

國立臺灣大學管理學院資訊管理學系



碩士論文

Department of Information Management

College of Management

National Taiwan University

Master Thesis

應用時間演化主題多管道潛藏狄利克雷分配推薦主題標籤

Recommending Hashtags Using

Topics over Time Multiple Channel Latent Dirichlet Allocation

李健華

Chien-Hua Lee

指導教授：盧信銘 博士

Advisor: Hsin-Min Lu, Ph.D.

中華民國 103 年 7 月

July, 2014

## 誌謝



碩士班的兩年生涯即將結束，回想過去從台大工管系畢業進到資管所，一路走來受到家人、許多師長以及朋友們的幫助與鼓勵，才能讓我發掘自己對此領域的興趣，並從對程式語言一竅不通到有一定基礎，最後順利完成這篇碩士論文。

首先要感謝的是我的指導教授盧信銘博士。盧老師在我碩士班求學的階段中，指引我追求知識的方向；在我做研究的過程中，給予指導並幫助解決我所碰到的困難與問題，使我能朝向正確的道路前進。除了盧老師之外，也要感謝口試委員李昇暉老師和曹承礎老師，讓我從其他角度重新審視我的研究，幫助我完善我的論文。實驗室的朋友們也是我生活中不可或缺的一部份。宇泰學長與群融學弟在我撰寫論文時給予了許多幫助；日鑫學長在口試當天為我們加油打氣；承鑫和凱迪更是一路走來的兩位好戰友，不管是在課業上或是生活上，我們都互相幫助，一起成長。還要特別感謝女朋友芳如的體諒與包容，並且總是在我最需要幫助的時候，第一時間給予我支持，讓我有動力繼續前進。最後則是要謝謝每一位朋友與研究所的同學，因為有你們的加入，才能讓我的碩士生活如此豐富與精彩。

碩士班即將告一段落，再次感謝所有人的幫助與鼓勵，讓我能夠成長茁壯。希望在之後的生涯中，能夠不負學校給予我們的能力，並帶著大家的期待與祝福，發揮所長，回饋社會。

最後，謹以此文獻給我最摯愛的父母。

## 中文摘要



隨著社交網絡的盛行，有越來越多的使用者加入，其中所包含的資訊量更是迅速的成長。為了有效且快速的分類和搜尋推文(tweet)，推特(Twitter)的用戶使用主題標籤(hashtag)來標記並歸類推文。由於添加主題標籤不是一項自動化的程序，絕大部分的推文都沒有使用主題標籤，在我們的研究中更只有15%的推文有使用，大大的降低其價值。故本研究希望提出一個主題標籤的推薦系統，在使用者輸入完推文後，能自動產生一組合適的主題標籤以供選擇，提升主題標籤的覆蓋率。

本研究以主題模型(topic model)為基礎，加入時間群集(temporal clustering)的方法，提出時間演化主題多管道潛藏狄利克雷分配(Topic over Time Multiple Channel Latent Dirichlet Allocation，簡稱 TOT-MCLDA)。此模型根據可觀察的推文資訊，針對不同時間下的潛藏主題做分群，並預測適合的主題標籤。

本研究使用三年期的推特資料進行實驗，實驗結果證明 TOT-MCLDA 表現優於先前研究所提出的推薦系統，能顯著的提升推薦的準確率。此外，TOT-MCLDA 所建立之推文與主題標籤之間的關聯，也可作為基礎應用於其他相關研究上，增加可信度。

關鍵字：社交網絡、推特、主題標籤、推薦系統、主題模型

# Abstract



Along with the development of social network and the sustainable user growth, the explosion of contents provides tons of information. In order to efficiently and effectively classify tweets, users of Twitter can make use of hashtags to mark and categorize their tweets. However, most of the tweets do not contain hashtags. In addition, our research shows that there are only 15% of tweets contain hashtags, which greatly reduce the value of hashtags. Therefore, our research aims to develop a hashtag recommendation system to automatically provide hashtags according to the content of the tweet.

Our research mode is constructed based on Mixed Membership Model. We further extend the model by incorporating the temporal clustering effect and propose the result model, Topics over Time Multiple Channel Latent Dirichlet Allocation (TOT-MCLDA). The insight of our model is that the text words and hashtags from one tweet have the same latent topic condition factors. In addition, tweets posted in the same period of time have higher relevance. Hence, we can make use of the tweet contents to recommend hashtags by its latent topics. Experimental results on a 3-year Twitter dataset demonstrate that the proposed method can outperform some state-of-the-art methods.

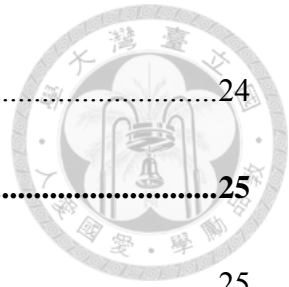
Keywords: Social Network, Twitter, Hashtags, Recommendation System, Topic Model

# Contents



誌謝 .....	i
中文摘要 .....	ii
Abstract.....	iii
Contents .....	iv
List of Figures.....	vi
List of Tables .....	viii
<b>Chapter 1 Introduction.....</b>	<b>1</b>
<b>Chapter 2 Literature Review .....</b>	<b>5</b>
2.1 Recommendation Systems.....	5
2.1.1 Non-personalized Recommendation Systems.....	5
2.1.2 Personalized Recommendation Systems.....	6
2.1.2.1 Memory-based Collaborative Filtering.....	7
2.1.2.2 Model-based Collaborative Filtering .....	8
2.1.2.3 Content-based approach .....	9
2.2 Hashtag Recommendation Systems.....	11
2.2.1 Naïve Bayes Method .....	11
2.2.2 Similarity Approach .....	13
2.2.3 Topic Model Based Method .....	17

2.3	Summary.....	24
<b>Chapter 3</b>	<b>Methodology.....</b>	<b>25</b>
3.1	TOT-MCLDA.....	25
3.1.1	Topics over Time.....	25
3.1.2	Mixed Membership Model.....	29
3.1.3	Topics over Time Multiple Channel Latent Dirichlet Allocation.....	31
3.2	Baseline Model.....	41
3.3	Metrics.....	42
<b>Chapter 4</b>	<b>Data Selection and Experimental Results.....</b>	<b>43</b>
4.1	Data Selection and Preprocess.....	43
4.2	Performance Evaluation.....	46
4.3	Parameter estimation.....	47
4.4	Experimental results of Twitter data.....	48
4.4.1	Analyses of Recommendation Lists.....	55
4.4.2	Analyses of Topic Distributions.....	68
<b>Chapter 5</b>	<b>Conclusions and Future Work.....</b>	<b>80</b>
	Reference.....	82
	Appendix A.....	86



# List of Figures



Figure 2.1.1: The amount of usage of three hashtags (#nfl, #nba, #mlb) over different time period.....	3
Figure 2.2.1: Graphical structure of LDA .....	18
Figure 3.1.1: Graphical structure of Topics over Time.....	26
Figure 3.1.2: Graphical structure of Mixed Membership Model .....	30
Figure 3.1.3: Graphical structure of Multiple Channel Latent Dirichlet Allocation .....	32
Figure 3.1.4: TOT-MCLDA.....	33
Figure 4.1.1: The distribution of text words' count.....	45
Figure 4.1.2: The distribution of hashtags' count.....	45
Figure 4.4.1: The histogram of hit rate on recommending top 10, 20, and 50 hashtags .	54
Figure 4.4.2: TOT-MCLDA topic 46 distributed over time (The fitted beta PDF is shown by the red line). .....	70
Figure 4.4.3: MCLDA topic 71 distributed over time (The fitted beta PDF is shown by the red line; Beta distribution is fit in a post-hoc fashion). .....	71
Figure 4.4.4: TOT-MCLDA topic 6 distributed over time (The fitted beta PDF is shown by the red line).....	74
Figure 4.4.5: MCLDA topic 96 distributed over time (The fitted beta PDF is shown by	

the red line; Beta distribution is fit in a post-hoc fashion). ..... 75

Figure 4.4.6: TOT-MCLDA topic 120 distributed over time (The fitted beta PDF is shown by the red line). ..... 78





# List of Tables



Table 3.1.1: Notation of variables of TOT-MCLDA .....	34
Table 3.3.1: An example of hit rate .....	42
Table 4.1.1: Basic statistics of the dataset .....	44
Table 4.4.1: Hit rate of each model on recommending top 10 hashtags (%).....	48
Table 4.4.2: Two tailed paired t-test of the hit rate difference between two models on recommending top 10 hashtags (%) .....	49
Table 4.4.3: Hit rate of each model on recommending top 20 hashtags (%).....	50
Table 4.4.4: Two tailed paired t-test of the hit rate difference between two models on recommending top 20 hashtags (%) .....	51
Table 4.4.5: Hit rate of each model on recommending top 50 hashtags (%).....	52
Table 4.4.6: Two tailed paired t-test of the hit rate difference between two models on recommending top 50 hashtags (%) .....	53
Table 4.4.7: Data of ID 4846085206 tweet .....	55
Table 4.4.8: SIM hashtags ranking of ID 4846085206 tweet.....	56
Table 4.4.9: LDA, MCLDA, TOT-MCLDA hashtags ranking of ID 4846085206 tweet	56
Table 4.4.10: Data of ID 91523345744543745 tweet .....	57
Table 4.4.11: SIM hashtags ranking of ID 91523345744543745 tweet.....	58



Table 4.4.12: LDA hashtags ranking of ID 91523345744543745 tweet.....	58
Table 4.4.13: MCLDA hashtags ranking of ID 91523345744543745 tweet.....	59
Table 4.4.14: TOT-MCLDA hashtags ranking of ID 91523345744543745 tweet .....	59
Table 4.4.15: Data of ID 55991762501644288 tweet .....	60
Table 4.4.16: SIM hashtags ranking of ID 55991762501644288 tweet.....	61
Table 4.4.17: LDA hashtags ranking of ID 55991762501644288 tweet.....	62
Table 4.4.18: MCLDA hashtags ranking of ID 55991762501644288 tweet.....	62
Table 4.4.19: TOT-MCLDA hashtags ranking of ID 55991762501644288 tweet .....	63
Table 4.4.20: Data of ID 92703923739164673 tweet .....	64
Table 4.4.21: SIM hashtags ranking of ID 92703923739164673 tweet.....	65
Table 4.4.22: LDA hashtags ranking of ID 92703923739164673 tweet.....	66
Table 4.4.23: MCLDA hashtags ranking of ID 92703923739164673 tweet.....	66
Table 4.4.24: TOT-MCLDA hashtags ranking of ID 92703923739164673 tweet .....	67
Table 4.4.25: TOT-MCLDA text word distribution of topic 46 sorted by probability ....	69
Table 4.4.26: TOT-MCLDA hashtag distribution of topic 46 sorted by probability .....	69
Table 4.4.27: MCLDA text word distribution of topic 71 sorted by probability.....	70
Table 4.4.28: MCLDA hashtag distribution of topic 71 sorted by probability.....	71
Table 4.4.29: TOT-MCLDA text word distribution of topic 6 sorted by probability .....	73
Table 4.4.30: TOT-MCLDA hashtag distribution of topic 6 sorted by probability .....	73

Table 4.4.31: MCLDA text word distribution of topic 96 sorted by probability.....74

Table 4.4.32: MCLDA hashtag distribution of topic 96 sorted by probability.....75

Table 4.4.33: TOT-MCLDA text word distribution of topic 120 sorted by probability ..77

Table 4.4.34: TOT-MCLDA hashtag distribution of topic 120 sorted by probability .....77



# Chapter 1 Introduction



As the increasing popularity of online social media venues, e.g., Facebook and Google+, users have created a huge amount of short text messages. Among these social networks, Twitter is one of the biggest and most popular microblogging websites. According to recent statistics, there are more than 135,000 new users joining Twitter every day. In addition, there are about 645 million active registered users posting 58 million tweets every day.

Discussions on microblogging websites can influence the accessibility and visibility of similar issues. In Twitter, user can follow other users or retweet other users' tweet. With the effect of network externality, an issue will be spread out rapidly. Such phenomenon intensifies the information circulation.

However, the increasing amount of user-generated content may cause the categorizing and searching more difficult. For this reason, Twitter presents the use of "hashtag". In Twitter, registered users are limited to post 140-character messages (i.e., tweets). Along with the general text, hashtags, which are arbitrary words prepended with hash symbol #, can also be inserted into each tweet. These hashtags can be seen as a categorization of their tweets. Several usages of hashtags have been proposed. Some users search for certain topics by the use of hashtags. Others use hashtags as keywords

to highlight or tag important issues. This makes the connection between relevant tweets much easier and helps the conversation between similar users more effective.



Although hashtags are useful, we find that there are only 15% of tweets contain at least one hashtag. Since the addition of hashtags is manual and custom, different user habit may result in different meaning. For example, users familiar with technology will use **#apple** as a representative of Apple Inc., while others treats it as the fruit. This vocabulary gap may cause confusion to users. Furthermore, the informality and non-systematicness of hashtags could also make the situation more complex. There are many synonymous hashtags being used for describing the same semantic information, e.g. **#mlb** and **#majorleaguebaseball**.

In order to deal with the above-mentioned problems, some hashtag recommendation systems have been proposed. Mazzia and Juett (2009) recommend suitable hashtags by considering the probability of each hashtag class based on Bayes' rule given the words in the tweet. Zangerle et al. (2011) use a term frequency - inverse document frequency (TF-IDF) based method to find similar tweets and recommend the top-ranked hashtags over three ranking approaches. Kywe et al. (2012) consider both user preference and tweet content to find a personalized set of hashtags. Godin et al. (2013) introduce a topic model based approach to recommend hashtags.

However, previous studies did not consider the trends in hashtags. Most of the



events discussed on Twitter are time-sensitive. For example, the regular season of National Football League (NFL) is from September to December, in Figure 2.1.1 we can see that the amount of usage of **#nfl** gradually rises from the end of August, to a peak at December, and finally drop at January of the following year. This implies that the meaning and representation of hashtags might vary over time. Therefore, it is important to consider time feature while designing models in order to capture the implicit effects.

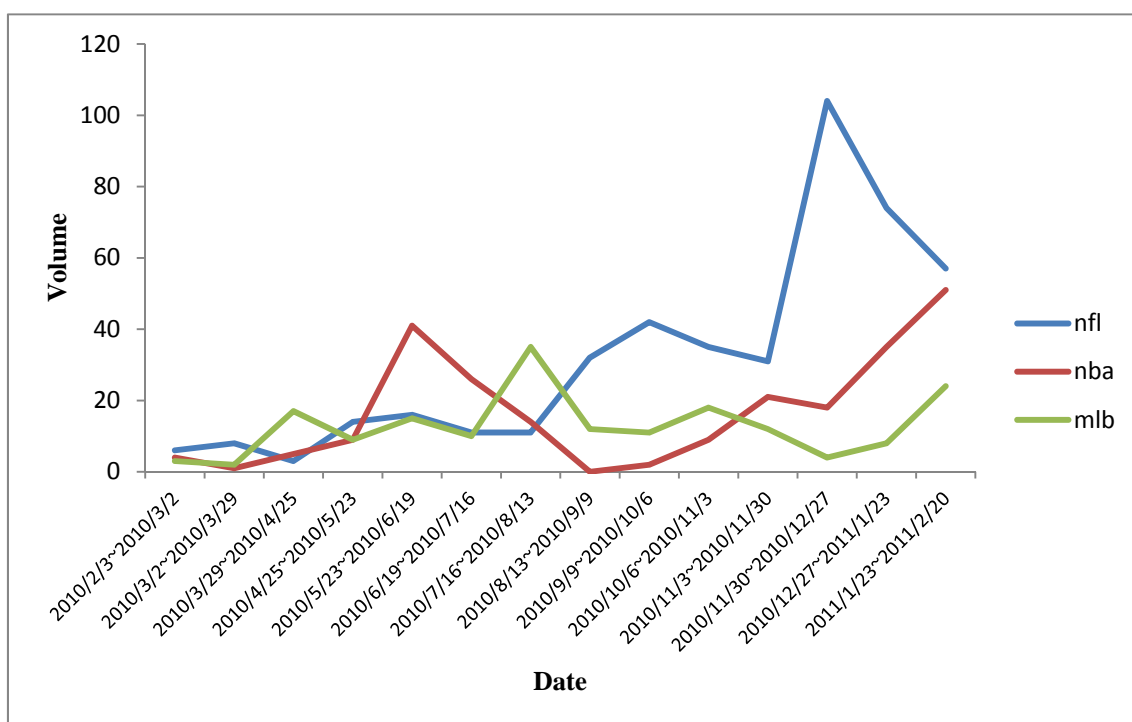



Figure 2.1.1: The amount of usage of three hashtags (**#nfl**, **#nba**, **#mlb**) over different time period.



To solve the problem, we provide Topics over Time Multiple Channel Latent Dirichlet Allocation (TOT-MCLDA), a mixture model aggregating both Multiple Channel Latent Dirichlet Allocation (MCLDA) and Topics over Time (TOT), to recommend a set of hashtags based on the underlying topics of the given tweets. This approach not only incorporates the consideration of time feature, but also provides opportunities for adding different types of context to enrich the input information and strengthen the model. We aim to use the latent topics of text and hashtags to recommend suitable hashtags to newly generated tweets.

We apply our method on a 3-year Twitter data to generate hashtags for new tweets. The baseline models we used are SIM, LDA, and MCLDA. The performance is measured through 10-fold cross-validation. The metric we use is hit rate. The experiments show that TOT-MCLDA has a hit rate of 27.625%, which significantly outperforms other three methods. We further analyze the recommendations given by each method to investigate the differences.

The thesis includes 5 sections. First, we review some related work in chapter 2. We propose our method in chapter 3. Chapter 4 presents the dataset information, the experiment design, and the results. Chapter 5 concludes.

## Chapter 2 Literature Review



Our study is related to the research field of hashtag recommendation systems and topic models. This chapter summarizes the related literatures.

### 2.1 Recommendation Systems

Recommendation systems aim to predict the preference of a user towards items or objects that have never been considered before (Herlocker et al., 2004). There are two types of recommendation systems, non-personalized recommendation systems and personalized recommendation systems. Further introductions are shown as follow.

