World Wide Web
DOI 10.1007/s11280-016-0417-x

Using time-sensitive interactions to improve topic derivation in twitter

Robertus Nugroho[1] · Weiliang Zhao[2] · Jian Yang[2] · Cecile Paris[3] · Surya Nepal[3]

Received: 29 February 2016 / Revised: 31 August 2016 / Accepted: 8 September 2016

Abstract Twitter has become one of the most popular social media platforms, widely used for discussion and information dissemination on all kinds of topics. As a result, both business and academics have researched methods to identify the topics being discussed on Twitter. Those methods can be employed for a number of applications, including emergency management, advertisements, and corporate/government communication. However, deriving topics from this short text based and highly dynamic environment remains a huge challenge. Most current methods use the content of tweets as the only source for topic derivation. Recently, tweet interactions have been considered for improving the quality of topic derivation. In this paper, we propose a method that considers both content and interactions with a temporal aspect to further improve the quality of topic derivation. The impact of the temporal aspect in user/tweet interactions is analyzed based on several Twitter datasets. The proposed method incorporates time when it clusters tweets and identifies representative terms for each topic. Experimental results show that the inclusion of the temporal aspect

✉ Robertus Nugroho
robertus.nugroho@students.mq.edu.au

Weiliang Zhao
weiliang.zhao@mq.edu.au

Jian Yang
jian.yang@mq.edu.au

Cecile Paris
cecile.paris@data61.csiro.au

Surya Nepal
surya.nepal@data61.csiro.au

[1] Department of Computing, Macquarie University and CSIRO Data61, Sydney, NSW, Australia

[2] Department of Computing, Macquarie University, Sydney, NSW, Australia

[3] CSIRO Data61 Australia, Sydney, NSW, Australia

in the interactions results in a significant improvement in the quality of topic derivation comparing to existing baseline methods.

Keywords Topic derivation · Temporal aspect in twitter · Joint matrix factorization

1 Introduction

Twitter has evolved from a microblogging platform to a medium that enables people to interact with each other in a conversation-like manner. With more than 300 million monthly active users[1], Twitter becomes one of the most popular social media platforms and it provides real-time information and opinions [11]. Topic derivation from Twitter, to understand what people are talking about, is essential for a wide range of applications such as emergency, social awareness, health monitoring, and market analysis, and it is of interest to many organizations [33].

Topic derivation is the process of determining the main topics of a collection of Twitter messages (tweets) and choosing a set of terms to represent each topic [20]. Deriving topics from Twitter is challenging for two reasons: firstly, tweets are short (140 characters maximum) and often include informal language (e.g., emoticons, abbreviations) and misspellings, leading to a sparsity problem when considering term co-occurrences in tweets. Secondly, the Twitter environment is highly dynamic with topics changing quickly over time.

Existing topic derivation methods based on term co-occurrences, such as LDA [2], PLSA [9] and NMF [16], suffer from sparsity problem. The relationship between correlated terms has been exploited for addressing this problem [10, 35, 36]. Due to the fact that the original tweet content is used as the only source of information, the sparsity problem remains. Vosecky et al. [32] proposed a method to incorporate linked external resources to augment the tweet content. The study in [24] went beyond terms and exploited content based social features such as hashtag, emoticons, and urls. These approaches have not considered the information hidden in the social interactions amongst posts in the Twitter environment.

In our previous work [23], we proposed a topic derivation model that exploits both interaction features and content similarity. The intuition behind the use of these interaction features such as *mention*, *reply*, and *retweet* to identify topics is that they are typically employed to indicate that the posts are part of a conversation, and all posts pertaining to a conversation are likely to be on the same topic. Our experiments showed that the model resulted in better quality in comparison with existing work in topic derivation. Our method, however, did not take time into consideration. As Twitter is a highly dynamic environment, this omission potentially reduces the performance of the method in terms of topic quality. To address the dynamic aspect of Twitter, some approaches have exploited temporal aspect of to the tweet content or associated hashtags, e.g., [3, 26], and [29]. To the best of our knowledge, the temporal aspect of the posts' *interactions* have not been explored for topic derivation in a collection of tweets.

While taking conversations into account as discussed in [23] can improve topic derivation quality, conversations typically are time-sensitive. For example, two tweets with the mentions of the same users nearly at the same time are more likely to be about the same topic than two posts with mentions of same users within a long time interval. Therefore

[1] https://about.twitter.com/company, accessed 9 February 2016.

incorporating the temporal aspect when looking at the interactions may further improve the quality of topic derivation.

We reported our preliminary results of incorporating the temporal aspect in topic derivation in [22]. This paper provides a comprehensive description of the proposed approach with several major extensions including (1) adding a statistical analysis of the impact of time on tweet clustering, particularly for 'mention', (2) adding a detailed explanation of the update rules for the tweet-topic matrix and topic-term matrix, (3) performing an additional annotation task for our *tweetMarch* dataset, this time with an analysis of inter-annotator agreement, (4) providing additional experimental results over two publicly available datasets *TREC2014* and *tweetSanders*, (5) conducting an analysis of our proposed method to deal with the varying nature of topics in the timeline, (6) discussing the rationale of the incorporation of temporal aspects in tweet interactions, and (7) providing a comprehensive motivating example. This work is summarized as:

- We discuss the relationships between topics and interaction features (*mention*, *reply* and *retweet*) using a dataset obtained by collecting tweets over a month. We found that the *mention* is time-sensitive with respect to topic assignation.
- We model the time sensitivity of *mention* as an exponential decay according to the time difference of two tweets with the same mention. The decay parameter is based on an analysis of tweets that include a mention. This time sensitivity model is then incorporated in the tweet relationship model in order to influence the matrix inter-joint factorization for topic derivation.
- We conducted a comprehensive set of experiments to evaluate the proposed new model with three different Twitter datasets, using widely accepted evaluation metrics for topic derivation. The results show that the new time-sensitive method results in a significant improvement of the quality of topic derivation comparing with well-known baseline methods and our previous work [23].
- We also performed the evaluation of our method by scrutinizing tweets grouped in a series of time periods. The results show that our proposed method can cope with the dynamic tweet stream better than the baseline methods.

The rest of the paper is organized as follows. Section2 provides a motivating example. Section 3 analyses the different temporal sensitivities of *mentions*, *replies*, and *retweets*. Section 4 explains a method to measure the relationships between tweets by incorporating the temporal aspect. Section 5 describes the topic derivation process which incorporates a time aspect when considering the interactions amongst tweets. Section 6 reports on a series of experiments. Related work is discussed in Section 7, and we conclude in Section 8.

2 Motivating example

Twitter adopts a non-mutual relationship between users through the following-follower mechanism.[2] User a can follow user b, but user b does not need to follow back user a. When user a follows user b, user a is subscribed to user b's tweets. All of the Twitter posts from user b will appear on the home timeline[3] of user a. The default setting of a Twitter user account is *public*. As long as it is not changed to *private*, other users are able to see all

[2]Twitter FAQs about following (https://support.twitter.com/articles/14019, accessed 4 February 2016).

[3]What's a Twitter timeline? (https://support.twitter.com/articles/164083, accessed 4 February 2016).

Table 1 Tweet examples

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

of his/her posts. This default setting allows Twitter users to interact with other users using *reply*, *retweet*, or *mention*, when they do not follow each other. These interactions form the implicit relationships between tweets in a particular conversation or about a particular topic.

Table 1 shows some tweet examples that illustrate typical interactions between users. 7 users are involved within these 6 tweets. A user can post a 'self-contained' tweet [6], i.e., a tweet that has no reference to other tweets except potentially through the same hashtag. t_1 is an example of a self-contained tweet. A tweet then can be "replied to" by other users, or retweeted. Users can also initiate a conversation using 'mention' within their posts.

In Table 1, t_2 is an example of a reply to t_1. A reply tweet usually starts with the author's username of the original tweet. More specific information about the reply status, such as the replied tweet id and the replied tweet author id can be found from the new tweet's fields[4] in a *JSON* format.[5] A *reply* turns up in a discussion between users. t_3 is a retweet of t_1. A retweet is a mechanism to re-post/share another user's tweet. Usually, a retweet has a 'RT' in the beginning of the text to indicate that the tweet is a re-post of another user's tweet. More detailed information about the retweet status is available from the tweet's field. Finally, t_4, t_5, and t_6 are examples of tweets that use the mention feature. Tweets with mentions contain another user's username in their text content. t_4, t_5, and t_6 mention *@user5* in their tweets. If we only read the text, it is difficult to differentiate between mention and reply. A reply itself can be seen as a mention due to the availability of the username in the text. If we look at the tweet's field, we can differentiate a reply tweet from a mention, as the latter has an empty reply status. A mention is used to involve other users in a discussion on a particular topic. A tweet can also contain a *hashtag*, which is a word starting with the # symbol. In Table 1, t_4 is an example of a tweet that has several hashtags (*#polo, #giddyup, #youcantpolosolo*). Hashtags have been widely adopted in social networks to bookmark the content, or to associate user interests in particular topics [37]. However hashtags may not be

[4] https://dev.twitter.com/overview/api/tweets, accessed 6 February 2016.

[5] JSON (JavaScript Object Notation), is a syntax for storing and exchanging data. It is an easier-to-use alternative to XML. (http://www.w3schools.com/json/default.asp, accessed 6 February 2016).

directly linked to topics. For example, if we have a hastagh *#Adelaide*, it indicates a location rather than a topic.

Figure 1 provides a graphical illustration of the relationships between the tweets shown in Table 1. In this figure, we group all tweets in the collection into three time windows based on their timestamps. The first time window is the tweets that were posted between 5.30 PM to 6.00 PM in 12 January 2015. t_1 and t_2 are in this time window. The second time window is between 6.30 PM to 7.00 PM in 12 January 2015. t_3 and t_4 are in this time window. The last time window is between 11.30 AM and 12.00 PM in 13 January 2015. t_5 and t_6 are in this time window.

