JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 35, 675-696 (2019)
DOI: 10.6688/JISE.201905_35(3).0011

Corpus-based Topic Derivation and Timestamp-based Popular Hashtag Prediction in Twitter

SHARATH KUMAR B R, KUOCHEN WANG AND SHI-MIN SHEN
Department of Computer Science
National Chiao Tung University
Hsinchu, 300 Taiwan
E-mail: kwang@cs.nctu.edu.tw

With the use of the Internet, mobile platforms, online commerce, and social media services, the footprints of human behavior can be easily recorded in the digital world, which generates data on an extremely large scale. Twitter as a big data social network becomes one of the most important sources for capturing up-to-date events happened in the world. Deriving topics from Twitter is important for various applications, such as situation awareness, market analysis, content filtering, and recommendations. However, topic derivation with high purity in Twitter is hard to achieve because tweets are limited to 140 characters. Previous works on topic derivation in Twitter suffer from low purity. In this paper, we propose corpus-based topic derivation (CTD) approach that combines a Twitter corpus and LF-LDA, which is a text processing model to identify topics and clusters of similar hashtags. We use asymmetric topic LF-LDA to obtain better purity of topics. Compared to intJNMF, a representative related work, the purity (F-measure) of our proposed CTD increases from 5.26% (27.81%) to 11.32% (34.28%) for 20 to 100 topics. We also propose a timestamp-based popular hashtags prediction (TPHP) approach by creating trending hashtags lists (THLs), which are lists of hashtags used by many users and make use of timestamps in tweets. We use the edit distance to find the difference between consecutive THLs. Then the difference can be used to calculate volatilety to find how people react to real world events. Compared to Hybrid+, a representative related work, the mean average precision (MAP) of our TPHP increases by 19.45% (week-day), 15.08% (week-week) and 16.95% (month-week).

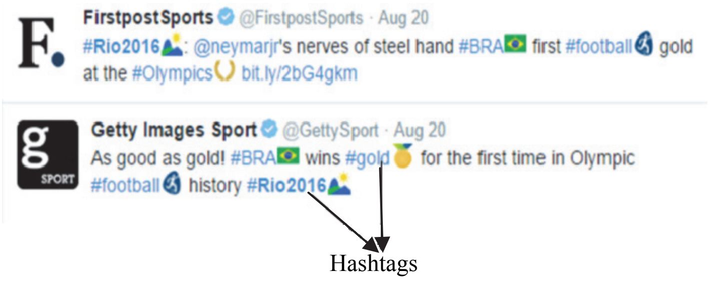
Keywords: corpus, popular hashtag prediction, timestamp, topic derivation, twitter

1. INTRODUCTION

With the use of the Internet, mobile platforms, online commerce, and social media services, the tendency of human behaviors can be easily recorded in the digital world. Through social media, they generate large amounts of data. In Twitter on average 500 million tweets are produced each day [19]. Detecting topics in Twitter streams has been attracting a lot of attention recently. Topic detection is very useful in assisting companies and political parties in understanding user's opinions and needs. An example of tweets and hashtags is shown in Fig. 1.

Twitter as a big data social network becomes one of the most important sources for capturing up-to-date events happened in the world. Due to short (140 characters) nature of tweets it is difficult to derive topics with good purity. Purity is used to evaluate the quality of a derived topic cluster, which means how many of the words in a topic cluster belong to that topic [16]. There are related work on topic derivation [2, 4] in Twitter and

Received August 31, 2017; revised April 23 & December 11, 2018; accepted December 28, 2018.
Communicated by Hung-Yu Kao.



Firstpost Sports @FirstpostSports · Aug 20
#Rio2016: @neymarjr's nerves of steel hand #BRA first #football gold at the #Olympics bit.ly/2bG4gkm

Getty Images Sport @GettySport · Aug 20
As good as gold! #BRA wins #gold for the first time in Olympic #football history #Rio2016

Fig. 1. An example of tweets and hashtags.

popular hashtag prediction [9]. However, there is room for improvements on the purity of topic derivation and the mean average precision (MAP) of popular hashtag prediction. How many of predicted hashtags are present in the actual list of hashtags gives the MAP of predictions.

Therefore, we propose a corpus-based topic derivation (CTD) approach to improve the purity of topic derivation. We use semantic hashtags for better topic derivation. Semantic hashtags mean hashtags that belongs to some categories, such as technologies, sports, politics, *etc*. Semantic hashtag classification [24] is a method of understanding the way people think. It means finding out a person's hashtag preference is important for learning the personality and general preference of the person. For example, if a user uses hashtags #running, #health, #fitness, #diet in the user's tweets, then we can say that the user has general preference towards health. The proposed CTD identifies topics and clusters of similar hashtags by integrating a Twitter GloVe corpus with asymmetric topic LF-LDA (Latent Feature-Latent Dirichlet Allocation) [17]. We use an asymmetric topic model to make topics different from one another and to reduce repeated words between topics and there is more similarity between words within a topic. This increases the purity of the resulting topics.

In addition, we also propose a timestamp-based popular hashtag prediction (TPHP) approach to predict popular hashtags and to improve the MAP. The proposed TPHP creates trending hashtags lists (THLs) for every 60 minutes. We calculate the edit distance between two consecutive THLs and then use it to calculate volatility to find how people react to real world events. Volatility is a tendency to change quickly and unpredictably, especially for the worse [23]. Here volatility shows how often trending topics change over time. The lowest volatility (minimum) implies the focused attention of users (people). We take note of when the volatility drops to the minimum and what corresponding event occurred on that day. A significant drop in volatility indicates focused attention of the users. The proposed TPHP achieves better MAP for popular hashtag prediction.

