What and With Whom? Identifying Topics in Twitter Through Both Interactions and Text

Robertus Nugroho, Jian Yang, Weiliang Zhao, Cecile Paris, and Surya Nepal

Abstract—The overwhelming amount of information continuously flowing through the Twitter environment makes topic derivation essential. It indeed plays a valuable role in a variety of Twitter-based applications, including content recommendations, news summarization, market analysis, etc. Topic derivation methods are typically based on semantic features of tweet contents. Because tweets are short by nature, such methods suffer from data sparsity. To alleviate this problem, this paper proposes a topic derivation method that incorporates tweet text similarity and interactions measures. Besides the tweet contents, the approach takes into account several types of interactions amongst tweets: Tweets which mention the same people, replies and retweets. Topic derivation is done through a two-step matrix factorization process. We conducted a number of experiments on several Twitter datasets to reveal both the individual and integrated effects of the various features being considered. Our experimental results against TREC2014 and our self collected tweetMarch datasets demonstrate that the proposed method is able to provide more than 30 percent improvement compared to other advanced topic derivation methods.

Index Terms—Twitter, topic derivation, joint-NMF, tweets interactions

With around 6 thousands messages (tweets) per second,[1] Twitter has become a phenomenal platform for information dissemination, covering a wide range of topics. However, with this very large, unstructured and redundant big data, the information stream can easily overload users [1]. Thus, having effective methods to derive topics from Twitter is critical for a wide range of services, such as determining the hot issues, forecasting events, marketing, and recommending specific items. It is also important to enable the study of issues related to complex social networks.

Deriving topics from Twitter is a process of clustering tweets based on topic similarity by determining the main topic of every tweet, and, at the same time, retrieving a list of keywords to represent every topic [2]. Topic derivation on a document collection is typically done by identifying the latent thematic structures of the collection and choosing a set of representative words for every structure. Popular topic derivation methods include *Probabilistic Latent Semantic Analysis (PLSA)* [3], *Non-negative Matrix Factorization (NMF)* [4], and *Latent Dirichlet Allocation (LDA)* [5]. In these methods, each term in the documents is observed to find its semantic relationships and similarities with terms in other documents. As these methods exploit only the text of a document, they tend to have their best performance when there is a high frequency of co-occurring terms, such as in a traditional document collection.

In Twitter, however, the frequency of co-occurring terms amongst tweets is normally very low, as a tweet is limited to only 140 characters.[2] This leads to an extremely sparse relationship matrix between the collection of tweets and the unique terms available in these tweets. As a result, the quality of topic derivation decreases [6].

We illustrate this through the example shown in Table 1. There, we have 6 tweets connected through various interactions.[3] Fig. 1a shows the relationships between the tweets and all the terms available in the collection. We can see that t_1 is related to t_3 since all the terms in t_1 are available in t_3. Similarly, t_1, t_3 and t_6 are related to t_4 due to the fact that they have "#Canberra" as a common hashtag. Also, t_4 and t_2 have a common term: "Australia". In contrast, t_5 does not have any relationship with other tweets as there are no common terms amongst them. As illustrated in Fig. 1a, we can see that there are not many terms overlapping across those tweets.

Fig. 1b shows the relationships amongst the tweets in this example, formed by the Twitter interaction features such as *mention*, *reply* and *retweet*. We see that t_1 and t_2 are part of a conversation about politics. User a mentions user b in her tweet t_1, and user b then replies in tweet t_2. The relationships are indicated by the mention and reply features. We observe, however, that the tweets do not share any terms. t_1 is retweeted by user c in t_3. The *retweet* shows an obvious

[1] http://www.internetlivestats.com/twitter-statistics/, accessed February 23, 2017.

[2] https://dev.twitter.com/overview/api/counting-characters, accessed February 23, 2017.

[3] We made up this simple example for illustration purposes.

- R. Nugroho is with the Department of Computing, Macquarie University, Sydney, NSW 2109, Australia, and the CSIRO Data61, Eveleigh, NSW 2015, Australia. E-mail: robertus.nugroho@students.mq.edu.au.
- J. Yang and W. Zhao are with the Department of Computing, Macquarie University, Sydney, NSW 2109, Australia. E-mail: {jian.yang, weiliang.zhao}@mq.edu.au.
- C. Paris and S. Nepal are with the CSIRO Data61, Eveleigh, NSW 2015, Australia. E-mail: {cecile.paris, surya.nepal}@data61.csiro.au.

Manuscript received 22 Aug. 2016; revised 23 Feb. 2017; accepted 6 Apr. 2017. Date of publication 24 Apr. 2017; date of current version 12 June 2020.
Digital Object Identifier no. 10.1109/TSC.2017.2696531

TABLE 1
Motivating Example

Id	User	Tweets
t_1	a	New senate, exciting times in #Canberra @b
t_2	b	@a true, and what a start with the census in Australia!
t_3	c	RT @a New senate, exciting times in #Canberra @b
t_4	d	#Floriade in #Canberra, biggest celebration of spring in Australia
t_5	e	@d any special event in particular worth coming for?
t_6	d	@e NightFest always has fantastic performers and great tasting pates from #Canberra and surrounding areas

relationship between t_1 and t_3, as both tweets contain mostly similar terms. By looking at the interactions between the tweets, we also see that t_4, t_5, and t_6 share a similar topic about the Floriade event. t_5 by user e is a reply to t_4, and user d then replies back to t_5 in her tweet t_6.

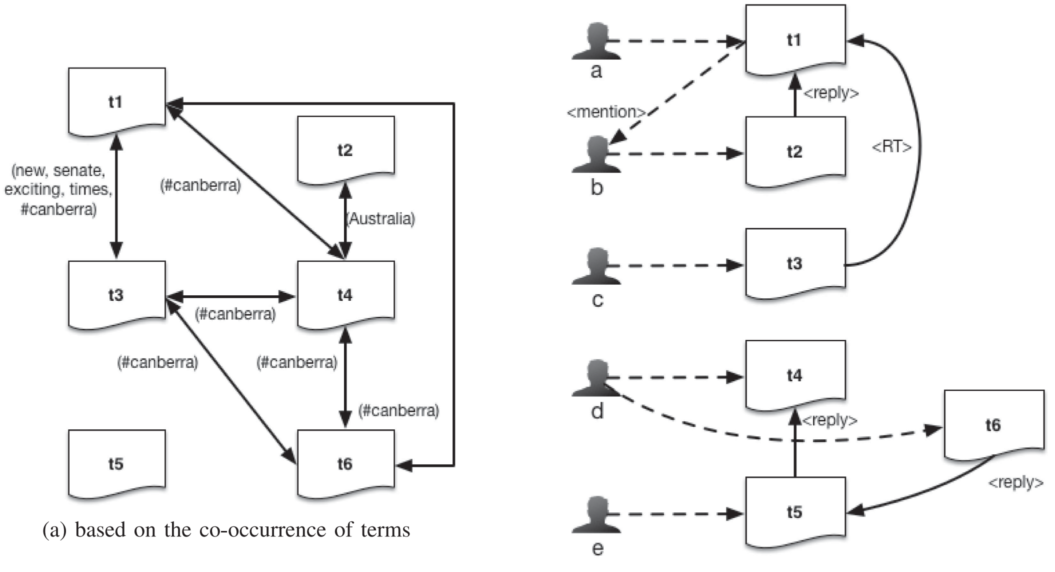
Interaction features (*mention*, *reply* or *retweet* (*RT*)) amongst tweets are strong indications that those tweets are part of a discussion or a conversation about a particular topic. Thus, using these features should enable us to achieve a significant improvement on topic derivation quality. In this simple example, we see two main topics: One concerning politics, and one about the Floriade celebration that is being held in Canberra. However, if only the contents of the tweets are exploited, the topics that will be derived will most likely be: #Canberra and special event, since t_1, t_2, t_3, t_4, t_6 are in the same group due to shared terms, and t_5 is isolated.

Researchers have proposed various topic derivation methods in the Twitter environment [7], [8], [9], [10], [11]. Most of the proposed methods are still focused on the exploitation of the tweets' content, which is extremely sparse. Recent works by [12] and [13] included Twitter's social features, but the involved social features were still limited to content based interactions such as *urls* and *hashtag*.

To deal with the extreme sparsity of a term relationship matrix in the Twitter environment, we propose a novel approach. It incorporates both the interaction features and content similarity to derive topics from a collection of tweets. Using two consecutive non-negative matrix factorization (NMF) processes to cluster the tweets based on topics and the cluster results to derive the keywords representations for each topic, the proposed approach outperforms other advanced baseline methods. The inclusion of the interactions along with the content similarity in our method provides more ability to alleviate the sparsity problem compared with other existing methods by adding more information about topical connectivity between tweets.