In Figure 1, tweets t_1 and t_2 are related to each other since t_2 is a reply of tweet t_1. t_3 is related to t_1 as its retweet, although they are in different time windows. t_1, t_2, and t_3 are

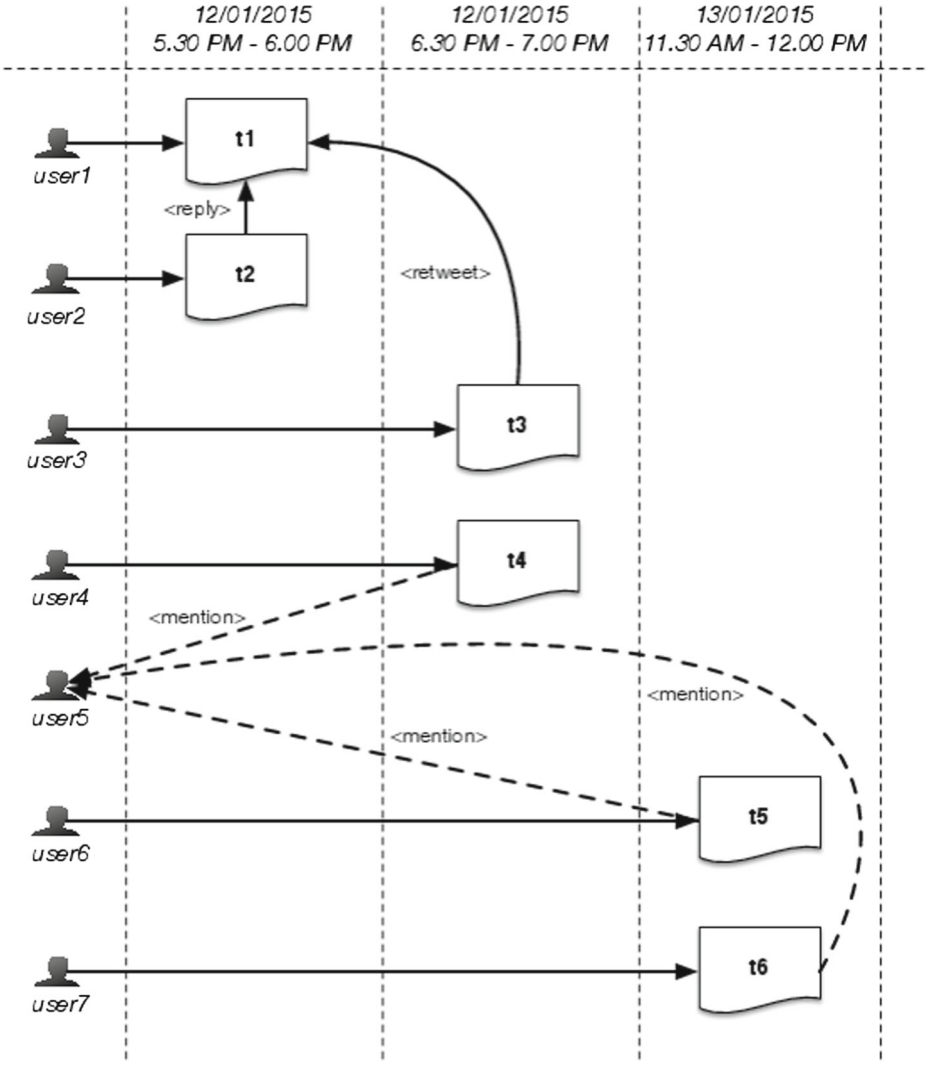


Figure 1 Relationships between tweets based on interactions

talking about the same topic: '*pizza*'. We can see that both replies and retweets are likely to be on the same topic as the original post. Replies and retweets are explicit interaction between two tweets.

t_4, t_5, and t_6 are connected to each other due to the fact that they mention the same user (*@user5*). But t_4 talks about a completely different topic than t_5 and t_6. Tweet t_4 talks about the 'shirt label', while t_5 and t_6 talk about 'Portarlington Mussel Festival'. If we look at the tweets' timestamp in Table 1, we find that t_5 and t_6 have little time difference if compared to t_4 posted a day before. These examples show that the mention feature can help to determine the topical relationship between tweets, although they do not share similar terms. They also illustrate that two tweets which mention the same user are likely to be on the same topic *if* they occur around the same time. Time thus plays an important role when attempting to link tweets that mention the same people.

The tweet set in Table 1 and its graphical illustration in Figure 1 show the importance of the mention, reply and retweet in building relationship between tweets to group them into topics. By understanding the connections between tweets, topic derivation in Twitter becomes more accurate despite the minimal terms overlapping frequency. The examples above show that the time should be taken into account in order to improve the quality of topic derivation in the highly dynamic Twitter environment. In the next section, we investigate how time impacts on the interactions when comes to grouping tweets.

3 Time in tweet interactions: an analysis

In this section, we analyze the impact of time on user interactions for the same topic, based on *mention*, *reply*, and *retweet*. We analyze tweets of a Twitter dataset to see how time

Table 2 Top 15 Twitter users in Australia and all related tweets (i.e., tweets that involve these top 15 Twitter users, either by mentioning them, replying to them or retweeting their posts) between Jan 12, 2015 and Feb 12, 2015

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

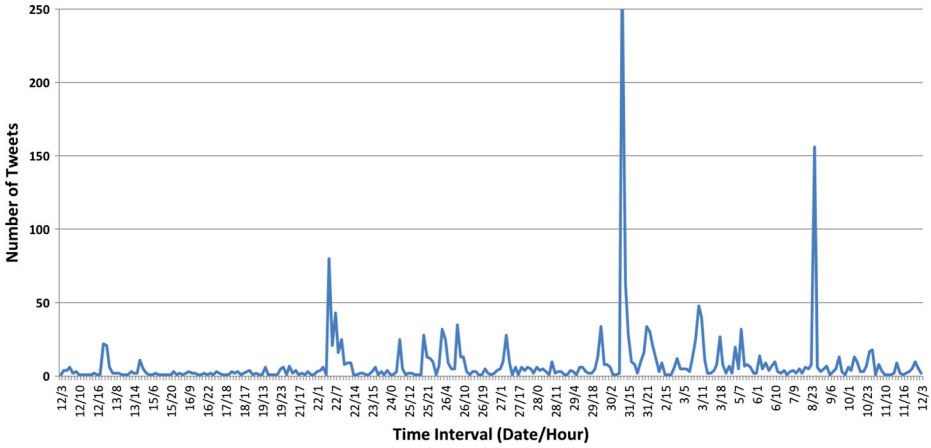


Figure 2 Tweets mentioning user @MrKRudd with a 3 hour time intervals

affects the topic similarity between tweets. We obtained the dataset as follows. Using the Twitter's streaming API,[6] we retrieved all tweets from the top 15 Twitter users in Australia[7] in January 2015 and all the tweets that mention those users (including reply and retweet tweets) during the period of January 12, 2015 to February 12, 2015. The resulting dataset consists of more than 6 million tweets, with about 800 thousand users. The details of the dataset are shown in Table 2.

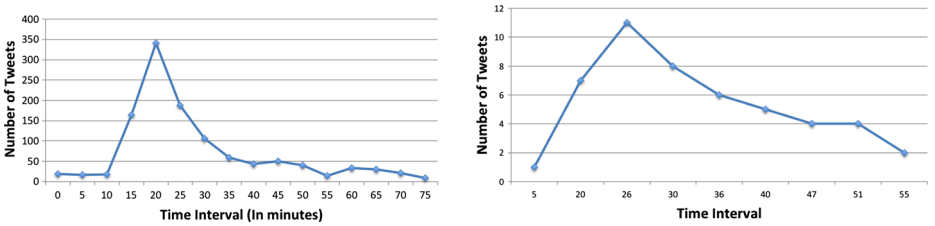
Our investigation starts with an analysis of individual user mentions at different level of time granularity to see how mentions are distributed over time. We look at the topics in the dataset to see if there is a relationship between mentions and topics. We find that, for all users, when the number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

As an example, Figure 2 shows the distributions of the tweets that mention *@MrKRudd* in a 3 hour time interval. We can see that there are several fluctuations within different time intervals. We find that each peak in Figure 2 (an indication of a sharp increase in the number of tweets mentioning *@MrKRudd*) is strongly related to a particular topic. For example, on January 22, 2015 at 7am (22/7), most of the tweets mentioning *@MrKRudd* were talking about the *"plain packaging act"*. The tweets at 3 PM on January 31, 2015 were about *"Queensland votes"*, and the tweets at 11 PM on February 08, 2015 were about "*the end of Kevin Rudd's leadership in February 2012*".

We see from the figure that the number of tweets with the same mention reaches a peak and then fades away (decay). Figure 3 shows the subset of the distributions of the tweets that mention (a) *@CodySimpson* and (b) *@MClarke23* on 5-minute intervals. The specific distributions are different, reaching their peaks and decaying at different rates. What they have in common, however, is that each peak indicates a specific topic. The peak in Figure 3a is related with the topic: "*Cody's birthday*"; and the peak in Figure 3b is related with the topic: "*the absence of Michael Clarke on treatment issue*".

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

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



(a) @CodySimpson (b) @MClarke23

Figure 3 Tweet distributions of tweets mentioning (**a**) *@CodySimpson* and (**b**) *@MClarke23* on 5 minutes time intervals within 1 hour

We perform a statistical analysis on all the variations of the tweet distributions, using a 5 minutes interval. We sum up the number of tweets from all users by choosing the subset of the tweet distributions starting from the closest lowest point before a peak and ending at the lowest point after the peak. Figure 4 shows this sum. We can see that most of the mentions related to a particular topic reach a peak in about 15 minutes and then gradually fade away. An exponential function is adopted to model the process of fading away. The exponential function has a parameter to control its decaying. This parameter is how long the mention frequency decays from its peak to the peak's half value. It can be expressed as:

$$a = i_{t_{max}/2} - i_{t_{max}} \tag{1}$$

where $i_{t_{max}}$ is the time when the tweet mention distribution reaches its peak, and $i_{t_{max}/2}$ is the time when the tweet mention distribution reaches half of the peak value after the peak. In Figure 4, the number of tweets in the highest point (t_{max}) is 367,368, and it is reached after 15 minutes ($i_{t_{max}}$). Then, $i_{t_{max}/2}$ is calculated as the time to reach 183,684 after the peak, which is 37 minutes. So, a for Figure 4 will be 22 minutes (1,320 seconds). This a will be used in the exponential function that models the temporal aspect in the mention behavior in Twitter.

