely to be on the same topic as the original post.

t_4, t_5, and t_6 are connected to each other due to the fact that they mention the same user (*@user5*). But t_4 talks about a completely different topic than t_5 and t_6. Tweet t_4 talks about the 'shirt label', while t_5 and t_6 talk about 'Portarlington Mussel Festival'. If we look at the tweets' timestamp in Table 6.1, we find that t_5 and t_6

have little time difference if compared to t_4 posted a day before. These examples illustrate that two tweets which mention the same user are likely to be on the same topic *if* they occur around the same time. Time thus plays an important role when attempting to link tweets that mention the same people. In the next section, we investigate how time impacts on the interactions when comes to grouping tweets.

Table 6.2: Top 15 Twitter users in Australia and all related tweets (i.e., tweets that involve these top 15 Twitter users, either by mentioning them, replying to them or retweeting their posts) between 12 January 2015 and 12 February 2015

Username	related tweets	users involved	followers
@CodySimpson	**388,970**	**69,246**	**7,384,541**
@5SOS	**2,068,129**	**258,292**	**6,619,112**
@Calumn5SOS	**2,330,628**	**340,686**	**5,154,177**
@luke_brooks	**583,999**	**56,908**	**2,242,597**
@example	**8,464**	**5,208**	**2,107,484**
@KyrieIrving	**46,896**	**33,311**	**2,064,137**
@BrooksBeau	**819,423**	**95,879**	**1,932,857**
@jascurtissmith	**3,318**	**1,368**	**1,831,271**
@MrKRudd	**2,249**	**1,553**	**1,524,455**
@allisimpson	**88,504**	**20,107**	**1,418,732**
@claireholt	**5,413**	**2,497**	**1,299,287**
@MClarke23	**2,442**	**1,525**	**1,293,651**
@DarrynLyons	**1,154**	**390**	**1,143,222**
@hillsongunited	**3,456**	**2,455**	**969,020**
@imacelebrity	**1,675**	**1,340**	**894,187**
@JordanJansen	**10,774**	**2,512**	**759,192**

6.3 Time in Tweet Interactions: An Analysis

In this section, we analyze tweets in a Twitter dataset to see how time affects the topic similarity between tweets and their interactions. We obtained the dataset as follows. Using the Twitter's streaming API[1], we retrieved all tweets from the top 15

[1]https://dev.twitter.com/streaming/overview

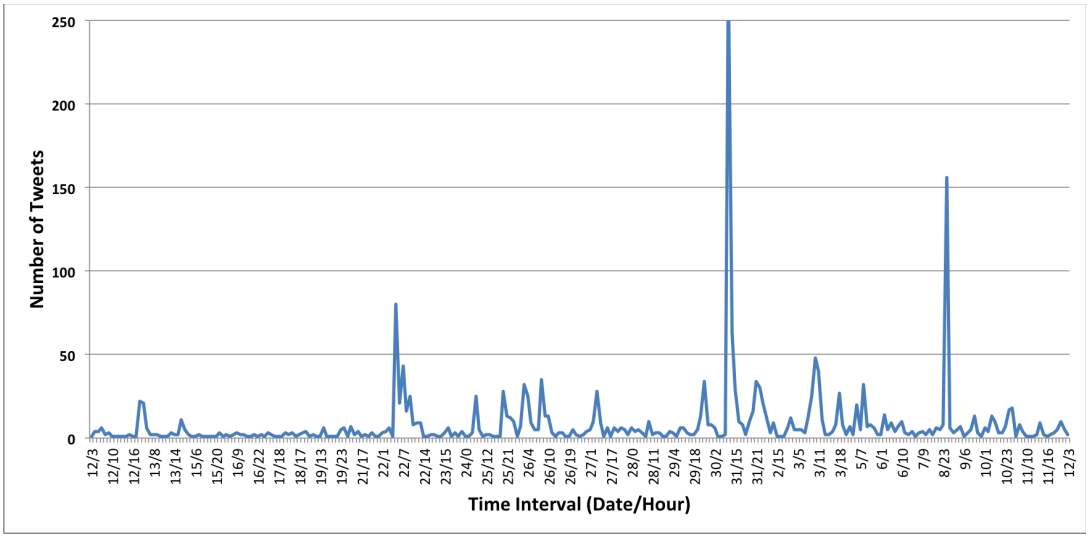


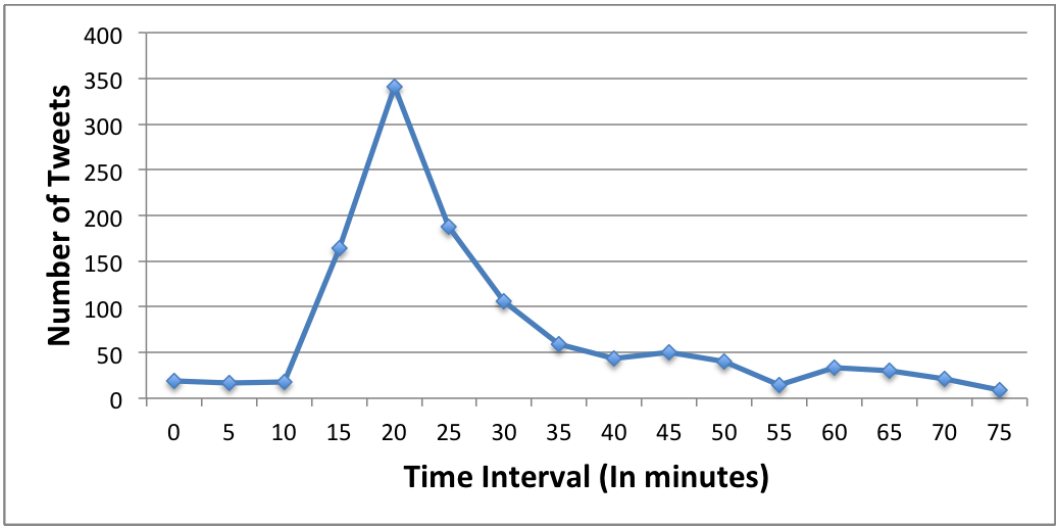
Figure 6.2: Tweets mentioning user @MrKRudd between 12 January 2015, 3AM to 12 February 2015, 3AM. Each interval in a 3 hour interval.

Twitter users in Australia[2] in January 2015 and all the tweets that mention those users (including replies and retweets) during the period of 12 January 2015 to 12 February 2015. The resulting dataset consists of more than 6 million tweets, with about 800 thousand users. The details of the dataset are shown in Table 6.2.

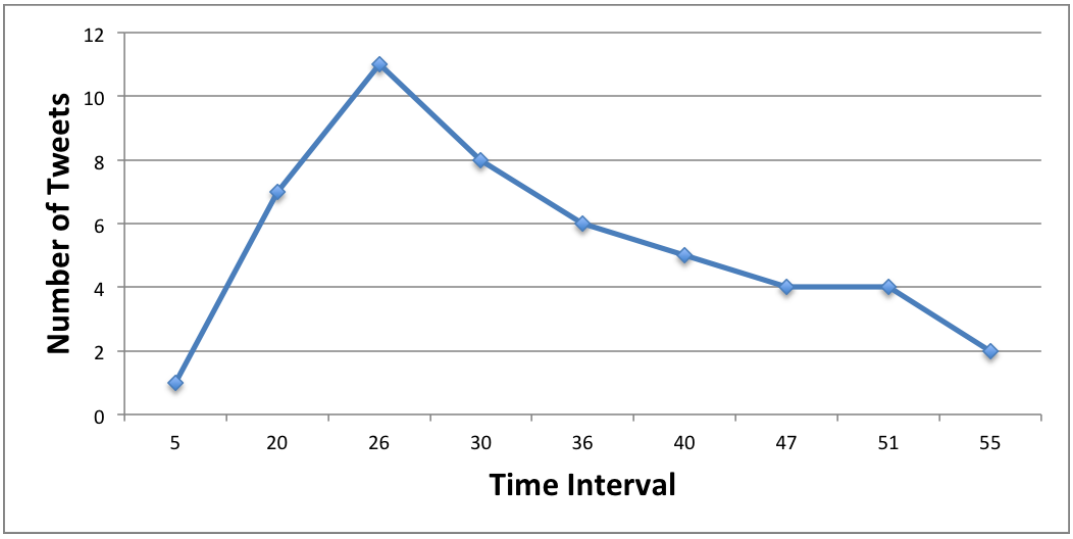
Our investigation starts with an analysis of individual user mentions at a different level of time granularity to see how mentions are distributed over time. We look at the topics in the dataset to see if there is a relationship between mentions and topics. We find that, for all users, when the number of mentions of a specific user rises at a particular time, most of the tweets published at that time are on the same topic.

As an example, Figure 6.2 shows the distributions of the tweets that mention *@MrKRudd* in a 3 hour time interval. There are several fluctuations within different time intervals. Each peak in Figure 6.2 (an indication of a sharp increase in the number of tweets mentioning *@MrKRudd*) is strongly related to a particular topic. For example, on 22 January 2015 at 7am (22/7), most of the tweets mentioning

[2]https://followerwonk.com/bio/?q_type=all&l=Australia, accessed January 11, 2015, ordered by number of followers



(a) @CodySimpson



(b) @MClarke23

Figure 6.3: Tweet distributions of tweets mentioning (a) *@CodySimpson* and (b) *@MClarke23* with 5 minutes time intervals within 1 hour

@MrKRudd were talking about the "*plain packaging act*". The tweets at 3 PM on 31 January 2015 were about "*Queensland votes*", and the tweets at 11 PM on 08 February 2015 were about "*the end of Kevin Rudd's leadership in February 2012*".

We see from the figure that the number of tweets with the same mention reaches a peak and then fades away (decay). Figure 6.3 shows the subset of the distributions of the tweets that mention (a) *@CodySimpson* and (b) *@MClarke23* with 5-minute intervals. The specific distributions are different, reaching their peaks and decaying at different rates. What they have in common, however, is that each peak indicates a specific topic. The peak in Figure 6.3a is related to the topic: "*Cody's birthday*"; and the peak in Figure 6.3b is related to the topic: "*the absence of Michael Clarke on treatment issue*".