#### 2.1.1 Non-personalized Recommendation Systems

Non-personalized recommendation systems give out a list of items ranked by averaging other users' rating or opinion. Since the recommendations are independent to the user, the recommended item list is identical for each user.

Non-personalized recommendation systems are widely used in several social networking websites because it can give out an overall opinion of items or products of all the users without changing the rank by every user; in addition, the analysis process is simple and the collection of data is easy. These methods usually recommend items by



popularity, e.g., click through rate (CTR) of links or purchase rate of products. For example, MTV (<http://www.mtv.com/music/>) recommends top 10 most popular artists to users based on the total number of viewing times of their music videos.



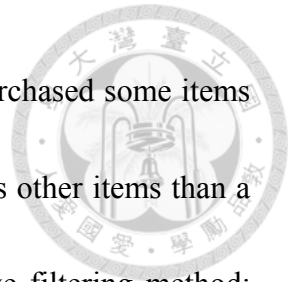
Uddin et al. (2011) developed a tag recommendation system based on the use of Adapted PageRank (Hotho et al., 2006). The underlying principle is that a post assigned with important tags by important users becomes important as well. Since this method generates ranking list without utilizing any personal information, it is a non-personalized tag recommendation system. Since non-personalized recommendation systems give a fixed list of recommendations for all users, it might not appeal to everyone (Anderson and Hiralall, 2011).

### 2.1.2 Personalized Recommendation Systems

Compared to non-personalized recommendation systems, personalized recommendation systems consider the preferences of an individual user to generate a personalized recommendation list. Personalized recommendation systems are mainly separated into two categories, collaborative filtering and content-based approach (Kywe et al., 2012).

Collaborative filtering is a method makes use of users' habit, experience, or preference to recommend useful items or information to users. It assumes that user with

similar preference will rate or buy items similarly, e.g., if user A purchased some items same as user B previously did, user A is more likely to buy user B's other items than a randomly picked user. There are two general types of collaborative filtering method: memory-based and model-based.



### 2.1.2.1 Memory-based Collaborative Filtering

Memory-based collaborative filtering is a method that searches into the database to find users with similar features or attributes. There are two general classes of collaborative filtering, user-based collaborative filtering and item-based collaborative filtering.

User-based collaborative filtering calculates the similarity between two users and recommends items that are rated good or commonly purchased by the most similar users. A similar user is defined as those who shares similar preference with the target user. Cosine similarity (Salton and McGill, 1983) and Pearson correlation coefficient (Resnick et al., 1994) are popular similarity-based approaches. This type of method was firstly developed by Goldberg et al. (1992). It was designed for the purpose of solving the information overloading problems in Xerox's Palo Alto Research Center. It filters out e-mail that is not related to the user. However, user-based collaborative filtering does not scale well when user number gets larger because the computation cost of



searching for similar users becomes too high.

In contrast to user-based approach, item-based collaborative filtering aims to calculate the correlation between two items. Similar items are defined as those that are co-rated high or often co-purchased by users. Then the most similar items correspond to the item purchased by the target user are recommended. Item-based approach overcomes the scalability problem in user-based approach. In Sarwar et al.'s (2001) study, as the model size getting larger, only a slight increase on the run time of recommendation system.

#### 2.1.2.2 Model-based Collaborative Filtering

Since some recommendation systems generate recommendations on the basis of large datasets, memory-based collaborative filtering is not always scalable and efficient. Model-based collaborative filtering is designed to build a model based on the dataset. This method extracts information from the dataset and develops a model for further recommendation without using the whole dataset every time. Some model-based collaborative filtering algorithms have been proposed: Breese et al. (1988) developed a clustering model based on the use of naïve Bayes to classify similar users into the same group; Chen and George (1999) designed a Bayesian network model to construct the relationship between each items, and the neighboring items are recommended; Koren et

al. (2009) use matrix factorization to reduce the dimensionality of users and items, and generate a rank based on latent factors.

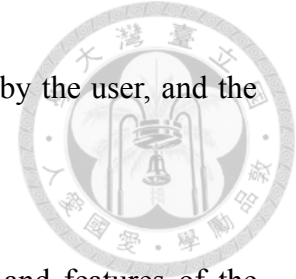


Although collaborative filtering performs quite well in lots of circumstances, it still faces some problems. The size of the dataset affects the performance of the recommendations directly. Data sparsity may cause the problem of cold start, which means new user or new item has no sufficient factors to compare and link with similar users or items in the dataset. Besides, large dataset may reduce the efficiency of memory-based approaches. In addition, user who is not consistently agreed or disagree with any group of users will make the recommendation systems nearly impossible to generate recommendations, i.e., grey sheep problem.

### 2.1.2.3 Content-based approach

Content-based recommendation system was first proposed by Balabanović and Shoham (1997). It assumes that items with similar features will be rated or purchased by users with similar preferences. Compared to collaborative filtering, content-based approach depends heavily on the item features and user profile that contains the information related to those features. A user profile of preferences is built to express the type of item the user likes, which is usually estimated based on the transactional data or rating history of the user. Therefore, content-based recommendation system will

compare candidate items with items previously rated or purchased by the user, and the most similar items are recommended.



Since content-based approach makes use of the descriptions and features of the items, it performs well if the items can be represented as a set of features properly.

Therefore, the quality and suitability of the features are important.

However, if the features of items are not available, they will need to be added manually. Besides, it cannot recommend new items different from the classes of the user preference if the user does not involve in new classes. Same as collaborative filtering, content-based approach also faces the problem of cold start while new user joins in.



## 2.2 Hashtag Recommendation Systems

Although lots of recommendation systems developed for social networks have been proposed, only a few studies focus on the problem of hashtag recommendation. This provides a huge space for improvement and future development.

### 2.2.1 Naïve Bayes Method

Mazzia et al. (2009) developed a naïve Bayes method that compares all hashtags by calculating their probabilities according to the text. It assumes each word in a tweet is independent.

In this method, given a target tweet  $d_k = \{w_{k,1}, \dots, w_{k,N}\}$ , they first estimate the maximum a posteriori probability of each hashtag by

$$P(h_i | w_{k,1}, \dots, w_{k,N}) = \frac{P(h_i)P(w_{k,1}|h_i)P(w_{k,2}|h_i) \dots P(w_{k,n}|h_i) \dots P(w_{k,N}|h_i)}{P(w_{k,1}, \dots, w_{k,N})}$$

Equation 2.2.1.1

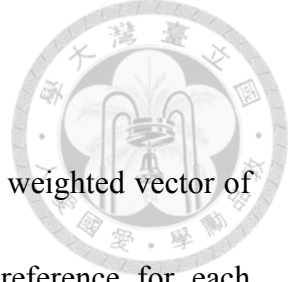
where  $h_i$  represents the  $i^{\text{th}}$  hashtag;  $w_{k,1}, \dots, w_{k,N}$  represent the text in the target tweet;  $P(h_i | w_{k,1}, \dots, w_{k,N})$  is the probability of using hashtag  $h_i$  given the text in the target tweet;  $P(h_i)$  is the ratio of the number of times hashtag  $h_i$  is used in the whole dataset to the total number of hashtags used in the whole dataset;  $P(w_{k,n} | h_i)$  is the

probability of using  $w_{k,n}$  given hashtag  $h_i$ , this is calculated from the existing dataset of tweets.



Then they rank all the hashtags by their probability and recommend the top-K hashtags with highest probability to the target tweet, where K denotes the size of recommended hashtags that will be presented.

This method is a type of model-based recommendation systems, it takes advantage of scalability and efficiency.



## 2.2.2 Similarity Approach

This approach makes use of TF-IDF to represent a tweet into a weighted vector of words  $d_k = \{e_{k,1}, \dots, e_{k,|W|}\}$  in a word vocabulary  $W$  and the preference for each hashtag of a user into a weighted vector of hashtags  $u_j = \{e_{j,1}, \dots, e_{j,|H|}\}$  in a hashtag dictionary  $H$ , where

$$e_{k,i} = \frac{\text{Freq}_{k,i}}{\text{Max}_k} * \log\left(\frac{|D|}{n_i}\right)$$

Equation 2.2.2.1


$$e_{j,i} = \frac{\text{Freq}_{j,i}}{\text{Max}_j} * \log\left(\frac{|U|}{n_i}\right)$$

Equation 2.2.2.2

where  $\text{Freq}_{k,i}$  is the frequency of word  $w_{k,i}$  in target tweet  $d_k$ ;  $\text{Max}_k$  is the total number of text in the target tweet  $d_k$ ;  $|D|$  is the total number of tweets in the whole dataset;  $n_i$  is the number of tweets in which word  $w_i$  appears;  $\text{Freq}_{j,i}$  is the frequency of hashtag  $h_i$  used by user  $u_j$ ;  $\text{Max}_j$  is the total number of hashtags used by user  $u_j$ ;  $|U|$  is the total number of users;  $n_i$  is the number of users who use hashtag  $h_i$  before.

Zangerle et al. (2011) proposed a hashtag recommendation system that





recommends hashtags in the similar tweets. It assumes that similar tweets hold similar meanings and hashtag distributions. The similarity score between two tweets is measured by the sum of all TF-IDF of all words occurring within the target tweet. The more the text of the target tweet is matched, the higher the TF-IDF score is. The final set of similar tweets is consisting of those tweets which have the highest score.

Then they extract all the hashtags in the final set of similar tweets. Since these hashtags have to be ranked, they evaluated three ranking methods:

*OverallPopularity*: This method ranks hashtags by considering their number of occurrence in the whole dataset.

*RecommendationPopularityRank*: This method ranks hashtags by considering their number of occurrence in the final set of similar tweets.

*SimilarityRank*: This method ranks hashtags by ranking the retrieved set of similar tweets using the TF-IDF score. The hashtags contained in the most similar tweet are recommended.

Their experiments show that *SimilarityRank* performs the best in recommending five hashtags.

In order to take into account personal preferences when recommending hashtags, Kywe et al. (2012) introduced a personalized hashtag recommendation system utilizing both tweet content and user preference of hashtags. The similarity score between the



target tweet  $d_k$  and another tweet  $d_j$  is calculated by cosine similarity, where

$$\text{Similarity}(d_k, d_j) = \frac{d_k \cdot d_j}{\|d_k\| \cdot \|d_j\|}$$

Equation 2.2.2.3

The candidate hashtags are those contained in the top-X most similar tweets, which is denoted by *HashtagsOfTweets*. Furthermore, the similarity score between the target user  $u_j$  and another user  $u_k$  is calculated by cosine similarity, where

$$\text{Similarity}(u_j, u_k) = \frac{u_j \cdot u_k}{\|u_j\| \cdot \|u_k\|}$$

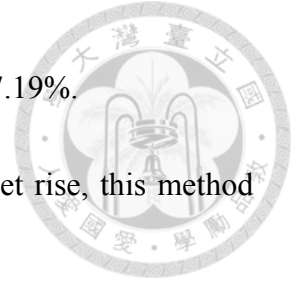
Equation 2.2.2.4

The candidate hashtags are those used by the top-Y most similar users, which is denoted by *HashtagsOfUsers*.

The candidate hashtags to be recommended are those that are in the union of *HashtagsOfTweets* and *HashtagsOfUsers*. Then these candidate hashtags are ranked by frequency, which is calculated by adding the number of times the hashtag used in top-X similar tweets with the number of times it used by top-Y similar users. Last, the top ranked hashtags are recommended to the target user and the target tweet. The results of

experiments show the hit rate of top ten recommended hashtags is 37.19%.

However, as the total amount of tweets and users in the dataset rise, this method will face the problem of scalability.



### 2.2.3 Topic Model Based Method

Godin et al. (2013) proposed a topic model based method making use of Latent Dirichlet Allocation (LDA) for general hashtag recommendation. The recommended hashtags are given by sampling the top ranked words that resemble the general topic of the target tweet based on the probability.

Latent Dirichlet Allocation was first proposed by Blei et al. (2003), it is a generative model. It assumes that document is composed of several topics, which means that each document has its own topic distribution; in addition, each topic is a probability distribution of words. Therefore, the content of document is decided by these two distribution. The graphical structure of LDA is shown in Figure 2.2.1.



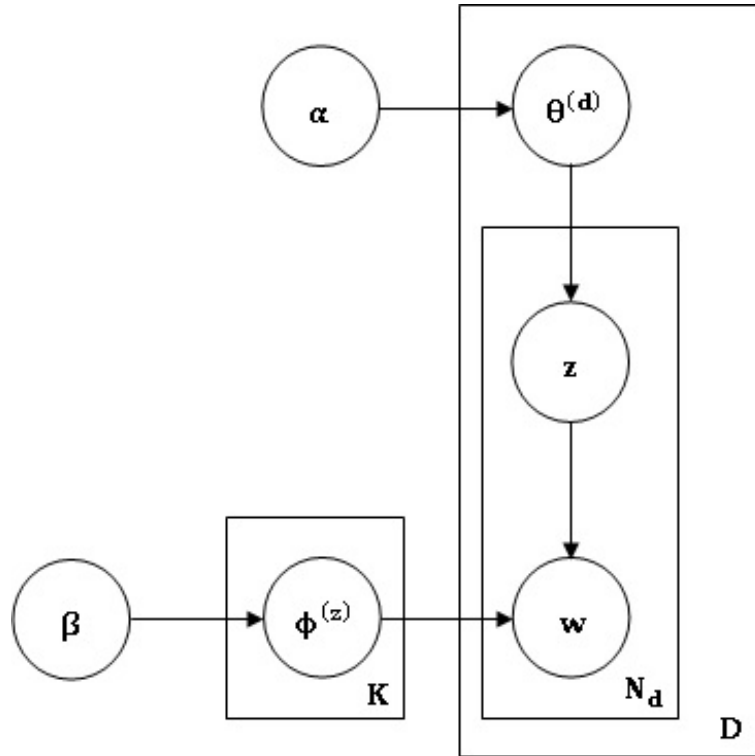


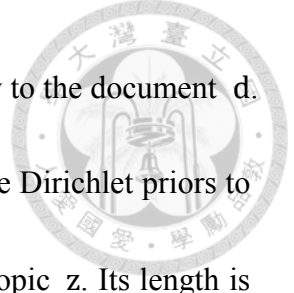
Figure 2.2.1: Graphical structure of LDA

If we represent the words in the corpus as a vector  $W$  and the corresponding latent topic as  $Z$ , then the generative probability can be presented as

$$P(W, Z, \theta, \phi | \alpha, \beta) = \prod_{k=1}^K P(\phi^{(k)} | \beta) \prod_{d=1}^D P(\theta^{(d)} | \alpha) \prod_{i=1}^{N_d} P(z_i^{(d)} | \theta^{(d)}) P(w_i^{(d)} | \phi^{(z_i^{(d)})})$$

Equation 2.2.3.1

This model assumes that the prior probability distribution of documents over topics and topics over words are both Dirichlet distributions. The vector  $\alpha$  is the Dirichlet



priors to  $\theta^{(d)}$ , which is the multinomial distribution of topics specify to the document  $d$ .

Its length is the number of topics, denoted by  $K$ . The vector  $\beta$  is the Dirichlet priors to

$\phi^{(z)}$ , which is the multinomial distribution of words specify to the topic  $z$ . Its length is

the number of unique words, denoted by  $V$ . The generation of a document involves in

the following process:

- (1) For each topic  $k$ , draw a distribution over words  $\phi^{(k)} \sim \text{Dir}(\beta)$
- (2) For each document  $d$ ,
  - (a) Draw a vector of topic proportions  $\theta^{(d)} \sim \text{Dir}(\alpha)$
  - (b) For each word,
    - (i) Draw a topic assignment  $z_i^{(d)} \sim \text{Mult}(\theta^{(d)})$ ,  $z_i^{(d)} \in \{1, \dots, K\}$
    - (ii) Draw a word  $w_i^{(d)} \sim \text{Mult}(\phi^{(z_i^{(d)})})$ ,  $w_i^{(d)} \in \{1, \dots, V\}$

The estimation of  $\theta$  and  $\phi$  is involved with latent variable, so the distribution is intractable to compute. In view of the problem, some approximate inferences are proposed, e.g., Laplace Approximation, Variational Approximation, and Markov chain Monte Carlo (Blei et al., 2003).

Since the Dirichlet distribution is the conjugate prior of the multinomial distribution, it is allowed to compute the joint distribution  $P(W, Z)$  by integrating out  $\theta$  and  $\phi$  (Griffiths and Steyvers, 2004). Given  $P(W, Z) = P(W|Z)P(Z)$ , since  $\theta$  and  $\phi$  only appear in the first and second terms respectively, the integral of  $\theta$  and  $\phi$  is separable. We first integrate out  $\phi$  from  $P(W|Z)$  and get



$$P(W|Z) = \left( \frac{\Gamma(V\beta)}{\Gamma(\beta)^V} \right)^K \prod_{k=1}^K \frac{\prod_{v=1}^V \Gamma(n_{(\cdot),(k)}^{(\cdot),(v)} + \beta)}{\Gamma(n_{(\cdot),(k)}^{(\cdot),(k)} + V\beta)}$$

Equation 2.2.3.2

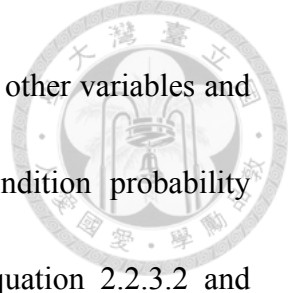
where  $n_{(\cdot),(k)}^{(\cdot),(v)}$  represents the number of times word  $v$  has been assigned to topic  $k$  in the vector of assignments  $Z$ ;  $n_{(\cdot),(k)}^{(\cdot),(k)}$  represents the number of times topic  $k$  appears in the vector of assignments  $Z$ . Then we integrate out  $\theta$  from  $P(Z)$  and get

$$P(Z) = \left( \frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K} \right)^D \prod_{d=1}^D \frac{\prod_{k=1}^K \Gamma(n_{(\cdot),(k)}^{(d),(k)} + \alpha)}{\Gamma(n_{(\cdot),(k)}^{(d),(k)} + K\alpha)}$$

Equation 2.2.3.3

where  $n_{(\cdot),(k)}^{(d),(k)}$  represents the number of times topic  $k$  appears in document  $d$ ;  $n_{(\cdot),(k)}^{(d),(k)}$  represents the number of words in document  $d$ .