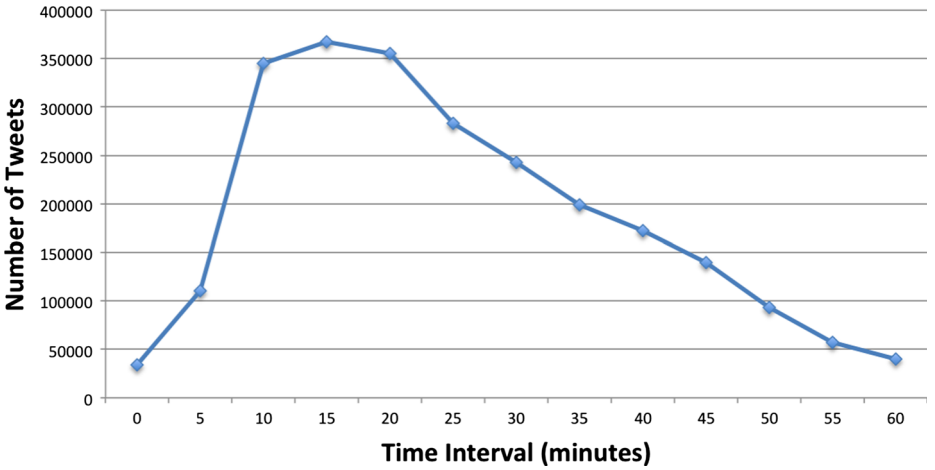
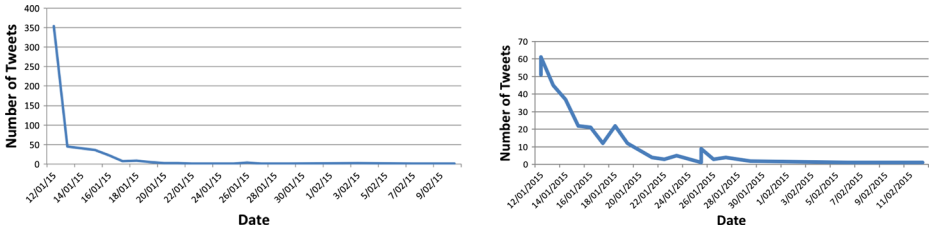


Figure 4 The sum of all fluctuations in all tweet mention distributions with 5-minute time intervals



(a) RT of @CodySimpson (b) RT to @luke_brooks

Figure 5 Tweet distributions of retweet to a tweet by (**a**) @*CodySimpson* and (**b**) @*luke_brooks* within a 1 month period

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

Figure 5a shows the tweet distributions of a retweet to a tweet by @*CodySimpson*: ("*It's the 11th back home in Aus. I m officially 18.*"). The tweet was retweeted for 494 times in total, with 354 retweets on the first day, 22 on the third day, and the remaining scattered over time. The tweet distributions of a retweet to tweet by @*luke_brooks* shown in Figure 5b shows similar trends. The highest number of retweet happened in the first day with around 112 tweets, 61 tweets on the second day, and still 45 tweets on the third day. The original tweet is still being retweeted several times during this 1 month period. Irrespective of the time elapsed, the retweets are still on the same topic.

Figure 6 shows the tweet distribution of the replies to a tweet by @5SOS ("*Getting lots and lots of ideas for songs! Ready to write a new record!!*"). The total number of replies was 7414 tweets, with a peak on the first day but continuing the following day (with still 291 replies on the next day). The analysis of the reply and retweet behavior supports our previous statement that both replies and retweets can be classified as explicit interaction

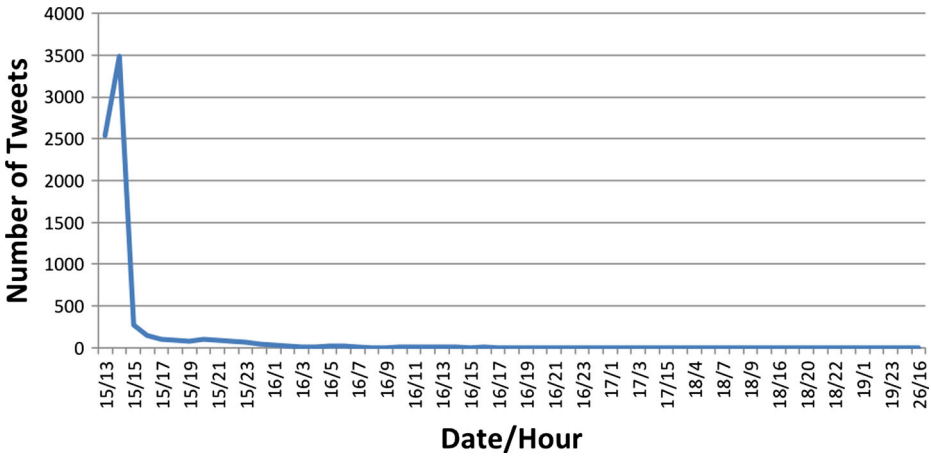


Figure 6 Tweet distribution of a reply to a tweet by *@5SOS* within a 1 month period

between two tweets which show the participation of users in a discussion about particular topic.

4 Measuring relationships between tweets

A tweet is represented by a tuple $t = \langle P_t, rtp_t, C_t, i_t \rangle$, where P_t is the union of the author and people mentioned in the tweets, rtp_t is the reply and retweet information, C_t is the set of terms contained in the tweet (including hashtags), and i_t is the timestamp of the tweet. We denote a relationship between two tweets t_i and t_j as $R(t_i, t_j)$. A zero value (0) of R means that there is no relation between them, and a higher value indicates the relationship is stronger. The relationship R is constructed based on the combination of three components as interactions based on people ($po(P_{t_i}, P_{t_j})$), common actions ($act(rtp_{t_i}, rtp_{t_j})$), and content similarity ($sim(C_{t_i}, C_{t_j})$). It is expressed as:

$$R(t_i, t_j) = po(P_{t_i}, P_{t_j}) + act(rtp_{t_i}, rtp_{t_j}) + sim(C_{t_i}, C_{t_j}) \, . \tag{2}$$

Interaction based on people $po(P_{t_i}, P_{t_j})$ is modeled as the number of common mentioned people in tweets t_i and t_j divided by the total number of people involved in both tweets. As discussed in Section 3, tweets that mention similar users within a particular period are more likely to share the same topic. So, for the definition of interactions based on people, we improve on our previous work [23] by adding a temporal factor $f(i_{t_i} - it_j)$. The people based interaction is now calculated as follows:

$$po(P_{t_i}, P_{t_j}) = \frac{|P_{t_i} \cap P_{t_j}|}{|P_{t_i} \cup P_{t_j}|} f(i_{t_i} - it_j)$$

$$where \quad f(i_{t_i} - it_j) = e^{-\frac{1}{a}|i_{t_i} - i_{t_j}|}, \tag{3}$$

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

The interaction based on user actions, denoted as $act(rtp_{t_i}, rtp_{t_j})$, is based on the *retweet* and *reply* relationship between two tweets. As previously discussed, temporal aspect has no effect on these relationships. Replies and retweets are a clear indication of discussion activities on the same topic. If tweet A is a *retweet* or *reply* of tweet B (or vice versa), or if both tweets are *replying* to or *retweeting* the same tweet, $act(rtp_{t_i}, rtp_{t_j})$ will be 1 (indicating a strong relationship), otherwise it is 0. We denote rtp_t as the ID of the retweeted or replied tweet in tweet t.

$$act(rtp_{t_i}, rtp_{t_j}) = \begin{cases} 1, (rtp_{t_i} = j) \; or \; (i = rtp_{t_j}) \\ \quad or \; (rtp_{t_i} = rtp_{t_j}) \\ 0, \; otherwise \end{cases} \tag{4}$$

As there are a large number of self-contained tweets (i.e., tweets with no relation to any other tweet), our model for topic derivation takes content similarity between tweets into account. Before calculating the content similarity, we perform some preprocessing steps to remove all terms/characters that potentially lower the topic identification processes performance, such as emoticons, punctuations and terms with fewer than 3 characters. As we consider hashtaghs as an important part of the text that helps identify users' interests on particular topics, hashtags are included and kept unchanged. We do not normalize the hashtags into their original words (e.g., '#youcantpolosolo' to 'you cant polo solo'), to avoid losing the connectivity between tweets that have the same unique hashtag. As a result, the formula

treats hashtags and its plain words differently (e.g. '#Sydney' and 'Sydney'). In this preprocessing step, we also remove stop-words, so only the content-full words are left in the tweet. All terms in the tweets collection are then stemmed and tokenized. As tweets are short, two tweets sharing at least one (non-stop) word are likely to be on the same topic. $sim(C_{t_i}, C_{t_j})$ denotes the similarity between tweets t_i and t_j, measured by *cosine similarity* [27].

$$\begin{aligned} sim(C_{t_i}, C_{t_j}) &= \frac{C_{t_i}.C_{t_j}}{\|C_{t_i}\|\|C_{t_j}\|} \\ &= \frac{\sum_{x=1}^{n}(C_{t_i})_x \times (C_{t_j})_x}{\sqrt{\sum_{x=1}^{n}((C_{t_i})_x)^2} \times \sqrt{\sum_{x=1}^{n}((C_{t_j})_x)^2}} \end{aligned} \tag{5}$$

$R(t_i, t_j)$ represents the strength of the relationships between tweets t_i and t_j. The mention based interaction, the action based interaction, and the content similarity contribute to the $R(t_i, t_j)$ value. Tweets that are connected by the action based interaction normally have a high degree of content similarity, but it is not always true. For example, in a reply, the reply tweet may have a totally different content comparing with that of the original tweet. A tweet and its reply definitely have a relationship. If only content similarity is considered, this relationship would be lost. Furthermore, if a tweet and its reply have a higher degree of content similarity, the relationship will be stronger. The combination of the three components makes the $R(t_i, t_j)$ covers a wide range of situations when quantifying tweet relationships.