We perform a statistical analysis on all the variations of the tweet distributions, using a 5 minutes interval. We sum up the number of tweets from all users by choosing the subset of the tweet distributions starting from the closest lowest point before a peak and ending at the lowest point after the peak. Figure 6.4 shows this sum. Most of the mentions related to a particular topic reach a peak within about 15 minutes and then gradually fade away. A half-life exponential decay function is adopted to model the process of fading away. The exponential function has a parameter to control its decaying. This parameter is how long the mention frequency decays from its peak to the peak's half value. It can be expressed as:

$$a = i_{t_{max}/2} - i_{t_{max}} \tag{6.1}$$

where $i_{t_{max}}$ is the time when the tweet mention distribution reaches its peak, and $i_{t_{max}/2}$ is the time when the tweet mention distribution reaches half of the peak value after the peak. In Figure 6.4, the number of tweets in the highest point (t_{max}) is 367,368, and it is reached after 15 minutes ($i_{t_{max}}$). Then, $i_{t_{max}/2}$ is calculated as the time to reach 183,684 after the peak, which is 37 minutes. So, a for Figure 6.4 will

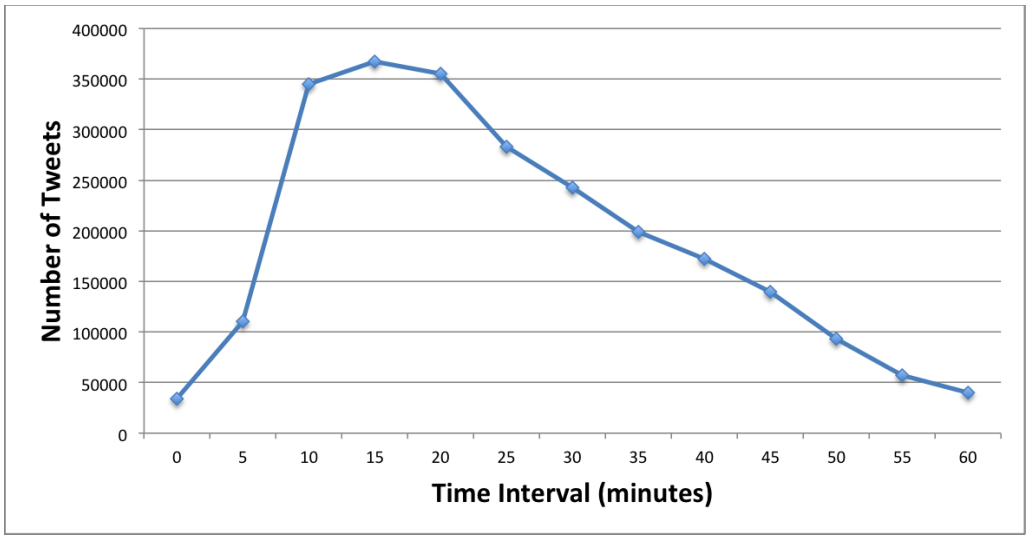


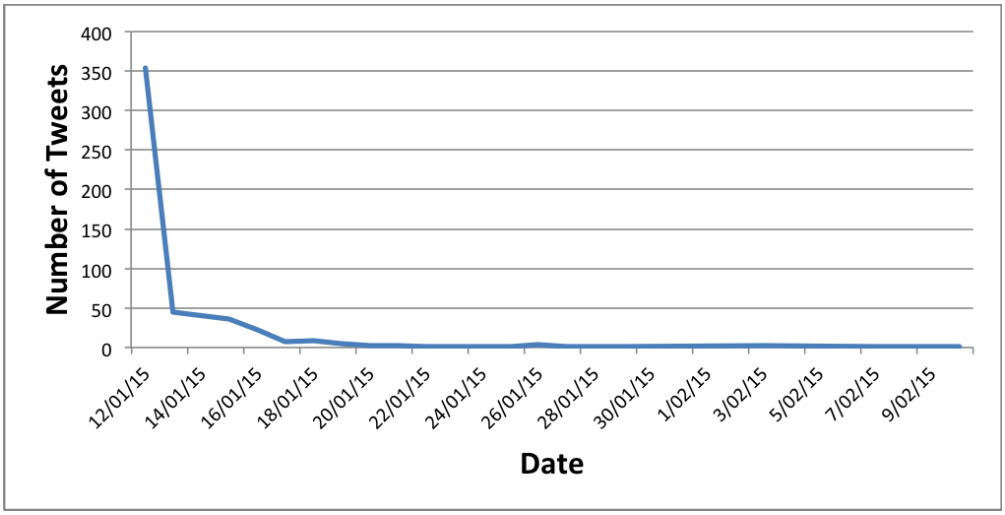
Figure 6.4: The sum of all fluctuations in all tweet mention distributions with 5-minute time intervals

be 22 minutes (1,320 seconds). This a will be used in the exponential function that models the temporal aspect in the mention behavior in Twitter.

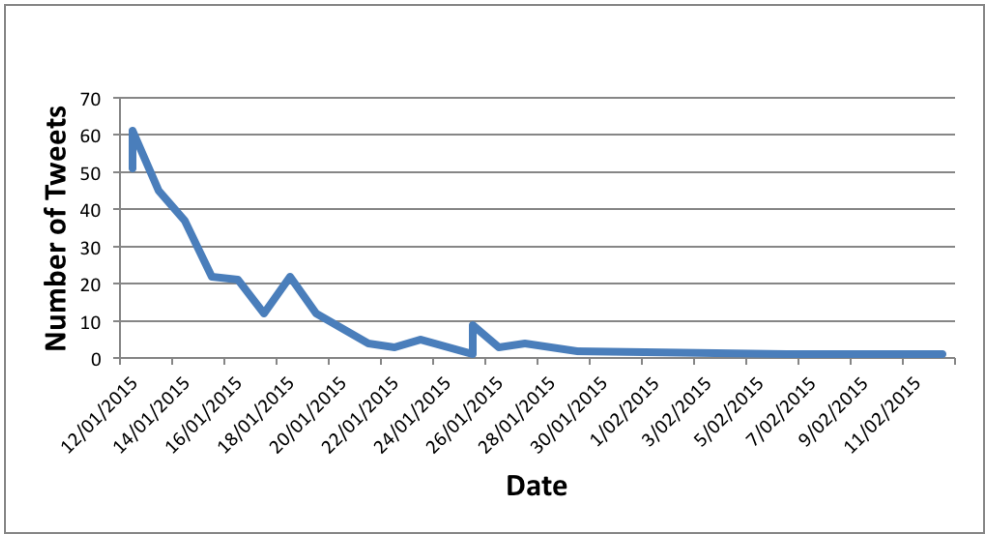
In contrast to the mention behavior, replies or retweets have no clear temporal relationships with the original tweet in terms of topic similarity. The analysis of the dataset shows that a *retweet* or a *reply* could occur long time after the original tweet and still be on the same topic.

For example, Figure 6.5a shows the tweet distributions of a retweet to a tweet by *@CodySimpson*: ("*It's the 11th back home in Aus. I m officially 18.*"). The tweet was retweeted for 494 times in total, with 354 retweets on the first day, 22 on the third day, and the remaining scattered over time. The tweet distributions of a retweet to tweet by *@luke_brooks* shown in Figure 6.5b shows similar trends. The highest number of retweets happened in the first day with around 112 tweets, followed by 61 tweets on the second day, and 45 tweets on the third day. The original tweet is still being retweeted several times during this 1 month period. Irrespective of the time elapsed, the retweets are still on the same topic.

Figure 6.6 shows the tweet distribution of the replies to a tweet by @5SOS



(a) RT of @CodySimpson



(b) RT to @luke_brooks

Figure 6.5: Tweet distributions of retweets to a tweet by (a) *@CodySimpson* and (b) *@luke_brooks* within a 1 month period

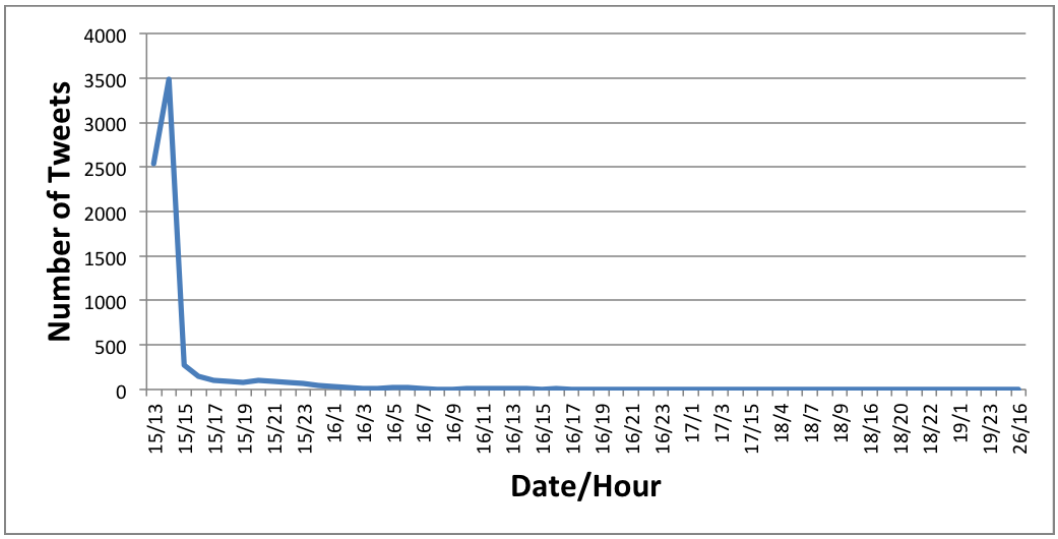


Figure 6.6: Tweet distribution of replies to a tweet by *@5SOS* within a 1 month period

("*Getting lots and lots of ideas for songs! Ready to write a new record!!*"). The total number of replies was 7414 tweets, with a peak on the first day but continuing the following day (291 replies). The analysis of the reply and retweet behavior supports our previous statement that both replies and retweets can be classified as explicit interactions between two tweets which show the participation of users in a discussion about a particular topic.