2. RELATED WORK

We first classify and review existing topic derivation and popular hashtag prediction techniques. Then we compare our proposed CTD with these topic derivation techniques qualitatively. We also qualitatively compare our proposed TPHP with these popular hashtag prediction methods.

Topic derivation uses a statistical model for discovering the topics that occur in a collection of documents. Some text mining techniques are frequently used for discovery of hidden semantic structures in a text body. Given that a document is about a particular topic, one would expect particular words to appear in the document more or less frequently. A document includes multiple topics in different proportions. The topics produced by topic derivation techniques are clusters of similar words. Generally, topic derivation techniques can be classified as unsupervised, supervised and semi-supervised learning. In unsupervised learning, labels are not given and we try to extract information in general out of data we have given [2, 4, 17]. This will work well when we have large documents that contain many words. However, in case of small documents or texts of short length, the purity of resulting topics is low. Supervised learning includes labelled data [9]. The precision of supervised learning depends on the quality of labelled data. However, the cost of labelling data is high. Semi-supervised learning includes both labeled and unlabeled data, usually a small amount of labeled data and a large amount of unlabeled data. Use of labeled and unlabeled data can produce improvement in purity. Fig. 2 shows classification of topic derivation and popular hashtag prediction techniques.

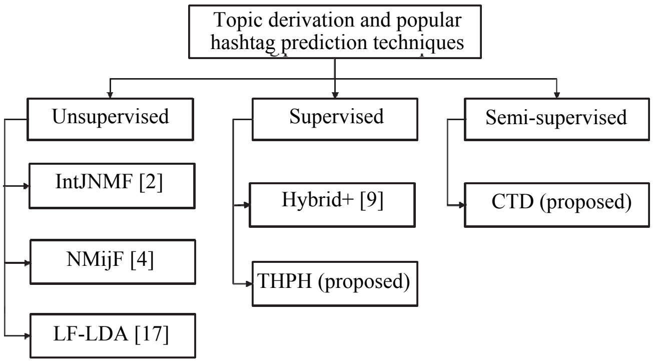


Fig. 2. Classification of topic derivation and popular hashtag prediction techniques.

Topic derivation is used to find topics in documents. LDA [18], a popular algorithm, is used for text summarization of large documents. Latent Dirichlet Allocation (LDA) is a generative statistical model that allows sets of observations to be explained by unobserved groups that explain why some parts of the data are similar [17]. LDA is a three-level hierarchical Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of topics [17]. Each topic, in turn, is modeled as an infinite mixture over an underlying set of topic probabilities [17]. This algorithm takes the advantage of a large number of words in large documents. However, it is not feasible to find topics in small documents. LF-LDA [17] is a text processing model which performs well in small documents. LF-LDA is formed by taking the original Dirichlet multinomial topic models LDA and replacing their topic-to-word Dirichlet multinomial component that generates words from topics with a two-component mixture of a topic-toword Dirichlet multinomial component and a latent feature component [17]. As shown in Fig. 4, asymmetric LF-LDA is different from LF-LDA since in asymmetric LF-LDA we make use of asymmetric topic property. This means that words in a topic cluster are more similar to each other and topic clusters are different to each other. For example, if we have two clusters of music and movies, hashtags in each cluster are more similar to each other and topic clusters are more different in the case of asymmetric LF-LDA. For the same example, in LF-LDA, topics in these two clusters are not well separated to each other. That is, more hashtags are common between these two clusters. In intJNMF [2], topics are derived by performing a two-step matrix factorization over semantic features of tweets and interaction between tweets. This method did not consider the time factor and self-contained tweets (tweets without interaction). Note that in Twitter, majority of tweets are self-contained tweets. In NMijF [4], authors conduct co-factorization jointly over Twitter interaction features and content attributes within a single iterative-update process. The tweet-relationship matrix combines interaction features and content attributes of tweets and can provide more information regarding the clustering characteristics of tweets. But the timestamps of tweets were not considered, which results in low purity of topics. The comparison of the proposed CTD with related work is summarized in Table 1.

Hybrid+ [9] conducted popular hashtag prediction and it can tell whether a predicted hashtag has been adopted by users. Popular hashtags are predicted using explicit features and latent factor models together. But its MAP of predicted hashtags is low. The comparison of the proposed TPHP with Hybrid+ [9] is summarized in Table 2.

Table 1. Comparison of proposed CTD with related work.

Approach	intJNMF [2]	NMijF [4]	LF-LDA [17]	CTD (proposed)
Hashtag type	Hashtags not used	Hashtags not used	N/A	Semantic hashtags
Algorithm	intJNMF	NMijF	LF-LDA	Asymmetric topic LF-LDA
Use a corpus (vector representation of words)	No	No	No	Yes
Pros	Incorporation of interactions among tweets helps relieve the sparsity problem	Includes content attributes of tweets to deal with the sparsity problem	Can derive topics from small documents	Because of using an asymmetric topic model, topics are well separated to one another
Cons	Does not include self-contained tweets	Takes more time in cases with large amounts of data	Purity of resulting topics is low	New hashtags are not included in the corpus

Table 2. Comparison of proposed TPHP with related work.

Approach	Hybrid+ [9]	TPHP (proposed)
Hashtag type	Single hashtags	Semantic hashtags
Model	Latent variable model	Timestamp based THL model
Pros	Content-based hashtag recommendation	Considers hashtag trends for every 60 minutes
Cons	The MAP of predicted hashtags is low	Cannot recommend hashtags based on user location

3. DESIGN APPROACHES

We propose corpus-based topic derivation (CTD) and timestamp-based popular hashtag prediction (TPHP) approaches for Twitter. The design architecture is first described in section 3.1. Then the two approaches are described in sections 3.2 and 3.3.