The values of all the relationships amongst tweets form a tweet-to-tweet relationship matrix $A \in \mathbb{R}^{m \times m}$, where $a_{ij} = f(R(t_i, t_j))$. $f(R(t_i, t_j))$ is a *sigmoid function* [31] that normalizes the value of $R(t_i, t_j)$ for a smoother relationship distribution.

$$f(R(t_i, t_j)) = \begin{cases} \frac{1}{1+e^{-R(t_i,t_j)}}, R(t_i, t_j) > 0 \\ 0, \ otherwise \end{cases} \tag{6}$$

By incorporating a temporal factor in people based interactions, we obtain a more accurate tweet-to-tweet relationship matrix. This matrix will be used to improve the topic derivation by jointly factorizing it with tweet-to-term matrix as discussed in the next section.

5 Matrix *inter-joint* factorization for topic derivation

LDA [2] based methods have been very popular to derive topics from document collections. With the 'bag of words' assumption, LDA models a document as a mixture of topics. It relies on the frequency of overlapping terms, which unfortunately is very low in Twitter. Approaches that extend LDA for the Twitter environment still mostly focus on exploiting tweets' content, or only consider limited interaction features. As a result, they still suffer from the sparsity issue.

In our previous work [20], we proposed the *eLDA* and *intLDA* methods to deal with the sparsity problem. eLDA expands the tweet content by adding non-existent words from all connected tweets. The connections are built up based on interaction features. The intLDA directly incorporates the observed tweet relationships in the process of learning the tweet-topic distribution θ. The relationships between tweets are used as an additional variable to adjust the tweet-topic distribution. If two tweets are related, then the tweet-topic distributions of those two tweets are simultaneously adjusted. The evaluation results showed that intLDA outperformed the original LDA and other advanced baseline methods. However, we believe that intLDA can be further improved by taking account of the relationship weight

between tweets rather than just connected or unconnected in intLDA. In our previous work [21], we found that a non-negative matrix factorization (NMF) based approach [16] can achieve the same objective as LDA based methods, but is more flexible when incorporating the strength of relationship between tweets.

NMF is one of the most effective method to uncover the hidden thematic structure or latent features of a relationship-based matrix by factorizing the matrix into its lower dimensional representation [16]. It is very popular in the domain of unsupervised clustering [8, 12, 13, 28]. NMF is highly flexible and can be implemented in a distributed [18] or online system [34]. For topic derivation on Twitter, NMF methods are able to give both the tweet clusters of potential topics (topic-tweet) and the topic-term for each topic [23, 36]. To deal with the sparsity issue in the Twitter environment, we proposed the Non-Negative Matrix inter-joint Factorization (*NMijF*) [23] to get a high quality topic derivation by incorporating the relationships between tweets. This method did not take temporal aspect into account.

In this paper, we incorporate a time aspect into the NMijF process described in [23]. We denote the resulting new method as *tNMijF*. Like the method on which it is based, tNMijF is an inter-joint factorization of a non-negative symmetric matrix $A \in \mathbb{R}^{m \times m}$ and another non-negative matrix $V \in \mathbb{R}^{m \times n}$ within a unified process. In our implementation, matrix A is the new tweet-to-tweet relationship matrix discussed in the previous section (which includes a temporal aspect), and V is the tweet-to-term matrix which contains the relationship between tweets and the unique terms appearing in all tweets in the dataset. Each element in V is calculated using the *tf-idf* function described in [19]. We briefly describe the process here. More details can be found in [23].

As shown in Figure 7, the tweet-to-tweet matrix A is factorized to the tweet-topic matrix W as a base and W^T as the coefficient matrix. *Within the same process*, the tweet-to-term matrix V is factorized to the shared topic-tweet matrix W and topic-term matrix Y as the coefficient. In this method, matrices A and V share the tweet-topic matrix W. Hence, by implementing tNMijF, we can directly retrieve the main topic of a tweet from the topic-tweet matrix W *and* the top-n representative terms for each topic from the topic-term matrix Y within a unified process.

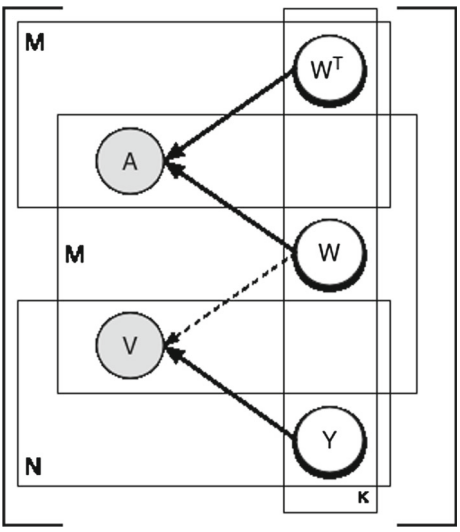


Figure 7 Graphical Model of *tNMijF*

The tweet-to-tweet matrix A is much more dense than the tweet-to-term matrix V. In the best case (all terms are connected), the density of A will be equal to V. The sparsity of V could heavily penalize the quality of topic derivation. So, to handle this problem, the effect of matrix V in the factorization process to derive matrix W needs to be reduced. We implement the scale parameter α to control the effect in every iteration to achieve the objective function.

The inter-joint factorization process in tNMijF aims at finding the minimum divergence ($\mathscr{D}$) of $A \approx WW^T$ and $V \approx WY$. The graphical model for tNMijF is shown on Figure 7, with the following objective function ($\mathscr{T}_{tNMijF}$):

$$\begin{aligned}\mathscr{T}_{tNMijF} &= \mathscr{D}(A\|WW^T) + \alpha\mathscr{D}(V\|WY) \\ &= \sum_{im} d(a_{im}|(ww^T)_{im}) + \alpha \sum_{mn} d(v_{mn}|(wy)_{mn})\end{aligned} \tag{7}$$

where there exists at least one element w and y in each of the matrices W and Y such that $w \geq 0$ and $y \geq 0$, and the scaling parameter α satisfies $0 \leq \alpha \leq 1$.

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

$$\begin{aligned}d(a_{im}|(ww^T)_{im}) &= a_{im} \log \frac{a_{im}}{(ww^T)_{im}} - a_{im} + (ww^T)_{im}, \textit{and} \\ d(v_{mn}|(wy)_{mn}) &= v_{mn} \log \frac{v_{mn}}{(wy)_{mn}} - v_{mn} + (wy)_{mn}\end{aligned} \tag{8}$$

To derive the multiplicative update rules for every element in each iteration, we follow the parameter estimation procedure from [30] by introducing auxiliary variables $r_{i,m,k}$ and $s_{m,n,k}$ ($\sum_k r_{i,m,k} = 1$, $\sum_k s_{m,n,k} = 1$), and use the Jensen's inequality [14] to derive the upper bound $\mathscr{F}$ of $\mathscr{T}_{tNMijF}$

$$\begin{aligned}\mathscr{T}_{tNMijF} &= \mathscr{D}(A\|WW^T) + \alpha\mathscr{D}(V\|WY) \qquad (9)\\ &\leq \sum_{im}((ww^T)_{im} - a_{im}\sum_k r_{i,m,k} \log \frac{w_{i,k}w_{k,m}^T}{r_{i,m,k}}) \\ &\quad +\alpha\sum_{mn}((wy)_{mn} - v_{mm}\sum_k s_{m,n,k} \log \frac{w_{m,k}y_{k,n}}{s_{m,n,k}}) \\ &\cong \mathscr{F} \qquad (10)\end{aligned}$$

Equality is achieved if and only if:

$$r_{i,m,k} = \frac{w_{i,k}w_{k,m}^T}{\sum_k w_{i,k}w_{k,m}^T}, s_{m,n,k} = \frac{w_{m,k}y_{k,n}}{\sum_k w_{m,k}y_{k,n}} \tag{11}$$

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

$$\frac{\partial\mathscr{F}}{\partial w_{ik}} = \sum_{m=1}^{M}(w_{k,m}^T - a_{i,m}\frac{r_{i,m,k}}{w_{i,k}}) + \alpha\sum_{n=1}^{N}(y_{k,n} - v_{i,n}\frac{s_{m,n,k}}{w_{i,k}}) \tag{12}$$

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

$$w_{i,k} = \frac{\sum_{m=1}^{M} a_{i,m}r_{i,m,k} + \alpha\sum_{n=1}^{N} v_{i,n}s_{m,n,k}}{\sum_{m=1}^{M} w_{k,m}^T + \alpha\sum_{n=1}^{N} y_{k,n}} \tag{13}$$

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

$$\hat{w}_{i,k} = w_{i,k} \frac{(\sum_{m=1}^{M} \frac{a_{i,m}}{(ww^T)_{i,m}} w_{k,m}^T + \alpha \sum_{n=1}^{N} \frac{v_{i,n}}{(wy)_{i,n}} y_{k,n})}{\sum_{m=1}^{M} w_{k,m}^T + \alpha \sum_{n=1}^{N} y_{k,n}} \tag{14}$$

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

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

$$\hat{y}_{k,n} = y_{k,n} \frac{(\sum_{m=1}^{M} \frac{w_{k,m}}{(wy)_{k,m}} w_{k,m})}{\sum_{m=1}^{M} w_{k,m}} \tag{15}$$

6 Experiments

We now analyse our experiments with the new time-sensitive model. We first present our experimental datasets, followed by the baseline methods and the evaluation metrics we employed. Then, we provide the results with a discussion.

6.1 Datasets

In this work, we use three datasets, *TREC2014*, *tweetSanders*, and *tweetMarch*, to evaluate our proposed method. The first two are available online and widely used in the evaluation of social network analysis methods. tweetMarch is a corpus we collected. These datasets have different characteristics, especially related to the availability of interaction, number of topics involved, and the average density of various types of relationships.

The TREC2014 dataset is provided by *The Text REtrieval Conference* (TREC), a community co-sponsored by National Institute of Standards and Technology (NIST) and U.S. Department of Defense. This dataset is available online at http://trec.nist.gov/data/microblog2014.html. It consists of more than 50000 tweets, and each of these tweets belongs to one of the 55 available topics (MB171 to MB225). We use the *Twitter REST API*[8] to download the tweets based on their given IDs. 40951 tweets out of the 50000 tweets in the TREC2014 dataset could be downloaded. This could be due to different reasons: for example, the tweet might have been deleted, or its status might have been changed to "*protected*". This dataset has 3,463 replies, and 0 retweet. The tweets are from 35,670 users. The tweets are labeled with their topics

The tweetSanders dataset[9] includes more than 5,000 tweets. From 5,513 tweet IDs that are available from the list, we could download 4,572 tweets. These tweets are labeled with respect to one of 4 topics: *Apple*, *Microsoft*, *Google*, *Twitter*. This dataset has 297 *reply* tweets and 269 *retweets*. These tweets are from 3711 different users.