6.4 Measuring Relationships between tweets

As described in Section 5.2.2 of Chapter 5, a tweet is represented by a tuple $t = \langle U_t, rtp_t, C_t, i_t \rangle$, where U_t is the union of the author and users mentioned in the tweets, rtp_t is the reply and retweet information, C_t is the set of terms contained in the tweet (including hashtags), and i_t is the timestamp of the posted tweet. The relationship between two tweets t_i and t_j is denoted as $R(t_i, t_j)$, where a zero value (0) of R means that there is no relation between them, and a higher value indicates the relationship is stronger. The relationship R is constructed based on the joint probability of three components: reply-retweet ($act(t_i, t_j)$), user mention

($m(t_i, t_j)$), and content similarity ($sim(t_i, t_j)$). It is expressed as:

$$R(t_i, t_j) = \begin{cases} 1, P(act(t_i, t_j)) > 0 \\ P(m(t_i, t_j) \cup sim(t_i, t_j)),\ otherwise \end{cases} \tag{6.2}$$

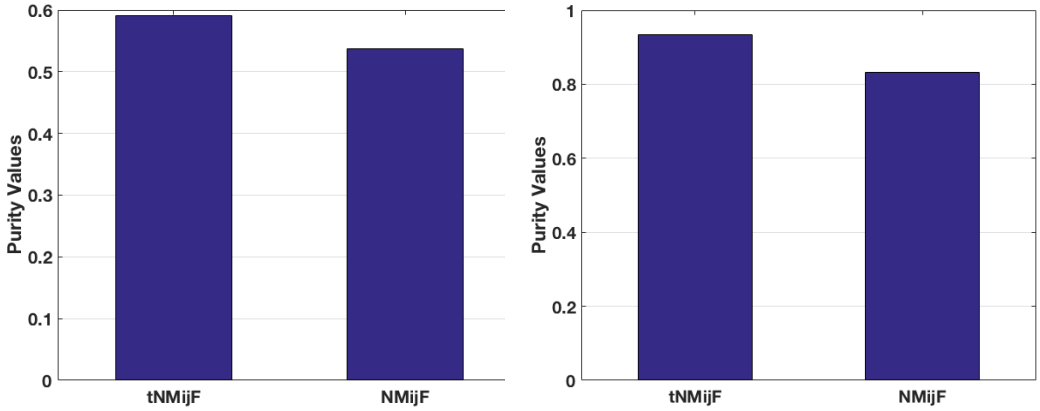
$$\begin{aligned} where \quad & P(m(t_i, t_j) \cup sim(t_i, t_j)) \\ & = P(m(t_i, t_j)) + P(sim(t_i, t_j)) - P((m(t_i, t_j) \cap sim(t_i, t_j))), \\ and \quad & P(m(t_i, t_j) \cap sim(t_i, t_j)) = P(m(t_i, t_j)) x P(sim(t_i, t_j)) \end{aligned} \tag{6.3}$$

As discussed in section 6.3, tweets that mention similar users within a particular period are more likely to share the same topic. So, we improve our definition of interactions based on user mentions by adding a temporal factor $f(i_{t_i} - it_j)$. Interactions based on user mentions $m(t_i, t_j)$ are modeled as the number of common mentioned people in tweets t_i and t_j divided by the total number of people involved in both tweets:

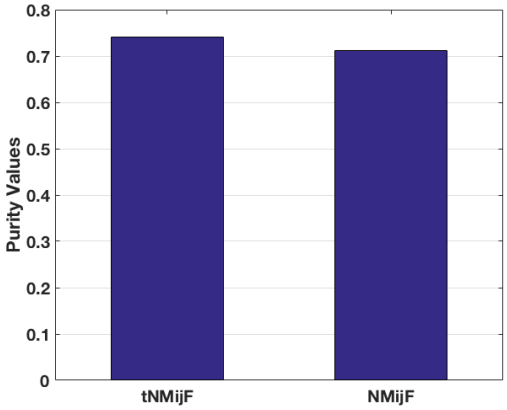
$$\begin{aligned} P(m(t_i, t_j)) &= \frac{|U_{t_i} \cap U_{t_j}|}{|U_{t_i} \cup U_{t_j}|} f(i_{t_i} - it_j) \\ where \quad & f(i_{t_i} - it_j) = e^{-\frac{1}{a}|i_{t_i} - i_{t_j}|}, \end{aligned} \tag{6.4}$$

$f(i_{t_i} - it_j)$ is an exponential function that models the temporal aspect of the mention behavior in Twitter. Its parameter, a, was defined in the previous section. $f(i_{t_i} - it_j)$ controls the decay rate of the temporal effect.

As before, the values of all the relationships amongst tweets form a tweet-to-tweet relationship matrix $A \in \mathbb{R}^{m \times m}$, where $a_{ij} = R(t_i, t_j)$. By incorporating a temporal factor in user mentions based interactions, we obtain a more accurate tweet-to-tweet relationship matrix. It is used to improve the topic derivation by jointly factorizing it with tweet-to-term matrix using the NMijF approach as discussed in the previous chapter.



(a) TREC2014, $k = 55$ (b) tweetSanders, $k = 4$



(c) tweetMarch, $k = 6$

Figure 6.7: Purity evaluation results for the three datasets

6.5 Experiments

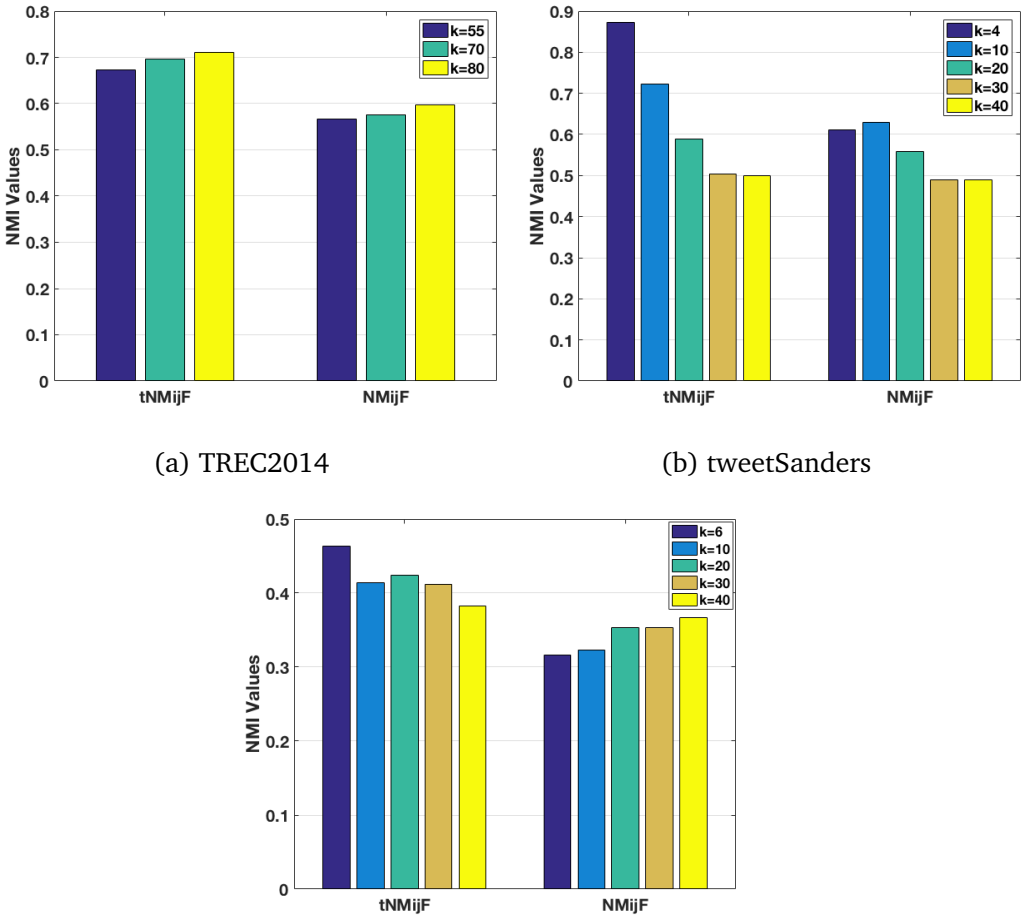
We now present our experiments with this new time-sensitive model. We compare our new model with our previous NMijF version which does not include the time factor. The one with the new model is denoted as *tNMijF*. We also evaluate the performance of our method on dealing with the dynamic nature of the tweet stream and the varying nature of topics in the stream.

6.5.1 Results and Discussion

As before, we run the new method and baseline methods for 30 times over all of the datasets and tune all the parameters for the best performance. Similar to our previous experiments, the scaling parameter α is set to 0.1 to ensure that the sparsity of tweet-to-term matrix V does not heavily penalize the topic-tweet matrix W and still gives good results when factorizing the topic-term matrix H.

Figure 6.7 shows the evaluation results using the purity metric for the three datasets. For TREC2014, our proposed method tNMijF results in about 3% improvement over our previous not-time-sensitive model. It is worth noting that, in this dataset, the number of tweets that have any interactions is very small, and mentions make up only around 0.02% of the relationships between tweets. There are no retweets at all in this dataset. The improvement we obtain suggests that our new time-sensitive method performs better even with few interaction features. When the percentage of tweets using mentions is higher, the improvement is also higher. In the tweetMarch dataset, for example, the interactions based on user mentions is about 0.24% of all linked tweets in the tweet-to-tweet matrix, and improvement in purity is around 10%. Both tNMijF and NMijF are able to obtain a very high purity value in the tweetSanders dataset, with the purity value 0.934 and 0.832, respectively. This is due to the high density of the tweet-to-tweet matrix in tweetSanders.

For the NMI evaluation, tNMijF results in roughly a 5-10% improvement com-



(a) TREC2014

(b) tweetSanders

(c) tweetMarch

Figure 6.8: NMI evaluation results for the three datasets

pared to NMijF, see Figure 6.8. We use different numbers of topics to test the performance of the methods. As shown by all subfigures in Figure 6.8, tNMijF constantly outperforms the original NMijF.

Table 6.3 shows the results of the pairwise F-Measure metrics. The inclusion of the temporal aspect function improves both precision and recall in comparison to the NMijF in all datasets. tNMijF consistently provides the best results for both precision and recall. At all experiments, the proposed method outperforms NMijF, which does not take the temporal aspect into account.