3.1 Design Architecture

Twitter is an online social networking service that enables users to send and read short messages (140 characters), called "tweets." Registered users can read and post tweets. In the proposed CTD and TPHP, we use tweets from Twitter. Fig. 3 shows the design architecture of the proposed CTD (the left portion of Fig. 3) and the proposed TPHP (the right portion of Fig. 3). In the proposed CTD, we use a Twitter corpus with asymmetric topic LF-LDA to derive topics associated with hashtags. A corpus is vector representations of words, which is used for topic derivation. A tweet is a message sent using Twitter. Hashtags are words that appear in tweets. GloVe corpus is the pre-trained word vectors of tweets. It is an unsupervised learning algorithm for obtaining vector representations for words [14]. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations show interesting linear substructures of the word vector space [14]. Please note that if we do not have hashtags in pre-trained GloVe vectors, the data we test would have out-of-vocabulary (OOV) hashtags [25]. Such OOV hashtags are set to an "UNKNOWN" word and are assigned to the same vector. This works well because we usually have a small percentage of OOV hashtags. The asymmetric topic LF-LDA is asymmetric over topics and symmetric over words. By using asymmetric topic LF-LDA with a Twitter corpus, we can get topics and semantic hashtags related to these topics.

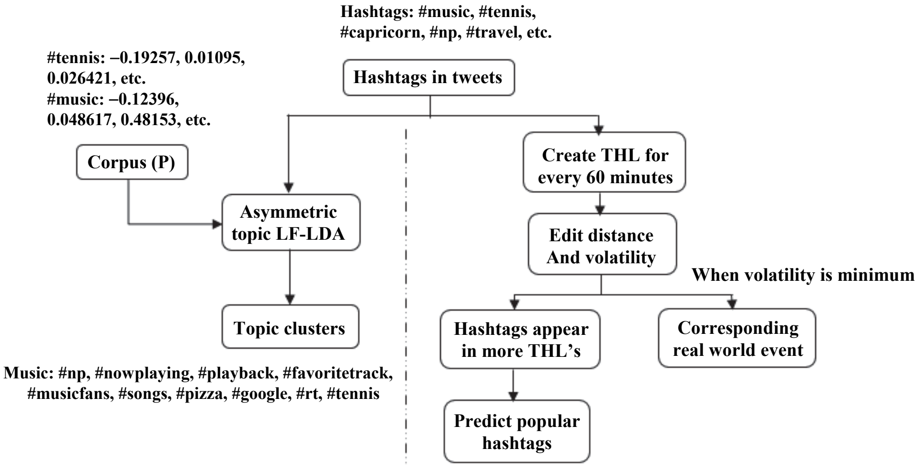

Hashtags: #music, #tennis, #capricorn, #np, #travel, etc.
#tennis: −0.19257, 0.01095, 0.026421, etc.
#music: −0.12396, 0.048617, 0.48153, etc.
Hashtags in tweets
Corpus (P)
Create THL for every 60 minutes
Asymmetric topic LF-LDA
Edit distance And volatility
When volatility is minimum
Topic clusters
Hashtags appear in more THL's
Corresponding real world event
Music: #np, #nowplaying, #playback, #favoritetrack, #musicfans, #songs, #pizza, #google, #rt, #tennis
Predict popular hashtags

Fig. 3. Design architecture of the proposed CTD (the left portion) and the proposed TPHP (the right portion).

In the proposed TPHP, we create trending hashtags lists (THLs) using timestamps. We create THL for every 60 minutes. It helps us to track the trends in Twitter because we get 24 THLs each day and we collect hashtags that appear in more THLs on that day. If a hashtag is a top trending hashtag and it appears in all 24 THLs, then we select that hashtag for recommendation. This is repeated for all days of training data and we collect hashtags that appear in more THLs on training data and make prediction based on this collection of hashtags. In addition, we calculate the edit distance between two consecutive THLs and then use it to calculate volatility to find how people react to real world events.

3.2 Proposed CTD

Twitter is used by millions of people and each user may use different hashtags in his/her tweets which are related to the same topic. To reduce the overlapping between topics, we shift attention from single hashtags to more general categories: clusters of semantically similar hashtags. For example, hashtags like #np, #nowplaying, and #musicfans are semantic hashtags related to music. Because of short (140 characters) nature of tweets, topic derivation needs both labeled and unlabeled data to derive topics with better purity. Therefore, we make use of labeled and unlabeled data in the proposed CTD to improve the purity of topics.

Conventional topic modeling algorithms, such as LDA, infer document-to-topic and topic-to-word distributions from the co-occurrence of words within documents. But when the training corpus of documents is small or when the documents are short like tweets, the resulting distributions might be based on little evidence. Topic models have also been constructed using latent features (LFs) [21, 22]. LF vectors have been used for a wide range of NLP (natural language processing) tasks. The combination of values permitted by LFs forms a high dimensional space that makes LFs be well suited to model topics. Rather than relying only on a multinomial or LF model, LF-LDA explores how to take advantage of both LFs and multinomial models by using an LF representation trained on a large external corpus to supplement a multinomial topic model estimated from a smaller corpus. In Twitter, tweets are limited to only 140 characters; hence, LF-LDA can be used for topic derivation for texts of short length.

We use a Twitter corpus with asymmetric topic LF-LDA to find the clusters of semantic hashtags and topics associated with these hashtags. A corpus is a vector representation of a large collection of words. By using the GloVe corpus with asymmetric topic LF-LDA, we get topics and semantic hashtags related to these topics. Table 3 shows topics and examples of semantic hashtags related to these topics.

Table 3. Topics and examples of semantic hashtags.