TREC2014 and tweetSanders have been annotated with respect to topics, and thus can be directly used as the gold standard in evaluating the topic-based tweet clustering. However, it seems that only important tweets related to the assigned topic were included in these datasets. We thus assembled a new labeled dataset. We collected data from 03 March 2014 until 07 March 2014 using the Twitter Streaming API and denote this collected dataset as

[8] https://dev.twitter.com/rest/public, accessed 9 February 2016.

[9] http://www.sananalytics.com/ accessed January 20, 2014.

Table 3 Description of the datasets

Dataset	# tweets	# users	# reply	# retweet	# Topics
TREC2014	40,951	35,670	3,463	0	55
tweetSanders	4,572	3,711	297	269	4
tweetMarch	729,334	509,713	12,221	101,272	6

tweetMarch. This dataset includes both well-structured tweets and tweets with misspelled words or with full of emoticons, url and other noise.

As shown in Table 3, tweetMarch has 729,334 tweets, and it involves 509,713 users around the world. It has 12,221 *reply* tweets and 101,272 *retweets*. For evaluation purposes, we invited two annotators to label each tweet from the first 10,000 tweets of the tweetMarch dataset (the tweets are kept in order of the time they were posted). Each tweet is labeled by both annotators with a topic from 6 available topics: *food, day activities, life expressions, people communications, politics,* and *travel and transport*. The labeling scheme for the tweetMarch dataset is different from our previous works in [22], where we only had one annotater per tweet. For the 10,000 annotated tweets, the annotators agreed in 83 % of tweets. The *kappa* value [7] of these labeled tweets is 0.77. Based on Landis and Koch interpretation [25], the *kappa* value of 0.77 is categorized as *substantial agreement*.

We only used tweets in English in the experiments. For all these three datasets, the same preprocessing method is employed to remove irrelevant terms or characters (emoticons, punctuations, and terms that less than 3 characters), and stop-words. Then, each term is stemmed using the NLTK python package[10] followed by the tokenization of all tweets and terms. As mentioned before, hashtags are kept unchanged.

The different characteristics between the datasets are shown by the percentage of non-zero element in various types of relationships, which are presented in Table 4. The *tweet-to-tweet* matrix (A) represents the relationship between tweets by our definition, which was discussed previously in Section 4. The *tweet-to-term* matrix (V) is calculated using *tf-idf* function [27]. For the *term-to-term* matrix (T), we use the *positive point mutual information (PPMI)* function described in [36]. From Table 4, we can see that our definition of the tweet-to-tweet relationship provides the most dense non-zero element in comparison with other types of relationships, even if the number of tweets that have interactions is low (e.g., TREC2014 with only around 8 % tweets that are replies and no retweets at all). The tweetSanders dataset is the least sparse on all relationships compared with the other datasets due to its apparent involved topics.

In order to prove our hypothesis that 'mention' is important to topic based tweet clustering, for each dataset, we provide statistics for the first 1000, 3000, and 5000 (ordered by the time they were posted). The black bar represents the number of pairs of tweets that are linked by 'mention'. The gray bar shows the number of pairs of tweets that are linked by 'mention' and actually are under the same topic according to the evaluation labels of the involved tweets. Figure 8 shows that tweets that are connected by 'mention' are highly possible to be under the same topic.

Figure 8 also reflects that the possibility of being under the same topic for tweets connected by the mention becomes smaller when the difference of posting times between tweets becomes bigger. Figure 8a for dataset TREC2014 and Figure 8b for dataset tweetMarch

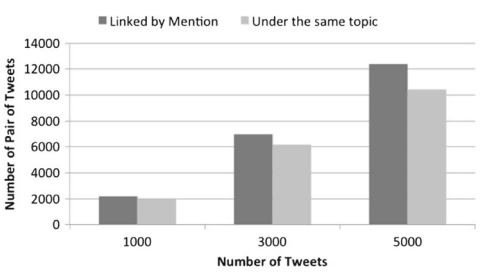
[10] http://www.nltk.org/.

Table 4 Density comparison between the tweet-to-tweet relationship matrix (A), tweet-to-term matrix (V), and term-to-term matrix (T)

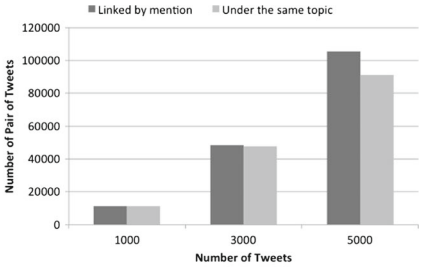
Dataset	A	V	T
TREC2014	2.695 %	0.056 %	0.249 %
tweetSanders	23.887 %	0.089 %	0.301 %
tweetMarch	12.842 %	0.075 %	0.271 %

show the same trend that the possibility for tweets to be under the same topic becomes smaller when the number of tweets becomes bigger. Note that the subsets of tweets used in these observations are ordered by the timestamp. The bigger number of tweets means that more tweets with bigger difference of posting times between tweets are included. For both datasets, there are thousands of users involved and most of them are ordinary users as opposed to special users (e.g., celebrities). Temporal aspect can be utilized not only for special users but also for ordinary users. The time function provided in Section 4 can thus help to improve the quality of topic derivation when incorporating interactions between tweets.

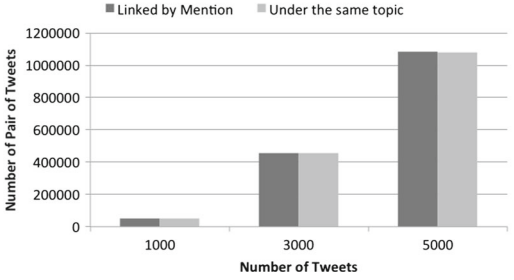
Figure 8c for dataset tweetSanders shows that the possibility of being under the same topic for tweets connected by the mention is very high (almost the same as that of the labeled tweets for evaluation). The topics in this dataset are very obvious. For example, when people talk topics about *apple, google, microsoft* or *twitter*, they almost always mention the usernames *@apple, @google, @microsoft* or *@twitter* in their tweets.



(a) TREC2014



(b) tweetMarch



(c) tweetSanders

Figure 8 Number of pairs of tweets linked by mention vs number of pairs of tweets linked by mention and labeled under same topics

6.2 Evaluation metrics

For evaluation purposes, we used several baseline methods:

- *NMijF* [23]. This is our previous model. It takes into account tweet' interactions and employs a non negative inter-joint factorization, but it is not time-sensitive. We use this method as a baseline to see the impact of the temporal aspect. While we have already shown that *NMijF* improves on the next three baselines, *TNMF*, *LDA* and *NMF*, we still include them for completeness sake.
- *TNMF* [36]. This topic derivation method incorporates a term-to-term correlation matrix to improve the quality of the results using matrix factorization techniques.
- *LDA* [2]. The most popular method in topic derivation. It has a "bag of words" assumption and works solely on the content of the document.
- *NMF* [16]. This is the basic method of matrix factorization. It directly factorizes the tweet-to-term matrix into topic-tweet and topic-term matrix.

We conducted the evaluations on the quality of topic derivation produced by all the methods by measuring the accuracy level of a cluster in comparison with our evaluation set. We compared the clustering result W from N tweets with an evaluation set of classes C. In particular, three complementary metrics were used in the evaluation on cluster quality [19]: Pairwise *F Measure*, *Purity* and *Normalized Mutual Information (NMI)*.

Purity evaluates the extent to which tweets are assigned the correct clusters (topics) based on our labeled datasets. This metric is defined in (16) below, in which totally incorrect clustering has a purity value of 0, and a perfect clustering has a purity value of 1. Perfect clustering means that, for all clusters, the tweets that are in the cluster are also grouped into one topic in the labeled dataset. To compute purity, we sum the maximum value from the mapping of each cluster in W that has the highest number of matched elements with the class from evaluation set C, and divide by the total number of involved tweets.

$$purity(W, C) = \frac{1}{N} \sum_i \max_j |w_i \cap c_j| \,. \tag{16}$$

Note that purity will not be penalized if the method produces more clusters than the labeled dataset. We illustrate this metric with our motivating example of Table 1. Assume if the evaluation set C is $C = \{\{t_1, t_2, t_3\}, \{t_4\}, \{t_5, t_6\}\}$, purity will be 1 if no tweet is assigned to a cluster with non-matching elements regardless of the number of clusters. So, $W = \{\{t_1, t_2\}, \{t_3\}, \{t_4\}, \{t_5, t_6\}\}$ will have purity value 1, eventhough W has more clusters than C. The purity value will go down if there is at least a pair of non-matching elements in the same cluster, for example if $W = \{\{t_1, t_2, t_4\}, \{t_3\}, \{t_5, t_6\}\}$.

Purity is a transparent measure of cluster quality. However, it is not designed to consider the trade-off between the quality of the clusters against the number of resulting clusters. To see the performance of the methods on different numbers of clusters, we employ NMI. NMI measures the mutual information shared between clusters and classes $I(W; C)$, normalized by the entropy of clusters $H(W)$ and classes $H(C)$. Similar to purity, the value of NMI will be ranged between 0 and 1.

$$NMI(W, C) = \frac{I(W; C)}{[H(W) + H(C)]/2} \,. \tag{17}$$

In this metric, mutual information $I(W, C)$ is a measure to quantify the statistical information shared by the pair of clusters W and C [5], which is defined in (18) below.

$$I(W, C) = \sum_k \sum_j P(w_k \cap c_j) \log \frac{P(w_k \cap c_j)}{P(w_k) P(c_j)} \tag{18}$$

where k and j are the number of clusters in W and C respectively. $P(w_k)$ is the probability of a tweet being in cluster w_k, $P(c_j)$ is the probability of a tweet being in cluster c_j, and $P(w_k \cap c_j)$ is the probability of a tweet being in both cluster w_k and gold standard c_j. The calculation of the entropy of clusters $H(W)$ and classes $H(C)$ are shown in equation below.

$$\begin{aligned} H(W) &= -\sum_k P(w_k) \log P(w_k), \\ H(C) &= -\sum_j P(c_j) \log P(c_j) \end{aligned} \tag{19}$$

From our previous example, we know that if $W = \{\{t_1, t_2\}, \{t_3\}, \{t_4\}, \{t_5, t_6\}\}$, the purity value will be 1 although the number of clusters in W is higher than in the evaluation set C. With NMI, $NMI(W, C)$ is 0.86, due to the different numbers of clusters. However, with NMI, the metric value does not necessarily mean that it will be decreased if the number of clusters is higher. The quality of the clusters will affect the result as well. For example, if $W = \{\{t_1, t_2, t_4\}, \{t_3\}, \{t_5, t_6\}\}$, the NMI value is 0.685.