Table 6.3: *Precision (p), Recall (r) and F-Measure (F-M)* for the three datasets

Method	p	r	F-M
tNMijF	**0.392**	**0.378**	**0.385**
NMijF	0.369	0.344	0.356

(a) TREC2014, $k = 55$

Method	p	r	F-M
tNMijF	**0.871**	**0.984**	**0.924**
NMijF	0.767	0.977	0.818

(b) tweetSanders, $k = 4$

Method	p	r	F-M
tNMijF	**0.610**	**0.385**	**0.469**
NMijF	0.581	0.329	0.420

(c) tweetMarch, $k = 6$

6.5.2 Tweet distributions and purity evaluations over time periods

The evaluations of tNMijF in the previous subsection are for static collections of tweets. In fact, in an online environment like Twitter, topics may have a lot of changes over a time period. This subsection focuses on the dynamic nature of the Twitter stream and the varying nature of topics in this stream. A tweet stream is divided into a series of time periods, and the performance of our proposed method is evaluated by measuring the accuracy of topic derivation over the timeline. The TREC2014 dataset is used to demonstrate the tweet distributions and the purity evaluations with different topic derivation methods.

TREC2014 has been used for the purpose of temporal-based information retrieval [65]. Here, we consider the tweets that belong to the first ten topics (MB171 to MB180) in TREC2014. The total number of tweets is 7126. These tweets are sorted by the posting time in an ascending order. We put the first 7000 sorted tweets into 7 temporal groups (T1 to T7). Each group has 1000 tweets in a period of time. These time periods have quite similar length (about one week).

For the sake of completeness, we include several other methods from the previous chapters as baselines. We applied our latest method and baseline methods for tweets

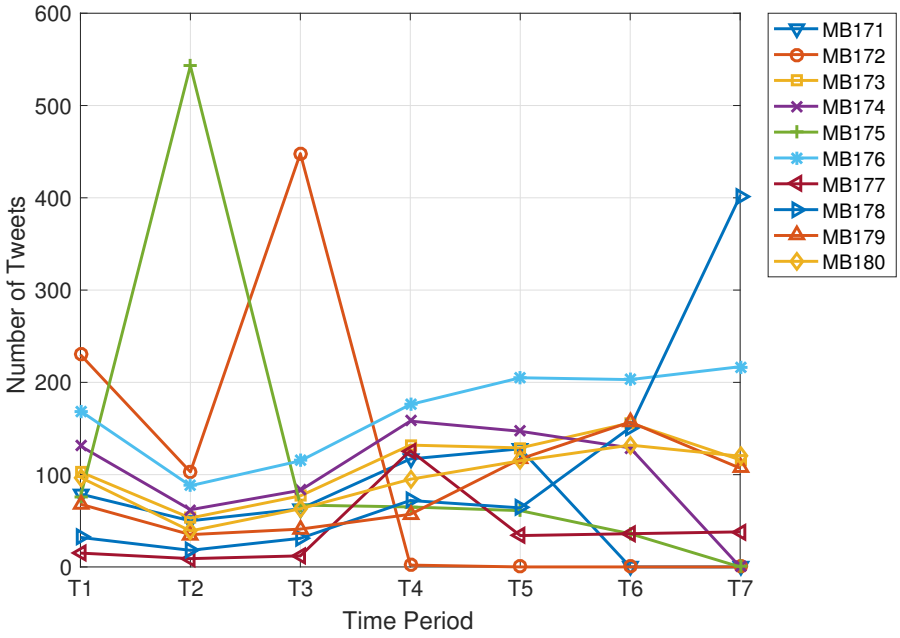


Figure 6.9: Tweet distributions over time periods for labeled topics

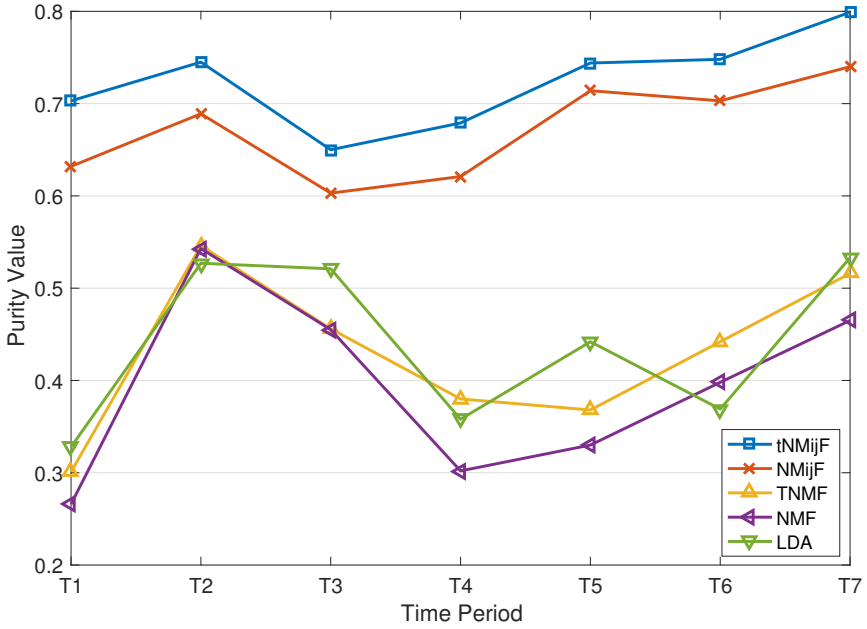


Figure 6.10: Purity evaluation results for different time periods

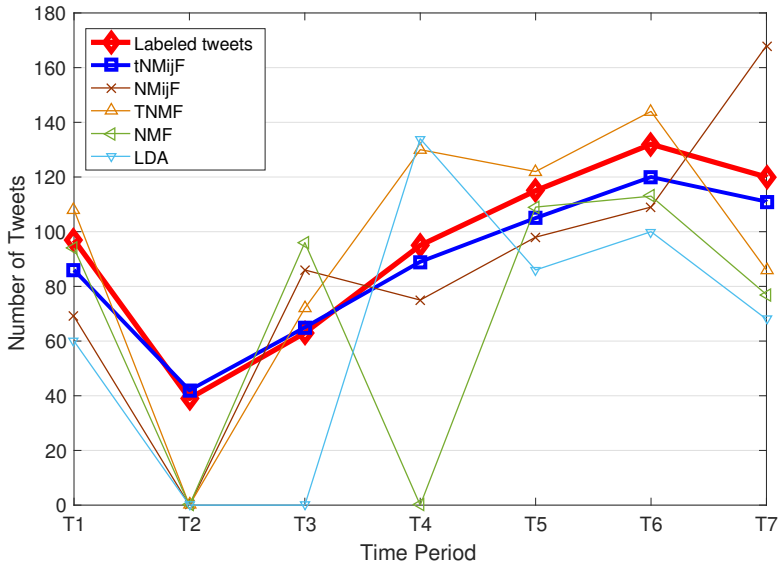


Figure 6.11: Tweet distributions over time periods for the topic MB180 with different topic derivation methods

in each group to derive topics and carried out the purity evaluation. The tweet distributions over the time periods for labeled topics are shown in Figure 6.9. The purity evaluation results for the different time periods are shown in Figure 6.10. The tweet distributions over the time periods for the topic MB180 are shown in Figure 6.11.

To evaluate the performance of a method, it is necessary to examine the numbers of tweets that belong to a specific topic over different time periods. For all methods, the purity values in $T2$ are quite high due to the fact that more than 500 tweets belong to the topic MB175 (see Figure 6.9). For a specific time period, when there is no topic with a dominant number of tweets, the purity values are quite low for all baseline methods. Our time-sensitive method performs very well in such a situation comparing with these baseline methods. In Figure 6.11, the line with the diamond symbols shows the numbers of labeled tweets that belong to the topic MB180; the line with square symbols shows the numbers of tweets that belong to the topic MB180 by using our new method. The other lines show the results by using the

baseline methods. When the number of tweets belonging to a topic is low, the baseline methods and the original NMijF could not get any reasonable results. The topic is totally missing. Our new proposed method tNMijF, which takes the time into account, provides a very accurate result. At $T2$, 39 tweets are labeled under the topic MB180; 42 tweets are put under this topic using our method.

Consistent with the evaluations against the static tweet collections in the previous subsection, our proposed method tNMijF achieves the best performance over all time periods. These results show that the varying nature of topics in a timeline will not strongly affect the accuracy improvement brought in by our method. This analysis indicates that our final method can cope with dynamic tweet streams better than existing methods. It can derive topics by processing the tweet streams as a series of tweet groups and achieve good results with 74% accuracy on average.

6.6 Discussion

In this chapter, we have investigated the effect of time on user interactions for topic derivation in Twitter. We found that the user mention is time-sensitive with respect to the topic assignment. We modeled the time sensitivity of mentions as an exponential decay according to the time difference of two tweets with the same mentions. We have proposed a new topic derivation method that incorporates this time factor. We conducted a set of experiments on the 3 different datasets described in Chapter 3.

Our results show that incorporating a temporal aspect on the interaction features can improve the quality of topic derivation results. In particular, the proposed method results in a consistent improvement in the quality of topic derivation over both well-known baseline methods and our prior method, which was not time-sensitive. NMijF with time factor can also be implemented to derive topics by processing the tweet streams as a series of tweet groups and achieve good results.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

In this thesis, we proposed a novel method to improve the quality of topic derivation in a Twitter environment. This method incorporates tweet content similarity and interactions measures. Topic derivation is the unsupervised task of clustering tweets based on their main topics and listing the most important keywords to represent the identified topics. Because posts in Twitter are short and the environment is highly dynamic, the task of deriving topics from the post remains a huge challenge. Most existing methods are based on the semantic features of tweet contents as the only source of information. Because tweets are short by nature, such methods suffer from data sparsity, which, in turn, hurts the quality of topic derivation [29].

Current topic derivation methods often model a document as a mixture of topics. The methods determine the most likely distribution of words per topic and the most likely distribution of topics in documents. After the process, it is straightforward to determine the most important topics in a document and the most salient keywords for a topic. Our investigations find a marked predominance of one topic per tweet, which is not surprising given how short tweets are. It is thus sensible to characterize a tweet by its most important topic.