Griffiths and Steyvers (2004) further use Markov chain Monte Carlo to infer  $P(Z|W)$ . Markov chain is constructed by sampling from the variables  $Z$ , and it will converge to the target distribution after several transitions (Gilks et al., 1996; Newman and Barkema, 1999; Liu, 2001). Gibbs Sampling (Geman and Geman, 1984) is used as the sampling method. The next state is reached by sequentially sampling all variables



from their distribution when conditioned on the current values of all other variables and the data. Therefore, in order to apply Gibbs Sampling, the condition probability  $P(Z_i|Z_{-i}, W)$  is needed, which can be obtained by combining equation 2.2.3.2 and equation 2.2.3.3:

$$P(Z_i = k|Z_{-i}, W) \propto \frac{n_{(-i),(k)}^{(v)} + \beta}{n_{(-i),(k)}^{(v)} + V\beta} \frac{n_{(-i),(k)}^{(d),(\cdot)} + \alpha}{n_{(-i),(\cdot)}^{(d),(\cdot)} + K\alpha}$$

Equation 2.2.3.4

where  $n_{(-i),(k)}^{(v)}$  represents the number of times word  $v$  has been assigned to topic  $k$  in the vector of assignments  $Z$  without considering current position  $i$ ;  $n_{(-i),(k)}^{(v)}$  represents the number of times topic  $k$  appears in the vector of assignments  $Z$  without considering current position  $i$ ;  $n_{(-i),(k)}^{(d),(\cdot)}$  represents the number of times topic  $k$  appears in document  $d$  without considering current position  $i$ ;  $n_{(-i),(\cdot)}^{(d),(\cdot)}$  represents the number of words in document  $d$  without considering current position  $i$ .

Given a set of recorded sweeps, the estimation of  $\theta$  and  $\phi$  can be computed via:

$$\hat{\phi}_v^{(k)} = \frac{n_{(\cdot),(k)}^{(v)} + \beta}{n_{(\cdot),(k)}^{(v)} + V\beta}$$

Equation 2.2.3.5





$$\hat{\theta}^{(d)}_k = \frac{n_{(\cdot),(k)}^{(d),(\cdot)} + \alpha}{n_{(\cdot),(\cdot)}^{(d),(\cdot)} + K\alpha}$$

Equation 2.2.3.6

Based on these two estimations, we can construct the probability distribution of documents over topics and topics over words.

In Godin et al.'s (2013) study, the data collection is a collection of tweets. A document corresponds to one tweet. The words of the document are both text words and hashtags of the tweet. After training the model, we can get the topic distribution  $\hat{\theta}^{(d)}$  of each tweet  $d$ , where  $\hat{\theta}^{(d)} = \{\hat{\theta}^{(d)}_1, \hat{\theta}^{(d)}_2, \dots, \hat{\theta}^{(d)}_K\}$ ,  $d = 1, 2, \dots, D$ ; the word distribution  $\hat{\phi}^{(k)}$  of each topic  $k$ , where  $\hat{\phi}^{(k)} = \{\hat{\phi}^{(k)}_1, \hat{\phi}^{(k)}_2, \dots, \hat{\phi}^{(k)}_V\}$ ,  $k = 1, 2, \dots, K$ .

Given a target tweet  $\tilde{d}$ , in order to determine the topic distribution of the tweet, they again made use of Collapsed Gibbs Sampling and the trained model. The conditional distribution is now equal to:

$$P(\tilde{z}_i = k | \tilde{z}_{-i}, \tilde{w}, z, w) \propto \frac{n_{(-i),(k)}^{(\tilde{d}),(\tilde{w}_i)} + n_{(\cdot),(k)}^{(\cdot),(\tilde{w}_i)} + \beta \frac{n_{(-i),(\cdot)}^{(\tilde{d}),(\cdot)} + \alpha}{n_{(-i),(\cdot)}^{(\tilde{d}),(\cdot)} + n_{(\cdot),(\cdot)}^{(\cdot),(\cdot)} + V\beta \frac{n_{(-i),(\cdot)}^{(\tilde{d}),(\cdot)} + K\alpha}$$

Equation 2.2.3.7



where  $n_{(-i),(k)}^{(\tilde{d}),(\tilde{w}_i)}$  represents the number of times word  $\tilde{w}_i$  has been assigned to topic  $k$  in the vector of assignments  $Z_{\tilde{d}}$  without considering current position  $i$ ;  $n_{(-i),(k)}^{(\tilde{d}),(\cdot)}$  represents the number of times topic  $k$  appears in the vector of assignments  $Z_{\tilde{d}}$  without considering current position  $i$ . Given a set of recorded sweeps, the estimation of  $\hat{\theta}_{\cdot k}^{(\tilde{d})}$  can be computed via:

$$\hat{\theta}_{\cdot k}^{(\tilde{d})} = \frac{\alpha + n_{(\cdot),(k)}^{(\tilde{d}),(\cdot)}}{K\alpha + n_{(\cdot),(\cdot)}^{(\tilde{d}),(\cdot)}}$$

Equation 2.2.3.8

Now the topic distribution  $\hat{\theta}^{(\tilde{d})}$  of the target tweet  $\tilde{d}$  is obtained, where  $\hat{\theta}^{(\tilde{d})} = \{\hat{\theta}_{\cdot 1}^{(\tilde{d})}, \hat{\theta}_{\cdot 2}^{(\tilde{d})}, \dots, \hat{\theta}_{\cdot K}^{(\tilde{d})}\}$ .

According to the number of hashtags they want to recommend, they sample through the topic distribution  $\hat{\theta}^{(\tilde{d})}$  of the target tweet  $\tilde{d}$  for a topic  $k$  and select a top word from  $\hat{\phi}^{(k)}$  in ranked order. The results of experiments show that the hit rate of top five recommend hashtags is 80%.

## 2.3 Summary

All the hashtag recommendation systems previously mentioned show reasonable performance. However, the patterns in a large dataset collected over an abundant of time are usually dynamic. The patterns present in the early part of the collection are not in effect later. Since the above mentioned method construct models without considering the feature of timestamp, they cannot properly distinguish the latent difference of tweets in different time period. Besides, the data and information in tweets are usually sparse and complicated. Therefore, our research aim to develop a flexible temporal clustering method which not only consider time feature to deal with the problem of polysemous words in different time period, but also give a space for incorporation of more features' information in tweets.



## Chapter 3 Methodology



In this chapter, we will first introduce the proposed model in detail. Next, we will summarize the baseline models, metrics, and the experimental design.

### 3.1 TOT-MCLDA

Our research model is similar to the Mixed Membership Model (Erosheva et al, 2004). We further extend the model by incorporating the temporal clustering effect proposed by Wang and McCallum (2006). The result of model, Topics over Time Multiple Channel Latent Dirichlet Allocation (TOT-MCLDA), is capable of identifying temporal clustering effects in textual data sources that contains more than one type of contexts.

#### 3.1.1 Topics over Time

Wang and McCallum (2006) proposed an extension model of LDA, i.e., Topics over Time (TOT), which explicitly models time jointly with word co-occurrence patterns. It assumes that the distribution over topics is influenced by both word co-occurrences and the document's timestamp. The graphical structure of TOT is shown

in Figure 3.1.1.

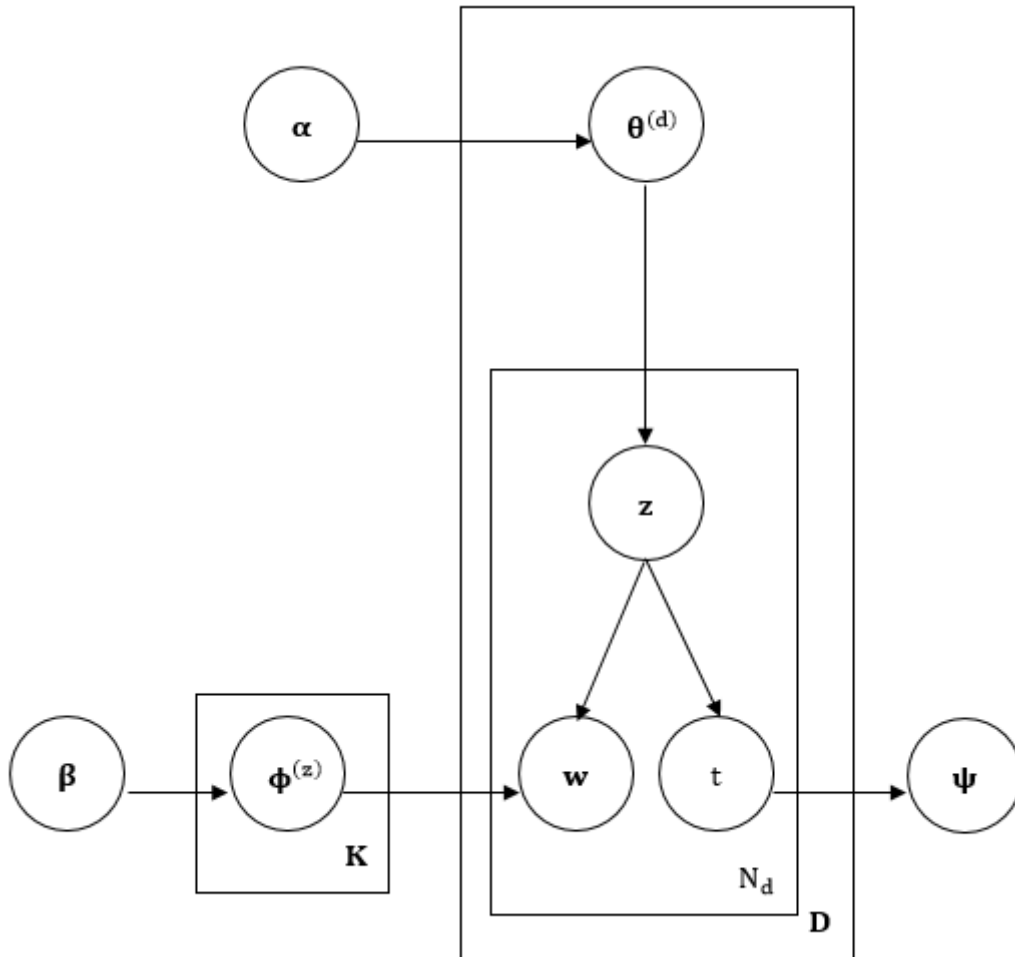
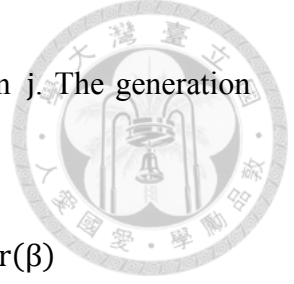


Figure 3.1.1: Graphical structure of Topics over Time

The vector  $\alpha$  is the Dirichlet priors to  $\theta^{(d)}$ , which is the multinomial distribution of topics specify to the document  $d$ . Its length is the number of topics, denoted by  $K$ . The vector  $\beta$  is the Dirichlet priors to  $\phi^{(z)}$ , which is the multinomial distribution of words specify to the topic  $z$ . Its length is the number of unique words, denoted by  $V$ .



The scalar  $t_j$  is the timestamp associated with word  $w_j$  at position  $j$ . The generation

of a document involves in the following process:

- (1) For each topic  $k$ , draw a distribution over words  $\phi^{(k)} \sim \text{Dir}(\beta)$
- (2) For each document  $d$ ,
  - (a) Draw a vector of topic proportions  $\theta^{(d)} \sim \text{Dir}(\alpha)$
  - (b) For each word,
    - (i) Draw a topic assignment  $z_i^{(d)} \sim \text{Mult}(\theta^{(d)})$ ,  $z_i^{(d)} \in \{1, \dots, K\}$
    - (ii) Draw a word  $w_i^{(d)} \sim \text{Mult}(\phi^{(z_i^{(d)})})$ ,  $w_i^{(d)} \in \{1, \dots, V\}$
    - (iii) Draw a timestamp  $t_i^{(d)} \sim \text{Beta}(\psi^{(z_i^{(d)})})$

The time range of the data used for parameter estimation in this model is first normalized to a range from 0 to 1 in order to employ the Beta distribution. The model can be completed by inferring the posterior probability  $P(Z|W, T)$ . Same as the inference procedure mentioned above in LDA, this model makes use of Markov chain Monte Carlo and Collapsed Gibbs Sampling. The conditional probability is presented:

$$P(Z_i = k | Z_{-i}, W, T) \propto \frac{n_{(-i),(k)}^{(\cdot),(v)} + \beta \quad n_{(-i),(k)}^{(d),(v)} + \alpha \quad t_i^{\psi_{k1}-1} (1 - t_i)^{\psi_{k2}-1}}{n_{(-i),(k)}^{(\cdot),(v)} + V\beta \quad n_{(-i),(k)}^{(d),(v)} + K\alpha \quad B(\psi_{k1}, \psi_{k2})}$$

Equation 3.1.1.1

where  $n_{(-i),(k)}^{(\cdot),(v)}$  represents the number of times word  $v$  has been assigned to topic  $k$  in the vector of assignments  $Z$  without considering current position  $i$ ;  $n_{(-i),(k)}^{(d),(v)}$

represents the number of times topic  $k$  appears in the vector of assignments  $Z$  without considering current position  $i$ ;  $n_{(-i),(k)}^{(d),(.)}$  represents the number of times topic  $k$  appears in document  $d$  without considering current position  $i$ ;  $n_{(-i),(.)}^{(d),(.)}$  represents the number of words in document  $d$  without considering current position  $i$ ;  $t_i$  represents the timestamp of the current position  $i$ . In addition,  $\Psi$  is updated after each sweep of Collapsed Gibbs Sampling by the method of moments:

$$\psi_{k1} = \bar{t}_k \left( \frac{\bar{t}_k(1 - \bar{t}_k)}{s_k^2} - 1 \right)$$

Equation 3.1.1.2

$$\psi_{k2} = (1 - \bar{t}_k) \left( \frac{\bar{t}_k(1 - \bar{t}_k)}{s_k^2} - 1 \right)$$

Equation 3.1.1.3

where  $\bar{t}_k$  and  $s_k^2$  indicate the sample mean and the biased sample variance of the timestamps belonging to topic  $k$  respectively.

Given a set of recorded sweeps, the estimation of  $\theta$  and  $\phi$  can be computed via:



$$\hat{\phi}_{\nu}^{(k)} = \frac{n_{(\cdot),(\nu)}^{(\cdot),(k)} + \beta}{n_{(\cdot),(\nu)}^{(\cdot),(k)} + V\beta}$$

Equation 3.1.1.4

$$\hat{\theta}_k^{(d)} = \frac{n_{(\cdot),(\nu)}^{(d),(\cdot)} + \alpha}{n_{(\cdot),(\nu)}^{(d),(\cdot)} + K\alpha}$$

Equation 3.1.1.5

Based on these two estimations, we can construct the probability distribution of documents over topics and topics over words.

### 3.1.2 Mixed Membership Model

Mixed Membership Model (Erosheva et al., 2004) is an extension model of LDA that take into account both feature of words and research paper citations in the document to capture the notion that documents that share the same hyperlinks and same words, tend to be on the same topic. The graphical structure of Mixed Membership Model is shown in Figure 3.1.2.



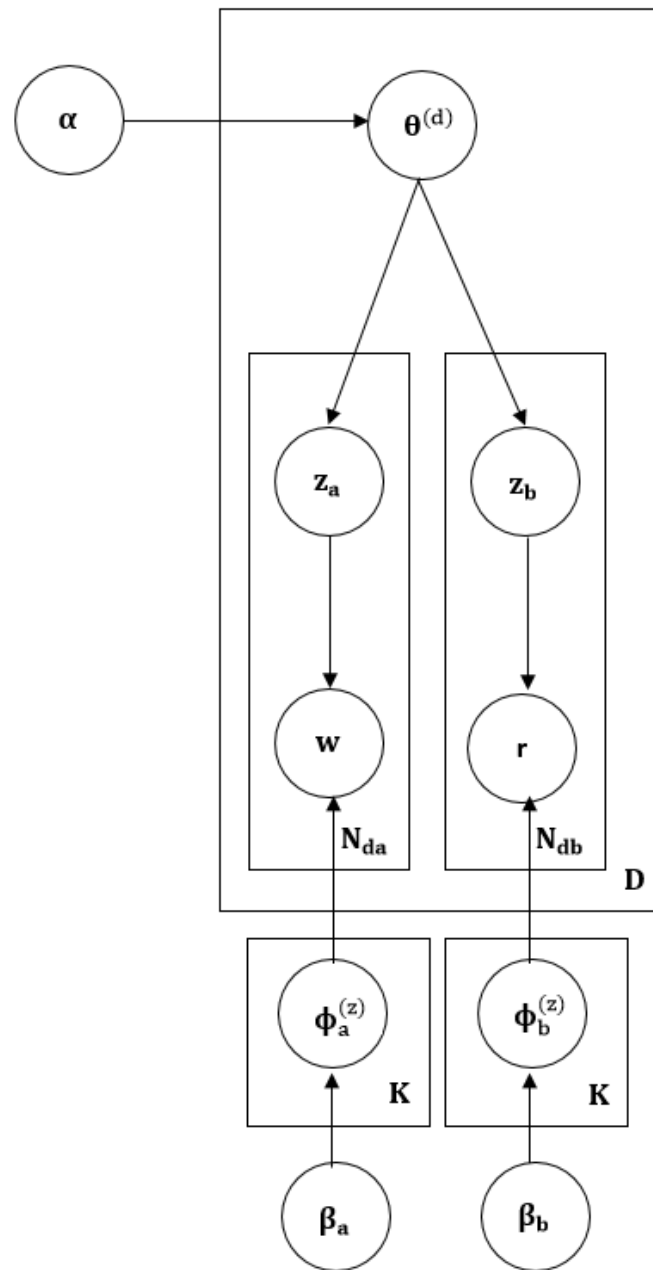
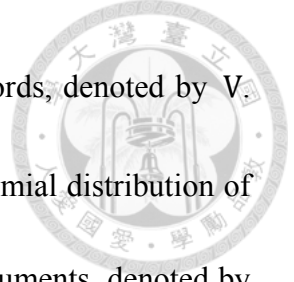


Figure 3.1.2: Graphical structure of Mixed Membership Model

The vector  $\alpha$  is the Dirichlet priors to  $\theta^{(d)}$ , which is the multinomial distribution of topics specify to the document  $d$ . Its length is the number of topics, denoted by  $K$ .