We used the Pairwise F-Measure to measure the accuracy of the clustering result by analyzing the harmonic mean of both precision and recall. In this metrics, precision p is defined as the fraction of pairs of tweets correctly put in the same cluster, and recall r is the fraction of actual pairs of tweets that were identified. The formula of pairwise F-Measure is shown in (20), and the definition of both precision and recall are shown in (21)

$$F = 2 \times \frac{p \times r}{p + r}. \tag{20}$$

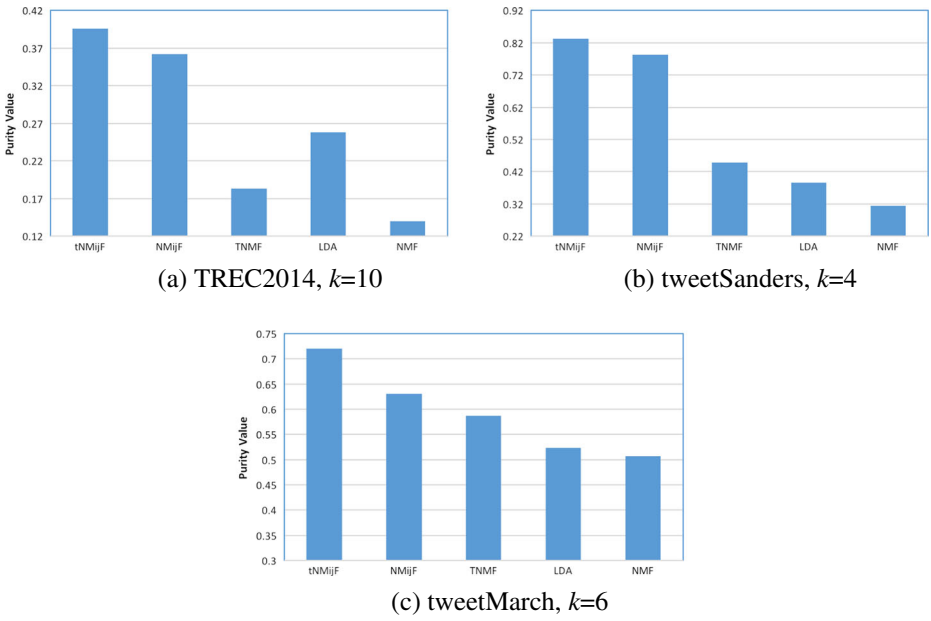
$$p = \frac{TP}{TP + FP}, \quad r = \frac{TP}{TP + FN} \tag{21}$$

where TP (*True Positive*) is the number of pairs of tweets from a cluster in the evaluation set which are assigned to the same cluster in the output. TN (*True Negative*) is the number of pairs of tweets from different clusters in the evaluation set that are assigned to different clusters. The *False Positive* (FP) is the number of pairs of tweets that should not be in the same cluster, but are assigned to the same cluster. *False Negative* (FN) is the number of pairs of tweets that should be in the same cluster, but are assigned to different clusters.

6.3 Results and discussion

We ran the proposed method and baseline methods for 20 times over all of the datasets and tuned all the parameters for the best performance. The average density (non-zero element) of the tweet-to-term matrix V for all three datasets is only 0.07 %, which is far below our tweet-to-tweet relationship matrix with 13.14 % density. The scaling parameter $\alpha = 0.1$ was found to be the best for all of the matrix inter-joint factorization processes, as the matrix V is very sparse. This α value ensures that the sparsity of V does not heavily penalize the topic-tweet matrix W and still gives good results when factorizing the topic-term matrix Y.

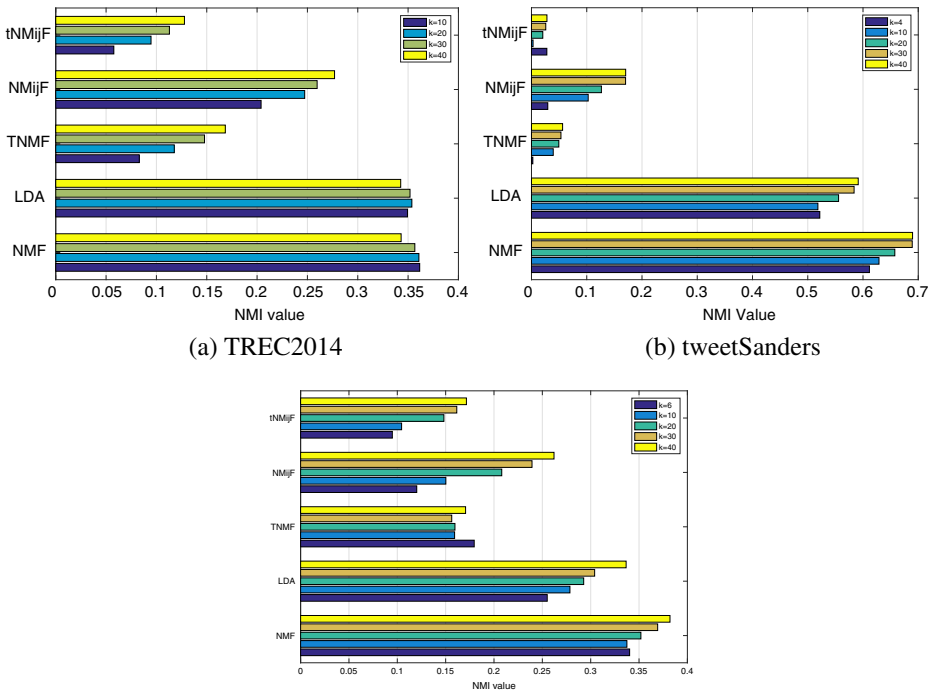
Figure 9 shows the evaluation results using the purity metric for all three datasets. For the TREC2014 dataset, we test all tweets that belong to the first ten topics (MB171 to MB180).



(a) TREC2014, k=10 (b) tweetSanders, k=4

(c) tweetMarch, k=6

Figure 9 Purity evaluation results of three datasets



(a) TREC2014 (b) tweetSanders

(c) tweetMarch

Figure 10 NMI evaluation results of three datasets

With this dataset, our proposed method tNMijF gives about 3 % improvement over our previous work, and 10-25 % over the other baseline methods. It is worth noting that, in the TREC2014 dataset, the number of tweets that have interactions is very small. The 'mention' made only around 0.02 % of the relationship between tweets, and there are no retweets at all in this dataset. The improvement we obtained suggests that our proposed method is able to give better performance even when the number of the involved interaction features is low. When the percentage of tweets with the mention is high, the improvement over our previous work is also higher. In the tweetMarch dataset, the interaction based on people is about 0.24 % of all linked tweets in the tweet-to-tweet matrix. The purity improvement is around 10 % over our original NMijF, and 15-30 % over the other baseline methods. Both tNMijF and NMijF are able to obtain a very high purity value in the tweetSanders dataset in comparison with other methods, with the purity value 0.83 and 0.78 respectively. The high density of tweet-to-tweet matrix in tweetSanders, as shown in Table 4, strongly supports the improvement of quality of the resulting clusters.

For the NMI evaluation, tNMijF results in roughly a 5-10 % improvement compared to NMijF, and 90-200 % improvement over the other methods, TNMF, LDA and NMF. Figure 10 shows the results of this NMI evaluation for all datasets. We use several different numbers of topics to test the performance of all methods using the NMI metric. As shown by all subfigures in Figure 10, tNMijF provides more stable results in any number of topics, with a positive trend in comparison with other baseline methods.

Table 5 shows the results of the pairwise F-Measure metrics. It can be seen that the inclusion of the temporal aspect function improves both precision and recall in comparison to the baseline methods in all datasets. tNMijF consistently provides the best results for both precision and recall. At any configuration, the proposed method outperforms NMijF as our previous work, which does not take the temporal feature into account.

In our previous work [22], we did not compute the annotator agreement to evaluate the quality of the labeling results. In this paper, we provide an annotator agreement analysis over the first 10,000 tweets (ordered by posting time). For about 83 % tweets, different annotators gave the same label. Figure 11 shows the comparison of the evaluation results on the tweetMarch dataset with new labeling scheme (with annotator agreement computation)

Table 5 *Precision (p), Recall (r) and F-Measure (F-M)* for three datasets

Method	p	r	F-M
tNMijF	**0.174**	**0.665**	**0.276**
NMijF	0.168	0.643	0.267
TNMF	0.047	0.176	0.074
LDA	0.091	0.343	0.144
NMF	0.040	0.147	0.063

(a) TREC2014, $k = 10$

Method	p	r	F-M
tNMijF	**0.767**	**0.876**	**0.818**
NMijF	0.738	0.778	0.757
TNMF	0.288	0.336	0.310
LDA	0.272	0.300	0.285
NMF	0.262	0.299	0.280

(b) tweetSanders, $k = 4$

Method	p	r	F-M
tNMijF	**0.615**	**0.351**	**0.447**
NMijF	0.564	0.313	0.402
TNMF	0.341	0.196	0.249
LDA	0.373	0.215	0.273
NMF	0.314	0.149	0.202

(c) tweetMarch, $k = 6$

(a) TREC2014, $k = 10$
(b) tweetSanders, $k = 4$
(c) tweetMarch, $k = 6$

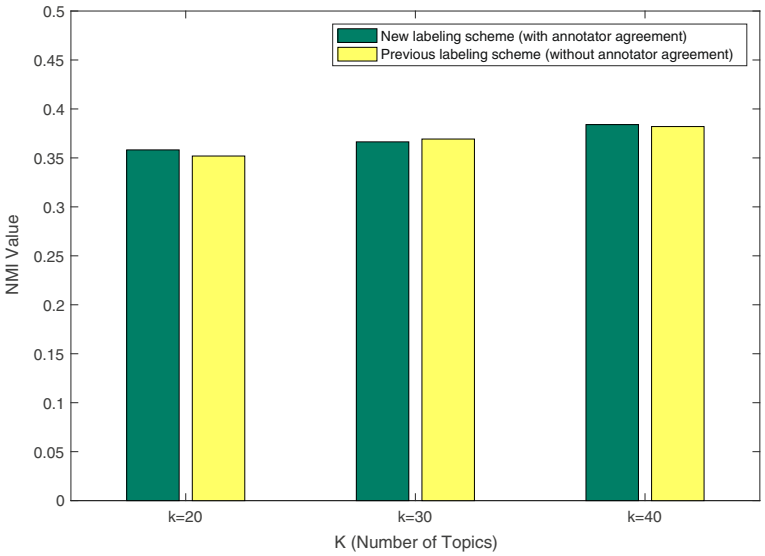


Figure 11 NMI evaluation over the tweetMarch dataset with tNMijF method

and previous labeling scheme. For all different k (number of topics) values, the NMI values are quite close.