This paper expands on [14] in the following ways: (1) it provides a detailed explanation of how the relationships amongst tweets are obtained and measured; (2) it describes the rationale for the extension of the matrix inter-joint factorization algorithm and presents its implementation; (3) we evaluated our method on a publicly available additional dataset; (4) we present a careful characterization of our two datasets in terms of the interaction features they contain, to better understand the impact of these features on the method; and, finally, (5) we describe a set of experiments designed to identify the impact of each feature. Our contributions can be summarized as follows:

- We model the tweet-to-tweet relationship as a combination of tweet-content similarity and social interactions between users through mentions, replies and retweets. Both content similarities and interaction relationships are taken into account in the topic derivation.
- We develop a topic derivation method for a collection of tweets based on the non-negative matrix factorization method *intJNMF*. The method takes account of the tweet-relationship matrix and directly



(a) based on the co-occurrence of terms

(b) based on tweets interactions

Fig. 1. Relationship between tweets illustrations for topic derivation.

uses its tweet-topic latent factors to infer the keywords representation for every topic.

- We carry out comprehensive experiments on two Twitter datasets to evaluate our proposed method using various metrics. The experimental results show that incorporating the relationships amongst tweets into the process can alleviate the sparsity problem and thus improve the quality of the derived topics. We also observe the impact of each feature and find that their combination achieves the best performance.

We organize the rest of the paper as follows. Section 2 describes our approach, including how we define relationships amongst tweets, how the tweets are clustered using these interactions and how topic words are derived. Section 3 presents our experiments to evaluate the method. Section 4 reviews the related work, and Section 5 provides the conclusions and potential future work.

1 Deriving Topics by Incorporating Interactions

In this section, we present a detailed discussion about our new topic derivation approach, which will be referred to as *intJNMF*. Different from other methods that focus only on exploiting content, our approach also takes the conversational based Twitter social interactions into account. In the first section, we discuss our model of relationships between tweets that will form the tweet-relationship matrix. The topic derivation process is then discussed in the subsequent sections. It includes two main steps: (1) obtaining the latent tweet-topic matrix as a cluster of tweets by performing matrix factorization over the tweet-relationship matrix; (2) inferring the keywords representation for every topic by utilizing the obtained tweet-topic matrix in the factorization process of the sparse tweet-term matrix.

1.1 Measuring Relationships between Tweets

A tweet is *self-contained* if it does not contain any reference to other tweets except through the same hashtag [15]. For example, in Table 1, t_4 is self-contained. Alternatively, there can be several types of social interactions among tweets. For example, a tweet may include a *mention* (e.g., t_1 in Table 1), a *reply* (e.g., t_2, t_5, t_6), or a *retweet* (e.g., t_3). A *mention* is an interaction to include other Twitter users in a discussion about particular topic (e.g., $@b$ in t_1). It can also be used to initiate a conversation with other users. While a *reply* can be considered as a part of a conversation, a *retweet* (*RT*) is an action to share a tweet with one's friends (followers). Finally, a tweet can contain a *hashtag*. A hashtag is a specific word starting with the hash (#) symbol. In Table 1, hashtags can be found in t_1, t_3, t_4 and t_6 (i.e., #Canberra, #Floriade). Hashtags are sometimes used as proxies for topics [16], but similar hashtags do not necessarily mean similar topics, and hashtags often cannot directly represent topics. For example, in t_1, the hashtag #Canberra indicates a location, which is not the real topic of the tweet. We still consider the inclusion of hashtags as an important feature in content-based similarity as they indicate indirect relationships amongst tweets. All features mentioned above form important underlying networks in the Twitter environment.

The social interactions can be classified into two parts: Interactions based on people and interactions based on actions. *Mention* is an example of an interaction based on people. If there are two or more tweets mentioning the same users, there is a higher possibility that they have a similar topic in comparison with tweets without any interactions. Interactions based on actions include *replies* and *retweets* features. When a tweet is a reply or a retweet of another specific tweet, it is very likely to share the same topic. Recently, a new feature was added in Twitter allowing users to add a comment when they want to retweet a tweet. This new feature makes retweet look like a reply with the original tweet as its quotation. Not all tweets involve social interaction, however. To deal with self-contained tweets, we use the content similarity (including hashtags) to measure the relationship of these tweets to others.

To see how the social interactions between tweets and the content similarity are able to represent the topical connectivity, we study two labeled Twitter datasets with different characteristics in terms of the number of interactions involved, the number of topics and the relationships density. We evaluate the level of topical accuracy of a pair of tweets connected by either people based interactions or action based interactions, or content similarity.

1.1.1 Datasets

We use two datasets: *tweetMarch* and *TREC2014* to analyze the topical relationships between tweets. *tweetMarch* is a corpus of tweets we collected for our research, and *TREC2014* is available online at http://trec.nist.gov/data/microblog2014.html. These two datasets will also be used for the purpose of the evaluation presented in Section 2.

Each dataset has different characteristics, especially in relation to the availability of interaction features and the density of term co-occurrences. Our first step is to perform some pre-processing on the posts. We remove all characters that are irrelevant for topic representation (punctuations, emoticons), stop-words and all terms with fewer than 3 characters. Then, all remaining terms are stemmed using the python NLTK package, followed by the tokenization of all tweets and terms. As previously mentioned, all hashtags are kept unchanged. In our experiments, we only include English tweets by filtering the tweets through the language information in their metadata.

The *tweetMarch* dataset was collected between 03 March 2014 and 07 March 2014, using the Twitter Streaming API,[4] It has 729,334 tweets from 599,713 different users. 12,221 posts are reply tweets and 101,272 retweets. In this corpus, the tweets are kept in the order of the time they were posted. Two annotators were invited to label the first 10,000 tweets into 6 different topics: *food, day activities, life expressions, people communications, politics,* and *travel and transport*. Both annotators agreed in 83 percent of the tweets. The kappa value [17] is 0.77, which measures the qualitative inter-rater agreement. This value is categorized as substantial agreement based on the Landis and Koch interpretation [18].

Table 2 shows the number of connections between tweets for each type of feature. We see that the interactions based

[4] https://dev.twitter.com/streaming/overview, accessed February 23, 2017.

TABLE 2
Number of Connections between Tweets in the *tweetMarch* Dataset (for Each Interaction Type)

# of tweets	people	actions	content	all
5000	43497	7874	2201094	2207719
10000	132735	17238	8711010	8728951
15000	225470	22447	20191171	20219567
20000	368151	27287	37003316	37046269
25000	564435	33070	57921730	57988129

TABLE 3
Density Comparison of the Tweet-Relationship Matrix (A), Tweet-Term Matrix (V), and Term-Term Matrix (T) from the *tweetMarch* Dataset

# of tweets	# of terms	A	V	T
5000	5417	17.662%	0.125%	0.379%
10000	8031	17.458%	0.084%	0.298%
15000	10489	17.973%	0.065%	0.255%
20000	12491	18.523%	0.055%	0.229%
25000	14067	18.556%	0.049%	0.214%

on people form around 1.31 percent of connections between tweets on average. Since the number of replies or retweets are very low, action based interactions only connect 0.16 percent of the tweets on average. The highest number of connections between tweets is presented by the similarity of tweet-content. This is due to the high number of self-contained tweets. While there are few interactions based on people and action, taking them into account in the topic derivation process still has a high impact on the quality of the topic derived, as will be seen below based on our experimental results. The discussion of the impact of each interaction feature can be found later in Section 2.3.1.

Table 3 shows the comparison of the density between our tweet-relationship matrix (A), the commonly used tweet-term matrix (V), and the term-term matrix (T) [9] for this dataset. The tweet-term matrix is computed with the *tf-idf* function [19], and, for the term-term matrix, we use the *positive point mutual information (PPMI)* function [9]. As shown in Table 3, the tweet-relationship matrix (A) has the highest density with 18.03 percent of non-zero element on average for different number of tweets in the subset of dataset. The tweet-term relationship is the most sparse with only 0.08 percent non-zero element, followed by the term-term matrix with 0.28 percent density on average. This analysis suggests that our definition of tweet interactions is able to significantly improve the density of the matrix over other regular types of relationships.

The *TREC2014* dataset is provided by *The Text REtrieval Conference* (TREC),[5] a community co-sponsored by the National Institute of Standards and Technology (NIST) and U.S. Department of Defense. TREC2014 consists of more than 50,000 tweet IDs, with each of the tweets belonging to one of 55 available topics.[6] To download the tweets based on a given ID, we use Twitter REST API.[7] From the list of IDs available in the *TREC2014* dataset, only 46,572 tweets can be downloaded. This could be due to different reasons: For example, the tweet has been deleted or the status of tweet has been changed to 'protected'. These downloaded tweets were authored by a total of 35,670 users.

Table 4 shows the density of several type of relationships within the TREC2014 dataset. Here we see that the tweet-relationship matrix (A) has the highest density over the other type of relationships. It is interesting to note that, from the total tweets available in TREC2014, there are only 3,463 reply tweets and no retweets. Yet, the density of the tweet-relationship matrix is still far higher than the density of the tweet-term and term-term matrices.