The vector  $\beta_a$  is the Dirichlet priors to  $\phi_a^{(z)}$ , which is the multinomial distribution of



words specify to the topic  $z$ . Its length is the number of unique words, denoted by  $V$ . The vector  $\beta_b$  is the Dirichlet priors to  $\phi_b^{(z)}$ , which is the multinomial distribution of citations specify to the topic  $z$ . Its length is the total number of documents, denoted by

[D]. The generation of a document involves in the following process:

- (1) For each topic  $k$ ,
  - (a) Draw a distribution over words  $\phi_a^{(k)} \sim \text{Dir}(\beta_a)$
  - (b) Draw a distribution over citations  $\phi_b^{(k)} \sim \text{Dir}(\beta_b)$
- (2) For each document  $d$ ,
  - (a) Draw a vector of topic proportions  $\theta^{(d)} \sim \text{Dir}(\alpha)$
  - (b) For each word,
    - (i) Draw a topic assignment  $z_i^{(d)} \sim \text{Mult}(\theta^{(d)})$ ,  $z_i^{(d)} \in \{1, \dots, K\}$
    - (ii) Draw a word  $w_i^{(d)} \sim \text{Mult}\left(\phi_a^{(z_i^{(d)})}\right)$ ,  $w_i^{(d)} \in \{1, \dots, V\}$
  - (c) For each citation,
    - (i) Draw a topic assignment  $z_i^{(d)} \sim \text{Mult}(\theta^{(d)})$ ,  $z_i^{(d)} \in \{1, \dots, K\}$
    - (ii) Draw a citation  $r_i^{(d)} \sim \text{Mult}\left(\phi_b^{(z_i^{(d)})}\right)$ ,  $r_i^{(d)} \in \{1, \dots, |D|\}$

### 3.1.3 Topics over Time Multiple Channel Latent Dirichlet Allocation

Based on the structure of Mixed Membership Model, our research extends it to multiple channels, namely Multiple Channel Latent Dirichlet Allocation (MCLDA). The graphical structure of MCLDA is shown in Figure 3.1.3.

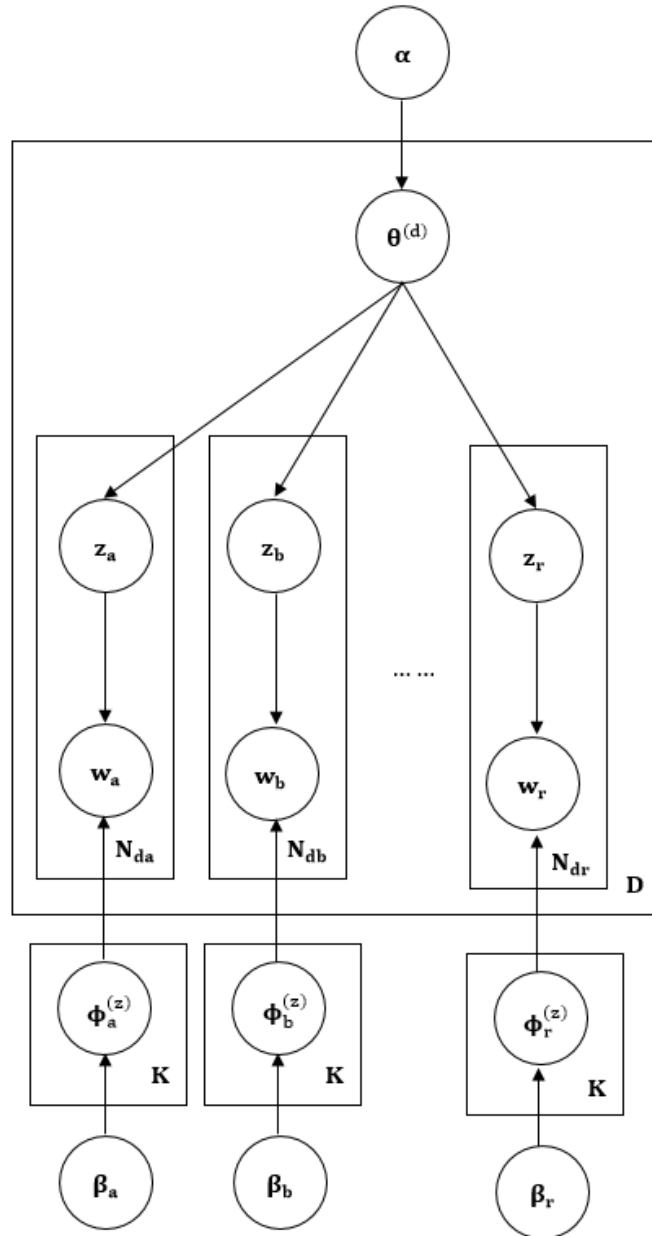


Figure 3.1.3: Graphical structure of Multiple Channel Latent Dirichlet Allocation

Then we implement the technique of injection of time feature in Topics over Time and develop Topics over Time Multiple Channel Latent Dirichlet Allocation. The graphical structure of MCLDA is shown in Figure 3.1.4. The explanation of each symbol is shown in Table 3.1.1.

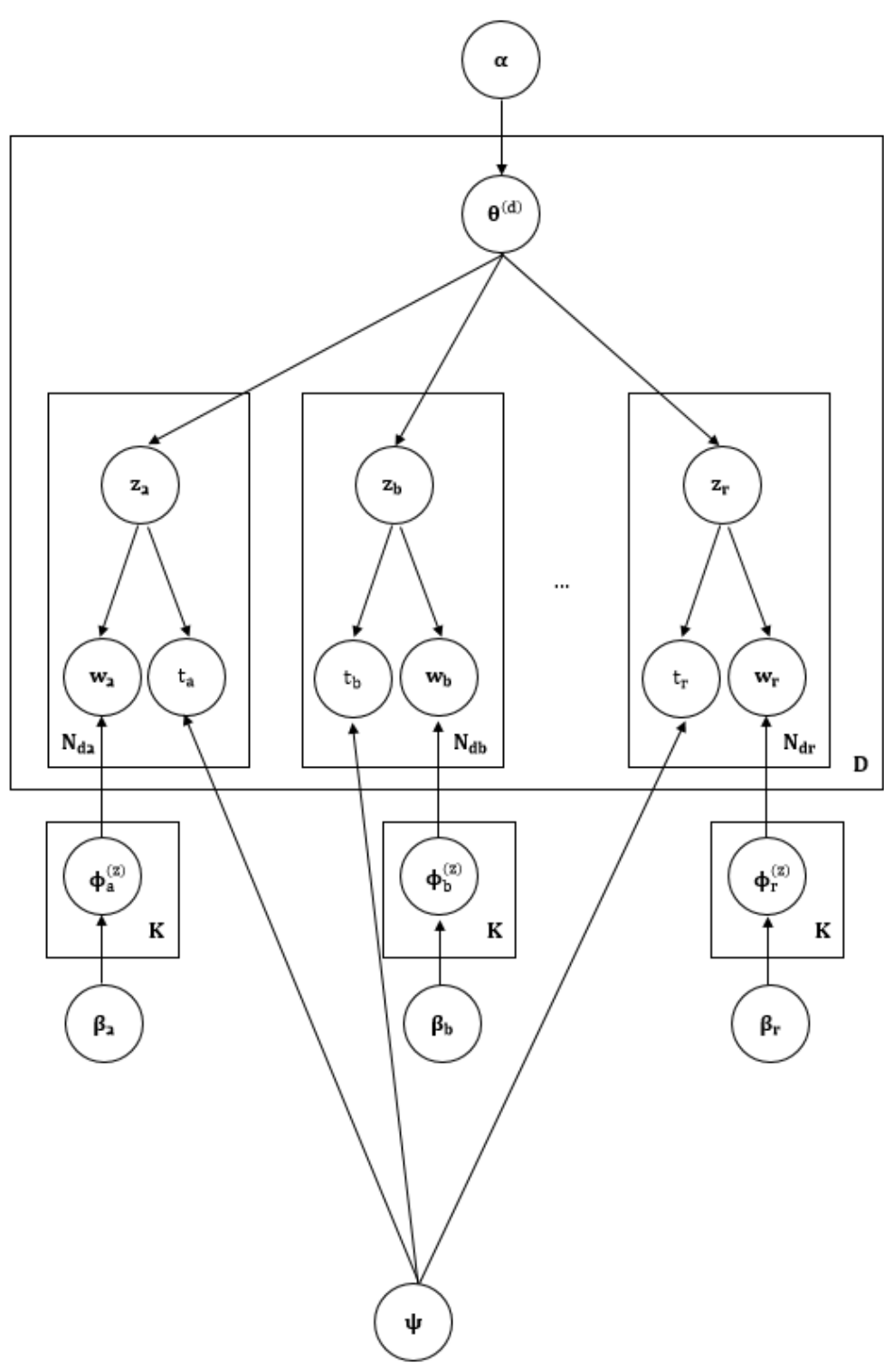


Figure 3.1.4: TOT-MCLDA

Table 3.1.1: Notation of variables of TOT-MCLDA

Symbol	Description
$\alpha$	Dirichlet priors to the multinomial distribution $\theta^{(d)}$
$\theta^{(d)}$	Multinomial distribution of topics specify to the document $d$
$\beta_r$	Dirichlet priors to the multinomial distribution $\phi_r^{(z)}$ of $r^{\text{th}}$ type of context
$\phi_r^{(z)}$	Multinomial distribution of words of $r^{\text{th}}$ type of context specify to the topic $z$
$z_{ri}$	Topic associated with the word $w_{ri}$ of $r^{\text{th}}$ type of context
$w_{ri}$	The $i^{\text{th}}$ word of $r^{\text{th}}$ type of context
$V_r$	The number of unique words of $r^{\text{th}}$ type of context
$t_{ri}$	Timestamp associated with the word $w_{ri}$ of $r^{\text{th}}$ type of context
$\psi$	The beta distribution of time
$K$	Number of topics
$D$	Number of documents
$N_{dr}$	Number of words of $r^{\text{th}}$ type of context in document $d$



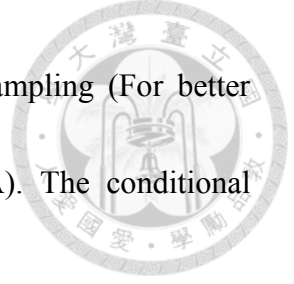
The generation of a document involves in the following process:

- (1) For each topic  $k$ ,
  - (a) Draw a distribution over words  $\phi_a^{(k)} \sim \text{Dir}(\beta_a)$
  - (b) Draw a distribution over citations  $\phi_b^{(k)} \sim \text{Dir}(\beta_b)$
  - (c) ...
  - (d) Draw a distribution over citations  $\phi_r^{(k)} \sim \text{Dir}(\beta_r)$
- (2) For each document  $d$ ,
  - (a) Draw a vector of topic proportions  $\theta^{(d)} \sim \text{Dir}(\alpha)$
  - (b) For each word of context  $a$ ,
    - (i) Draw a topic assignment  $z_{ai}^{(d)} \sim \text{Mult}(\theta^{(d)})$ ,  $z_{ai}^{(d)} \in \{1, \dots, K\}$
    - (ii) Draw a word  $w_{ai}^{(d)} \sim \text{Mult}\left(\phi_a^{(z_{ai}^{(d)})}\right)$ ,  $w_{ai}^{(d)} \in \{1, \dots, V_a\}$
    - (iii) Draw a timestamp  $t_{ai}^{(d)} \sim \text{Beta}\left(\psi_a^{(z_{ai}^{(d)})}\right)$
  - (c) For each word of context  $b$ ,
    - (i) Draw a topic assignment  $z_{bi}^{(d)} \sim \text{Mult}(\theta^{(d)})$ ,  $z_{bi}^{(d)} \in \{1, \dots, K\}$
    - (ii) Draw a word  $w_{bi}^{(d)} \sim \text{Mult}\left(\phi_b^{(z_{bi}^{(d)})}\right)$ ,  $w_{bi}^{(d)} \in \{1, \dots, V_b\}$
    - (iii) Draw a timestamp  $t_{bi}^{(d)} \sim \text{Beta}\left(\psi_b^{(z_{bi}^{(d)})}\right)$
  - (d) ...
  - (e) For each word of context  $r$ ,
    - (i) Draw a topic assignment  $z_{ri}^{(d)} \sim \text{Mult}(\theta^{(d)})$ ,  $z_{ri}^{(d)} \in \{1, \dots, K\}$
    - (ii) Draw a word  $w_{ri}^{(d)} \sim \text{Mult}\left(\phi_r^{(z_{ri}^{(d)})}\right)$ ,  $w_{ri}^{(d)} \in \{1, \dots, V_b\}$
    - (iii) Draw a timestamp  $t_{ri}^{(d)} \sim \text{Beta}\left(\psi_r^{(z_{ri}^{(d)})}\right)$

The time range of the data used for parameter estimation in this model is first normalized to a range from 0 to 1 in order to employ the Beta distribution.

TOT-MCLDA can be completed by inferring the posterior probability

$P(Z|W_a, W_b, \dots, W_r, \dots, T)$ . Same as the inference procedure mentioned above, we can



make use of Markov chain Monte Carlo and Collapsed Gibbs Sampling (For better understanding of derivation process, please refer to Appendix A). The conditional probability of channel  $r$  is presented:

$$P(z_{ri} = k | z_a, z_b, z_c, \dots, z_{r-i}, \dots, w_a, w_b, w_c, \dots, w_r, \dots, t_a, t_b, t_c, \dots, t_r, \dots)$$

$$\propto \frac{\beta_r + n_{(-i),(k)}^{(R),(w_{ri})}}{V_r \beta_r + n_{(-i),(k)}^{(R),(\cdot)}} \frac{\alpha + n_{(\cdot),(k)}^{(da),(\cdot)} + \dots + n_{(-i),(k)}^{(dr),(\cdot)} + \dots}{K\alpha + n_{(\cdot),(\cdot)}^{(da),(\cdot)} + \dots + n_{(-i),(\cdot)}^{(dr),(\cdot)} + \dots} t_{ri}^{\psi_{k1}-1} (1 - t_{ri})^{\psi_{k2}-1} B(\psi_{k1}, \psi_{k2})$$

Equation 3.1.1.6

where  $n_{(-i),(k)}^{(R),(w_{ri})}$  represents the number of times word  $w_{ri}$  has been assigned to topic  $k$  in the vector of assignments  $Z_r$  without considering current position  $i$ ;  $n_{(-i),(k)}^{(R),(\cdot)}$  represents the number of times topic  $k$  appears in the vector of assignments  $Z_r$  without considering current position  $i$ ;  $n_{(\cdot),(k)}^{(dx),(\cdot)}$  represents the number of times topic  $k$  appears in document  $dx$ , where  $x$  is all type of context except for  $r$ ;  $n_{(\cdot),(\cdot)}^{(dx),(\cdot)}$  represents the number of words in document  $dx$ , where  $x$  is all type of context except for  $r$ ;  $n_{(-i),(k)}^{(dr),(\cdot)}$  represents the number of times topic  $k$  appears in document  $dr$  without considering current position  $i$ ;  $n_{(-i),(\cdot)}^{(dr),(\cdot)}$  represents the number of words in document  $dr$  without considering current position  $i$ ;  $t_{ri}$  represents the timestamp of the current position  $i$ . In addition,  $\Psi$  is updated after each sweep of Collapsed Gibbs

Sampling by the method of moments:



$$\psi_{k1} = \bar{t}_k \left( \frac{\bar{t}_k(1 - \bar{t}_k)}{s_k^2} - 1 \right)$$

Equation 3.1.1.7

$$\psi_{k2} = (1 - \bar{t}_k) \left( \frac{\bar{t}_k(1 - \bar{t}_k)}{s_k^2} - 1 \right)$$

Equation 3.1.1.8

where  $\bar{t}_k$  and  $s_k^2$  indicate the sample mean and the biased sample variance of the timestamps belonging to topic  $k$  respectively.

Given a set of recorded sweeps, the estimation of  $\phi_r$  for  $r^{\text{th}}$  type of context and  $\theta$  can be computed via:

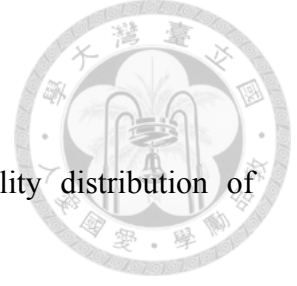
$$\widehat{\phi}_r^{(k)} = \frac{\beta_r + n_{(\cdot), (k)}^{(R), (v)}}{V_r \beta_r + n_{(\cdot), (k)}^{(R), (\cdot)}}$$

Equation 3.1.1.9

$$\widehat{\theta}_k^{(d)} = \frac{\alpha + n_{(\cdot), (k)}^{(da), (\cdot)} + \dots + n_{(\cdot), (k)}^{(dr), (\cdot)} + \dots}{K\alpha + n_{(\cdot), (\cdot)}^{(da), (\cdot)} + \dots + n_{(\cdot), (\cdot)}^{(dr), (\cdot)} + \dots}$$

Equation 3.1.1.10



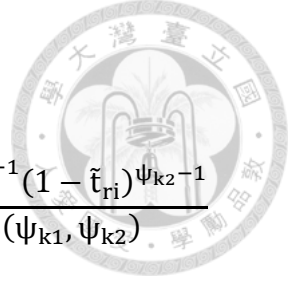


Based on these estimations, we can construct the probability distribution of documents over topics and topics over words.

In our research, we consider both text and hashtags in tweets as two types of context, and posting time of tweets as timestamp to construct TOT-MCLDA. The data collection is a collection of tweets. Each document corresponds to one tweet. The words of context type **a** of the document are text of the tweet. The words of context type **b** of the document are hashtags of the tweet. The timestamp of the document is the posting time of the tweet.