Topic	Example of semantic hashtags
Technology	#google, #microsoft, #supercomputers, #ibm, #wikipedia, #pinterest, #startuptip, #topworkplace
Politics	#tcot, #p2, #top, #usgovernment, #dems, #owe, #politics, #teaparty
Lifestyle	#pizza, #pepsi, #cheese, #health, #vacation, #caribbean, #ford, #honda, #volkswagen, #hm, #timberland
Twitter-specific	#followme, #followback, #teamfollowback, #followfriday, #friday, #justretweet, #instantfollower, #rt
Mobile devices	#apple, #galaxy4, #note3, #iosapp, #androidgames, #releases. #ipadgames

As shown in left-hand side of Fig. 3, we feed hashtags in tweets and GloVe corpus into the asymmetric LF-LDA model to get clusters of hashtags related to topics. Both the corpus and tweets are used at the same time in asymmetric LF-LDA. That is, asymmetric LF-LDA uses the vector representation of words in the corpus and hashtags in tweets to generate a cluster of hashtags that are related to one topic. In GloVe, for example, a hashtag #tennis is represented in 25d (d: dimensions) as #tennis –0.19257, 0.01095, 0.026421, –1.3744, –0.41477, 0.57898, 0.068704, –0.15461, 0.08917, –1.0976, –0.56159, –0.31318, 1.1077, –1.5308, 0.54875, 0.53085, –1.0322, 0.10249, 0.55959, –0.13588, –2.9936, 0.18661, 1.0769, 0.61615, –0.22304. Similarly, all hashtags in the corpus have their vector representations. For example, when we feed corpus #tennis –0.19257, 0.01095, 0.02642, …, #music –0.12396, –0.048617, 0.48153, … and hashtags #music, #tennis, #capricorn, #np, #travel, … in a tweet into asymmetric LF-LDA, we get a topic cluster, Music, which contains hashtags #np, #nowplaying, #playback, #favoritetrack, #musicfans, #songs, #pizza, #google, #rt, #tennis.

```
Initialize the world-topic variables $z_{d_i}$ using the LDA
sampling algorithm
for iteration iter = 1, 2, … do
    for topic $t = 1, 2, …, T$ do
        $\tau_t = \arg\max_{\tau_t} P(\tau_t | \boldsymbol{Z}, \boldsymbol{S})$
for document d = 1, 2, …, |D| do
    for world index $i = 1, 2, …, w_{d_i}, …, N_d$ do
        sample $z_{d_i}$ from
        $P(z_{d_i} = t, z_{d_i} \mid \boldsymbol{Z}_{\neg d_i}, \boldsymbol{S}_{\neg d_i}, \boldsymbol{\tau}, \boldsymbol{\omega})$
```

Fig. 4. Asymmetric topic LF-LDA algorithm [17].

Fig. 4 shows the algorithm of asymmetric topic LF-LDA. The algorithm generates a topic, details as follows. For document *d*, for each *i*th word w_{d_i}, the model chooses a topic indicator z_{d_i} from corpus P. Here we use z_{d_i} instead of s_{d_i} in LF-LDA [17]. Note that s_{d_i} is a binary indicator of *i*th word w_{d_i} to indicates whether a word is generated by latent feature or Dirichlet multinomial. Since we use latent feature only, we use z_{d_i}. By using z_{d_i}, which is a topic indicator, generated topics can be well separated to one another. Thus, a topic can be generated from the chosen words by the word-to-topic model.

It is significant to find the diversity of user interests. The diversity of user interests means to find whether a user has a focused interest (interested in only one topic) or diverse interests (interested in many topics). Given a user, we track the hashtags that the user adopted within certain hours after a hashtag is created and then we can find whether the user has a focused interest or diverse interests. We consider attributes like number of early adopters who adopted the hashtag, timestamp, and number of tweets that a user has produced. It is important to distinguish users with focused or diverse interests. This is because once a recommendation is made, the adoption probability is higher for users with focused interests than for users with diverse interest.

We can find popular or viral hashtags on particular days based on the time at which the hashtags are created. Popularity of a hashtag can be found, as follows:

- A hashtag is selected based on the number of early adopters of that hashtag within *t* hours after the birth of that hashtag. We may track adoption events for *t* = 4, 12, and 24 hours since the birth of hashtag.
- Hashtags are ranked by number of users who used that hashtag and are sorted in descending order.
- The most popular hashtags are said to be viral because they are adopted by more users.

3.3 Proposed TPHP

A word, phrase or topic that is mentioned at a greater rate than others is said to be a trending topic in Twitter [20]. These topics help Twitter and their users to understand what events are happening in the world and what people's opinions are about these events. A trending hashtags list (THL) is a list of top trending hashtags that are used by many users and are sorted in descending order based on total number of times these hashtags are used. Using timestamps, we create THLs between time intervals of 60 minutes on each day. We get 24 THLs on each day and we consider each THL as a state in automata. Here we take a THL as a state in automata and each state is a list of top 10 trending hashtags. The automata transits from one state to another state as time progresses and results in changes in the THLs. Each transition is the result of an action. The action is a set of new trends that are trying to break into a THL. When a hashtag is used by many users at the next time interval, then that hashtag will be inserted to a place in a THL based on its times of occurrence. This results in transition of THLs from one state to another state. Fig. 5 shows the state transition of THLs and the edit distance next to each edge between two consecutive THLs, which will be defined later.