1.1.2 Topical Relationship

To investigate the prevalence of topical relationships when pair of tweets are connected by interactions and content similarity, we conducted a topical connectivity analysis on both the tweetMarch and TREC2014 labeled datasets. Fig. 2 shows the percentage of pairs of tweets under the same topic when connected by action based interactions (reply and retweet), mention based interactions, and content similarity.

99 percent of pairs of tweets that are connected by the action based interaction (reply and retweet) are about the same topic. 80 percent of pairs of tweets that are connected by the mention based interaction are about the same topic. 51 percent of pairs of tweets that are connected by the content similarity are about the same topic. Further analysis also shows that the chance of being about the same topics when connected by content similarity is much higher if two tweets have two or more terms in common. Unfortunately, more than 90 percent of tweets that are paired by content similarity have only one term in common.

In both datasets, tweets with interactions are only around 20 percent. Content similarity is still the most important feature to build up the relationship matrix. Due to the fact that pair of tweets connected by action based interactions (reply-retweet) and mention based interactions are more likely to be about the same topic, we can predict that the incorporation of interaction features in the relationship matrix will improve the quality of the topic derivation compared to only considering the content similarity.

1.1.3 Relationship Formulation

A tweet is defined as a tuple $t = \langle P_t, rtp_t, C_t \rangle$, where P_t is all users mentioned in the tweet including its original author, rtp_t is the reply and retweet information, and C_t is the set of

TABLE 4
Density Comparison of the Tweet-Relationship Matrix (A), Tweet-Term Matrix (V), and Term-Term Matrix (T) from *TREC2014* Dataset

# of tweets	# of terms	A	V	T
5000	6793	2.698%	0.090%	0.317%
10000	10019	2.699%	0.061%	0.267%
15000	12647	2.680%	0.049%	0.237%
20000	14870	2.696%	0.0415%	0.218%
25000	16848	2.703%	0.0367%	0.205%

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

[6] List of topics are available at http://trec.nist.gov/data/microblog2014.html, accessed February 23, 2017.

[7] https://dev.twitter.com/rest/public, accessed Feburary 23, 2017.

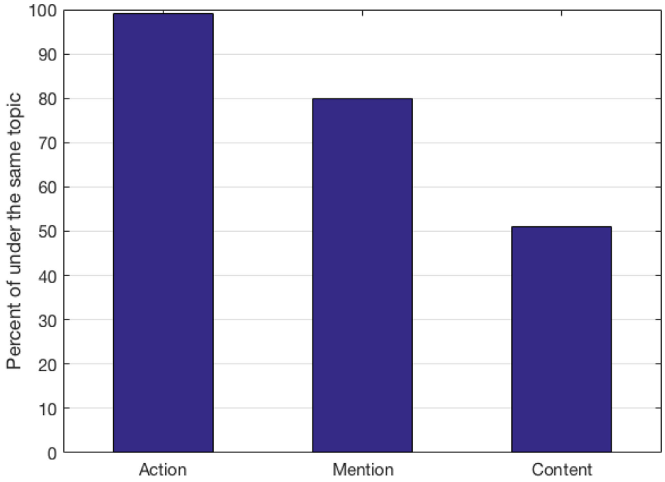


Fig. 2. Percentage of pair of tweets to be about the same topic when connected by action based interaction, mention based interaction, and content similarity for both datasets.

terms from the tweet, including hashtags. The relationship between two tweets t_i and t_j is then denoted as $R(t_i, t_j)$. It is a combination of three interactions: People ($po(P_{t_i}, P_{t_j})$), actions ($act(rtp_{t_i}, rtp_{t_j})$) and content similarity ($sim(C_{t_i}, C_{t_j})$). The value of the relationship between tweets range from 0 to 1, where 0 means no relationship, and a higher value of $R(t_i, t_j)$ means a stronger relationship between two tweets. $R(t_i, t_j)$ is defined as follows:

$$\begin{aligned} R(t_i, t_j) &= po(P_{t_i}, P_{t_j}) + act(rtp_{t_i}, rtp_{t_j}) \\ &\quad + sim(C_{t_i}, C_{t_j}). \end{aligned} \tag{1}$$

The component of the relationship from the interactions based on people $po(P_{t_i}, P_{t_j})$ is defined as the intersection of P_{t_i} and P_{t_j} (i.e., people mentioned in both tweets including the authors) divided by the total number of all users involved in both tweets

$$po(P_{t_i}, P_{t_j}) = \frac{|P_{t_i} \cap P_{t_j}|}{|P_{t_i} \cup P_{t_j}|}. \tag{2}$$

In the motivating example (see Table 1), $P_{t_4} = \{d\}$ and $P_{t_5} = \{d, e\}$. d is the common user mentioned in both tweets, so $po(P_{t_4}, P_{t_5})$ will be 0.5.

The relationship from the actions based interactions includes the activity of *retweet* and *reply* between tweets t_i and t_j. We denote this component as $act(rtp_{t_i}, rtp_{t_j})$. This type of interactions is the most apparent feature that indicates the existence of a relationship between tweets. If tweet t_i is a *retweet* or *reply* of tweet t_j or vice versa, or if both tweets are *replying* to or *retweeting* the same tweet, the value of $act(rtp_{t_i}, rtp_{t_j})$ will be 1, otherwise it is 0. When $act(rtp_{t_i}, rtp_{t_j})$ equals to 1, it indicates that those two tweets are on the same topic. rtp_t is the ID of a retweeted or replied tweet in tweet t

$$act(rtp_{t_i}, rtp_{t_j}) = \begin{cases} 1, (rtp_{t_i} = j) \ or \ (i = rtp_{t_j}) \\ \qquad or \ (rtp_{t_i} = rtp_{t_j}) \\ 0, \ otherwise. \end{cases} \tag{3}$$

The value of $act(t_1, t_2)$ in Table 1 will be 1 since t_2 is a reply of t_1. $act(t_2, t_3)$ is also 1 as both t_2 and t_3 refer to the same tweet t_1.

The relationship from the tweet-content is based on content similarity. $sim(C_{t_i}, C_{t_j})$ denotes the similarity of the tweet-content between tweet t_i and t_j, measured using the *cosine similarity* formula [19]. In the preprocessing steps, all terms/characters that potentially degrade the performance of topic identification processes (i.e., emoticons, punctuations and terms with fewer than 3 characters) are removed. We also remove stop words, and are thus left only with the content-full words. Hashtags are included and kept unchanged

$$\begin{aligned} sim(C_{t_i}, C_{t_j}) &= \frac{C_{t_i}.C_{t_j}}{\|C_{t_i}\|\|C_{t_j}\|} \\ &= \frac{\sum_{x=1}^{n} (C_{t_i})_x \times (C_{t_j})_x}{\sqrt{\sum_{x=1}^{n} ((C_{t_i})_x)^2} \times \sqrt{\sum_{x=1}^{n} ((C_{t_j})_x)^2}}. \end{aligned} \tag{4}$$

Having all of the three components, we can calculate the relationship among the tweets ($R(t_i, t_j)$) as shown in equation (1). All values of ($R(t_i, t_j)$) form a tweet-relationship matrix $A \in \mathbb{R}^{m \times m}$, where $a_{ij} = f(R(t_i, t_j))$. $f(x)$ is a *sigmoid function* [20] to normalize the value of each element in matrix A for a better relationship distribution

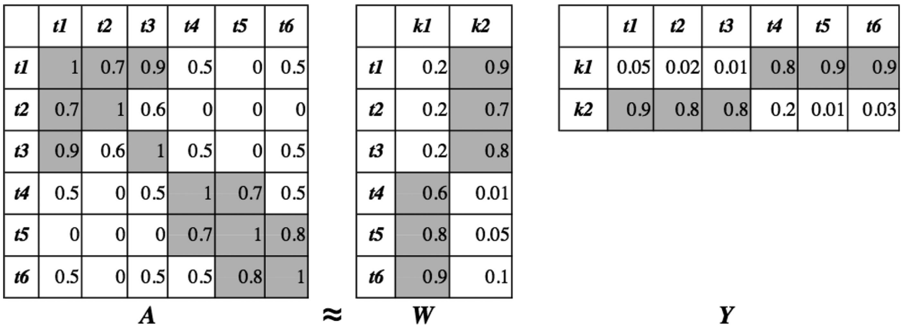
$$f(x) = \begin{cases} \frac{1}{1+e^{-x}}, x > 0 \\ 0, \ otherwise \end{cases}. \tag{5}$$

1.2 Clustering Tweets

In our proposed approach, deriving topics from Twitter is done through two consecutive steps: (1) cluster the tweets by deriving the latent tweet-topic matrix from the tweet-relationship matrix, and (2) learn the keywords representation for every topic by using the derived latent tweet-topic from previous step. Both steps utilize the NMF technique, so we call our topic derivation method *intJNMF*. This section discusses the first step of this approach.