After training the model, we can get the topic distribution  $\hat{\theta}^{(d)}$  of each tweet  $d$ , where  $\hat{\theta}^{(d)} = \{\hat{\theta}^{(d)}_1, \hat{\theta}^{(d)}_2, \dots, \hat{\theta}^{(d)}_K\}$ ,  $d = 1, 2, \dots, D$ ; the text distribution  $\hat{\phi}_a^{(k)}$  of each topic  $k$ , where  $\hat{\phi}_a^{(k)} = \{\hat{\phi}_a^{(k)}_1, \hat{\phi}_a^{(k)}_2, \dots, \hat{\phi}_a^{(k)}_{v_a}\}$ ,  $k = 1, 2, \dots, K$ ; the hashtag distribution  $\hat{\phi}_b^{(k)}$  of each topic  $k$ , where  $\hat{\phi}_b^{(k)} = \{\hat{\phi}_b^{(k)}_1, \hat{\phi}_b^{(k)}_2, \dots, \hat{\phi}_b^{(k)}_{v_b}\}$ ,  $k = 1, 2, \dots, K$ .

To determine the topic distribution of a new tweet, we again make use of Collapsed Gibbs Sampling and the trained text distribution  $\hat{\phi}_a^{(k)}$ . Given a target tweet  $\tilde{d}$ , we can compute the conditional distribution of text words:



$$P(\tilde{z}_{ai} = k | \tilde{z}_{a-i}, \tilde{w}_a, \tilde{t}_{ri}, z_a, z_b, w_a, w_b, \psi)$$

$$\propto \frac{\beta_a + n_{(\cdot),(k)}^{(A),(\tilde{w}_{ai})} + n_{(-i),(k)}^{(\tilde{d}a),(\tilde{w}_{ai})}}{V_a \beta_a + n_{(\cdot),(k)}^{(A),(\cdot)} + n_{(-i),(k)}^{(\tilde{d}a),(\cdot)}} \frac{\alpha + n_{(-i),(k)}^{(\tilde{d}a),(\cdot)}}{K\alpha + n_{(-i),(\cdot)}^{(\tilde{d}a),(\cdot)}} \tilde{t}_{ri}^{\psi_{k1}-1} (1 - \tilde{t}_{ri})^{\psi_{k2}-1} B(\psi_{k1}, \psi_{k2})$$

Equation 3.1.1.11

where  $n_{(-i),(k)}^{(\tilde{d}a),(\tilde{w}_{ai})}$  represents the number of times word  $\tilde{w}_{ai}$  has been assigned to topic  $k$  in the vector of assignments  $Z_{\tilde{d}a}$  without considering current position  $i$ ;  $n_{(-i),(k)}^{(\tilde{d}a),(\cdot)}$  represents the number of times topic  $k$  appears in the vector of assignments  $Z_{\tilde{d}a}$  without considering current position  $i$ . Given a set of recorded sweeps, the estimation of  $\hat{\theta}_{(k)}^{(\tilde{d})}$  can be computed via:

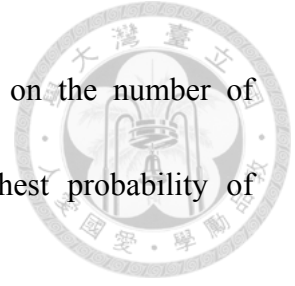
$$\hat{\theta}_{(k)}^{(\tilde{d})} = \frac{\alpha + n_{(\cdot),(k)}^{(\tilde{d}a),(\cdot)}}{K\alpha + n_{(\cdot),(\cdot)}^{(\tilde{d}a),(\cdot)}}$$

Equation 3.1.1.12

Now the topic distribution  $\hat{\theta}^{(\tilde{d})}$  of the target tweet  $\tilde{d}$  is obtained, where  $\hat{\theta}^{(\tilde{d})} = \{\hat{\theta}_{(1)}^{(\tilde{d})}, \hat{\theta}_{(2)}^{(\tilde{d})}, \dots, \hat{\theta}_{(K)}^{(\tilde{d})}\}$ .

In order to recommend suitable hashtags, we stack all vector  $\widehat{\Phi}_b^{(k)}$  by the order of  $k = 1, 2, \dots, K$  and obtain matrix  $\widehat{\Phi}_b$  with dimension  $K \times V_b$ . By calculating the inner product of  $\hat{\theta}^{(\tilde{d})}$  and  $\widehat{\Phi}_b$ , we can get a vector  $X$ . The element  $x_v$  in  $X$  represent the

probability of hashtag  $v$  appearing in the target tweet  $\tilde{d}$ . Based on the number of hashtags we want to present, we select the hashtags with highest probability of appearance and recommend to the target tweet  $\tilde{d}$ .



## 3.2 Baseline Model



In our research, we compare Topics over Time Multiple Channel Latent Dirichlet Allocation to three baseline models: similarity approach, Latent Dirichlet Allocation, and Multiple Channel Latent Dirichlet Allocation.

As for similarity approach, we make use of *HashtagsOfTweets* proposed by Kywe et al. (2012) to find suitable hashtags in the top-X similar tweets. The implementation of LDA is the same as Godin et al. (2013) proposed. The implementation of MCLDA is the same as we mentioned in the process of recommending hashtags in TOT-MCLDA.



### 3.3 Metrics

In our research, we use hit rate to evaluate the performance of each model. The formula of hit rate is shown below,

$$\text{Hit Rate} = \frac{\text{Number of Hits}}{\text{Number of Target Tweets}}$$

Equation 3.3.1

For a target tweet, a set of recommended hashtags will be generated by a model. If the set of recommended hashtags contain at least one of the ground truth hashtags, a hit occur. E.g., if five target tweets and the sets of recommended hashtags correspond to each tweet are shown in Table 3.3.1, the hit rate will be  $\frac{3}{5} * 100\% = 60\%$ .

Table 3.3.1: An example of hit rate

Target tweet No.	Ground truth hashtag in target tweet	Recommended hashtags	Hit
1	#soccer	#soccer, #ball, #fifa	✓
2	#nba	#basketball, #nba, #spurs	✓
3	#coffee	#cake, #cream, #tiramisu	
4	#van	#car, #truck, #van	✓
5	#love	#feet, #ankle, #leg	
Number of hits			3

## Chapter 4 Data Selection and Experimental Results

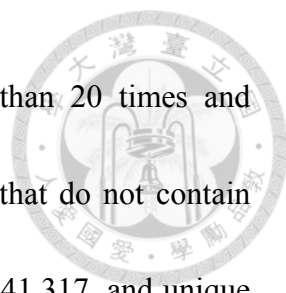


In this chapter, we will first present the research testbed. Then the experimental results of TOT-MCLDA and baseline models are given.

### 4.1 Data Selection and Preprocess

In our research, we adopt the Twitter dataset collected by Li et al. (2012). The dataset was originally collected in May 2011. There are 61,732,967 tweets in the whole dataset. Each record contains the following elements: Type (status), Origin (original content), Text (processed content), URL (URL tweet), ID (tweet id), Time (creation time), RetCount (retweet count), Favorite (favorite), MentionedEntities (mentioned user id), and Hashtags (hashtags).

Given several examples of the original data, we construct the research testbed by going through the following procedure. First, we removed stopwords from the Text feature and deleted tweets that do not contain either text or hashtags. The remaining dataset contains 12,257,039 tweets. Next, we randomly sampled 1,000,000 tweets from the dataset, which contains 193,992 unique text words and 211,861 unique hashtags. However, most of the text words and hashtags appear only a few times or once in the



dataset. Therefore, we removed those text words appeared less than 20 times and hashtags appeared less than 10 times. We then filtered out tweets that do not contain either text or hashtags. The number of tweets is now decreased to 741,317, and unique text words and hashtags are decreased to 20,292 and 16,839. Table 4.1.1 shows some basic statistics of the dataset. Figure 4.1.1 and Figure 4.1.2 show the phenomenon of Zipf's law of text words and hashtags respectively. In order to implement time feature, we normalized the creation time of each tweet to a range from 0 to 1 by using linear interpolation, where 0.0001 represents 2008/08/07 and 0.9999 represents 2011/08/03.

Table 4.1.1: Basic statistics of the dataset

Item	Value
Total number of tweets	741,317
Total number of text words	4,036,493
Total number of unique text words	20,292
Average number of text words per tweet	5.52
Standard deviation of text words per tweet	2.75
Total number of hashtags	1,059,024
Total number of unique hashtags	16,839
Average number of hashtag per tweet	1.45
Standard deviation of hashtag per tweet	0.91
Time duration	2008/08/07 ~ 2011/08/03

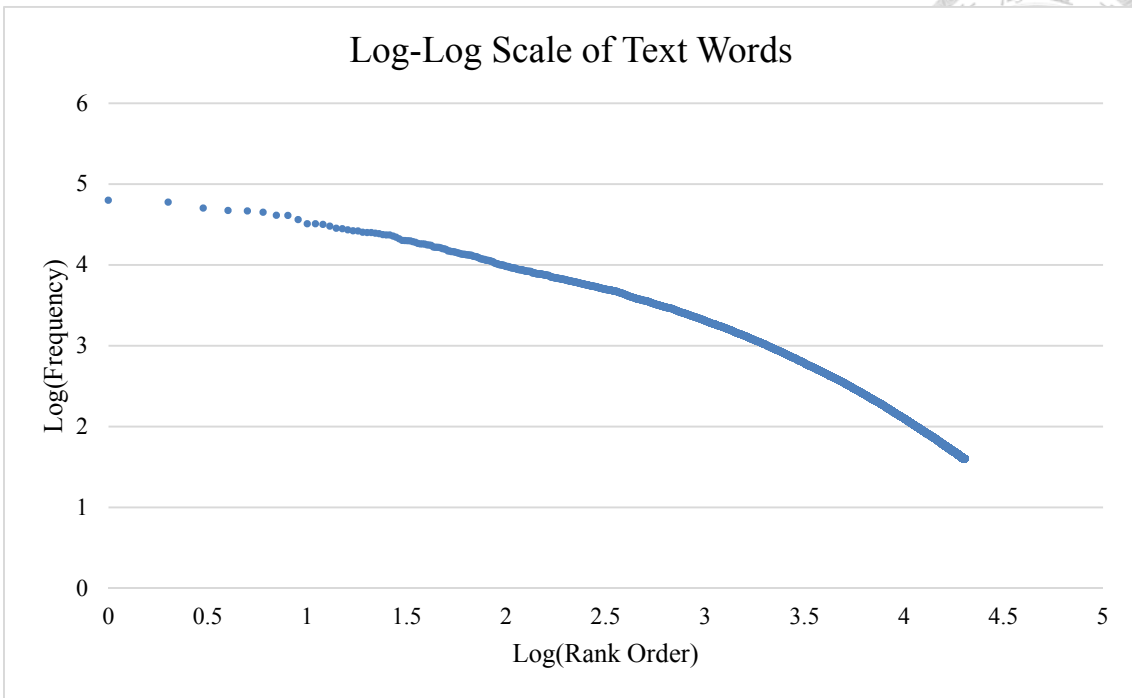


Figure 4.1.1: The distribution of text words' count

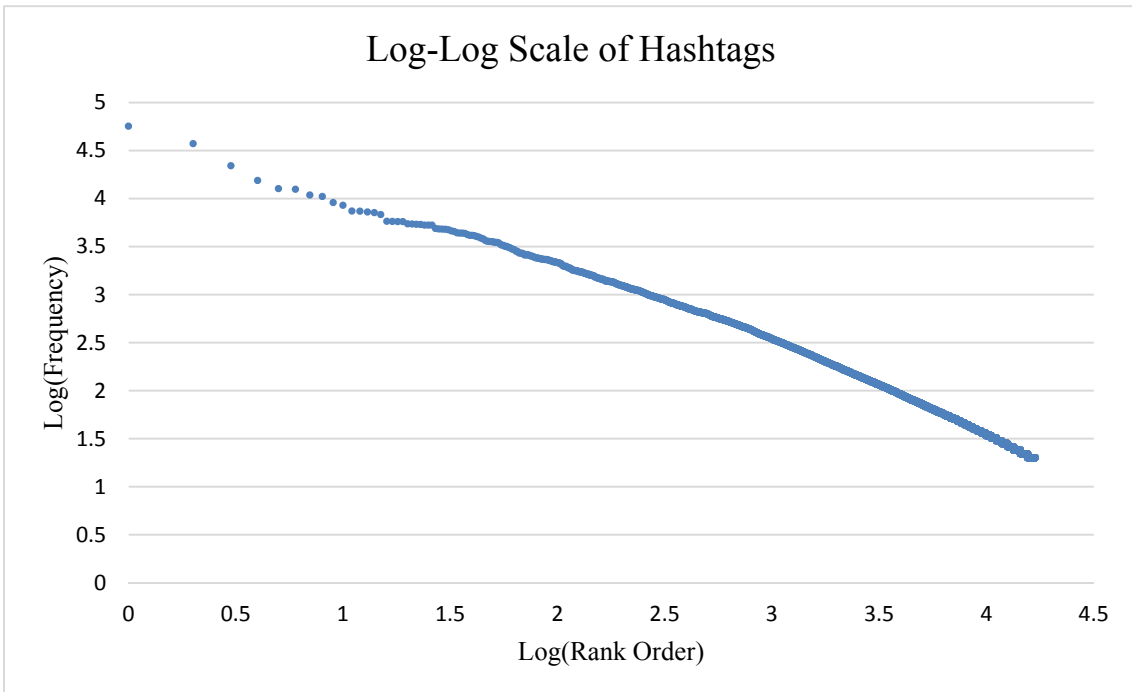


Figure 4.1.2: The distribution of hashtags' count



## 4.2 Performance Evaluation



In order to construct a more accurate evaluation process, we make use of 10-fold cross-validation method. We first randomly split the dataset into 10 equally sized subsets. Among these subsets, one subset is used as the validation data to test the model, and the remaining 9 subsets are used as the training data to build the model. The process is repeated 10 times and each subset is used only once as the validation data. After the process, each model will obtain 10 hit rate values corresponding to 10 subsets. Then we perform paired t-test to see if the performances of models have significant difference.

### 4.3 Parameter estimation



In the SIM model, we consider top 20 similar tweets according to the amount of similar tweets Kywe et al. (2012) used.

In LDA, MCLDA, and TOT-MCLDA, we need to prescribe  $\alpha$  (topic's Dirichlet prior),  $\beta_j$  (feature  $j$ 's Dirichlet prior), and  $K$  (the number of topics). The prior  $\alpha$  is usually set to  $50/K$  and the prior  $\beta$  is set to 0.1 (Griffiths and Steyvers, 2004). However, the number of text words and hashtags in a tweet is too small, large  $\alpha$  or  $\beta$  might contort the Dirichlet distribution. Therefore, we apply  $\alpha = \{5/K, 1/K\}$  and  $\beta = \{0.1, 0.01, 0.001\}$ . In order to estimate suitable  $\alpha$  and  $\beta$ , we randomly sampled a subset from our dataset and ran 10-fold cross-validation on different sets of parameter ( $\alpha \times \beta$ ). Topic number  $K$  is set to 200, which is used by Godin et al. (2013) in their research. The result shows that the suitable set of parameters is  $\alpha = 1/K$  and  $\beta = 0.001$ .

## 4.4 Experimental results of Twitter data



In our experiments, we recommend top 10, 20, and 50 hashtags to each target tweet.

Table 4.4.1 shows the performance of each model on recommending top 10 hashtags. In addition, the two tailed paired t-test of the difference of hit rate in Table 4.4.2 shows that the performance of any two models is significantly different. As a result of fact, we can infer that TOT-MCLDA performs the best.

Table 4.4.1: Hit rate of each model on recommending top 10 hashtags (%)

Fold NO.	SIM	LDA	MCLDA	TOT-MCLDA
1	24.31	22.14	24.14	24.86
2	27.53	26.01	27.86	27.96
3	24.91	24.77	25.37	28.44
4	27.83	25.37	28.34	29.01
5	25.51	24.44	25.23	26.12
6	28.76	27.35	29.26	29.98
7	26.37	26.22	26.24	28.07
8	24.64	24.43	25.02	25.66
9	27.67	27.32	27.77	27.83
10	25.75	25.34	25.98	28.32
<b>Avg.</b>	<b>26.328</b>	<b>25.339</b>	<b>26.521</b>	<b>27.625</b>

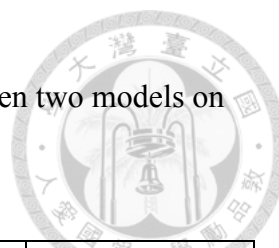


Table 4.4.2: Two tailed paired t-test of the hit rate difference between two models on recommending top 10 hashtags (%)

	SIM	LDA	MCLDA	TOT-MCLDA
SIM	-	-	-	-
LDA	0.99**	-	-	-
MCLDA	-0.19*	-1.18**	-	-
TOT-MCLDA	-1.30**	-2.29***	-1.10**	-

*Note.* Significant at: \* $p < 0.05$ , \*\* $p < 0.01$ , \*\*\* $p < 0.001$ . The minuend is the model

in row, and the subtrahend is the model in column.



Table 4.4.3 shows the performance of each model on recommending top 20 hashtags. In addition, the two tailed paired t-test of the difference of hit rate in Table 4.4.4 shows that the performance of any two models is significantly different. Therefore, we can infer that TOT-MCLDA performs the best.

Table 4.4.3: Hit rate of each model on recommending top 20 hashtags (%)

Fold NO.	SIM	LDA	MCLDA	TOT-MCLDA
1	30.01	29.06	30.28	30.55
2	33.09	32.27	33.45	33.48
3	31.02	30.98	32.98	35.42
4	33.55	33.26	34.26	34.89
5	32.09	32.02	33.16	33.83
6	35.04	34.21	35.26	35.75
7	32.16	32.01	32.19	34.13
8	30.67	30.43	30.99	31.02
9	33.51	32.17	33.73	33.77
10	31.89	31.14	32.73	34.62
<b>Avg.</b>	<b>32.303</b>	<b>31.755</b>	<b>32.903</b>	<b>33.746</b>

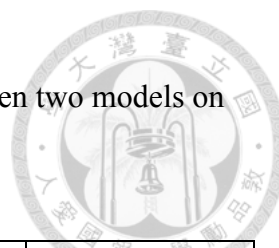


Table 4.4.4: Two tailed paired t-test of the hit rate difference between two models on recommending top 20 hashtags (%)

	SIM	LDA	MCLDA	TOT-MCLDA
SIM	-	-	-	-
LDA	0.55**	-	-	-
MCLDA	-0.6**	-1.15***	-	-
TOT-MCLDA	-1.44**	-1.99***	-0.84*	-

*Note.* Significant at: \* $p < 0.05$ , \*\* $p < 0.01$ , \*\*\* $p < 0.001$ . The minuend is the model in row, and the subtrahend is the model in column.