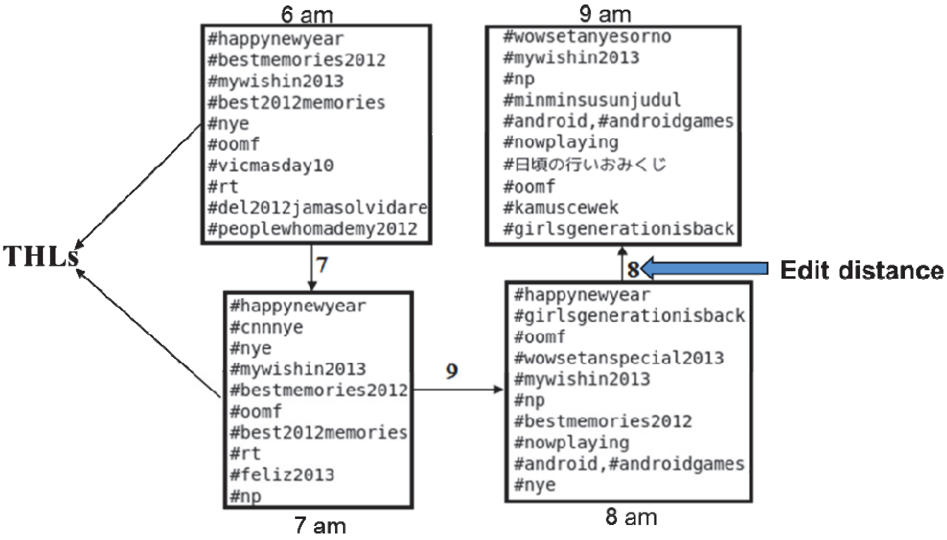


Fig. 5. State transition of THLs and edit distance next to each edge between two consecutive THLs.

For popular hashtag prediction we create THLs each day and we get top trending hashtags in every 60 minutes. The hashtags are sorted in descending order based on the total number of users using these hashtags and we take top hashtags in each THL. We collect the hashtags that appear in more THLs on that day. For example, if a hashtag is a top trending hashtag and it appears in all 24 THLs, then we select that hashtag for recommendation. This is repeated for all days of training data and we collect hashtags that appear in more THLs on training data and make prediction based on this collection of hashtags.

Trending hashtags help us find what events happened in the world and how people reacted to these events. THLs can be used to find real world events. We use edit distance [6] to find the difference between THLs. Edit distance is a way of quantifying how dissimilar two strings (*e.g.*, words) are to each other by counting the minimum number of operations required to transform one string into the other [6]. In our case, we take each hashtag as a letter in a string. For example, in Fig. 5, if we take THL at 6 am, there are 10 hashtags. Here we take each hashtag as a letter (*e.g.*, #happynewyear as 'a', #bestmemories2012 as 'b', *etc.*) in a string. Hence, in Fig. 5, each THL is considered as a string of length 10. Now we find the minimum number of operations required to transform one THL into the other THL, which is the edit distance of these two THLs. Hence, we can compute the edit distance between two consecutive THLs. In Fig. 5, the edit distance is shown next to each edge.

The edit distance between two consecutive THLs can be computed using Eq. (1) [6]. Here a and b are THLs at time t and $t + 1$, respectively. i and j are the lengths of THLs a and b, respectively. In this case, i and j are both 10 because the length of a THL is 10. Fig. 5 shows THLs and the edit distance between them. In the figure, we can see that the edit distance between 6:00 am and 7:00 am is 7. Similarly, the edit distance between 7:00 am and 8:00 am is 9 and that between 8:00 am and 9:00 am is 8. The pseudo code for computing the edit distance is given in Fig. 6.

$$\text{Edit}_{(a,b)}(i,j)\begin{cases}0, & i=j=0\\ i, & j=0 \text{ and } i>0\\ j, & i=0 \text{ and} j>0\\ \min\begin{cases}\text{Edit}(a,b) & (i-1,j)+1, \text{ otherwise}\\ \text{Edit}(a,b) & (i,j-1)+1\\ \text{Edit}(a,b) & (i-1,j-1)+1_{(a_i\neq b_j)}\end{cases}\end{cases} \tag{1}$$

```
Edit distance (a, b)
    int m[i, j] = 0
    for i ← 1 to |a|
    do m[i, 0] = i
    for j ← 1 to |b|
    do m[0, j] = j
    for i ← 1 to |a|
    do for j ← 1 to |b|
            do m[i, j] = min{m[i – 1, j – 1] + if (a[i] = b[j]) then 0 else 1
                             m[i – 1, j] + 1,
                             m[i, j – 1] + 1}
            return m[|a|, |b|]
```

Fig. 6. Pseudo code for computing the edit distance.

We use volatility to find how people react to real world events and it is calculated using the edit distance between THLs. Volatility is summation of edit distances between THLs and it is calculated using Eq. (2). Volatility becomes low when there are not much difference between THLs. This indicates that many people are talking about some particular incident or event. When we find low volatility, we know that some real world events or incidents have happened.

$$Volatility = \sum_{t=1}^{T-1} Edit_{(a,b)}(X_{t+1}, X_t) \tag{2}$$

where

X_t: THL at time t and X_{t+1}: THL at time $t+1$
$T = (24 * 60)$ / time interval (in minutes)

Time interval between THLs is 60 minutes in this case; therefore

$T = (24 * 60) / 60 = 24$

4. EVALUTION

We first describe our experiment setup. Then we compare the proposed CTD and the proposed TPHP approaches with representative related work quantitatively. Finally, we discuss experiment results and potential applications of our results.

4.1 Experiment Setup

All the experiments were conducted on a system with Intel Core i7-3537U 2.00 GHz CPU and 8 GB memory. Tables 4 and 5 show the experiment setups for the proposed CTD and the proposed TPHP, respectively. We used publicly available Twitter data [13] that were crawled from Twitter between Jan-01-2013 and Feb-07-2013. It contains around 200 million tweets. We used an IBM Infosphere Streams [3] tool for finding popular hashtags and for hashtag deduplication. For topic derivation, we used the GloVe [14] corpus. We created THLs using IBM Infosphere Streams [3] for every 60 minutes each day. Therefore, we get 24 THLs each day.

Table 4. Experiment setup for the proposed CTD.