The clusters of tweets are derived by factorizing the tweet-relationship matrix A into a lower dimensional representations of the latent tweet-topic matrix using NMF. NMF is a popular dimensional reduction technique, and one of its main application domains is unsupervised clustering [21], [22], [23], [24], [25]. NMF is guaranteed to converge to the local optima between the data matrix and its lower rank representations matrix [4] when minimizing their distance. There are quite a few methods that can be employed to achieve this objective, such as generalized Kullback-Leibler divergence [26], multiplicative update rule [4], Itakuro-Saito distance [27] and Alternating Least Squares (ALS) [28].

The tweet-relationship is modeled as the combination of various interactions and content similarity, and the relationships between tweets express their topical connectivity. The derived tweet-topic matrix represents the latent thematic structure of the relationships between tweets. It can be directly used to generate the topical clusters of the tweets. In our approach, matrix $A \in \mathbb{R}^{m \times m}$ is factorized into its lower dimensional tweet-topic matrix



	t1	t2	t3	t4	t5	t6
t1	1	0.7	0.9	0.5	0	0.5
t2	0.7	1	0.6	0	0	0
t3	0.9	0.6	1	0.5	0	0.5
t4	0.5	0	0.5	1	0.7	0.5
t5	0	0	0	0.7	1	0.8
t6	0.5	0	0.5	0.5	0.8	1

A ≈

	k1	k2
t1	0.2	0.9
t2	0.2	0.7
t3	0.2	0.8
t4	0.6	0.01
t5	0.8	0.05
t6	0.9	0.1

W

	t1	t2	t3	t4	t5	t6
k1	0.05	0.02	0.01	0.8	0.9	0.9
k2	0.9	0.8	0.8	0.2	0.01	0.03

Y

Fig. 3. Factorization of tweet-relationship matrix A into the latent matrix W and Y. The dark areas indicate the potential topical clusters of the tweets.

$W \in \mathbb{R}^{m\times k}$ and $Y \in \mathbb{R}^{k\times m}$ where k is the given number of clusters/topics. Since A is a symmetric matrix, either W and Y is able to show the potential cluster for every tweet. The objective of this factorization process is to minimize the divergence of A and WY so that $A \approx WY$. We employ the *Kullback-Leibler divergence* [26] to measure the divergence $D(A\|WY)$ [4]

$$D(A\|WY) = \sum_{ij}\left(a_{ij}\log\frac{a_{ij}}{(wy)_{ij}}\right) - a_{ij} + (wy)_{ij}. \tag{6}$$

The multiplicative update rules in each iteration for matrix W and Y are as follows:

$$W = W\frac{Y^T(A/(WY))}{Y^TI}, \qquad Y = Y\frac{(A/(WY))W^T}{IW^T}. \tag{7}$$

Fig. 3 shows the results of the factorization process of the tweet-relationship matrix A from the example of tweets available in Table 1. In this figure, W and Y are the latent tweet-topic matrices derived from A with the number of topics $k = 2$. These two matrices are the lower dimensional representations of the matrix A. We can see that, in matrix A, the strong connection between tweets are marked in the dark areas, and it also shows how the tweets are grouped. In both matrices W and Y, the representation of the relationships in k number of topics is consistent, for example, if, for every row in matrix W, we take the highest value to define the cluster membership. t_1, t_2 and t_3 are in cluster k_2, and t_4, t_5, and t_6 are in cluster k_1. In the next step, the tweet-topic matrix $W \in \mathbb{R}^{m\times k}$ is used as an additional information when learning the keywords representation to deal with the sparsity of the tweet-term matrix V.

1.3 Inferring Keywords Representation for Each Topic

The second step of our proposed approach is to infer the best keywords to represent every topic. In a general NMF, the representative keywords are captured by factorizing the tweet-term matrix directly into the tweet-topic matrix and the topic-term matrix. Each element in the tweet-term matrix is computed using the *term frequency-inverse document frequency* (tf-idf) metric [19]. This metric calculates the weight of every unique term in a tweet. The higher value of the tf-idf of a term in a tweet, the more important this term to the tweet. It is defined as follows:

$$tfidf(s,t,T) = tf(s,t) \times idf(s,T). \tag{8}$$

where $tf(s,t)$ is the frequency of term s in the tweet t and the inverse document frequency $idf(s,T)$ is the level of rarity of the term s in the whole collection of tweets T.

Our intJNMF method makes use of the tweet-term matrix to infer the representative keywords. In particular, we compute the tweet-term matrix V using the *tf-idf* value for every tweet and all unique terms in each of them. Furthermore, the tweet-term matrix $V \in \mathbb{R}^{m\times n}$ is then factorized into tweet-topic matrix $W \in \mathbb{R}^{m\times k}$ and the topic-term matrix $\widetilde{H} \in \mathbb{R}^{k\times n}$. m is the number of tweets in a collection, n is the number of unique terms, and k is the number of potential topics defined by user. The objective function of the second factorization process is $\min D(V\|W\widetilde{H})$ [4] where

$$\min\, D(V\|W\widetilde{H}) = \sum_{ij}\left(v_{ij}\log\frac{v_{ij}}{(w\tilde{h})_{ij}}\right) - v_{ij} + (w\tilde{h})_{ij}, \tag{9}$$

$$\widetilde{H} = H\frac{(V/(WH))W^T}{IW^T}. \tag{10}$$

The tweet-term matrix V will be very sparse. As shown in Tables 3 and 4 of Section 2, the average density of tweet-term matrix (the non-zero element in the tweet-term matrix which shows the availability of relationship between tweet and term) is less than 0.2 percent. Thus, to reduce the negative impact of this extreme sparsity, we modify the NMF approach when factorizing the matrix. First, we use the tweet-topic matrix derived from the previous step to initialize the matrix W. Second, during the iteration to minimize the divergence between matrix V and WH, we only update the matrix H and retain matrix W in its original value. Matrix W was derived from the tweet-relationship matrix A, which is much less sparse if compared to the tweet-term matrix V. Our investigation shows that each cluster from the derived matrix W in the first step of the algorithm provides the most accurate topic. The experiment will be reported in Section 2 below. The biased update rule for W in the second step will provide additional information for the process inferring the topic-term matrix H, and, in the same time, reduce the penalty of the extreme sparsity of the tweet-term matrix V. In every iteration, the update rule for matrix H is shown in equation (10).

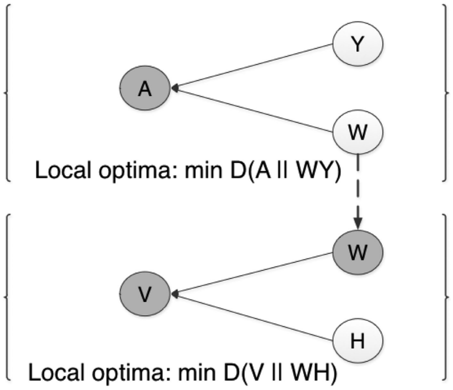


Fig. 4. *intJNMF* model.

The complete two-step process is illustrated in Fig. 4. From this figure, we can see the connection between the first factorization and the subsequent process. The second factorization process takes the matrix W from the previous step to infer matrix H without updating the matrix W. We call these consecutive steps as *Joint-NMF*. This model can also be expressed as follows:

$$A \approx WY \mapsto V \approx W\widetilde{H}, \tag{11}$$

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

$$\mathscr{T}_{Joint-1st-process} = D(A \| WY). \tag{12}$$

$$\mathscr{T}_{Joint-2nd-process} = D(V \| WH). \tag{13}$$

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

Table 5 shows the topic-term matrix H after performing Joint-NMF on V, the matrix built using the motivating example from Table 1. For readability purposes, words with a very low value for both topics (rows) are removed from the table. Thus, the keywords representation for topic k_1 can be inferred as *#floriade*, *celebration*, *spring*, *event*, *nightfest*. For cluster k_2, the best topic representation will be: *new, senate, exciting*. In topic derivation, a keyword is listed in several topics. In this case, '#canberra' can be included to represent both k_1 and k_2 as it has a high and almost similar value for both clusters.

The whole topic derivation process of intJNMF is described in the following *Algorithm 1*.

2 Experiments

In this section, we present our evaluation. We first discuss the evaluation metrics we employed, the baseline methods we considered and, finally, the results. In the last section we also discuss various setups used to test the impact of the interactions on the quality of topic derivation.