Table 4.4.5 shows the performance of each model on recommending top 50 hashtags. In addition, the two tailed paired t-test of the difference of hit rate is shown in Table 4.4.6. The performance of any two models is significantly different except for SIM and LDA. Hence, we can infer that TOT-MCLDA performs the best.

Table 4.4.5: Hit rate of each model on recommending top 50 hashtags (%)

Fold NO.	SIM	LDA	MCLDA	TOT-MCLDA
1	39.42	39.32	42.67	45.38
2	42.54	41.68	45.77	48.96
3	40.31	41.11	45.53	50.39
4	42.98	43.39	46.86	49.93
5	41.65	40.86	45.73	49.12
6	44.51	44.12	47.61	51.56
7	41.81	41.73	44.98	49.81
8	40.01	40.32	43.79	46.61
9	43.53	43.32	46.36	49.65
10	41.46	40.76	45.34	51.35
<b>Avg.</b>	<b>41.822</b>	<b>41.661</b>	<b>45.464</b>	<b>49.276</b>



Table 4.4.6: Two tailed paired t-test of the hit rate difference between two models on recommending top 50 hashtags (%)

	SIM	LDA	MCLDA	TOT-MCLDA
SIM	-	-	-	-
LDA	0.16	-	-	-
MCLDA	-3.64***	-3.8***	-	-
TOT-MCLDA	-7.45***	-7.62***	-3.81***	-

*Note.* Significant at: \* $p < 0.05$ , \*\* $p < 0.01$ , \*\*\* $p < 0.001$ . The minuend is the model in row, and the subtrahend is the model in column.





Figure 4.4.1 shows the histogram of hit rate of each model on recommending top 10, 20, and 50 hashtags. We can see that as the number of recommended hashtags grow, the performance of TOT-MCLDA becomes much better in comparison of other methods.

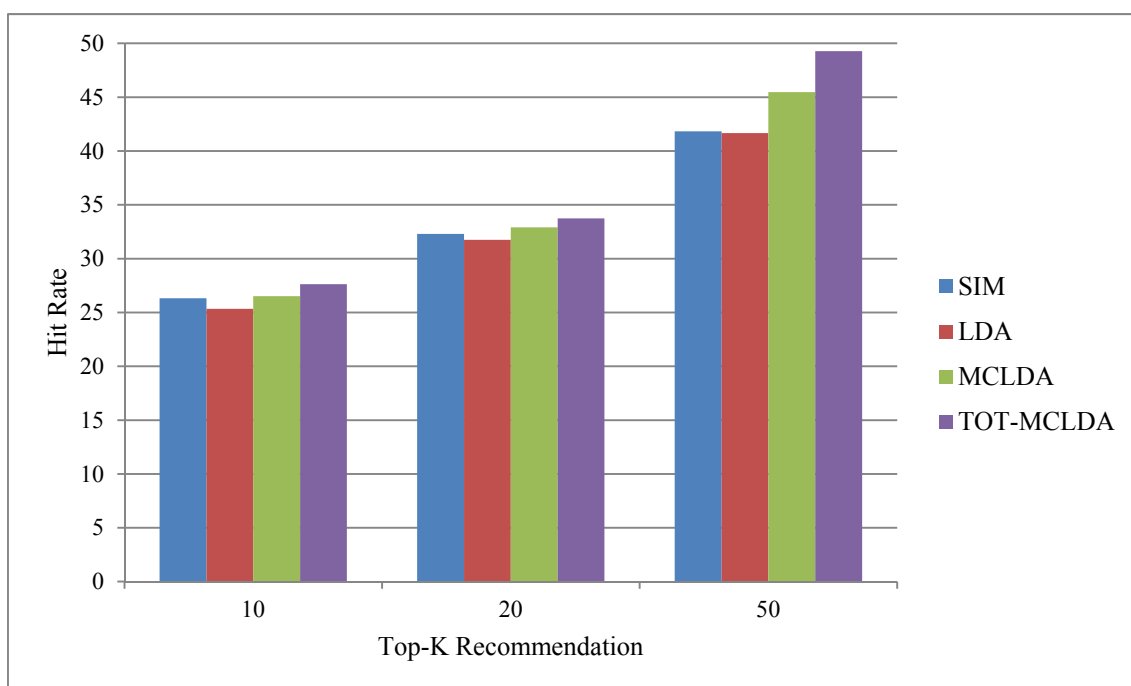


Figure 4.4.1: The histogram of hit rate on recommending top 10, 20, and 50 hashtags



#### 4.4.1 Analyses of Recommendation Lists

In order to discuss the results in detail, we select some tweets to see the performance of each model. Table 4.4.7 shows the data contained in the tweet of which ID is 4846085206. The tweet is talking about how wounded soldiers in Iraq inspire others by sharing their own experience. As for this tweet, SIM recommends #military at 3<sup>rd</sup> pick, LDA at 1<sup>st</sup> pick, MCLDA at 1<sup>st</sup> pick, and TOT-MCLDA at 1<sup>st</sup> pick.

Table 4.4.7: Data of ID 4846085206 tweet

<b>ID</b>	4846085206
<b>Original Content</b>	"Iraq Progress Inspires Returning Wounded: <a href="http://bit.ly/p94KH">http://bit.ly/p94KH</a> #military"
<b>Text Word</b>	iraq, progress, inspires, returning, wounded
<b>Hashtag</b>	#military
<b>Time</b>	2009-10-14 06:15:46+08

Since the text words in this tweet are very representative, all of the models perform well. Table 4.4.8 shows the recommendation list given by SIM. Since the most similar tweet uses #iava (i.e., Iraq and Afghanistan Veterans of America) and #iraq as the hashtags, the ranking of #military is receded. Table 4.4.9 shows the recommendation list given by LDA, MCLDA, and TOT-MCLDA. All of the three methods generate the same top 5 recommendation list. Since the latent topics of the tweet are quite unified, it is easy for these topic models to construct a suitable recommendation list.



Table 4.4.8: SIM hashtags ranking of ID 4846085206 tweet

Ranking	Hashtags
1	#iava
2	#iraq
3	<b>#military</b>
4	#veterans
5	#troops

Table 4.4.9: LDA, MCLDA, TOT-MCLDA hashtags ranking of ID 4846085206 tweet

Ranking	Hashtags
1	<b>#military</b>
2	#iraq
3	#afghanistan
4	#veterans
5	#usa

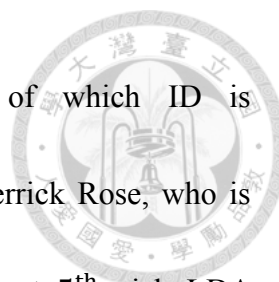


Table 4.4.10 shows the data contained in the tweet of which ID is 91523345744543745. The tweet is talking about a NBA player Derrick Rose, who is playing for Chicago Bulls. As for this tweet, SIM recommends #bulls at 5<sup>th</sup> pick, LDA at 7<sup>th</sup> pick, MCLDA at 3<sup>rd</sup> pick, and TOT-MCLDA at 2<sup>nd</sup> pick.

Table 4.4.10: Data of ID 91523345744543745 tweet

<b>ID</b>	91523345744543745
<b>Original Content</b>	"TrueHoop - By The Horns: Lots of Derrick Rose news <a href="http://ffd.me/n3ufvm">http://ffd.me/n3ufvm</a> #bulls"
<b>Text Word</b>	truehoop, horns, lots, derrick, rose, news
<b>Hashtag</b>	#bulls
<b>Time</b>	2011-07-14 23:04:00+08

Since the text words contain ‘derrick’ and ‘rose’, it is easy for SIM to find similar tweets which is also discussing the player. However, most of the top 10 similar tweets prefer hashtagging #derrickrose and #nba to #bulls, so the ranking of #bulls is receded.

Table 4.4.11 shows the ranking list.

Table 4.4.11: SIM hashtags ranking of ID 91523345744543745 tweet

Ranking	Hashtags
1	#derrickrose
2	#nba
3	#win
4	#winning
5	<b>#bulls</b>

As for LDA, MCLDA, and TOT-MCLDA, the consistency of the data makes them easier to accurately predict the latent topic of the tweet. However, LDA prefers recommending text words to hashtags since the amount of text words is higher. In contrast, MCLDA and TOT-MCLDA only pick hashtags to recommend, so the performances are better. Table 4.4.12, Table 4.4.13, and Table 4.4.14 show the ranking list given by LDA, MCLDA, and TOT-MCLDA respectively.

Table 4.4.12: LDA hashtags ranking of ID 91523345744543745 tweet

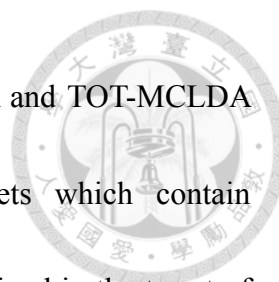
Ranking	Hashtags
1	#nba
2	#rose
3	#derrick
4	#derrickrose
5	#eastern
6	#chicago
7	<b>#bulls</b>

Table 4.4.13: MCLDA hashtags ranking of ID 91523345744543745 tweet

Ranking	Hashtags
1	#derrickrose
2	#nba
3	<b>#bulls</b>
4	#teambulls
5	#chicago

Table 4.4.14: TOT-MCLDA hashtags ranking of ID 91523345744543745 tweet

Ranking	Hashtags
1	#derrickrose
2	<b>#bulls</b>
3	#nba
4	#teambulls
5	#chicago



Since it is difficult to express the difference between MCLDA and TOT-MCLDA in the previous samples, we further select another two tweets which contain time-sensitive content to discuss. Table 4.4.15 shows the data contained in the tweet of which ID is 55991762501644288. The tweet is talking about what will happen to the travelers if federal government shuts down. This is originally refer to the event that the federal government might shut down if the United States Congress did not reach a deal on the 2011 United States federal budget in April, 2011. Some derived issues were also discussed heatedly, such as security problems of internet and society. In this case, SIM recommends #government at 5<sup>th</sup> pick, LDA at 11<sup>th</sup> pick, MCLDA at 9<sup>th</sup> pick, and TOT-MCLDA at 3<sup>rd</sup> pick.

Table 4.4.15: Data of ID 55991762501644288 tweet

<b>ID</b>	55991762501644288
<b>Original Content</b>	"If the #government shuts down, here's how @aoltravel says it could affect Ttravelers: <a href="http://ow.ly/4vbBz">http://ow.ly/4vbBz</a> "
<b>Text Word</b>	shuts, affect
<b>Hashtag</b>	#government
<b>Time</b>	2011-04-07 21:54:11+08

Since the only useful text words are ‘shuts’ and ‘affect’, there is little information contained in this tweet that can be used for hashtags prediction. Therefore, the performance of LDA and MCLDA is bad. In contrast, TOT-MCLDA can further make



use of the time feature to combine topics around April, 2011 and predict hashtags from a more concentrated distribution. Table 4.4.16 shows the ranking list given by SIM. Since some of the tweets are directly talking the issue, they are gathered in the top-20 similar tweets. However, most of them prefer hashtagging #ifgovernmentshutsdown, and some other non-relevant tweets are taken into consideration. Therefore, the ranking of #government is receded.

Table 4.4.16: SIM hashtags ranking of ID 55991762501644288 tweet

Ranking	Hashtags
1	#ifgovernmentshutsdown
2	#shutdown
3	#obama
4	#security
5	<b>#government</b>

Table 4.4.17, Table 4.4.18 and Table 4.4.19 show the ranking list given by LDA, MCLDA and TOT-MCLDA respectively. We can see that the ranking list given by TOT-MCLDA is more relevant to the event previously mentioned. However, the ranking list given by LDA and MCLDA seems to be more relevant to the Tea Party movement happened in September, 2009.



Table 4.4.17: LDA hashtags ranking of ID 55991762501644288 tweet

Ranking	Hashtags
1	#debt
2	#obama
3	#tax
4	#budget
5	#americans
6	#jobs
7	#teaparty
8	#credit
9	#money
10	#finance
11	<b>#government</b>

Table 4.4.18: MCLDA hashtags ranking of ID 55991762501644288 tweet

Ranking	Hashtags
1	#tcot
2	#obama
3	#debt
4	#tax
5	#economy
6	#taxes
7	#finance
8	#teaparty
9	<b>#government</b>

Table 4.4.19: TOT-MCLDA hashtags ranking of ID 55991762501644288 tweet

Ranking	Hashtags
1	#budget
2	#ifgovernmentshutsdown
3	<b>#government</b>
4	#governmentbudget
5	#security
6	#cybersecurity



Another example is shown in Table 4.4.20. The tweet of which ID is 92703923739164673 is talking about the Women World Cup in 2011. The final game between Japan and USA was held in Germany on July 18<sup>th</sup>. SIM recommends #wwc at 7<sup>th</sup> pick, LDA at 8<sup>th</sup> pick, MCLDA at 5<sup>th</sup> pick, and TOT-MCLDA at 1<sup>st</sup> pick.

Table 4.4.20: Data of ID 92703923739164673 tweet

<b>ID</b>	92703923739164673
<b>Original Content</b>	"Never been a huge soccer fan, but have to admit this World Cup Final is intense! Go Team USA #WWC"
<b>Text Word</b>	huge, soccer, fan, admit, world, cup, final, intense, team, usa
<b>Hashtag</b>	#wwc
<b>Time</b>	2011-07-18 05:15:12+08

Although the discussion of WWC is popular, FIFA World Cup 2010 is a confusing event to WWC. Tweet that does not include words relevant to WWC 2011, i.e., women or Germany, could be easily confused with FIFA World Cup 2010.

Table 4.4.21 shows the ranking list given by SIM. We can see that the recommendations are influenced by the event of FIFA World Cup 2010. This is due to the fact that top-20 similar tweets are mostly consisting of FIFA World Cup 2010 related tweets. For example, the original content in one of the most similar tweet is "Leaving!! What an amazing time! Now, I have to find a way to watch the Ghan Vs US world cup", of which hashtag is #fifa.



Table 4.4.21: SIM hashtags ranking of ID 92703923739164673 tweet

Ranking	Hashtags
1	#fifa
2	#worldcup
3	#worldcupfinal
4	#soccer
5	#fifaworldcup
6	#usa
7	#wwc
8	#womensworldcup

Both LDA and MCLDA are also immersed in the confusion. Table 4.4.22 and Table 4.4.23 show the ranking list given by LDA and MCLDA, respectively. We can see that the generated rankings are much more relevant to FIFA World Cup 2010. In this case, it is difficult to distinguish WWC 2011 and FIFA World Cup 2010, so the latent topic of the tweet is distorted. Since MCLDA additionally consider the feature of hashtag to build the model, it can generate a more suitable ranking list than LDA.

Table 4.4.22: LDA hashtags ranking of ID 92703923739164673 tweet

Ranking	Hashtags
1	#fifa
2	#worldcupfinal
3	#final
4	#soccer
5	#game
6	#worldcup
7	#usa
8	#wwc

Table 4.4.23: MCLDA hashtags ranking of ID 92703923739164673 tweet

Ranking	Hashtags
1	#fifa
2	#worldcup
3	#worldcupfinal
4	#usa
5	#wwc

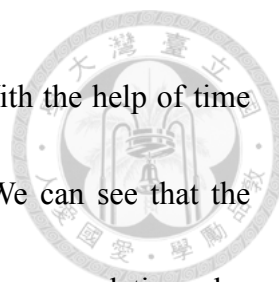
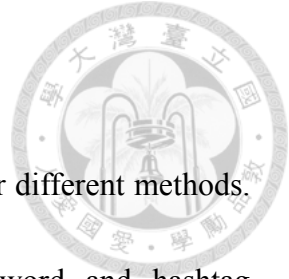


Table 4.4.24 shows the ranking list given by TOT-MCLDA. With the help of time feature, TOT-MCLDA can accurately generate suitable hashtags. We can see that the ranking list mostly focus on WWC 2011. Besides, the first recommendation also matches the target hashtag.

Table 4.4.24: TOT-MCLDA hashtags ranking of ID 92703923739164673 tweet

Ranking	Hashtags
1	#wwc
2	#womensworldcup
3	#worldcupfinal
4	#usa
5	#japan

These two time-sensitive examples clearly present how TOT-MCLDA performs better than other methods. In both cases, TOT-MCLDA successfully differentiates two polysemous issues with the help of time feature to distinguish different events.



#### 4.4.2 Analyses of Topic Distributions

We further inspect the hashtag distribution of similar topic over different methods.

Table 4.4.25 and Table 4.4.26 shows the TOT-MCLDA text word and hashtag distribution of topic 46. We can see that all hashtags are relevant to National Football League (#nfl) and team name (e.g., Jets, Redskins, and Bears etc.). Refer to Figure 4.4.2, the topic occurred every year from September to December. The peak of the beta distribution is located around October 2010. This is the time of regular season of NFL. Table 4.4.27 and Table 4.4.28 show the similar topic generated by MCLDA. However, Figure 4.4.3 shows that the topic is confused by other events occurred in March and April (when text words ‘players’ and ‘season’ were used again). The peak of the beta distribution is also left-shifted. This is the period when the regular season of National Basketball Association (NBA) ends. Hence, the hashtag distribution is mixed up with some other hashtags relevant to NBA and team name, e.g., #nba, #heat (Miami Heat), and #lakers (Los Angeles Lakers).