Examples of word-representations for several topics from TREC2014, tweetSanders and tweetMarch are listed in Tables 6, 7 and 8 respectively. Our proposed method presents better keywords for each topic, as it is able to provide more connected words that make the topic more readable [29]. NMF gives the worst performance for all datasets, since it gives many unrelated words to represent almost all topics. In Table 6 (TREC2014 dataset), all methods

Table 6 Top 5 terms for topics derived from the *TREC2014* dataset. (Words in italic have a high degree of connectivity with a specific topic, words in stroked have a low degree of connectivity with the topic)

Cluster/ Topic Number	Topics	*tNMijF*	*NMijF*	*TNMF*	*LDA*	*NMF*
MB171	Ron Weasley birthday	*ron* *weasley* *birthday* *happy* potter	*ron* *weasley* *harry* family *birthday*	potter *ron* harry *weasley* *birthday*	*ron* *weasley* *birthday* *happy* harry	*happy* *birthday* ~~member~~ *ron* *weasley*
MB172	Merging of US Air and American	*american* *airways* *airline* *air* *merger*	*american* *merger* *airways* *air* world	*airways* *american* high *airline* *deal*	*american* *airways* *airline* *merger* ~~watch~~	*american* *airways* *airline* *merger* ~~school~~
MB173	Muscle pain from statins	*pain* *muscle* ~~work~~ effect head	*pain* oldest ~~today~~ fat *statins*	*therapy* *pain* ~~regain~~ fat head	oldest *pain* effect ~~race~~ ~~dog~~	oldest eat ~~great~~ *pain* ~~star~~

seem to be able to list the keywords accurately in most topics. However, the objective is not only to list the keywords for each topic, but also to achieve high accuracy in the topic-based clustering. The results related to the cluster accuracy show that our proposed method performs better in this aspect.

6.4 Tweet distributions and purity evaluations over time periods

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

TREC2014 has been used for the purpose of temporal based information retrieval [17]. Here, we consider the tweets that belong to the first ten topics (MB171 to MB180) in TREC2014. The total number of tweets is 7126. These tweets are sorted by the posting time in an ascending order. We put the first 7000 sorted tweets into 7 temporal groups (T1 to T7). Each group has 1000 tweets in a period of time. These time periods have quite similar length (about one week).

We applied our proposed method and baseline methods for tweets in each group to derive topics and carried out the purity evaluation. The tweet distributions over the time periods for labeled topics are shown in Figure 12a. The purity evaluation results for the different time periods are shown in Figure 12b. Tweet distributions over the time periods for the topic MB180 are shown in Figure 12c.

To evaluate the performance of a method, it is necessary to scrutinize the numbers of tweets that belong to a specific topic over different time periods. For all methods, the purity values in $T2$ are quite high due to the fact that more than 500 tweets belong to the topic MB175 (see Figure 12a). For a specific time period, when there is no topic with a dominant number of tweets, the purity values are quite low for all baseline methods. Our proposed method performs very well in such a situation comparing with these baseline methods. In Figure 12c, the line with diamond symbols shows the numbers of labeled tweets that belong

Table 7 Top 5 terms for topics derived from the *tweetSanders* dataset

Topics	*tNMijF*	*NMijF*	*TNMF*	*LDA*	*NMF*
Apple	*iphone*	*iphone*	*iphone*	new	*iphone*
	apps	*apple*	~~#google~~	*iphone*	~~#twitter~~
	store	*store*	time	phone	*siri*
	siri	*#apple*	new	*#apple*	thank
	apple	love	*store*	*apps*	new
Google	*#google*	*#google*	*#google*	*#google*	*#google*
	#android	*#android*	*now*	~~#microsoft~~	*#android*
	sandwich	*sandwich*	cloud	~~twitter~~	*sandwich*
	cream	*nexus*	*#android*	*#android*	*cream*
	nexus	*android*	*nexus*	apps	*nexus*

Table 8 Top 5 terms for topics derived from the *tweetMarch* dataset

Topics	*tNMijF*	*NMijF*	*TNMF*	*LDA*	*NMF*
Travel/ transport	*traffic*	*lane*	*#traffic*	~~follow~~	*train*
	lane	*traffic*	*road*	*train*	*road*
	closed	*#traffic*	time	*#traffic*	*driver*
	stop	*accident*	*driver*	*driver*	time
	accident	closed	closed	~~gamer~~	closed
Politics	*politic*	*liberal*	*liberal*	textitabbot	*abbot*
	abbot	*obama*	*policy*	*politic*	*liberal*
	obama	people	*abbot*	~~time~~	*obama*
	govenrment	~~time~~	*government*	*obama*	~~big~~
	~~one~~	~~claim~~	~~time~~	*policy*	~~process~~
Food/Beverages	*tea*	*tea*	*sugar*	~~talk~~	order
	drink	*drink*	*tea*	*coffee*	*coffee*
	order	*coffee*	*coffee*	*drink*	*tea*
	coffee	~~hospital~~	~~talk~~	~~smoking~~	black
	~~table~~	order	~~reading~~	~~sleep~~	~~closed~~

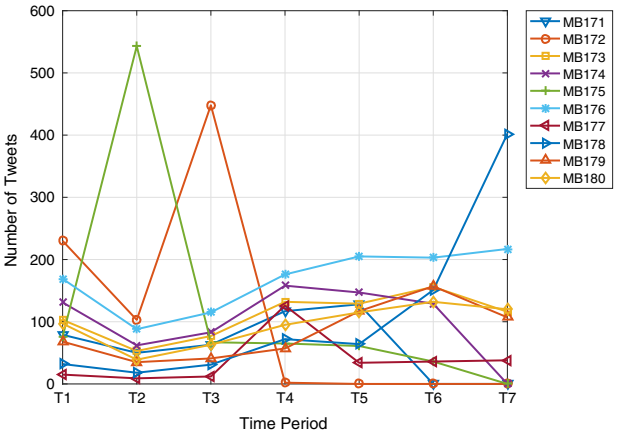
to the topic MB180; the line with square symbols shows the numbers of tweets that belong to the topic MB180 by using our proposed method. The other lines show the results by using the baseline methods. When the number of tweets that belong to a topic is low, the baseline methods and our previous method [23] could not get any reasonable result. The topic is totally missing. Our new proposed method tNMijF, which takes the time-sensitive interactions into account, provides a very accurate result. At $T2$, 39 tweets are labeled under the topic MB180; 42 tweets are put under this topic by using our method.

Consistent with the evaluations against the static tweet collections in the previous subsection, our proposed method tNMijF achieves the best performance over all time periods. These results show that the varying nature of topics in a timeline will not strongly affect the accuracy improvement brought in by our method. This analysis indicates that our proposed method can cope with dynamic tweet streams better than existing methods. Our proposed method can be implemented to derive topics by processing the tweet streams as a series of tweet groups and achieve good results with 0.74 accuracy on average.

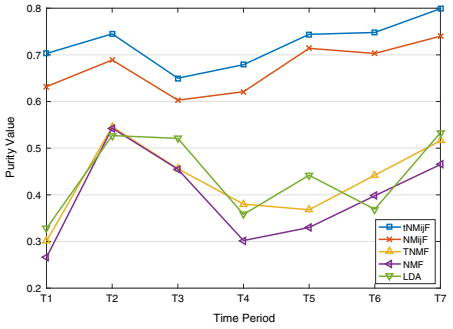
7 Related work

The short-in-content nature of Twitter presents a challenging problem for deriving the topics of a tweet collection. The limited length for each tweet renders the frequency of co-occurences between terms extremely low. This sparsity heavily penalizes the performance of the state-of-the art topic derivation methods such as LDA [2], PLSA [9] and NMF [16], as they generally work solely on content features.

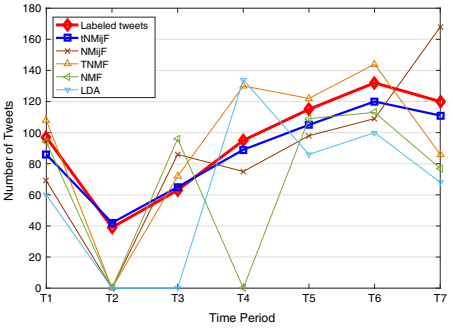
A lot of studies have been conducted to extend those popular methods to handle the sparsity issues. [24] proposed a variant of *labeled-LDA* to work on the Twitter environment with the hashtag and other content features (e.g., word distributions based on specific emoticons and social signal) as labels for a partially supervised topic learning process. [1] tackles the sparsity in Twitter real-time filtering by proposing a query expansion method to enrich the knowledge of the topic by deriving terms that are relevant from user's query and document



(a) Tweet distributions over time periods for labeled topics



(b) Purity evaluation results for different time periods



(c) Tweet distributions over time periods for the topic MB180 with different topic derivation methods

Figure 12 Tweet distributions and purity evaluations over time periods with dataset TREC2014

collections. The study of [32] addressed the problem of sparsity when modeling the multi-faceted topic in Twitter by augmenting the content with the help of hashtag-based semantic enrichment and auxiliary semantic from linked external sources. However, relying on external documents brings an extra burden when dealing with highly dynamic environments like Twitter. The users following-follower mechanism has also been investigated [4] for determining the popularity of authors to refine the topic learning process in Twitter. However, analyzing the relationships based on following-follower suffers from scalability issues in the Twitters streaming environment, since details of user information need to be queried apart from the dataset itself.