Algorithm 1. Topic Derivation Using *intJNMF*

```
INPUT: number of topics K, tweet-term matrix $V \in \mathbb{R}^{m \times n}$
OUTPUT: tweet-topic matrix $W \in \mathbb{R}^{m \times k}$ and topic-term matrix $H \in \mathbb{R}^{n \times k}$
get tweet-relationship matrix $A \in \mathbb{R}^{m \times m}$
initialize $W, Y$ and $H$
NMF on $A \approx W.Y$
repeat
    $H \leftarrow f(V, W, H)$
until $V \approx W.H$
return $W, H$
```

2.1 Baseline Methods

We use the following baseline methods:

- *intLDA* [2]: An extension of the LDA method with Gibbs Sampling. In this approach, related tweets are used directly in the process of sampling the tweet-topic distribution.
- *TNMF* [9]: An extension of NMF which incorporates the correlation between terms (term-term) matrix to derive the topics.
- *NMF* [4]: The most cited NMF method, which directly factorizes the tweet-term matrix V into the tweet-topic matrix V and the topic-term matrix H. Our NMF implementation is based on the KL divergence objective function.
- *LDA* [5]: The most popular topic derivation method with the "bag of words" assumption and with each document drawn from a mixture of several topics. Our LDA implementation is based on the Gibbs sampling inference strategy discussed in [29].

2.2 Evaluation Metrics

To evaluate the quality of derived topics, we measure the level of accuracy of the cluster results by comparing them with the labeled datasets. We use the following metrics: *Purity*, *Normalized Mutual Information (NMI)*, and *F-Measure*.

Purity [30] evaluates the extent to which tweets are assigned the correct topics based on our labeled datasets. The value of purity will be in the range of 0 and 1, where 1 is a perfect cluster. In this metric, every cluster derived from W (tweet-topic matrix) are assigned to the cluster from the evaluation dataset C with the maximum similarity. For every cluster, all correctly assigned tweets are counted, and the result is then divided by the total number of tweets involved in the evaluation. Perfect clustering means all

TABLE 5
Topic-Term Matrix (H) from Joint-NMF on $V \approx WH$

	new	senate	exciting	#canberra	census	#floriade	celebration	spring	event	nightfest
k_1	4.47e-10	4.15e-13	3.54e-15	0.17	2.31e-10	**0.59**	**0.35**	**0.57**	**0.55**	**0.43**
k_2	**0.55**	**0.51**	**0.54**	0.21	**0.43**	1.15e29	9.45e-30	7.82e-30	1.12e-12	2.32e-24

tweets are correctly assigned to a topic based on the evaluated set

$$purity(W,C) = \frac{1}{N}\sum_i \max_j |w_i \cap c_j|. \tag{14}$$

NMI [30] measures the accuracy of the cluster by computing the mutual information $I(W;C)$ divided by the average entropy of both clusters W and classes C. Similar to purity, this metric has values in the range of 0 to 1, where higher value means higher accuracy. Since it includes the normalization with entropy, this metric can also measure the trade-off of the quality of clusters on different setups (i.e., the number of clusters)

$$NMI(W,C) = \frac{I(W;C)}{[H(W)+H(C)]/2}. \tag{15}$$

Mutual information $I(W,C)$ is a measure to quantify the statistical information shared by a pair of clusters W and C [31], which is defined in equation below

$$I(W,C) = \sum_k \sum_j P(w_k \cap c_j)\log \frac{P(w_k \cap c_j)}{P(w_k)P(c_j)}, \tag{16}$$

where k and j are the number of clusters in W and C, respectively; w_k is the specific cluster at index k from the set of result clusters W, and c_j is the specific cluster at index k from the set of evaluation clusters C. $P(w_k)$ 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 result w_k and in the cluster from evaluation set c_j. The calculation of the entropy of clusters $H(W)$ and classes $H(C)$ are shown in equation below

$$\begin{aligned} H(W) &= -\sum_k P(w_k)\log P(w_k), \\ H(C) &= -\sum_j P(c_j)\log P(c_j). \end{aligned} \tag{17}$$

For the evaluation, we also include the pairwise F-Measure metric [30] which computes the harmonic mean of both precision p and recall r

$$F = 2 \times \frac{p \times r}{p + r}. \tag{18}$$

where

$$p = \frac{TP}{TP+FP}, \quad r = \frac{TP}{TP+FN} \tag{19}$$

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. *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.

2.3 Evaluation and Discussion

We analyze the impact of each feature on topic derivation by comparing our proposed approach against other baseline methods. In each experiment, we executed all methods with various numbers of expected topics. Each experiment executes 5 topic derivation methods for a particular number of expected topics. For both datasets, we set different numbers of topics to analyze the performance of the algorithms for different numbers of latent factors. For every k and every method, we ran the algorithms over both datasets 30 times, and take the average value of each evaluation metric for comparison.

2.3.1 Impact of Interaction Features

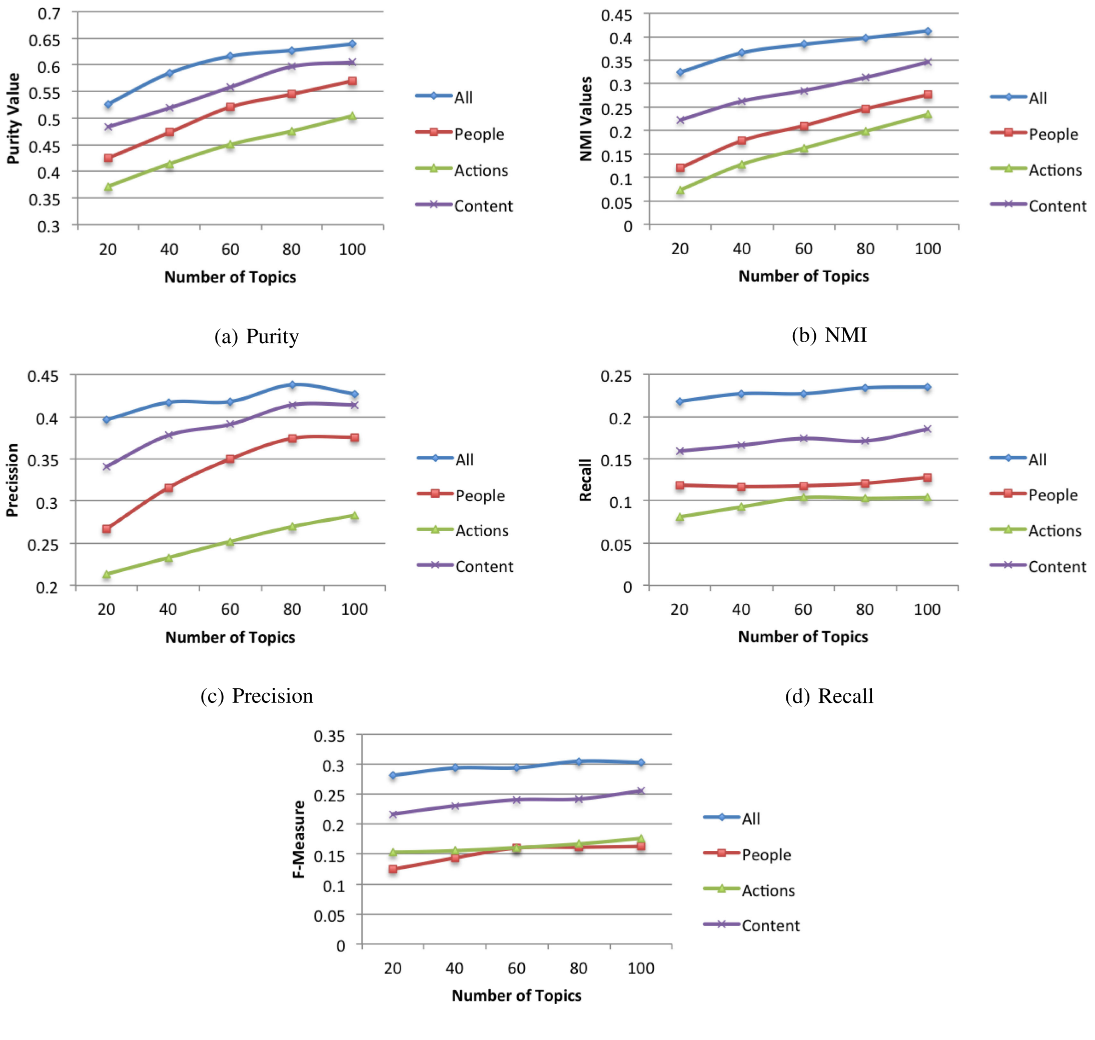
To see the impact on topic derivation of each individual feature of the relationship between tweets, we present their performance for various configurations and evaluation metrics over the *tweetMarch* dataset in Fig. 5. From each subfigure, we can see that the combination of all features provides the best results for all evaluations. All metrics show a similar trend, with content based similarity as the second best, followed by the people and actions based interactions. This trend matches with the number of connections between tweets from each feature as shown in Table 2. As there are very high percentages of content based relationship amongst the tweets, it is unquestionable that this feature will produce the highest tweet clusters accuracy in comparison with other individual features. However, when all three features are combined, there are significant improvements in all evaluation metrics.