Table 4.4.25: TOT-MCLDA text word distribution of topic 46 sorted by probability

Text Words	Probability
nfl	0.0337
players	0.0198
deal	0.0148
football	0.0141
draft	0.0140
lockout	0.0124
season	0.0104
free	0.0102
bears	0.0010
fans	0.0010

Table 4.4.26: TOT-MCLDA hashtag distribution of topic 46 sorted by probability

Hashtags	Probability
#nfl	0.2818
#twitnewsnow	0.1071
#jets	0.0293
#redskins	0.0273
#bears	0.0257
#eagles	0.0233
#packers	0.0232
#patriots	0.0207
#steelers	0.0196
#cowboys	0.0164



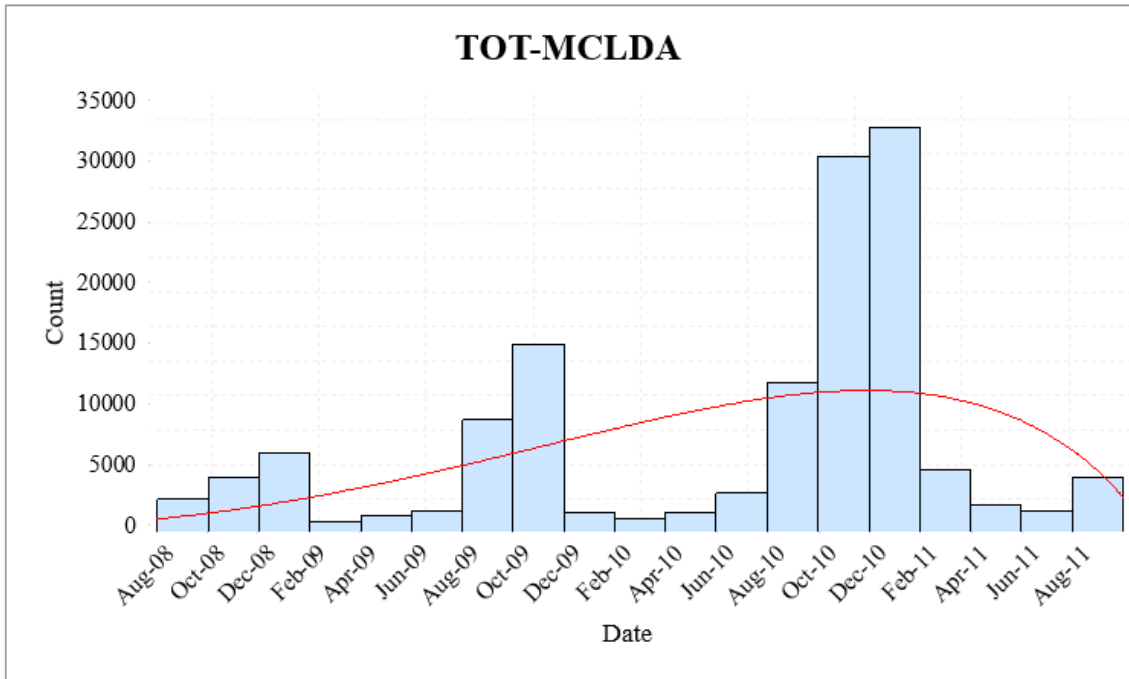


Figure 4.4.2: TOT-MCLDA topic 46 distributed over time (The fitted beta PDF is shown by the red line).

Table 4.4.27: MCLDA text word distribution of topic 71 sorted by probability

Text Words	Probability
nfl	0.0252
players	0.0128
football	0.0125
draft	0.0115
lockout	0.0106
season	0.0101
deal	0.0092
game	0.0091
fans	0.0077
team	0.0071

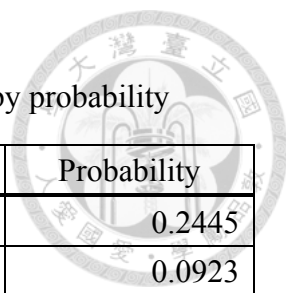


Table 4.4.28: MCLDA hashtag distribution of topic 71 sorted by probability

Hashtags	Probability
#nfl	0.2445
#twitnewsnow	0.0923
#fantasyfootball	0.0275
#nba	0.0244
#redskins	0.0229
#playoff	0.0200
#heat	0.0191
#eagles	0.0165
#redskins	0.0163
#lakers	0.0162

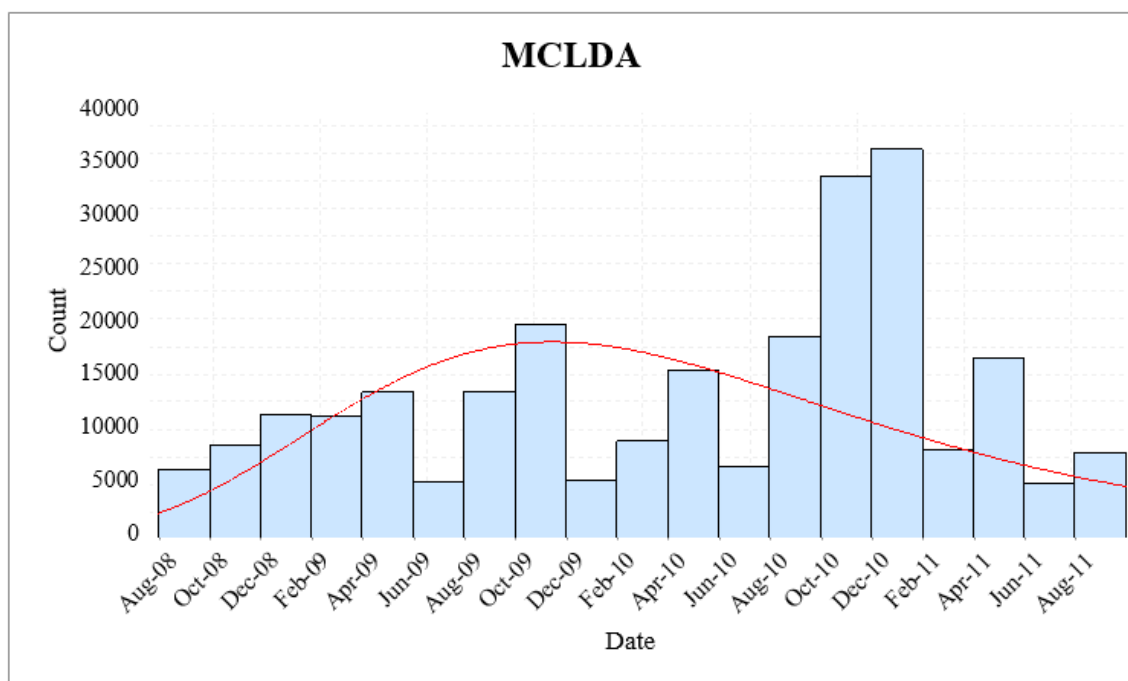


Figure 4.4.3: MCLDA topic 71 distributed over time (The fitted beta PDF is shown by the red line; Beta distribution is fit in a post-hoc fashion).




Table 4.4.29 and Table 4.4.30 shows the TOT-MCLDA text word and hashtag distribution of topic 6. The topic is mainly talking about the death of English singer Amy Winehouse. She died on July 23<sup>rd</sup> 2011 in London, England. The text word and hashtag distributions both accurately describe some of the keywords related to the topic. In Figure 4.4.4, we can see that TOT-MCLDA successfully localized the topic in time. The peak of the beta distribution is located around August 2011, few days after the tragedy. The similar topic generated by MCLDA is shown in Table 4.4.31 and Table 4.4.32. However, some of the text words and hashtags are irrelevant to the death of Amy Winehouse. For example, #sdcc, which is known as San Diego Comic Con, is highly raised due to its co-occurrence with text words ‘fun’ and ‘awesome’ (11<sup>th</sup> text word). In addition, Figure 4.4.5 shows that there are three peaks throughout the time. The first one is on July 2011. The second and third one is on July 2009 and July 2010, which are the month of San Diego Comic Con 2009 and San Diego Comic Con 2010, respectively. The peak of the beta distribution is also shifted to the date around July and August 2010.

Table 4.4.29: TOT-MCLDA text word distribution of topic 6 sorted by probability

Text Words	Probability
re	0.1387
accounts	0.0451
amy	0.0424
dead	0.0383
winehouse	0.0359
sad	0.0356
rip	0.0341
death	0.0255
died	0.0254
music	0.0217

Table 4.4.30: TOT-MCLDA hashtag distribution of topic 6 sorted by probability

Hashtags	Probability
#amywinehouse	0.0337
#rip	0.0198
#winehouse	0.0148
#sad	0.0141
#death	0.0140
#dead	0.0124
#london	0.0104
#music	0.0102
#sad	0.0010
#nowwatching	0.0010

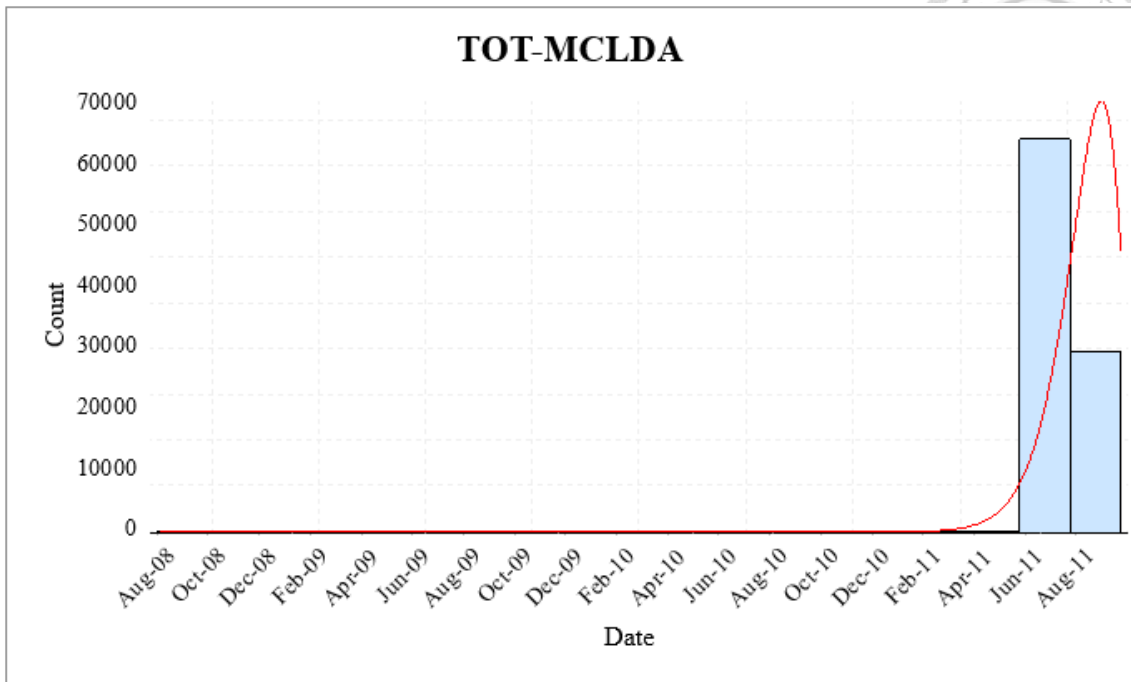


Figure 4.4.4: TOT-MCLDA topic 6 distributed over time (The fitted beta PDF is shown by the red line).

Table 4.4.31: MCLDA text word distribution of topic 96 sorted by probability

Text Words	Probability
amy	0.0226
love	0.0154
winehouse	0.0152
fun	0.0119
re	0.0105
time	0.0102
sad	0.0097
found	0.0096
thanks	0.0091
ll	0.0084

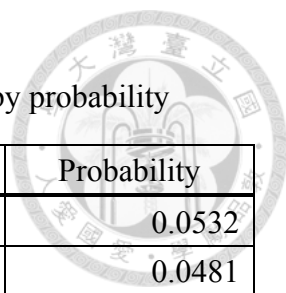


Table 4.4.32: MCLDA hashtag distribution of topic 96 sorted by probability

Hashtags	Probability
#amywinehouse	0.0532
#ff	0.0481
#fb	0.0417
#blogger	0.0412
#rip	0.0286
#sandiego	0.0178
#sdcc	0.0162
#mackidtips	0.0102
#winning	0.0098
#gno	0.0098

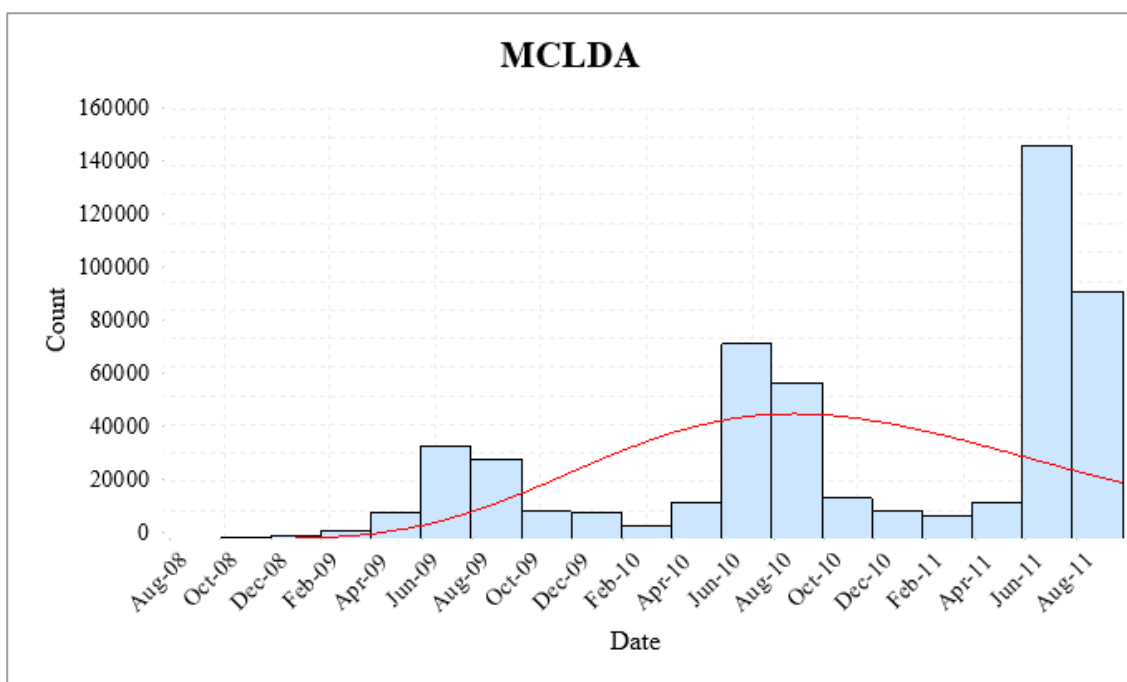
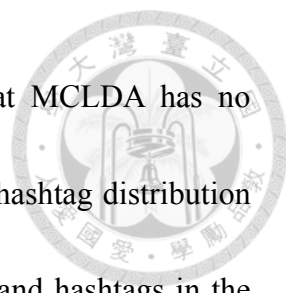


Figure 4.4.5: MCLDA topic 96 distributed over time (The fitted beta PDF is shown by the red line; Beta distribution is fit in a post-hoc fashion).



Topic 120 generated by TOT-MCLDA is a unique topic that MCLDA has no similar one. Table 4.4.33 and Table 4.4.34 show the text word and hashtag distribution of topic 120. It is hard to recognize the topic since the text words and hashtags in the distributions are widely used in lots of situation. However, according to Figure 4.4.6, TOT-MCLDA localized the topic on November. We can therefore infer that the topic is mainly about the SEMA show in Las Vegas, which is known as Specialty Equipment Market Association (SEMA) of the automobile aftermarket. This is the case that TOT-MCLDA makes use of time feature to find patterns hiding in commonly used words.

Table 4.4.33: TOT-MCLDA text word distribution of topic 120 sorted by probability

Text Words	Probability
la	0.0490
car	0.0480
de	0.0450
vegas	0.0276
las	0.0215
el	0.0174
drive	0.0162
en	0.0153
cars	0.0148
auto	0.0137

Table 4.4.34: TOT-MCLDA hashtag distribution of topic 120 sorted by probability

Hashtags	Probability
#cars	0.0855
#vegas	0.0636
#ford	0.0344
#auto	0.0230
#toyota	0.0216
#bmw	0.0170
#car	0.0162
#nissan	0.0161
#lasvegas	0.0155
#lexus	0.0144



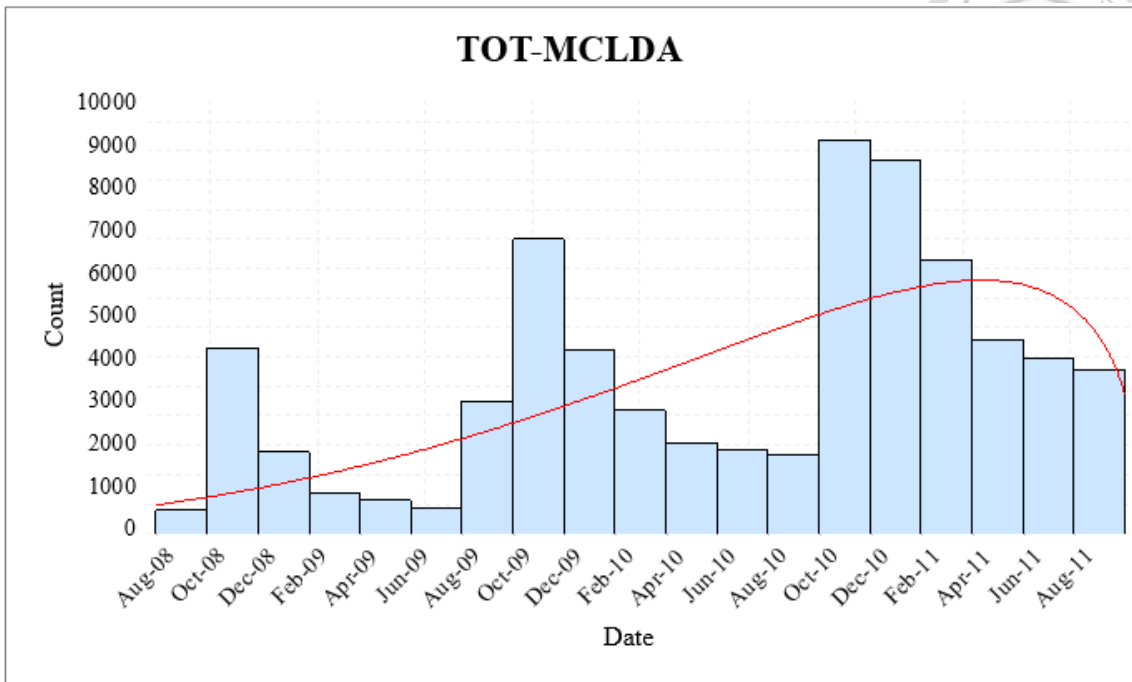
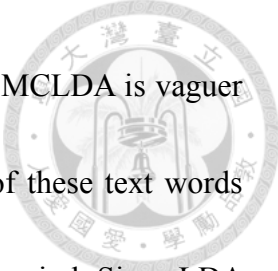


Figure 4.4.6: TOT-MCLDA topic 120 distributed over time (The fitted beta PDF is shown by the red line).