Most methods derive topics from the vocabulary used in the posts and their term co-occurrences. Our statistical analysis shows that the density of the term-to-term relationship matrix on a set of tweets is only 0.274% on average. Directly applying

methods that consider only the content of Twitter data may thus produce a poor characterization of the topics. Some techniques have been proposed to address the sparsity problem. They include content expansion (e.g., [98] [99], [44], [70], [122]), exploiting advanced semantic features of the tweet contents (e.g., [136], [135]), and incorporating content-based social features (e.g., [106], [104], [131]). Because Twitter is a highly changing environment, it is difficult to accurately predict what relevant content from external sources is to be added to the tweet. As for the exploitation of advanced semantic features of the tweets, including the limited content-based social features, it still potentially suffers from the sparsity problem.

Inspired by the limitations of those methods, we go beyond the tweet contents and incorporate the social interactions present in Twitter. Twitter offers several interactive features enabling users to interact with each other through tweets, such as *user mentions*, *replies*, *retweets*, and *hashtags*. User mentions and replies are helpful methods for initiating or joining a conversation on Twitter. Intuitively, all tweets belonging to the same conversation have a high chance of sharing the same topic even if no terms co-occur in their content. A retweet is a re-posting of someone else's tweet. Since retweets have many words in common with the original tweet, the term co-occurrence between the two tweets (original and retweet) will be high, and both tweets are likely to share a topic. Hashtag is another important content-based interaction feature in Twitter, popularly used to bookmark the content of a tweet or to present the users' interest to particular topics [138]. Our investigation shows that those components are able to characterize the topical relationships between tweets. The interactions are often associated with conversational activities between users through the tweets. We thus define the relationships between tweets as the interactions based on user mentions, replies-retweets and content similarity (including hashtags). In this thesis, we have shown that our definition of the tweet-to-tweet relationships results in a matrix with a much higher density than the other types of content-based relationships.

Our analysis on the topical connectivity between tweets shows that tweets linked by interactions and content similarity have different probabilities to be about the same topic. We observe that tweets linked by replies and retweets are almost always about the same topics, and tweets linked by user mentions and content similarity have quite a reasonable chance to be about the same topics. We also note that the number of tweets linked by user mentions and content similarity is much larger than the number of tweets linked by replies and retweets. We propose a joint probability model for the topical relationships between tweets to integrate the effects of the replies-retweets, user mentions, and the content similarity accurately. It provides a new foundation to build a more accurate tweet-to-tweet relationships matrix.

To incorporate the new joint probability model, we propose a Non-Negative inter-joint Matrix Factorization (NMijF) method. It factorizes two non-negative matrices, the symmetric tweet-to-tweet relationships matrix $A \in \mathbb{R}^{m \times m}$ and the tweet-to-term matrix $V \in \mathbb{R}^{m \times n}$, into latent tweet-topic matrix $W \in \mathbb{R}^{m \times k}$ and topic-term matrix $Y \in \mathbb{R}^{k \times n}$ in a unified iterative update process. NMijF takes a joint-approach in each iteration, that is, it updates the sharing latent matrix W only according to the tweet-to-tweet matrix A. A biased update rule for tweet-topic matrix W reduces the penalty due to the extreme sparsity of the tweet-to-term matrix V.

Experimental results against two publicly available Twitter datasets (TREC2014 and tweetSanders) and a dataset we collected (tweetMarch) demonstrate that our proposed NMijF method consistently outperforms all baseline methods on all evaluation metrics. The joint probability model for the tweet relationships has a positive impact on the quality of topic derivation. The experimental results reveal that the more accurate integration of content similarity, user mentions, and replies-retweets when measuring the relationship between tweets provides a significant improvement of the topic derivation quality in different situations.

Deriving topics from a collection of tweets is also problematic due to the highly dynamic environment. We investigate the tweets interactions' behavior to see

how time affects the topical connectivity between tweets. For this purpose, we retrieve all tweets from the top 15 Twitter users in Australia and their related tweets (related through user mentions, replies, and retweets) during 12 January 2015 until 12 February 2015. The collected dataset consists of more than 6 million tweets and involves around 800 thousands users. We conduct an analysis of the tweets interactions at different levels of granularity to see how the tweets and their interactions are distributed over time. We find that the replies or retweets interactions are not affected by time. Tweets that linked by replies or retweets are almost always about the same topic regardless of the posting time. In contrast, tweets linked by user mentions are sensitive to time. These tweets tend to be about the same topic, only when they are posted within the same period of time. Our statistical analysis shows that most of the user mentions related to a particular topic reach a peak in about 15 minutes and then gradually fade away.

We introduce the half-life exponential decay function to incorporate the time aspect of tweets interactions. The function models the process of topic fading away to provide a more precise relationship measurement when tweets are linked by user mentions. Experimental results show that the inclusion of this temporal aspect into the process results in further improvement on the quality of topic derivation in Twitter.

In an online environment like Twitter, topics may have a lot of changes over a time period. We also evaluate our work in the nature of the Twitter stream and the varying nature of topics in this stream. Consistent with the previous evaluations, our proposed method achieves the best performance over all time periods. This analysis indicates that our proposed method can cope with dynamic tweet streams better than existing methods. We conclude that incorporating the relationships between tweets and considering their time sensitivity are effective for dealing with the sparsity problem in Twitter and its dynamic environment.

7.2 Future Work

The work in this thesis focuses on addressing the sparsity problem and the dynamic nature of the Twitter environment with respect to the task of topic derivation. This study has raised several further questions, including how to dynamically choose the optimal number of topics, how to automatically summarize or label the topics, and the possibility to incorporate a more complex combination of different features while maintaining the scalability of the applications in an online environment. These are our new research directions.

A complex combination of different features includes improving the accuracy of the content similarity and investigating the trade-off between number of tweets and time period in the online processing. We are working on the integration of our proposed topic derivation approach and efficient automatic topic summarization method to identify topics evolution and tipped topics in various real-time application domains, such as in emergency management, situation monitoring, or marketing. Finally, we are investigating the possibility to apply the approach in other social media platforms with different characteristics.

Bibliography

[1] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau. Sentiment analysis of Twitter data. In *Proceedings of the Workshop on Languages in Social Media*, LSM '11, pages 30–38, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. ISBN 978-1-932432-96-1. URL `http://dl.acm.org/citation.cfm?id=2021109.2021114`.

[2] M. Albakour, C. Macdonald, I. Ounis, et al. On sparsity and drift for effective real-time filtering in microblogs. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pages 419–428. ACM, 2013.

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

[4] L. AlSumait, D. Barbarà, and C. Domeniconi. On-line LDA: Adaptive topic models for mining text streams with applications to topic detection and tracking. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, pages 3–12, Dec 2008. doi: 10.1109/ICDM.2008.140.

[5] J. R. Bellegarda. Exploiting both local and global constraints for multi-span statistical language modeling. In *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, 1998*, volume 2, pages 677–680 vol.2, May 1998. doi: 10.1109/ICASSP.1998.675355.

[6] J. R. Bellegarda, J. W. Butzberger, Y.-L. Chow, N. B. Coccaro, and D. Naik. A novel word clustering algorithm based on Latent Semantic Analysis. In

Proceedings of the 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings, volume 1, pages 172–175 vol. 1, May 1996. doi: 10.1109/ICASSP.1996.540318.

[7] S. K. Bista, S. Nepal, and C. Paris. Multifaceted visualisation of annotated social media data. In *Proceedings of the Big Data (BigData Congress), 2014 IEEE International Congress on*, pages 699–706. IEEE, 2014.

[8] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet Allocation. *The Journal of Machine Learning research*, 3:993–1022, 2003.

[9] D. M. Blei. Probabilistic topic models. *Communications of the ACM*, 55(4): 77–84, 2012.

[10] D. M. Blei and J. D. Lafferty. A correlated topic model of science. *The Annals of Applied Statistics*, pages 17–35, 2007.

[11] G. Cao, J.-Y. Nie, J. Gao, and S. Robertson. Selecting good expansion terms for pseudo-relevance feedback. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 243–250. ACM, 2008.

[12] N. Carlson. The real history of Twitter. `http://www.businessinsider.com.au/how-twitter-was-founded-2011-4`, April 2011. [Online, Accessed 6 October 2016].

[13] C. A. Cassa, R. Chunara, K. Mandl, and J. S. Brownstein. Twitter as a sentinel in emergency situations: lessons from the Boston marathon explosions. *PLOS Currents Disasters*, 2013.

[14] M. Cataldi, L. Di Caro, and C. Schifanella. Emerging topic detection on Twitter based on temporal and social terms evaluation. In *Proceedings of*

the Tenth International Workshop on Multimedia Data Mining, MDMKDD '10, pages 4:1–4:10, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0220-3. doi: 10.1145/1814245.1814249. URL `http://doi.acm.org/10.1145/1814245.1814249`.

[15] J. Chang, S. Gerrish, C. Wang, J. L. Boyd-Graber, and D. M. Blei. Reading tea leaves: How humans interpret topic models. In *Proceedings of The Advances in neural information processing systems*, pages 288–296, 2009.

[16] Y. Chen, H. Amiri, Z. Li, and T.-S. Chua. Emerging topic detection for organizations from microblogs. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, pages 43–52, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2034-4. doi: 10.1145/2484028.2484057. URL `http://doi.acm.org/10.1145/2484028.2484057`.

[17] X. Cheng, X. Yan, Y. Lan, and J. Guo. BTM: Topic modeling over short texts. *IEEE Transactions on Knowledge and Data Engineering*, 26(12):2928–2941, Dec 2014. ISSN 1041-4347. doi: 10.1109/TKDE.2014.2313872.

[18] F. Cheong and C. Cheong. Social media data mining: A social network analysis of tweets during the Australian 2010-2011 floods. In *Proceedings of the 15th Pacific Asia Conference on Information Systems (PACIS)*, pages 1–16. Queensland University of Technology, 2011.

[19] F. Chierichetti, J. M. Kleinberg, R. Kumar, M. Mahdian, and S. Pandey. Event detection via communication pattern analysis. In *ICWSM*, 2014.

[20] A. Cichocki, R. Zdunek, and S.-i. Amari. New algorithms for Non-negative Matrix Factorization in applications to blind source separation. In *Proceedings*

of the 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings, volume 5, pages V–V. IEEE, 2006.