We use several subsets of the tweetMarch dataset to further see the impact of incorporating social interactions to improve the quality of derived topics. Each subset has different proportions of reply and retweet tweets. The first subset has 10 percent; the second subset has 30 percent; and the third has 50 percent reply and retweet tweets. In this specific experiment, we compare our proposed intJNMF method with the same one, but without the incorporation of interactions and use only the content similarity function to compute the tweet-relationship matrix. The results are shown in Fig. 6. In this figure, we see that, at all subsets, the method that incorporates both the social interactions and content similarity outperforms the method that only considers the content.

2.3.2 Comparison with Baseline Methods

The purity and NMI evaluation results against the baseline methods on the tweetMarch dataset are shown in Figs. 7a and 7b, respectively. We evaluate the methods against the 10K tweets of the tweetMarch that have been manually annotated. We note that intJNMF significantly outperforms other methods with both metrics on all k numbers of topics. The intJNMF is able to provide 10-35 percent improvement on both Purity and NMI over the other baseline methods. In contrast, the original NMF method cannot achieve a good result. Directly factorizing the tweet-term matrix V into the tweet-topic W and topic-term H suffers from extreme sparsity. The LDA method has better results compared to the orignal NMF, but it is still inferior to our proposed method.

As shown in Table 6, the F-Measure results on the tweetMarch dataset also show a similar trend with other metrics. intJNMF performs again better than all the baseline methods. The F-Measure results on the tweetMarch dataset



(a) Purity

(b) NMI

(c) Precision

(d) Recall

(e) F-Measure

Fig. 5. Evaluation of the impact of each relationship feature in the *tweetMarch* dataset.

confirm that our intJNMF method is able to consistently outperform other baseline methods.

For the *TREC2014* dataset, we tested all tweets that belong to the first ten topics (*MB171* to *MB180*). Fig. 8 shows the results of the Purity and NMI evaluations on the *TREC2014* dataset for $k = 10$. In the purity test, the highest score is achieved by intJNMF with the value of 0.405, improving dramatically over that of our previous work, intLDA, which is in the second position with the value of 0.262. Both of these methods incorporate the interactions between tweets, and they are able to outperform other baseline methods that focus only on content.

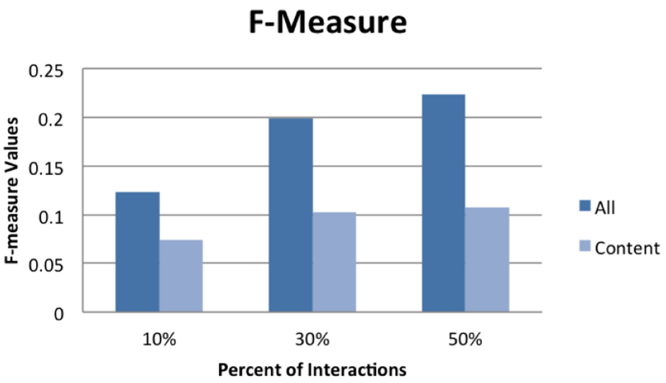
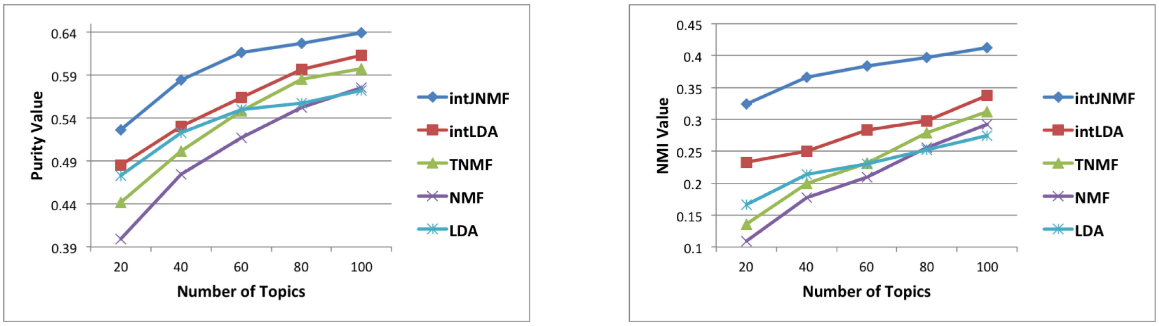


Fig. 6. Impact of interactions availability on three different subsets of *tweetMarch* evaluation set.

The NMI results for the *TREC2014* dataset are shown in Fig. 8b. They present a similar trend to the Purity results. intJNMF is in the top position with 0.367. The improvement is more than 40 percent compared to other baseline methods (intLDA:0.197, LDA:0.172, TNMF:0.083, and NMF:0.058).

The performance of intJNMF on *TREC2014* is confirmed with the precision, recall and f-measure results as shown in Table 7. We see that intJNMF again consistently achieves the best result. The most significant improvement is the recall measure with more than a 70 percent increase over the second best. The original NMF method has the worst performance.

Examples of word representations for several topics from the *tweetMarch* dataset are listed in Table 8. Labels for topics in Table 8 are done manually based on our labeled tweetMarch dataset. Our proposed method presents better keywords for each topic, as it is able to provide more connected words, making the topic more readable [32]. The NMF method has the worst performance, since it finds many unrelated words to represent almost all topics. Table 9 shows the top-5 topic-term for some topics from the



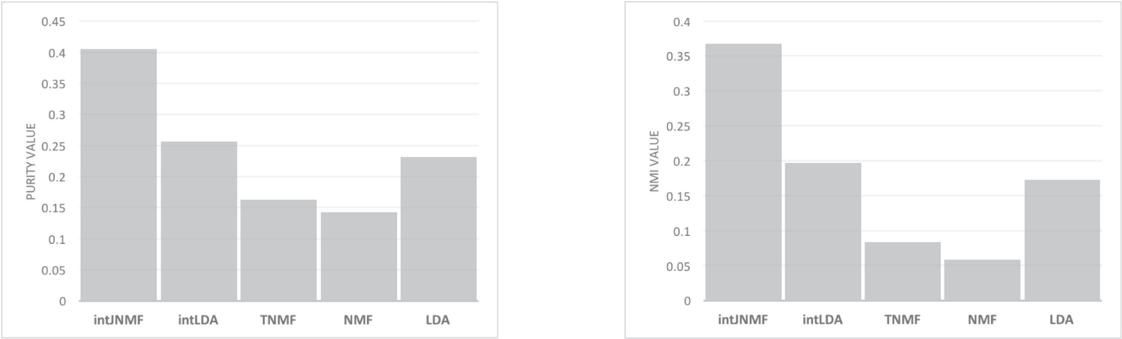
(a) Purity Score

(b) NMI Score

Fig. 7. *Purity* and *NMI* results on the *tweetMarch* dataset.

TABLE 6
Precision, Recall and F-Measure on *tweetMarch* Dataset for Topics $k = 20, 40, 60, 80, 100$

Method	$k = 20$			$k = 40$			$k = 60$			$k = 80$			$k = 100$		
	p	r	F-m	p	r	F-m	p	r	F-m	p	r	F-m	p	r	F-m
intJNMF	**0.396**	**0.218**	**0.280**	**0.417**	**0.227**	**0.292**	**0.418**	**0.227**	**0.293**	**0.438**	**0.234**	**0.304**	**0.427**	**0.235**	**0.302**
intLDA	0.274	0.193	0.222	0.277	0.179	0.212	0.269	0.169	0.203	0.279	0.153	0.192	0.267	0.153	0.188
TNMF	0.276	0.079	0.123	0.335	0.050	0.088	0.381	0.043	0.078	0.418	0.037	0.068	0.458	0.035	0.065
NMF	0.271	0.072	0.114	0.336	0.047	0.083	0.405	0.039	0.072	0.457	0.035	0.065	0.492	0.032	0.060
LDA	0.310	0.084	0.132	0.369	0.057	0.099	0.404	0.047	0.084	0.424	0.041	0.075	0.430	0.038	0.069



(a) Purity Score

(b) NMI Score

Fig. 8. *Purity* and *NMI* results on *TREC2014* dataset.

TREC2014 dataset. Labels for topics in Table 9 are provided by the TREC2014 dataset provider. In this table, all methods seem to be able to list the keywords accurately in most topics. Note, however, that, as explained previously, our new method achieved higher accuracy in the clustering.

To summarize, then, based on all evaluation results, our proposed method performs better. The improvement in topic derivation quality has shown that incorporating relationships between tweets is important to deal with the sparsity problem when considering term overlaps in the Twitter environment.

3 Related Work

In traditional media with lengthy content, most popular methods like PLSA [3], LDA [5] and NMF [4] focus only on content to derive the topics. However, because tweets are so short, there is typically little term overlap, resulting in very low numbers of term co-occurrences. This heavily hurts the quality of the derived topics. Some extensions were proposed to work in Twitter [7], [8], [9], [11], [12]. However, as they still mainly exploit only the content and/or limited interaction features, the sparsity remains a problem.