Dataset used	Publicly available Twitter data [13], containing 200 million tweets
Tool used	Using IBM Infosphere Streams [3] for finding popular hashtags and for hashtag deduplication
Corpus used	GloVe [14]

Table 5. Experiment setup for the proposed TPHP.

Dataset used	Publicly available Twitter data [13], containing 200 million tweets
Tool used	Using IBM Infosphere Streams [3] for creating THLs
Time interval between THLs	60 minutes

4.2 Experiment Results

To compare the proposed CTD with related work, we evaluate two parameters: purity [16] and F-measure [16]. To evaluate the proposed TPHP with related work we evaluate the mean average precision (MAP) [9].

First, we evaluate the purity. Purity [16] is used to find the quality of a derived topic cluster, which means how many of the derived hashtags in the derived topic actually belong to that topic cluster. A low-quality cluster of topics has purity 0 and a perfect cluster of topics have purity 1. It is calculated using Eq. (3).

$$Purity(K,C)=\tfrac{1}{N}\sum\nolimits_i \max\nolimits_j |k_i \cap c_j| \tag{3}$$

where $K = \{k_1, k_2, ..., k_i\}$ is the set of clusters, $C = \{c_1, c_2, \ldots, c_j\}$ is the set of classes, N is the total number of elements, k_i is the set of documents in k_i, and c_j is the set of documents in c_j in Eq. (3) [16].

For illustration, Table 6 shows example topics, hashtags and the number of related hashtags. Note that in Table 6 we can see that some hashtags are related to the associated topic and some do not. A topic is assigned based on a maximum number of hashtags that belong to the same cluster and are related to the topic. In Table 6, we can see that there are three topics and their derived hashtags. We can see the total number of hashtags that are related to the topic in the last column. Related hashtags are underlined in the table. For example, the purity of Table 6 is computed as,

$$Purity=\frac{4+3+5}{18}=0.666.$$

Table 6. Topics, hashtags and number of related hashtags.

Topic	Topic cluster	Number of hashtags related to the topic
Music	#nowplaying,#np, #songsthatilike, #musicfans, #aries, #tennis	4
Horoscope	#capricorn, #libra, #leo, #music, #retweet, #fb,	3
Sports	#soccer, #basketball, #baseball, #volleyball, #rugby, #followback	5

In Fig. 7, we compare the proposed CTD with intJNMF [2] and LF-LDA [17] in terms of purity. The purity of the proposed CTD is higher than that of intJNMF and LF-LDA for all cases. Note that we used a corpus with asymmetric topic LF-LDA. By using an asymmetric topic model, we minimize the similarity between topics. This makes topics look different from one another and words are not repeated between topics too much. In addition, because of symmetric words, there is more similarity between words within a topic. This also brings better purity of resulting topics. As the number of topics increases, purity also increases. Here the highest purity of topics for the proposed CTD is only around 0.64. This is because a social medium like Twitter includes users from many domains and they can use their own hashtags in their tweets. However, the resulted purity of topics is still useful because we are able to differentiate the topics from one another. Note we compare CTD with intJNMF because intJNMF's purity has been shown better compared to all other related work [2].

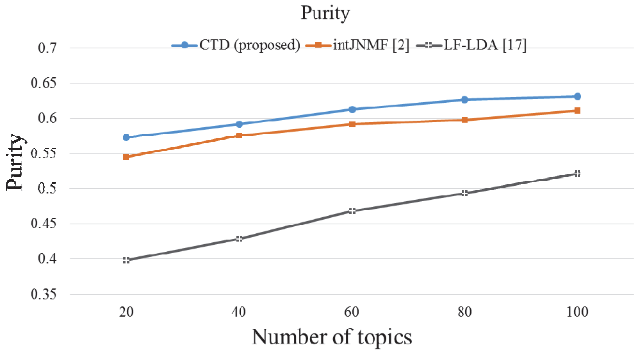


Fig. 7. Purity of topics.

Next, we evaluate the proposed CTD with intJNMF and LF-LDA in terms of F-measure. F-measure computes the harmonic mean of both precision p and recall r. Precision (p) is the number of hashtags correctly put in the same cluster and recall (r) is the actual number of hashtags that were identified correctly. Eq. (4) is used to calculate F-measure [16]:

$$F = 2\frac{p * r}{p + r}. \tag{4}$$

In a topic cluster, we will get hashtags. Among those hashtags, some are related to the topic and some are not related to the topic. By using these hashtags, we can calculate precision and recall. Here precision is the fraction of retrieved hashtags that are relevant to the topic and recall is the fraction of the relevant hashtags that are successfully retrieved. Then, F-measure can be calculated using precision and recall. Since the hashtags in the corpus are labeled, we can get a topic name for the topic cluster. For example, if we have a topic cluster "sports" with 20 hashtags, and we only retrieved 16 hashtags #soccer, #basketball, #baseball, #volleyball, #rugby, #bowling, #hockey, #polo, #badminton, #retweet, #songs, #np, #nye, #google, #followme and #leo. Among the 16 hashtags, only 9 hashtags (underlined) are related to sports. Then we calculate precision and recall as follows:

$$precision = \frac{9}{16} \approx 0.56 \qquad recall = \frac{9}{20} \approx 0.45.$$

Then, F-measure $= 2 \times \frac{0.56 \times 0.45}{0.56 + 0.45} = 0.49.$

The way acquiring the labeled data is described as follows. We used GloVe corpus.

The hashtags in this corpus is labeled prior to the training of the GloVe model. Regarding statistics of the hashtags, we used publicly available Twitter data that were crawled from Twitter between Jan-01-2013 and Feb-07-2013. It contains hashtags that were used in approximately 200 million tweets. Fig. 8 shows the comparison among the proposed CTD, intJNMF [2] and LF-LDA [17]. Assuming 95% of confidence level, errors (in terms of confidence interval) observed by applying the evaluation schema (*i.e.*, F-Measure) are marked in Fig. 8. The proposed CTD performs better than intJNMF [2] and LF-LDA for all cases in terms of F-measure. After 60 topics, the F-measure appears to be stable as we increase the number of topics.