In these three examples, we can see that the topic generated by MCLDA is vaguer than TOT-MCLDA. This is due to the fact that the co-occurrence of these text words and hashtags are frequent even though they are tweeted at different period. Since LDA and MCLDA construct distributions only based on the words co-occurrence, they cannot distinguish the difference. However, TOT-MCLDA is apt to concentrate the distributions based on the time the words appeared. Therefore, the text words distributions and hashtags distributions under same topic are more relevant to each other.

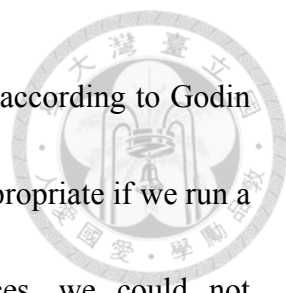
## Chapter 5 Conclusions and Future Work



By making use of time feature, TOT-MCLDA can not only generate more concentrated text word distribution and hashtag distribution over each topic, but also extract a more accurate and stable topic distribution over a target tweet. The probability of appearance of each hashtag is given by multiplying these distributions. According to the experiment results, we can see that TOT-MCLDA performs the best on recommending suitable hashtags to target tweets.

The main contribution of our research is that we introduce a hashtag recommendation system which can automatically generate suitable hashtags to users based on the tweet they post. It may further reduce the problem of lacking hashtags and increase the searchability of tweets. In addition, as more and more users use hashtags, further services can be introduced to the users. For example, products or goods can be recommended based on the hashtags the user uses. The scenario can additionally reinforce the usage of hashtags of users.

There are two main limitations in our research. First, the Twitter dataset is not entirely used. However, limited by the computing resources, the computing time will be too long and the memory will be not enough if we apply the experiment on all the data. Therefore, we could only randomly sample from the dataset and implemented our



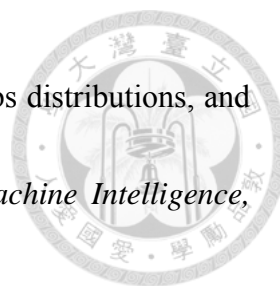
experiment. Second, the number of topics is given by a fix number according to Godin et al.'s (2013) study. Since the dataset is different, it will be more appropriate if we run a sensitive analysis. However, limited by the computing resources, we could not implement our model on topic number over 150. This can be improved if the equipment is updated in the future.

In order to recommend more suitable hashtags, some approaches can be taken in the future. First, since the topics discussed in Twitter are very diverse, the effect of topic number should be evaluated. We can further develop a more flexible model by changing the number of topics automatically. Another approach will be considering more types of feature in tweets to develop a better model. In our research, we introduce MCLDA to incorporate several types of data to form a model. However, we only make use of text words and hashtags. By considering more types of feature, we can reduce the effect of data sparsity in user generated content. Last, we can try to incorporate user preference and feedback to develop a more personalized approach and recommend more user-related hashtags.

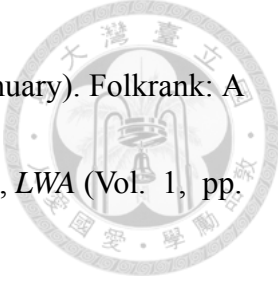
## Reference



- [1] Anderson, C., and Hiralall, M. (2011) Recommender systems for e-shops.
- [2] Balabanović, M., and Shoham, Y. (1997). Fab: content-based, collaborative recommendation. *Communications of the ACM*, 40(3), 66-72.
- [3] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *the Journal of machine Learning research*, 3, 993-1022.
- [4] Breese, J. S., Heckerman, D., and Kadie, C. (1998, July). Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence* (pp. 43-52). Morgan Kaufmann Publishers Inc..
- [5] Chen, Y. H., and George, E. I. (1999, January). A bayesian model for collaborative filtering. In *Proceedings of the 7th International Workshop on Artificial Intelligence and Statistics*. San Francisco: Morgan Kaufman Publishers, [<http://uncertainty99.microsoft.com/proceedings.htm>].
- [6] Erosheva, E., Fienberg, S., and Lafferty, J. (2004). Mixed-membership models of scientific publications. *Proceedings of the National Academy of Sciences of the United States of America*, 101(Suppl 1), 5220-5227.



- [7] Geman, S., and Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6), 721-741.
- [8] Gilks, W. R., Richardson, S., and Spiegelhalter, D. J. Markov Chain Monte Carlo in Practice. 1996. *New York: Chapman Hall/CRC*, 486.
- [9] Godin, F., Slavkovicj, V., De Neve, W., Schrauwen, B., and Van de Walle, R. (2013, May). Using topic models for Twitter hashtag recommendation. In *Proceedings of the 22nd international conference on World Wide Web companion* (pp. 593-596). International World Wide Web Conferences Steering Committee.
- [10] Goldberg, D., Nichols, D., Oki, B. M., and Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12), 61-70.
- [11] Griffiths, T. L., and Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National academy of Sciences of the United States of America*, 101(Suppl 1), 5228-5235.
- [12] Herlocker, J. L., Konstan, J. A., Terveen, L. G., and Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1), 5-53.

- 
- [13] Hotho, A., Jäschke, R., Schmitz, C., and Stumme, G. (2006, January). FolkRank: A ranking algorithm for folksonomies. In K. D. Althoff (Ed.), *LWA* (Vol. 1, pp. 111-114).
- [14] Koren, Y., Bell, R., and Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, 42(8), 30-37.
- [15] Kywe, S. M., Hoang, T. A., Lim, E. P., and Zhu, F. (2012). On recommending hashtags in twitter networks. In *Social Informatics* (pp. 337-350). Springer Berlin Heidelberg.
- [16] Li, R., Wang, S., Deng, H., Wang, R., & Chang, K. C. C. (2012, August). Towards social user profiling: unified and discriminative influence model for inferring home locations. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 1023-1031). ACM.
- [17] Liu, J. S. *Monte Carlo Strategies in Scientific Computing*. 2001. NY: Springer.
- [18] Mazzia, A., and Juett, J. (2009). Suggesting hashtags on twitter.
- [19] Newman, M. E., Barkema, G. T., and Newman, M. E. J. (1999). *Monte Carlo methods in statistical physics* (Vol. 13). Oxford: Clarendon Press.
- [20] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. (1994, October). GroupLens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative*



work (pp. 175-186). ACM.

[21] Salton, G., and McGill, M. J. (1983). Introduction to modern information retrieval.

[22] Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. (2001, April). Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web* (pp. 285-295). ACM.

[23] Uddin, M. M., Hassan, M. T., and Karim, A. (2011, December). Personalized versus non-personalized tag recommendation: A suitability study on three social networks. In *Multitopic Conference (INMIC), 2011 IEEE 14th International* (pp. 56-61). IEEE.

[24] Wang, X., & McCallum, A. (2006, August). 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* (pp. 424-433). ACM.

[25] Zangerle, E., Gassler, W., and Specht, G. (2011). Recommending #-tags in twitter. In *Proceedings of the Workshop on Semantic Adaptive Social Web (SASWeb 2011)*. *CEUR Workshop Proceedings* (Vol. 730, pp. 67-78).



# Appendix A

## Derivative of TOT-MCLDA



Set number of channel  $n = 3$  as an example

1.  $P(z_{ai} = k | z_{a-i}, z_b, z_c, w_a, w_b, w_c, t_a, t_b, t_c)$
2.  $P(z_{bi} = k | z_a, z_{b-i}, z_c, w_a, w_b, w_c, t_a, t_b, t_c)$
3.  $P(z_{ci} = k | z_a, z_b, z_{c-i}, w_a, w_b, w_c, t_a, t_b, t_c)$

1.

$$P(z_{ai} = k | z_{a-i}, z_b, z_c, w_a, w_b, w_c, t_a, t_b, t_c) \propto (1) * (2) * (3)$$

$$(1) P(w_{ai} | z_{ai} = k, z_{a-i}, z_b, z_c, w_{a-i}, w_b, w_c)$$

$$(2) P(z_{ai} = k | z_{a-i}, z_b, z_c, w_{a-i}, w_b, w_c)$$

$$(3) P(t_{ai} | z_{ai} = k, t_a, t_b, t_c)$$

(1)

$$\begin{aligned} & P(w_{ai} | z_{ai} = k, z_{a-i}, z_b, z_c, w_{a-i}, w_b, w_c) \\ &= \int P(w_{ai} | z_{ai} = k, \Phi_a^{(k)}) P(\Phi_a^{(k)} | z_{a-i}, w_{a-i}) d\Phi_a^{(k)} \\ &= \int \Phi_{a, w_{ai}}^{(k)} P(\Phi_a^{(k)} | z_{a-i}, w_{a-i}) d\Phi_a^{(k)} \end{aligned}$$

$$P(\Phi_a^{(k)} | z_{a-i}, w_{a-i}) \propto P(w_{a-i} | \Phi_a^{(k)}, z_{a-i}) P(\Phi_a^{(k)})$$



$$\propto \prod_{v=1}^{V_a} \Phi_{a,v}^{(k)} \beta_a^{-1+n_{(-i),(k)}^{(A),(v)}} \propto \text{Dirichlet}(\beta_a + n_{(-i),(k)}^{(A),(v)})$$

$$\begin{aligned} & P(w_{ai} | z_{ai} = k, z_{a-i}, z_b, z_c, w_{a-i}, w_b, w_c) \\ & \propto \int \Phi_{a,w_{ai}}^{(k)} \text{Dirichlet}(\beta_a + n_{(-i),(k)}^{(A),(v)}) d\Phi_a^{(k)} \\ & = \frac{\beta_a + n_{(-i),(k)}^{(A),(w_{ai})}}{\sum_{v=1}^{V_a} (\beta_a + n_{(-i),(k)}^{(A),(v)})} = \frac{\beta_a + n_{(-i),(k)}^{(A),(w_{ai})}}{V_a \beta_a + n_{(-i),(k)}^{(A),(.)}} \end{aligned}$$

(2)

$$\begin{aligned} & P(z_{ai} = k | z_{a-i}, z_b, z_c, w_{a-i}, w_b, w_c) \\ & = \int P(z_{ai} = k | \theta^{(d)}) P(\theta^{(d)} | z_{a-i}, z_b, z_c, w_{a-i}, w_b, w_c) d\theta^{(d)} \\ & = \int \theta_{z_{ai}}^{(d)} P(\theta^{(d)} | z_{a-i}, z_b, z_c, w_{a-i}, w_b, w_c) d\theta^{(d)} \end{aligned}$$

$$\begin{aligned} & P(\theta^{(d)} | z_{a-i}, z_b, z_c, w_{a-i}, w_b, w_c) \propto P(z_{a-i}, z_b, z_c | \theta^{(d)}) P(\theta^{(d)}) \\ & = P(z_{a-i} | \theta^{(d)}) P(z_b | \theta^{(d)}) P(z_c | \theta^{(d)}) P(\theta^{(d)}) \\ & \propto \prod_{k=1}^K \theta_k^{(d) \alpha - 1 + n_{(-i),(k)}^{(da),(.)} + n_{(.), (k)}^{(db),(.)} + n_{(.), (k)}^{(dc),(.)}} \\ & \propto \text{Dirichlet}(\alpha + n_{(-i),(k)}^{(da),(.)} + n_{(.), (k)}^{(db),(.)} + n_{(.), (k)}^{(dc),(.)}) \end{aligned}$$

$$\begin{aligned} & P(z_{ai} = k | z_{a-i}, z_b, z_c, w_{a-i}, w_b, w_c) \\ & \propto \int \theta_{z_{ai}}^{(d)} \text{Dirichlet}(\alpha + n_{(-i),(k)}^{(da),(.)} + n_{(.), (k)}^{(db),(.)} + n_{(.), (k)}^{(dc),(.)}) d\theta^{(d)} \end{aligned}$$



$$\begin{aligned}
&= \frac{\alpha + n_{(-i),(k)}^{(da),(\cdot)} + n_{(\cdot),(k)}^{(db),(\cdot)} + n_{(\cdot),(k)}^{(dc),(\cdot)}}{\sum_{k=1}^K \left( \alpha + n_{(-i),(k)}^{(da),(\cdot)} + n_{(\cdot),(k)}^{(db),(\cdot)} + n_{(\cdot),(k)}^{(dc),(\cdot)} \right)} \\
&= \frac{\alpha + n_{(-i),(k)}^{(da),(\cdot)} + n_{(\cdot),(k)}^{(db),(\cdot)} + n_{(\cdot),(k)}^{(dc),(\cdot)}}{K\alpha + n_{(-i),(\cdot)}^{(da),(\cdot)} + n_{(\cdot),(\cdot)}^{(db),(\cdot)} + n_{(\cdot),(\cdot)}^{(dc),(\cdot)}}
\end{aligned}$$

(3)

$$\begin{aligned}
&P(t_{ai} | z_{ai} = k, t_a, t_b, t_c) \\
&= P(t_{ai} | z_{ai} = k, \Psi_a^{(k)}) P(\Psi_a^{(k)} | t_{a-i}, t_b, t_c) \\
&= \frac{t_{ai}^{\psi_{k1}-1} (1 - t_{ai})^{\psi_{k2}-1}}{B(\psi_{k1}, \psi_{k2})}
\end{aligned}$$

$P(z_{ai} = k | z_{a-i}, z_b, z_c, w_a, w_b, w_c, t_a, t_b, t_c)$

$$\propto \frac{\beta_a + n_{(-i),(k)}^{(A),(w_{ai})}}{V_a \beta_a + n_{(-i),(k)}^{(A),(\cdot)}} \frac{\alpha + n_{(-i),(k)}^{(da),(\cdot)} + n_{(\cdot),(k)}^{(db),(\cdot)} + n_{(\cdot),(k)}^{(dc),(\cdot)}}{K\alpha + n_{(-i),(\cdot)}^{(da),(\cdot)} + n_{(\cdot),(\cdot)}^{(db),(\cdot)} + n_{(\cdot),(\cdot)}^{(dc),(\cdot)}} \frac{t_{ai}^{\psi_{k1}-1} (1 - t_{ai})^{\psi_{k2}-1}}{B(\psi_{k1}, \psi_{k2})}$$

$P(z_{bi} = k | z_a, z_{b-i}, z_c, w_a, w_b, w_c, t_a, t_b, t_c)$

$$\propto \frac{\beta_b + n_{(-i),(k)}^{(B),(w_{bi})}}{V_b \beta_b + n_{(-i),(k)}^{(B),(\cdot)}} \frac{\alpha + n_{(\cdot),(k)}^{(da),(\cdot)} + n_{(-i),(k)}^{(db),(\cdot)} + n_{(\cdot),(k)}^{(dc),(\cdot)}}{K\alpha + n_{(\cdot),(\cdot)}^{(da),(\cdot)} + n_{(-i),(\cdot)}^{(db),(\cdot)} + n_{(\cdot),(\cdot)}^{(dc),(\cdot)}} \frac{t_{bi}^{\psi_{k1}-1} (1 - t_{bi})^{\psi_{k2}-1}}{B(\psi_{k1}, \psi_{k2})}$$

$P(z_{ci} = k | z_a, z_b, z_{c-i}, w_a, w_b, w_c, t_a, t_b, t_c)$

$$\propto \frac{\beta_c + n_{(-i),(k)}^{(C),(w_{ci})}}{V_c \beta_c + n_{(-i),(k)}^{(C),(\cdot)}} \frac{\alpha + n_{(\cdot),(k)}^{(da),(\cdot)} + n_{(\cdot),(k)}^{(db),(\cdot)} + n_{(-i),(k)}^{(dc),(\cdot)}}{K\alpha + n_{(\cdot),(\cdot)}^{(da),(\cdot)} + n_{(\cdot),(\cdot)}^{(db),(\cdot)} + n_{(-i),(\cdot)}^{(dc),(\cdot)}} \frac{t_{ci}^{\psi_{k1}-1} (1 - t_{ci})^{\psi_{k2}-1}}{B(\psi_{k1}, \psi_{k2})}$$



Prove likewise, given a number of channel, if we focus on channel “r”:

$$P(z_{ri} = k | z_a, z_b, z_c, \dots, z_{r-i}, \dots, w_a, w_b, w_c, \dots, w_r, \dots, t_a, t_b, t_c, \dots, t_r, \dots)$$

$$\propto \frac{\beta_r + n_{(-i),(k)}^{(R),(w_{ri})}}{V_r \beta_r + n_{(-i),(k)}^{(R),(c)}} \frac{\alpha + n_{(c),(k)}^{(da),(c)} + \dots + n_{(-i),(k)}^{(dr),(c)} + \dots}{K\alpha + n_{(c),(c)}^{(da),(c)} + \dots + n_{(-i),(c)}^{(dr),(c)} + \dots} t_{ri}^{\psi_{k1}-1} (1 - t_{ri})^{\psi_{k2}-1} B(\psi_{k1}, \psi_{k2})$$

Where  $\psi_{k1}$  and  $\psi_{k2}$  are updated by the method of moment

$$\psi_{k1} = \bar{t}_k \left( \frac{\bar{t}_k(1 - \bar{t}_k)}{s_k^2} - 1 \right)$$

$$\psi_{k2} = (1 - \bar{t}_k) \left( \frac{\bar{t}_k(1 - \bar{t}_k)}{s_k^2} - 1 \right)$$

$\bar{t}_k$  is the sample mean of the timestamps w.r.t. topic k

$s_k^2$  is the biased sample variance of the timestamps w.r.t. topic k