Some studies tried to incorporate external sources for expanding the content to deal with sparsity issues. The study of [33] found that aggregating the sparse tweets into a single content to be processed by LDA could improve the quality of the topics. The study of [34] used Freebase on their knowledge-expansion based method to augment the content. The study of [13] expands the content based on the web document referred by the URL from the tweet. However, involving external documents processing is possibly not scalable in a highly dynamic environment such as Twitter. Furthermore, the addition of content from external resources is problematic as most of the tweet contents are informal and the added terms might not have any relation to the tweet topic [35].

The study of [9] extended the original NMF method to incorporate the term-correlation matrix. This matrix is built by computing the positive mutual information value for each pair of unique terms available in the tweets collection. The

TABLE 7
Precision, Recall, and F-Measure on *TREC2014* Dataset with $k = 10$

Method	precision	recall	F-Measure
intJNMF	**0.179**	**0.677**	**0.283**
intLDA	0.090	0.335	0.143
TNMF	0.048	0.179	0.076
NMF	0.040	0.148	0.063
LDA	0.080	0.294	0.126

TABLE 8
Top-5 Topic-Term for Some Topics discovered on the *tweetMarch* Dataset

Topic Labels	Representative words				
	NMijF	intLDA	TNMF	NMF	LDA
Travel/ transport	*train* #traffic *accident* *driver* *road*	*accident* *road* #traffic *train* closed	*#traffic* *road* **time** *driver* closed	**follow** *train* #traffic *driver* **gamer**	*train* *road* *driver* closed **time**
Politics	*liberal* *obama* *government* people **big**	*liberal* *obama* people **chance** *policy*	*policy* *liberal* **big** *government* **cold**	**gain** *politic* **high** *government* *obama*	**high** *liberal* *obama* **big** **process**
Food/ Beverages	*tea* *drink* *order* sweet *coffee*	order *tea* cold **talk** *brown*	black **free** *coffee* **talk** **reading**	**talk** *coffee* *drink* **smoking** **sleep**	table *tea* *coffee* stop **closed**

Words in italic have high connectivity with the topics, ***bold*** *words has low connectivity with the topics.*

term-correlation matrix is then jointly factorized with the tweet-term matrix to derive both the clusters for the tweets and the keywords representation. However, the method still only considers content based semantic relationships. As discussed in the dataset characteristics, the term-correlation matrix built from tweet content can be very sparse.

The study of [36] evaluated the implementation of the Author-Topic (AT) model [37] and the Author-Recipient-Topic (ART) model [38] in microblogs environments. These two models are based on LDA. The AT method assumes that a document's topic distribution is influenced by the content and the set of authors. The ART model improves on the AT method by incorporating not only the author, but also the recipient of the document. The experiments showed that LDA is still the best in most cases. In a higher number of topics, AT and ART were only able to present very limited improvement over the original LDA method. Ramage in [12] reported the implementation of the *labeled LDA* method to derive topics from the Twitter environment. The labels are learned through limited content based interactions such as hashtags and other signals of social interactions, like emoticons and other specific terms.

Different from other approaches, the study of [39] tried to take the context of Twitter users (e.g., following/followers, mentions) into account, but ignores the content of the tweets. The study of [40] incorporated the users following/

TABLE 9
Top-5 Topic-Term for Some Topics Discovered on the *TREC2014* Dataset

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

followers characteristics and LDA based topic derivation process to identify influential users in Twitter. The study of [41] reported that discussed topics derived from tweets that have social interactions will have much higher credibility than if such interactions are not available. Recently, we investigated the temporal features of the interaction features related to the derivation of topics, and found that, for online/real-time situation, involving the time aspect of mention based interactions could improve the quality of the derived topics [42]. These studies inspired us to investigate the best way of incorporating both the text and social interactions to improve the quality of topic derivation.

Our work is rooted in the NMF algorithm. NMF is one of the most effective methods to perform dimensional reduction and uncover the hidden thematic structures or latent features of a relationship-based matrix [4]. The study of [8] shows that NMF is able to provide more consistent results over multiple runs than other popular topic derivation methods such as LDA. However, the low frequency of co-occurring and overlapping term in Twitter makes the general NMF algorithms produce poor quality topics. To overcome this problem, we propose a joint factorization process of tweet-relationship matrix and tweet-term matrix. The first step learns the clusters of tweets based on the relationship between tweets, and the second step infers the topic words by using both the clustering and the context information of the tweets. Our experiments have demonstrated that our proposed intJNMF method obtains far superior performance than general NMF methods.

4 Conclusions

In this paper, we presented a new topic derivation method for a tweet collection. Topic derivation is important to provide underlying services for many applications in various areas including marketing, emergency, and national security. With the incorporation of tweet text similarity and tweet interactions, the quality of topic derivation is significantly improved.

Our evaluation results demonstrate that each feature has a positive impact on the quality of topic derivation, and the best performance is achieved when the three types of features (interaction based on people, interaction based on user action, and similarity of tweet-content) are used. intJNMF consistently outperforms other advanced methods on all evaluation metrics. Our experiments reveal that the incorporation of the relationships amongst tweets helps to deal with the sparsity issue from the low frequency of co-occurring terms in Twitter.

We are now working on an incremental model of this method to work in a real-time fashion and the automatic topic labeling as well as finding the optimal number of topics for every run. Finally, we are also investigating more complex combinations of the features and adding temporal features to deal with the dynamics of the Twitter environment.

Acknowledgments

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

References

[1] S. K. Bista, S. Nepal, and C. Paris, “Multifaceted visualisation of annotated social media data,” in *Proc. IEEE Int. Congress Big Data*, Jun. 2014, pp. 699–706.

[2] R. Nugroho, D. Molla-Aliod, J. Yang, C. Paris, and S. Nepal, “Incorporating tweet relationships into topic derivation,” in *Proc. Conf. Pacific Assoc. Comput. Linguistics*, May 2015, pp. 177–190.

[3] T. Hofmann, “Probabilistic latent semantic indexing,” in *Proc. 22nd Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, Aug. 1999, pp. 50–57.

[4] D. Lee and H. Seung, “Algorithms for non-negative matrix factorization,” in *Proc. Advances Neural Inf. Process. Syst.*, 2000, pp. 556–562.

[5] D. Blei, A. Ng, and M. Jordan, “Latent dirichlet allocation,” *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, 2003.

[6] K. Erk, “Vector space models of word meaning and phrase meaning: A survey,” *Language Linguistics Compass*, vol. 6, no. 10, pp. 635–653, 2012.

[7] Y. Hu, A. John, F. Wang, and S. Kambhampati, “ET-LDA: Joint topic modeling for aligning events and their Twitter feedback,” in *Proc. AAAI Conf. Artif. Intell.*, Jul. 2012, pp. 59–65.

[8] J. Choo, C. Lee, C. K. Reddy, and H. Park, “Utopian: User-driven topic modeling based on interactive nonnegative matrix factorization,” *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 1992–2001, Dec. 2013.

[9] X. Yan, J. Guo, S. Liu, X. Cheng, and Y. Wang, “Learning topics in short texts by non-negative matrix factorization on term correlation matrix,” in *Proc. SIAM Int. Conf. Data Mining*, Jul. 2013, pp. 749–757.

[10] M. Albakour, C. Macdonald, and I. Ounis, “On sparsity and drift for effective real-time filtering in microblogs,” in *Proc. 22nd ACM Int. Conf. Inf. Knowl. Manag.*, Oct. 2013, pp. 419–428.

[11] J. Li, Z. Tai, R. Zhang, W. Yu, and L. Liu, “Online bursty event detection from microblog,” in *Proc. IEEE/ACM 7th Int. Conf. Utility Cloud Comput.*, Dec. 2014, pp. 865–870.

[12] D. Ramage, S. T. Dumais, and D. J. Liebling, “Characterizing microblogs with topic models,” *Proc. 4th Int. AAAI Conf. Web Social Media*, May 2010, pp. 130–137.

[13] J. Vosecky, D. Jiang, K. W.-T. Leung, K. Xing, and W. Ng, “Integrating social and auxiliary semantics for multifaceted topic modeling in Twitter,” *ACM Trans. Internet Technol.*, vol. 14, no. 4, 2014, Art. no. 27.

[14] R. Nugroho, J. Yang, Y. Zhong, C. Paris, and S. Nepal, “Deriving topics in Twitter by exploiting tweet interactions,” in *Proc. 4th IEEE Int. Congress Big Data*, Jul. 2015, pp. 87–94.

[15] A. de Moor, “Conversations in context: A Twitter case for social media systems design,” in *Proc. 6th Int. Conf. Semantic Syst.*, Sep. 2010, Art. no. 29.