The approach reported in [36] and [35] exploits the term co-occurrence patterns to improve the topic learning process in short text environments. Unfortunately, in the Twitter environment, the relationship between terms is very sparse and it only provides a small improvement with respect to density in comparison with the original tweet-to-term relationships [23].

To deal with the dynamic nature of the Twitter environment, several methods have been proposed by including temporal features. However, most of the works were aimed at implementing topic derivation in an incremental/online fashion to learn the movement of topics overtime. [26] proposed a time based regularization in NMF method to learn the topics in social media. [15] presented an online variant of LDA to periodically model the topics from Twitter based on time slices. The study in [3] introduced the content aging theory to mine the emerging topics from Twitter stream. Stilo et al. [29] proposed *Symbolix Aggregate approximation* (SAX) to discretize the temporal series of terms to discover the events from Twitter content. All these studies still focus on contents, and overlook the social features available in the Twitter environment. As a result, they still suffer from the sparsity issue.

Different from the topic derivation work that only takes content into account, our previous work [23] incorporated the relationships between tweets to deal with the sparsity problem and showed improvements in performance. The work presented in this paper builds on this foundation, adding a time dimension to the interactions. To the best of our knowledge, our proposed method is the first one to incorporate temporal aspect, social interactions and content in a unified model to derive topics from a collection of tweets.

8 Conclusion

In this paper, we investigate the effect of time on user interactions for topic derivation in Twitter. We propose a new topic derivation method that includes a time factor. It can simultaneously achieve the clustering of tweets based on topics and the identification of the representative terms for each topic. We conducted a set of experiments on 3 different datasets.

Our results show that incorporating a temporal aspect on the interaction features can improve the results quality of the topic derivation. In particular, the proposed method results in a consistent improvement in the quality of topic derivation over both well-known baseline methods and our prior method, which was not time-sensitive.

Acknowledgments This work is partially supported by the Indonesian Directorate General of Higher Education (DGHE), Macquarie University, CSIRO Data61, Australian Research Council LP120200231, and Australian Research Council DP140101369.

References

1. Albakour, M., Macdonald, C., Ounis, I., et al.: On sparsity and drift for effective real-time filtering in microblogs. In: Proceedings of the 22nd ACM International Conference on Information andamp; Knowledge Management (CIKM 2013), pp. 419–428 (2013)
2. Blei, D., Ng, A., Jordan, M.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
3. Cataldi, M., Di Caro, L., Schifanella, C.: Emerging topic detection on twitter based on temporal and social terms evaluation. In: Proceedings of the Tenth International Workshop on Multimedia Data Mining, pp. 4. ACM, Washington DC USA (2010)
4. Cha, Y., Bi, B., Hsieh, C.C., Cho, J.: Incorporating popularity in topic models for social network analysis. In: Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval, pp. 223-232. ACM, Dublin, Ireland (2013)
5. Cover, T.M., Thomas, J.A.: Elements of Information Theory. John Wiley & Sons (2012)
6. de Moor, A.: Conversations in context: a twitter case for social media systems design. In: Proceedings of the 6th International Conference on Semantic Systems, p. 29. ACM, New York, NY, USA (2010)
7. Fleiss, J.L.: Measuring nominal scale agreement among many raters. Psychol. Bull. **76**(5), 378 (1971)

8. He, Z., Xie, S., Zdunek, R., Zhou, G., Cichocki, A.: Symmetric nonnegative matrix factorization: Algorithms and applications to probabilistic clustering. IEEE Trans. Neural Netw. **22**(12), 2117–2131 (2011)
9. Hofmann, T.: Probabilistic latent semantic indexing. In: Proceedings of the 22nd Annual International ACM SIGIR conference on Research and Development in Information Retrieval, pp. 50-57. ACM, Berkeley, CA, USA (1999)
10. Hu, Y., John, A., Wang, F., Kambhampati, S.: Et-lda: Joint topic modeling for aligning events and their twitter feedback. In: AAAI Conference on Artificial Intelligence (AAAI 2012), vol. 12, pp. 59–65. Toronto, Ontario, Canada (2012)
11. Kietzmann, J.H., Hermkens, K., McCarthy, I.P., Silvestre, B.S.: Social media? get serious! understanding the functional building blocks of social media. Bus. Horiz. **54**(3), 241–251 (2011)
12. Kim, J., Park, H.: Sparse nonnegative matrix factorization for clustering (2008)
13. Kuang, D., Park, H., Ding, C.: Symmetric nonnegative matrix factorization for graph clustering. In: SIAM International Conference on Data Mining (SDM), vol. 12, pp. 106–117. SIAM, Anaheim, California, USA (2012)
14. Kuczma, M.: An introduction to the theory of functional equations and inequalities: Cauchy's equation and Jensen's inequality Springer Science & Business Media (2009)
15. Lau, J.H., Collier, N., Baldwin, T.: On-line trend analysis with topic models: #twitter trends detection topic model online. In: Proceedings of COLING 2012, pp. 1519–1534. The COLING 2012 Organizing Committee, Mumbai, India (2012). http://www.aclweb.org/anthology/C12-1093
16. Lee, D., Seung, H.: Algorithms for non-negative matrix factorization. Denver, CO, USA (2000)
17. Lin, J., Efron, M., Wang, Y., Sherman, G.: Overview of the trec-2014 microblog track. Tech. rep., DTIC Document (2014)
18. Liu, C., Yang, H.C., Fan, J., He, L.W., Wang, Y.M.: Distributed nonnegative matrix factorization for web-scale dyadic data analysis on mapreduce. In: Proceedings of the 19th International Conference on World Wide Web, WWW '10, pp. 681–690. ACM, New York, NY, USA (2010).
19. Manning, C., Raghavan, P., Schütze, H.: Introduction to Information Retrieval, vol. 1 Cambridge (2008)
20. Nugroho, R., Molla-Aliod, D., Yang, J., Paris, C., Nepal, S.: Incorporating tweet relationships into topic derivation. In: Conference of the Pacific Association for Computational Linguistics (PACLING 2015), p. 2015. PACLING, Bali, Indonesia (2015)
21. Nugroho, R., Yang, J., Zhong, Y., Paris, C., Nepal, S.: Deriving topics in twitter by exploiting tweet interactions. In: Proceedings of the 4th IEEE International Congress on Big Data. IEEE Services Computing Community, New York, USA (2015)
22. Nugroho, R., Zhao, W., Yang, J., Paris, C., Nepal, S., Mei, Y.: Time-sensitive topic derivation in twitter. In: Web Information Systems Engineering – WISE 2015: 16th International Conference, Miami, FL, USA, November 1-3, 2015, Proceedings, Part I, pp. 138-152. Springer International Publishing, Cham (2015)
23. Nugroho, R., Zhong, Y., Yang, J., Paris, C., Nepal, S.: Matrix inter-joint factorization - a new approach for topic derivation in twitter. In: Proceedings of the 4th IEEE International Congress on Big Data. IEEE Services Computing, New York, USA (2015)
24. Ramage, D., Dumais, S.T., Liebling, D.J.: Characterizing microblogs with topic models. The International AAAI Conference on Web and Social Media (ICWSM) **10**, 130–137 (2010)
25. Richard, J., Landis, G.G.K.: The measurement of observer agreement for categorical data. Biometrics **33**(1), 159–174 (1977)
26. Saha, A., Sindhwani, V.: Learning evolving and emerging topics in social media: a dynamic nmf approach with temporal regularization. In: Proceedings of the fifth ACM international conference on Web search and data mining (WSDM 2012), pp. 693-702. ACM, Seattle, Washington (2012)
27. Salton, G.: Automatic Text Processing. Addison-Wesley, The Transformation, Analysis, and Retrieval of Information by Computer (1989)
28. Shahnaz, F., Berry, M.W., Pauca, V.P., Plemmons, R.J.: Document clustering using nonnegative matrix factorization. Inf. Process. Manag. **42**(2), 373–386 (2006)
29. Stilo, G., Velardi, P.: Time makes sense: Event discovery in twitter using temporal similarity. In: Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)-vol. 02, pp. 186–193. IEEE Computer Society, Warsaw, Poland (2014)
30. Takeuchi, K., Ishiguro, K., Kimura, A., Sawada, H.: Non-negative multiple matrix factorization. In: Proceedings of the Twenty-Third international joint conference on Artificial Intelligence, pp. 1713–1720. AAAI Press (2013)
31. Von Seggern, D.H.: CRC Standard Curves and Surfaces with Mathematica CRC Press (2006)
32. Vosecky, J., Jiang, D., Leung, K.W.T., Xing, K., Ng, W.: Integrating social and auxiliary semantics for multifaceted topic modeling in twitter. ACM Trans. Internet Technol. (TOIT) **14**(4), 27 (2014)

33. Wan, S., Paris, C.: Improving government services with social media feedback. In: Proceedings of the 19th International Conference on Intelligent User Interfaces, IUI '14, pp. 27–36. ACM, New York, NY, USA (2014).
34. Wang, F., Li, P., König, A.C.: Efficient document clustering via online nonnegative matrix factorizations. In: SIAM International Conference on Data Mining (SDM), vol. 11, pp. 908-919. SIAM, Arizona, USA (2011)
35. Yan, X., Guo, J., Lan, Y., Cheng, X.: A biterm topic model for short texts. In: Proceedings of the 22nd international conference on World Wide Web (WWW 2013), pp. 1445-1456. International World Wide Web Conferences Steering Committee, Rio de Janeiro, Brazil (2013)
36. Yan, X., Guo, J., Liu, S., Cheng, X., Wang, Y.: Learning topics in short texts by non-negative matrix factorization on term correlation matrix. In: Proceedings of the SIAM International Conference on Data Mining (SIAM 2013). SDM, San Diego, California, USA (2013)
37. Yang, L., Sun, T., Zhang, M., Mei, Q.: We know what@ you# tag: Does the dual role affect hashtag adoption? In: Proceedings of the 21st International Conference on World Wide Web (WWW 2012), pp. 261–270. ACM, Lyon, France (2012)