[21] D. P. Council. Researh note: World leader ranking on Twitter. `http://www.digitaldaya.com/admin/modulos/galeria/pdfs/73/161_o59ontgs.pdf`, December 2015. [Online, Accessed 6 October 2016].

[22] T. M. Cover and J. A. Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[23] A. Culotta. Towards detecting influenza epidemics by analyzing Twitter messages. In *Proceedings of the First Workshop on Social Media Analytics*, SOMA '10, pages 115–122, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0217-3. doi: 10.1145/1964858.1964874. URL `http://doi.acm.org/10.1145/1964858.1964874`.

[24] A. de Moor. Conversations in context: a Twitter case for social media systems design. In *Proceedings of the 6th International Conference on Semantic Systems*, page 29. ACM, 2010.

[25] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by Latent Semantic Analysis. *Journal of the American society for information science*, 41(6):391, 1990.

[26] K. Devarajan. Nonnegative Matrix Factorization: an analytical and interpretive tool in computational biology. *PLoS Comput Biol*, 4(7):e1000029, 2008.

[27] L. Dietz, S. Bickel, and T. Scheffer. Unsupervised prediction of citation influences. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pages 233–240, New York, NY, USA, 2007. ACM. ISBN

978-1-59593-793-3. doi: 10.1145/1273496.1273526. URL http://doi.acm.org/10.1145/1273496.1273526.

[28] A. Dubey, A. Hefny, S. Williamson, and E. P. Xing. A nonparametric mixture model for topic modeling over time. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, pages 530–538. doi: 10.1137/1.9781611972832.59. URL http://epubs.siam.org/doi/abs/10.1137/1.9781611972832.59.

[29] K. Erk. Vector space models of word meaning and phrase meaning: A survey. *Language and Linguistics Compass*, 6(10):635–653, 2012.

[30] J. L. Fleiss. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378, 1971.

[31] G. E. Forsythe, C. B. Moler, and M. A. Malcolm. *Computer methods for mathematical computations*. Prentice-Hall, 1977.

[32] E. Gaussier and C. Goutte. Relation between PLSA and NMF and implications. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 601–602. ACM, 2005.

[33] M. Girolami and A. Kabán. On an equivalence between PLSI and LDA. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 433–434. ACM, 2003.

[34] Y. Gotoh and S. Renals. Document space models using Latent Semantic Analysis. 1997.

[35] T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National academy of Sciences*, 101(suppl 1):5228–5235, 2004.

[36] W. Guo, H. Li, H. Ji, and M. T. Diab. Linking tweets to news: A framework to enrich short text data in social media. In *Proceedings of the 2013 Association for Computational Linguistics Conference*, Sofia, Bulgaria.

[37] Z. He, S. Xie, R. Zdunek, G. Zhou, and A. Cichocki. Symmetric Nonnegative Matrix Factorization: Algorithms and applications to probabilistic clustering. *IEEE Transactions on Neural Networks*, 22(12):2117–2131, 2011.

[38] A. Hermida, S. C. Lewis, and R. Zamith. Sourcing the arab spring: a case study of andy carvin's sources on Twitter during the Tunisian and Egyptian revolutions. *Journal of Computer-Mediated Communication*, 19(3):479–499, 2014.

[39] B. Hofer, T. J. Lampoltshammer, and M. Belgiu. Demography of Twitter users in the city of london: An exploratory spatial data analysis approach. In *Modern Trends in Cartography*, pages 199–211. Springer, 2015.

[40] M. Hoffman, F. R. Bach, and D. M. Blei. Online learning for latent dirichlet allocation. In *advances in neural information processing systems*, pages 856–864, 2010.

[41] T. Hofmann. Probabilistic Latent Semantic Indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57. ACM, 1999.

[42] L. Hong and B. D. Davison. Empirical study of topic modeling in Twitter. In *Proceedings of the First Workshop on Social Media Analytics*, SOMA '10, pages 80–88, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0217-3. doi: 10.1145/1964858.1964870. URL `http://doi.acm.org/10.1145/1964858.1964870`.

[43] P. O. Hoyer. Non-negative Matrix Factorization with sparseness constraints. *Journal of machine learning research*, 5(Nov):1457–1469, 2004.

[44] X. Hu, N. Sun, C. Zhang, and T.-S. Chua. Exploiting internal and external semantics for the clustering of short texts using world knowledge. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, pages 919–928, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-512-3. doi: 10.1145/1645953.1646071. URL http://doi.acm.org/10.1145/1645953.1646071.

[45] M. Jamali and M. Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 135–142. ACM, 2010.

[46] O. Jin, N. N. Liu, K. Zhao, Y. Yu, and Q. Yang. Transferring topical knowledge from auxiliary long texts for short text clustering. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, CIKM '11, pages 775–784, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0717-8. doi: 10.1145/2063576.2063689. URL http://doi.acm.org/10.1145/2063576.2063689.

[47] J. J. Jones, J. E. Settle, R. M. Bond, C. J. Fariss, C. Marlow, and J. H. Fowler. Inferring tie strength from online directed behavior. *PloS one*, 8(1):e52168, 2013.

[48] B. Kang, J. O'Donovan, and T. Höllerer. Modeling topic specific credibility on Twitter. In *Proceedings of the 2012 ACM international conference on Intelligent User Interfaces*, pages 179–188. ACM, 2012.

[49] E. Kim, Y. Sung, and H. Kang. Brand followersâĂŹ retweeting behavior on

Twitter: How brand relationships influence brand electronic word-of-mouth. *Computers in Human Behavior*, 37:18–25, 2014.

[50] H. Kim and H. Park. Sparse Non-negative Matrix Factorizations via alternating non-negativity-constrained least squares for microarray data analysis. *Bioinformatics*, 23(12):1495–1502, 2007.

[51] J. Kim and H. Park. Sparse Nonnegative Matrix Factorization for clustering. 2008.

[52] M. Kim, J. Yoo, K. Kang, and S. Choi. Nonnegative matrix partial co-factorization for spectral and temporal drum source separation. *IEEE Journal of Selected Topics in Signal Processing*, 5(6):1192–1204, Oct 2011. ISSN 1932-4553. doi: 10.1109/JSTSP.2011.2158803.

[53] K. Kireyev, L. Palen, and K. Anderson. Applications of topics models to analysis of disaster-related Twitter data. In *Proceedings of the NIPS Workshop on Applications for Topic Models: Text and Beyond*, volume 1. Canada: Whistler, 2009.

[54] E. Kouloumpis, T. Wilson, and J. D. Moore. Twitter sentiment analysis: The good the bad and the omg! In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*, volume 11, pages 538–541, Barcelona, Spain, 2011.

[55] M. Krieger and D. Ahn. Tweetmotif: exploratory search and topic summarization for Twitter. In *In Proc. of AAAI Conference on Weblogs and Social*, 2010.

[56] D. Kuang, H. Park, and C. Ding. Symmetric Nonnegative Matrix Factorization for graph clustering. In *Proceedings of the 2012 SIAM International Conference on Data Mining*, volume 12, pages 106–117, California, USA, 2012. SIAM.

[57] M. Kuczma. *An introduction to the theory of functional equations and inequalities: Cauchy's equation and Jensen's inequality*. Springer Science & Business Media, 2009.

[58] S. Kullback. *Information theory and statistics*. Courier Dover Publications, 1997.

[59] M. Kulldorff, R. Heffernan, J. Hartman, R. Assunçao, and F. Mostashari. A space–time permutation scan statistic for disease outbreak detection. *Plos med*, 2(3):e59, 2005.

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

[61] J. H. Lau, N. Collier, and T. Baldwin. On-line trend analysis with topic models:\# Twitter trends detection topic model online. In *Proceedings of the 24th International Conference on Computational Linguistics*, Mumbai, India, Dec .

[62] D. Lee and H. Seung. Algorithms for Non-negative Matrix Factorization. In *Proceedings of the Advances in Neural Information Processing Systems 13 (NIPS 2000)*, pages 556–562, Denver, CO, USA, 2000.

[63] D. D. Lee and H. S. Seung. Learning the parts of objects by Non-negative Matrix Factorization. *Nature*, 401(6755):788–791, 1999.

[64] W. Li, Y. Feng, D. Li, and Z. Yu. Micro-blog topic detection method based on BTM topic model and K-means clustering algorithm. *Automatic Control and Computer Sciences*, 50(4):271–277, 2016. ISSN 1558-108X. doi: 10.3103/S0146411616040040. URL `http://dx.doi.org/10.3103/S0146411616040040`.

[65] J. Lin, M. Efron, Y. Wang, and G. Sherman. Overview of the trec-2014 microblog track. Technical report, NIST, 2014. URL http://trec.nist.gov/pubs/trec23/trec2014.html.

[66] C. Liu, H.-c. Yang, J. Fan, L.-W. He, and Y.-M. Wang. Distributed Nonnegative Matrix Factorization for web-scale dyadic data analysis on mapreduce. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 681–690, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-799-8. doi: 10.1145/1772690.1772760. URL http://doi.acm.org/10.1145/1772690.1772760.

[67] F. Liu, Y. Liu, and F. Weng. Why is "sxsw" trending?: Exploring multiple text sources for Twitter topic summarization. In *Proceedings of the Workshop on Languages in Social Media*, LSM '11, pages 66–75, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. ISBN 978-1-932432-96-1. URL http://dl.acm.org/citation.cfm?id=2021109.2021118.

[68] J. Liu, C. Wu, and W. Liu. Bayesian probabilistic matrix factorization with social relations and item contents for recommendation. *Decision Support Systems*, 55(3):838 – 850, 2013. ISSN 0167-9236. doi: https://doi.org/10.1016/j.dss.2013.04.002. URL http://www.sciencedirect.com/science/article/pii/S0167923613000912.

[69] Y. Liu, A. Niculescu-Mizil, and W. Gryc. Topic-link LDA: Joint models of topic and author community. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 665–672, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-516-1. doi: 10.1145/1553374.1553460. URL http://doi.acm.org/10.1145/1553374.1553460.

[70] C. Lv, R. Qiang, F. Fan, and J. Yang. Proceedings of the information retrieval technology: 11th asia information retrieval societies conference, airs 2015,

brisbane, qld, australia, december 2-4, 2015. proceedings. pages 43–55, Cham, 2015. Springer International Publishing. ISBN 978-3-319-28940-3. doi: 10.1007/978-3-319-28940-3_4. URL http://dx.doi.org/10.1007/978-3-319-28940-3_4.