[16] L. Yang, T. Sun, M. Zhang, and Q. Mei, “We know what@ you# tag: Does the dual role affect hashtag adoption?” in *Proc. 21st Int. Conf. World Wide Web.*, Apr. 2012, pp. 261–270.

[17] J. L. Fleiss, “Measuring nominal scale agreement among many raters,” *Psychological Bulletin*, vol. 76, no. 5, 1971, Art. no. 378.

[18] G. G. K. J. Richard Landis , “The measurement of observer agreement for categorical data,” *Biometrics*, vol. 33, no. 1, pp. 159–174, 1977.

[19] G. Salton, *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Boston, MA, USA: Addison-Wesley, 1989.

[20] D. H. Von Seggern, *CRC Standard Curves and Surfaces with Mathematica*. Boca Raton, FL, USA: CRC Press, 2006.

[21] D. Kuang, H. Park, and C. Ding, “Symmetric nonnegative matrix factorization for graph clustering.” in *Proc. SIAM Int. Conf. Data Mining*, Apr. 2012, pp. 106–117.

[22] Z. He, S. Xie, R. Zdunek, G. Zhou, and A. Cichocki, “Symmetric nonnegative matrix factorization: Algorithms and applications to probabilistic clustering,” *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2117–2131, Dec. 2011.

[23] J. Kim and H. Park, “Sparse nonnegative matrix factorization for clustering,” Georgia Institute of Technology, Tech. Rep. GT-CSE-08-0, 2008.

[24] F. Shahnaz, M. W. Berry, V. P. Pauca, and R. J. Plemmons, “Document clustering using nonnegative matrix factorization,” *Inf. Process. Manag.*, vol. 42, no. 2, pp. 373–386, 2006.

[25] W. Xu, X. Liu, and Y. Gong, “Document clustering based on non-negative matrix factorization,” in *Proc. 26th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, Jul./Aug. 2003, pp. 267–273.

[26] S. Kullback, *Information Theory and Statistics*. Mineola, NY, USA: Dover Publications, 1997.

[27] A. Cichocki, R. Zdunek, A. H. Phan, and S.-i. Amari, *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multiway Data Analysis and Blind Source Separation*. Hoboken, NJ, USA: Wiley, 2009.

[28] A. Cichocki, R. Zdunek, and S.-i. Amari, "Hierarchical als algorithms for nonnegative matrix and 3d tensor factorization," in *Independent Component Analysis and Signal Separation*. Berlin, Germany: Springer, 2007, pp. 169–176.

[29] T. L. Griffiths and M. Steyvers, "Finding scientific topics," *Proc. Nat. Academy Sci.*, vol. 101, no. suppl 1, pp. 5228–5235, 2004.

[30] C. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*, Cambridge, UK: Cambridge University Press, 2008.

[31] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Hoboken, NJ, USA: Wiley, 2012.

[32] D. Mimno, H. Wallach, E. Talley, M. Leenders, and A. McCallum , "Optimizing semantic coherence in topic models," in *Proc. Conf. Empirical Methods Natural Language Process.*, Jul. 2011, pp. 262–272.

[33] L. Hong and B. D. Davison, "Empirical study of topic modeling in Twitter," in *Proc. 1st Workshop Social Media Analytics*, 2010, pp. 80–88. [Online]. Available: http://doi.acm.org/10.1145/1964858.1964870

[34] C. Lv, R. Qiang, F. Fan, and J. Yang, "Knowledge-Based Query Expansion in Real-Time Microblog Search," *Proc. Inf. Retrieval Technol.: 11th Asia Inf. Retrieval Societies Conf.*, 2015, pp. 43–55.

[35] G. Cao, J.-Y. Nie, J. Gao, and S. Robertson, "Selecting good expansion terms for pseudo-relevance feedback," in *Proc. 31st Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2008, pp. 243–250.

[36] N. F. N. Rajani, K. McArdle, and J. Baldridge, "Extracting topics based on authors, recipients and content in microblogs," in *Proc. 37th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, Jul. 2014, pp. 1171–1174.

[37] M. Steyvers, P. Smyth, M. Rosen-Zvi, and T. Griffiths, "Probabilistic author-topic models for information discovery," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2004, pp. 306–315.

[38] A. McCallum, A. Corrada-Emmanuel, and X. Wang, "Topic and role discovery in social networks," in *Proc. 19th int. Joint Conf. Artif. Intell.*, Edinburgh, Scotland, Jul.-Aug. 2005, pp. 786–791.

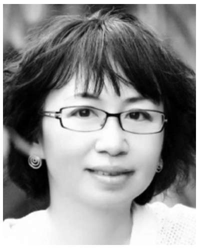
[39] R. Pochampally and V. Varma, "User context as a source of topic retrieval in Twitter," in *Proc. Workshop Enriching Inf. Retrieval*, Jul. 2011, pp. 1–3.

[40] J. Weng, E.-P. Lim, J. Jiang, and Q. He, "Twitterrank: Finding topic-sensitive influential twitterers," in *Proc. 3rd ACM Int. Conf. Web Search Data Mining*, 2010, pp. 261–270. [Online]. Available: http://doi.acm.org/10.1145/1718487.1718520

[41] B. Kang, J. O'Donovan, and T. Höllerer, "Modeling topic specific credibility on Twitter," in *Proc. ACM Int. Conf. Intell. User Interfaces*, Feb. 2012, pp. 179–188.

[42] R. Nugroho, W. Zhao, J. Yang, C. Paris, and S. Nepal, "Using time-sensitive interactions to improve topic derivation in Twitter," *World Wide Web*, pp. 1–27, 2016. [Online]. Available: http://dx.doi.org/10.1007/s11280–016-0417-x

Robertus Nugroho received the master's degree in computing and information technology from the University of New South Wales, Australia, in 2009. He working toward the PhD degree in the Department of Computing, Macquarie University, Australia. He was awarded a postgraduate studentship position at CSIRO Australia (2014-2017). In 2015 he received the best student paper award in IEEE BigData Congress 2015 and the best paper award at Web Information System Engineering (WISE) 2015 Conference. His current research interests include bigdata, social network analysis, and machine learning.



Jian Yang received the PhD degree in multidatabase systems area from Australian National University, in 1995. She is a professor in the Department of Computing, Macquarie University. Prior to joining Macquarie University, she was an associate professor with Tilburg University, The Netherlands (2000-2003), a senior research scientist in the Division of Mathematical and Information Science, CSIRO, Australia (1998-2000), and as a lecturer (assistant professor) in the Department of Computer Science, Australian Defence Force Academy, University of New South Wales (1993-1998). Her main research interests include Web service technology, business process management, interoperability, trust and security issues in digital libraries and e-commerce, social network.

Weiliang Zhao received the PhD degree from the School of Mathematics and Computing, University of Western Sydney, in 2009. He is a research fellow in the Department of Computing, Macquarie University. Before, he rejoined Macquarie University, he worked as a research fellow at University of Wollongong, data analyst at the Copyright Agency, software developer at ROAMZ, research fellow at Macquarie University, programmer at ANZ bank, and researcher at Chinese Academy of Science. His main research interests include social networks, service computing, trust management in distributed systems, and security in electronic commerce applications.

Cecile Paris received the bachelor's degree from the University of Berkeley (California) and the PhD degree from Columbia University (New York). She is the science leader for the Knowledge Discover and Management Research Group at Data61, CSIRO. Her expertise is in natural language processing, user modeling, social media analytics, social computing and, more generally, in artificial intelligence and communication. She is interested in understanding how people communicate, in facilitating communication with information and information environments and in making sense of big data. She has more than to 25 years of experience in research and research management, in CSIRO and other research laboratories overseas. Her group develops systems that are being used in government and in industry, in a wide variety of domains, including service delivery, digital libraries, e-research, mental health, business intelligence, media monitoring, customer-relationship, and service delivery. She is a fellow of the Australian Academy of Technology and Engineering (ATSE). She has authored more than 280 refereed technical papers. She is very active in the research community, in Australia and internationally, serving on numerous conference and workshop committees, on review boards of grant-giving bodies and journals.

Surya Nepal received the BE degree from the National Institute of Technology, Surat, India, the ME degree from the Asian Institute of Technology, Bangkok, Thailand, and the PhD degree from RMIT University, Australia. He is a principal research scientist at CSIRO Data61. His main research interest include the development and implementation of technologies in the area of distributed systems and social networks, with a specific focus on security, privacy, and trust. At CSIRO, he undertook research in the area of multimedia databases, web services and service oriented architectures, social networks, security, privacy and trust in collaborative environment and cloud systems and big data. He has more than 150 publications to his credit. Many of his works are published in top international journals and conferences such as VLDB, ICDE, ICWS, SCC, CoopIS, ICSOC, the *International Journals of Web Services Research*, the *IEEE Transactions on Service Computing*, *the ACM Computing Survey*, and the *ACM Transaction on Internet Technology*.