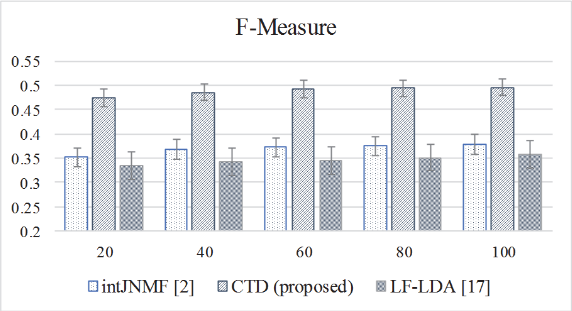


Fig. 8. F-measure of topics.

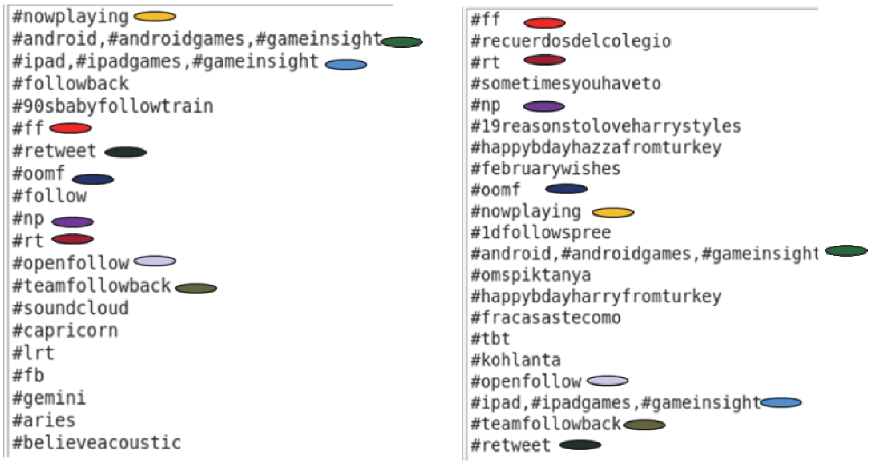
In Table 7, we can see that the proposed CTD took less running time than intJNMF, but took slightly more running time than LF-LDA. This is because the proposed CTD uses a Twitter corpus and LF-LDA does not use any corpus. In the case of intJNMF, it uses interaction between tweets to derive topics; hence, it took more time. Nevertheless, the proposed CTD performs better in terms of purity and F-measure, as shown in Figs. 7 and 8.

Table 7. Running time (in seconds) comparison of the three algorithms versus number of topics.

Number of topics	20	40	60	80	100
CTD (proposed)	93.8	97.6	100.8	101.2	101.5
LF-LDA [17]	88.5	92.4	96.7	97.2	98.1
intJNMF [2]	110.7	115.2	118.9	119.3	119.9

Now we compare the proposed TPHP with related work Hybrid+ [9] in terms of MAP [9]. MAP can be used to measure the performance of prediction. It shows how many predicted popular hashtags are in the actual list of popular hashtags. For illustration, Fig. 9 shows the predicted popular hashtags based on the training data from the last week of January 2013 and the actual popular hashtags on Feb-01-2013.

The prediction result in Fig. 9 shows that in a list of predicted 20 popular hashtags, 10 hashtags are present in the actual list of popular hashtags. This means that the proposed TPHP predicted 50% hashtags correctly. MAP is calculated by taking average precision (AP) of each prediction list. MAP is calculated using Eq. (5):



```
#nowplaying
#android,#androidgames,#gameinsight
#ipad,#ipadgames,#gameinsight
#followback
#90sbabyfollowtrain
#ff
#retweet
#oomf
#follow
#np
#rt
#openfollow
#teamfollowback
#soundcloud
#capricorn
#lrt
#fb
#gemini
#aries
#believeacoustic
```

```
#ff
#recuerdosdelcolegio
#rt
#sometimesyouhaveto
#np
#19reasonstoloveharrystyles
#happybdayhazzafromturkey
#februarywishes
#oomf
#nowplaying
#1dfollowspree
#android,#androidgames,#gameinsight
#omspiktanya
#happybdayharryfromturkey
#fracasastecomo
#tbt
#kohlanta
#openfollow
#ipad,#ipadgames,#gameinsight
#teamfollowback
#retweet
```

Fig. 9. Predicted popular hashtags (left) and actual hashtags (right).

$$MAP = \frac{\sum_{i=1}^{N} AP}{N}. \tag{5}$$

Here AP is average precision in a list and N is the number of recommended lists. AP is calculated using Eq. (6):

$$AP = \frac{Number\ of\ hashtags\ predicted\ correctly}{Total\ number\ of\ recommended\ hashtags}. \tag{6}$$

Note that the AP of the above predicted list in Fig. 9 is 0.5, calculated as follows:

$$AP = \frac{10}{20} = 0.5.$$

For example, suppose that we have 5 prediction lists with APs: 0.5, 0.57, 0.55, 0.42 and 0.45, then MAP can be calculated as

$$MAP = \frac{0.5 + 0.57 + 0.55 + 0.42 + 0.45}{5} = 0.498.$$

In our experiment, we divided our dataset into training and testing data as follows:

- Week-Day – Training data were from the last week of Jan-2013 and testing data were from Feb-01-2013.
- Week-Week – Training data were from the last week of Jan-2013 and testing data were from the first week of Feb-2013.
- Month-Week – Training data were from the full month of Jan-2013 and testing data were from the first week of Feb-2013.