[71] H. Ma, H. Yang, M. R. Lyu, and I. King. Sorec: social recommendation using probabilistic matrix factorization. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 931–940. ACM, 2008.

[72] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King. Recommender systems with social regularization. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 287–296. ACM, 2011.

[73] H.-F. Ma, Y.-X. Sun, M.-H.-Z. Jia, and Z.-C. Zhang. Microblog hot topic detection based on topic model using term correlation matrix. In *Proceedings of the 2014 International Conference on Machine Learning and Cybernetics*, volume 1, pages 126–130, July 2014. doi: 10.1109/ICMLC.2014.7009104.

[74] Z. Ma, W. Dou, X. Wang, and S. Akella. Tag-Latent Dirichlet Allocation: Understanding hashtags and their relationships. In *Proceedings of the Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences on*, volume 1, pages 260–267, Nov 2013. doi: 10.1109/WI-IAT.2013.38.

[75] L. L. Magoarou, A. Ozerov, and N. Q. K. Duong. Text-informed audio source separation using nonnegative matrix partial co-factorization. In *Proceedings of the 2013 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, Sept 2013. doi: 10.1109/MLSP.2013.6661995.

[76] J. I. Maletic and N. Valluri. Automatic software clustering via Latent Semantic Analysis. In *Proceedings of the 14th IEEE International Conference*

on Automated Software Engineering, 1999, pages 251–254, Oct 1999. doi: 10.1109/ASE.1999.802296.

[77] C. Manning, P. Raghavan, and H. Schütze. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge, 2008.

[78] T. Masada, S. Kiyasu, and S. Miyahara. Comparing LDA with pLSI as a dimensionality reduction method in document clustering. In *Large-Scale Knowledge Resources. Construction and Application*, pages 13–26. Springer, 2008.

[79] A. McCallum, A. Corrada-Emmanuel, and X. Wang. The author-recipient-topic model for topic and role discovery in social networks: Experiments with enron and academic email. 2005.

[80] A. McCallum, X. Wang, and A. Corrada-Emmanuel. Topic and role discovery in social networks with experiments on enron and academic email. *Journal of Artificial Intelligence Research*, 30:249–272, 2007.

[81] R. Mehrotra, S. Sanner, W. Buntine, and L. Xie. Improving LDA topic models for microblogs via tweet pooling and automatic labeling. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, pages 889–892, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2034-4. doi: 10.1145/2484028.2484166. URL `http://doi.acm.org/10.1145/2484028.2484166`.

[82] Y. Mei, Z. Zhang, W. Zhao, J. Yang, and R. Nugroho. A hybrid feature selection method for predicting user influence on Twitter. In J. Wang, W. Cellary, D. Wang, H. Wang, S.-C. Chen, T. Li, and Y. Zhang, editors, *Proceedings of the Web Information Systems Engineering – WISE 2015: 16th International Conference, Miami, FL, USA, November 1-3, 2015, Part I*, pages 478–492,

Cham, 2015. Springer International Publishing. ISBN 978-3-319-26190-4. doi: 10.1007/978-3-319-26190-4_32. URL http://dx.doi.org/10.1007/978-3-319-26190-4_32.

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

[84] S. A. Myers, A. Sharma, P. Gupta, and J. Lin. Information network or social network?: The structure of the Twitter follow graph. In *Proceedings of the companion publication of the 23rd international conference on World wide web companion*, pages 493–498. International World Wide Web Conferences Steering Committee, 2014.

[85] D. Q. Nguyen, R. Billingsley, L. Du, and M. Johnson. Improving topic models with latent feature word representations. *Transactions of the Association for Computational Linguistics*, 3:299–313, 2015. ISSN 2307-387X. URL https://transacl.org/ojs/index.php/tacl/article/view/582.

[86] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using em. *Machine learning*, 39(2-3): 103–134, 2000.

[87] R. Nugroho, J. Yang, Y. Zhong, C. Paris, and S. Nepal. Deriving topics in Twitter by exploiting tweet interactions. In *Proceedings of the 2015 IEEE International Congress on Big Data*, pages 87–94, June 2015. doi: 10.1109/BigDataCongress.2015.22.

[88] R. Nugroho, W. Zhao, J. Yang, C. Paris, S. Nepal, and Y. Mei. Time-sensitive topic derivation in Twitter. In *Proceedings of the Web Information Systems Engi-*

neering – WISE 2015: 16th International Conference, Miami, FL, USA, November 1-3, 2015, Proceedings, Part I, pages 138–152, Cham, 2015. Springer International Publishing.

[89] R. Nugroho, Y. Zhong, J. Yang, C. Paris, and S. Nepal. Matrix inter-joint factorization - a new approach for topic derivation in Twitter. In *Proceedings of the 2015 IEEE International Congress on Big Data*, pages 79–86, June 2015. doi: 10.1109/BigDataCongress.2015.21.

[90] R. Nugroho, D. Molla-Aliod, J. Yang, Y. Zhong, C. Paris, and S. Nepal. Incorporating tweet relationships into topic derivation. In K. Hasida and A. Purwarianti, editors, *Proceedings of the Computational Linguistics: 14th International Conference of the Pacific Association for Computational Linguistics, PACLING 2015, Bali, Indonesia, May 19-21, 2015, Revised Selected Papers*, pages 177–190, Singapore, 2016. Springer Singapore. ISBN 978-981-10-0515-2. doi: 10.1007/978-981-10-0515-2_13. URL `http://dx.doi.org/10.1007/978-981-10-0515-2_13`.

[91] R. Nugroho, W. Zhao, J. Yang, C. Paris, and S. Nepal. Using time-sensitive interactions to improve topic derivation in Twitter. *World Wide Web*, pages 1–27, 2016. ISSN 1573-1413. doi: 10.1007/s11280-016-0417-x. URL `http://dx.doi.org/10.1007/s11280-016-0417-x`.

[92] R. Nugroho, J. Yang, W. Zhao, C. Paris, and S. Nepal. What and with whom? identifying topics in Twitter through both interactions and text. *IEEE Transactions on Services Computing*, PP(99):1–1, 2017. ISSN 1939-1374. doi: 10.1109/TSC.2017.2696531.

[93] R. Nugroho, W. Zhao, J. Yang, C. Paris, and S. Nepal. The joint effects of tweet content similarity and tweet interactions for topic derivation. In *Proceedings of*

the 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS), pages 2338–2343, June 2017. doi: 10.1109/ICDCS.2017.60.

[94] A. Ostrow. Japan earthquake shakes Twitter users... and beyonce. `http://mashable.com/2009/08/12/japan-earthquake/#4IvI9oMp8kqd`, August 2009. [Online, Accessed 6 October 2016].

[95] O. Ozdikis, P. Senkul, and H. Oguztuzun. Semantic expansion of tweet contents for enhanced event detection in Twitter. In *Proceedings of the 2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 20–24, Aug 2012. doi: 10.1109/ASONAM.2012.14.

[96] A. Pak and P. Paroubek. Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of the 7th Language Resources and Evaluation Conference*, volume 10, pages 1320–1326, Valletta, Malta, 2010.

[97] I. Pentina, B. S. Gammoh, L. Zhang, and M. Mallin. Drivers and outcomes of brand relationship quality in the context of online social networks. *International Journal of Electronic Commerce*, 17(3):63–86, 2013.

[98] X.-H. Phan, L.-M. Nguyen, and S. Horiguchi. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *Proceedings of the 17th International Conference on World Wide Web*, WWW '08, pages 91–100, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-085-2. doi: 10.1145/1367497.1367510. URL `http://doi.acm.org/10.1145/1367497.1367510`.

[99] X. H. Phan, C. T. Nguyen, D. T. Le, L. M. Nguyen, S. Horiguchi, and Q. T. Ha. A hidden topic-based framework toward building applications with short

web documents. *IEEE Transactions on Knowledge and Data Engineering*, 23 (7):961–976, July 2011. ISSN 1041-4347. doi: 10.1109/TKDE.2010.27.

[100] R. Pochampally and V. Varma. User context as a source of topic retrieval in Twitter. In *Workshop on Enriching Information Retrieval (with ACM SIGIR)*, pages 1–3, 2011.

[101] M. Prateek and V. Vasudeva. Improved topic models for social media via community detection using user interaction and content similarity. In *Proceedings of the 2016 International FRUCT Conference on Intelligence, Social Media and Web (ISMW FRUCT)*, pages 1–7, Aug 2016. doi: 10.1109/FRUCT.2016.7584770.

[102] K. W. Prier, M. S. Smith, C. Giraud-Carrier, and C. L. Hanson. Identifying health-related topics on Twitter. In J. Salerno, S. J. Yang, D. Nau, and S.-K. Chai, editors, *Proceedings of the Social Computing, Behavioral-Cultural Modeling and Prediction: 4th International Conference, SBP 2011, College Park, MD, USA, March 29-31, 2011.*, pages 18–25, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg. ISBN 978-3-642-19656-0. doi: 10.1007/978-3-642-19656-0_4. URL http://dx.doi.org/10.1007/978-3-642-19656-0_4.

[103] M. Qiu, F. Zhu, and J. Jiang. It is not just what we say, but how we say them: LDA-based behavior-topic model. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, pages 794–802. doi: 10.1137/1.9781611972832.88. URL http://epubs.siam.org/doi/abs/10.1137/1.9781611972832.88.

[104] N. F. N. Rajani, K. McArdle, and J. Baldridge. Extracting topics based on authors, recipients and content in microblogs. In *Proceedings of the 37th*

International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR '14, pages 1171–1174, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2257-7. doi: 10.1145/2600428.2609537. URL http://doi.acm.org/10.1145/2600428.2609537.

[105] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*, EMNLP '09, pages 248–256, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics. ISBN 978-1-932432-59-6. URL http://dl.acm.org/citation.cfm?id=1699510.1699543.

[106] D. Ramage, S. T. Dumais, and D. J. Liebling. Characterizing microblogs with topic models. volume 10, pages 130–137, Washington DC, USA, May 2010. AAAI.

[107] A. Saha and V. Sindhwani. Learning evolving and emerging topics in social media: A dynamic nmf approach with temporal regularization. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*, WSDM '12, pages 693–702, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-0747-5. doi: 10.1145/2124295.2124376. URL http://doi.acm.org/10.1145/2124295.2124376.

[108] G. Salton. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of*. Addison-Wesley, 1989.

[109] F. Shahnaz, M. W. Berry, V. P. Pauca, and R. J. Plemmons. Document clustering using Nonnegative Matrix Factorization. *Information Processing & Management*, 42(2):373–386, 2006.

[110] B. Sharifi, M. A. Hutton, and J. K. Kalita. Experiments in microblog summarization. In *Proceedings of the 2010 IEEE Second International Conference on Social Computing*, pages 49–56, Aug 2010. doi: 10.1109/SocialCom.2010.17.

[111] M. Steyvers, P. Smyth, M. Rosen-Zvi, and T. Griffiths. Probabilistic author-topic models for information discovery. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 306–315, New York, NY, USA, 2004. ACM. ISBN 1-58113-888-1. doi: 10.1145/1014052.1014087. URL http://doi.acm.org/10.1145/1014052.1014087.

[112] G. Stilo and P. Velardi. Time makes sense: Event discovery in Twitter using temporal similarity. In *Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT) - Volume 02*, WI-IAT '14, pages 186–193, Washington, DC, USA, 2014. IEEE Computer Society. ISBN 978-1-4799-4143-8. doi: 10.1109/WI-IAT.2014.97. URL http://dx.doi.org/10.1109/WI-IAT.2014.97.

[113] A. Strehl and J. Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *The Journal of Machine Learning Research*, 3:583–617, 2003.

[114] K. Takeuchi, K. Ishiguro, A. Kimura, and H. Sawada. Non-negative multiple matrix factorization. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pages 1713–1720. AAAI Press, 2013.

[115] Y. Tang, X. Xiao, and Y. Shi. Influence maximization: Near-optimal time complexity meets practical efficiency. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, SIGMOD '14, pages 75–86, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2376-5. doi: 10.

1145/2588555.2593670. URL http://doi.acm.org/10.1145/2588555.2593670.

[116] L. Taslaman and B. Nilsson. A framework for regularized Non-negative Matrix Factorization, with application to the analysis of gene expression data. *PloS one*, 7(11):e46331, 2012.

[117] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 2012.

[118] O. Tsur, A. Littman, and A. Rappoport. Efficient clustering of short messages into general domains. In *Proceedings of the International AAAI Conference on Web and Social Media*, 2013. URL http://www.aaai.org/ocs/index.php/ICWSM/ICWSM13/paper/view/6103.

[119] Twitter. Twitter milestones. https://about.twitter.com/company/press/milestones. [Online, Accessed 6 October 2016].

[120] Twitter. #numbers. https://blog.twitter.com/2011/numbers, 2011. [Online, Accessed 6 October 2016].

[121] Twitter. Api overview. https://dev.twitter.com/overview/api, 2015. [Online, Accessed 28 September 2015].

[122] J. Vosecky, D. Jiang, K. W.-T. Leung, K. Xing, and W. Ng. Integrating social and auxiliary semantics for multifaceted topic modeling in Twitter. *ACM Transactions on Internet Technology (TOIT)*, 14(4):27, 2014.

[123] S. Wan and C. Paris. Improving government services with social media feedback. In *Proceedings of the 19th International Conference on Intelligent User Interfaces*, IUI '14, pages 27–36, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2184-6. doi: 10.1145/2557500.2557513. URL http://doi.acm.org/10.1145/2557500.2557513.

[124] F. Wang, P. Li, and A. C. König. Efficient document clustering via online Nonnegative Matrix Factorizations. In *Proceedings of the SIAM International Conference on Data Mining (SDM)*, volume 11, pages 908–919, Arizona, USA, April 2011. SIAM.

[125] X. Wang and A. McCallum. Topics over time: A non-markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, pages 424–433, New York, NY, USA, 2006. ACM. ISBN 1-59593-339-5. doi: 10.1145/1150402.1150450. URL http://doi.acm.org/10.1145/1150402.1150450.

[126] X. Wang, M. S. Gerber, and D. E. Brown. Automatic crime prediction using events extracted from Twitter posts. In S. J. Yang, A. M. Greenberg, and M. Endsley, editors, *Proceedings of the Social Computing, Behavioral - Cultural Modeling and Prediction: 5th International Conference, SBP 2012, College Park, MD, USA, April 3-5, 2012. Proceedings*, pages 231–238, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. ISBN 978-3-642-29047-3. doi: 10.1007/978-3-642-29047-3_28. URL http://dx.doi.org/10.1007/978-3-642-29047-3_28.

[127] X. Wang, D. Jin, X. Cao, L. Yang, and W. Zhang. Semantic community identification in large attribute networks. 2016. URL http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/11964.

[128] Y. Wang, E. Agichtein, and M. Benzi. TM-LDA: Efficient online modeling of latent topic transitions in social media. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pages 123–131, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-

1462-6. doi: 10.1145/2339530.2339552. URL http://doi.acm.org/10.1145/2339530.2339552.

[129] Y. Wang, J. Liu, J. Qu, Y. Huang, J. Chen, and X. Feng. Hashtag graph based topic model for tweet mining. In *Proceedings of the 2014 IEEE International Conference on Data Mining*, pages 1025–1030, Dec 2014. doi: 10.1109/ICDM.2014.60.

[130] J. Weng, E.-P. Lim, J. Jiang, and Q. He. Twitterrank: finding topic-sensitive influential Twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 261–270. ACM, 2010.

[131] H. Xia, J. Li, J. Tang, and M.-F. Moens. Plink-lda: Using link as prior information in topic modeling. In S.-g. Lee, Z. Peng, X. Zhou, Y.-S. Moon, R. Unland, and J. Yoo, editors, *Proceedings of the Database Systems for Advanced Applications: 17th International Conference, DASFAA 2012, Busan, South Korea, April 15-19, 2012, Part I*, pages 213–227, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. ISBN 978-3-642-29038-1. doi: 10.1007/978-3-642-29038-1_17. URL http://dx.doi.org/10.1007/978-3-642-29038-1_17.

[132] J. Xu, P. Liu, G. Wu, Z. Sun, B. Xu, and H. Hao. A fast matching method based on semantic similarity for short texts. In G. Zhou, J. Li, D. Zhao, and Y. Feng, editors, *Proceedings of the Natural Language Processing and Chinese Computing: Second CCF Conference, NLPCC 2013, Chongqing, China, November 15-19, 2013,*, pages 299–309, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. ISBN 978-3-642-41644-6. doi: 10.1007/978-3-642-41644-6_28. URL http://dx.doi.org/10.1007/978-3-642-41644-6_28.

[133] W. Xu, X. Liu, and Y. Gong. Document clustering based on Non-negative Matrix Factorization. In *Proceedings of the 26th annual international ACM*

SIGIR conference on Research and development in informaion retrieval, pages 267–273. ACM, 2003.

[134] D. Yajuan, C. Zhimin, W. Furu, Z. Ming, and H.-Y. Shum. Twitter topic summarization by ranking tweets using social influence and content quality. In *Proceedings of the 24th International Conference on Computational Linguistics*, pages 763–780, 2012.

[135] X. Yan, J. Guo, Y. Lan, and X. Cheng. A biterm topic model for short texts. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13, pages 1445–1456, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2035-1. doi: 10.1145/2488388.2488514. URL `http://doi.acm.org/10.1145/2488388.2488514`.

[136] X. Yan, J. Guo, S. Liu, X. Cheng, and Y. Wang. Learning topics in short texts by Non-negative Matrix Factorization on term correlation matrix. In *Proceedings of the SIAM International Conference on Data Mining (SIAM 2013)*, San Diego, California, USA, July 2013. SDM.

[137] J. Yang, J. McAuley, and J. Leskovec. Community detection in networks with node attributes. In *Proceedings of the 2013 IEEE 13th International Conference on Data Mining*, pages 1151–1156, Dec 2013. doi: 10.1109/ICDM.2013.167.

[138] L. Yang, T. Sun, M. Zhang, and Q. Mei. We know what@ you# tag: Does the dual role affect hashtag adoption? In *Proceedings of the 21st International Conference on World Wide Web (WWW 2012)*, pages 261–270, Lyon, France, April 2012. ACM.

[139] Z. Yang, Z. Yuan, and J. Laaksonen. Projective Non-negative Matrix Factorization with applications to facial image processing. *International Journal of Pattern Recognition and Artificial Intelligence*, 21(08):1353–1362, 2007.

[140] A. Yıldırım, S. Üsküdarlı, and A. Özgür. Identifying topics in microblogs using wikipedia. *PloS one*, 11(3):e0151885, 2016.

[141] M. Zhang, B. J. Jansen, and A. Chowdhury. Business engagement on Twitter: a path analysis. *Electronic Markets*, 21(3):161–175, 2011.

[142] Y. Zhao and G. Karypis. Criterion functions for document clustering: Experiments and analysis. Technical report, Citeseer, 2001.

[143] Y. Zhong, J. Yang, and R. Nugroho. Incorporating tie strength in robust social recommendation. In *Proceedings of the 4th IEEE International Congress on Big Data*, pages 63–70, New York, USA, July 2015. IEEE Services Computing Community.

[144] T. Zhou, H. Shan, A. Banerjee, and G. Sapiro. Kernelized probabilistic matrix factorization: Exploiting graphs and side information. In *Proceedings of the 2012 SIAM International Conference on Data Mining*, volume 12, pages 403–414, California, USA, 2012. SIAM.

[145] S. Zhu, K. Yu, Y. Chi, and Y. Gong. Combining content and link for classification using matrix factorization. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, pages 487–494, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-597-7. doi: 10.1145/1277741.1277825. URL `http://doi.acm.org/10.1145/1277741.1277825`.

[146] Y. Zuo, J. Zhao, and K. Xu. Word network topic model: a simple but general solution for short and imbalanced texts. *Knowledge and Information Systems*, 48(2):379–398, 2016. ISSN 0219-3116. doi: 10.1007/s10115-015-0882-z. URL `http://dx.doi.org/10.1007/s10115-015-0882-z`.