Fig. 10 shows the comparison between the proposed TPHP and Hybrid+ [9] in terms of MAP. As we can see the proposed TPHP performs better than Hybrid+ in all three cases. The Week-Day's MAP is higher because it uses one-week training data to predict one day. The Week-Week's MAP is lower compared to the other two cases because both training and testing data contains one-week data, and it shows that if we use the same amount of training and testing data, then the MAP decreases. The Month-Week's MAP is better than Week-Week's MAP because it uses more training data than the test data. From our results, we conclude that better prediction can be achieved by creating THLs for every 60 minutes using timestamp. Hashtag trends are considered for an entire day in Hybrid+ [9], but in our proposed TPHP we consider trends of hashtags for every 60 minutes, and by this we can achieve better MAP. Here we compared our proposed TPHP with Hybrid+ because Hybrid+ has been shown achieving better MAP compared other related methods [9]. Note that the highest MAP in our TPHP is only 0.48. This is because it is hard to achieve higher MAP because new hashtags are created by users every day.

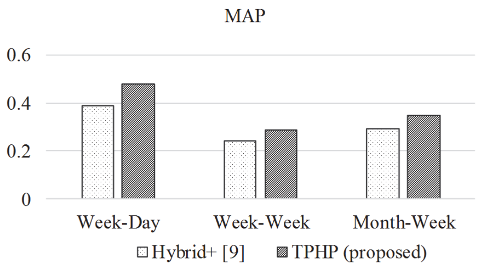


Fig. 10. MAP comparison.

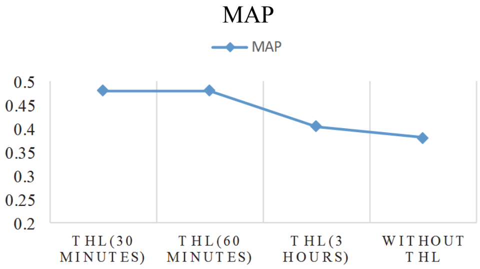


Fig. 11. Comparison of MAP with THL and without THL.

As shown in Fig. 11, a better MAP can be obtained by creating THLs for every 60 minutes; that is, we track the trending hashtags for every 60 minutes. Although the same MAP can be obtained by creating THLs for every 30 minutes, it generates more number of THLs and results in more calculations in computing the edit distance in comparison with the case for every 60 minutes. Therefore, a better MAP can be achieved by creating THLs for every 60 minutes.

In Table 8, we can see that the proposed TPHP took less running time compared to Hybrid+. This is because in the proposed TPHP we create THLs for every 60 minutes and it gives top trending hashtags for every 60 minutes. However, in the case of Hybrid+, it takes an entire day's hashtags and it results in higher running time. The Month-Week and Week-Week cases takes more running time than the Week-Day case as they used more tweet data for training and testing. Remind that the proposed TPHP performs better than Hybrid+ [9] in terms of MAP, as shown in Fig. 10.

Table 8. Running time (in seconds) comparison of the two algorithms for three cases of training and testing data.

Case	Hybrid+ [9]	TPHP (proposed)
Week-Day	119.3	110.7
Week-Week	217.8	190.0
Month-Week	431.6	377.2

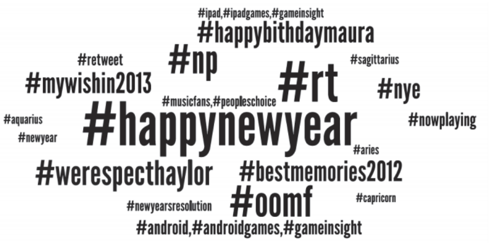


Fig. 12. A word cloud of popular hashtags on Jan-01-2013.

Fig. 12 shows a word cloud of popular hashtags on Jan-01-2013. Bigger the size of a word represents a more popular hashtag on that day. Here we can see that #happynewyear was used by more users. We considered the top trending hashtags on Jan-01-2013 and #happynewyear is the most popular hashtag on that day.

Fig. 13 shows popularity of hashtags over different periods. Here we compared #aries and #happynewyear. We checked popularity of these hashtags for three time intervals. As we can see in the figure, #aries is more popular between 6 am and 10 am even though the time interval is only 4 hours. In the case of #happynewyear, we compared between 6 am – 10 am and 6 pm – 6 am. Here we can see that in both time slots the numbers of hashtags are almost equal. For the case of time interval 6 am – 10 am, although the duration is only 4 hours, there is more number of hashtags used in that time slot.

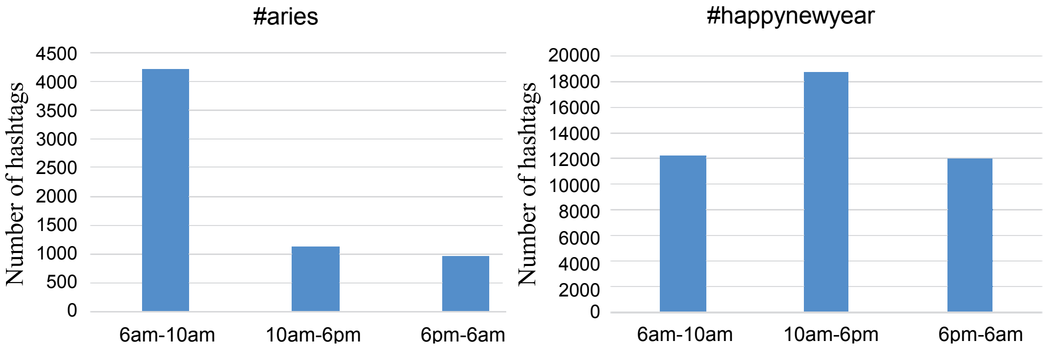


Fig. 13. Popularity of hashtags over different periods.

In Twitter, for the same topic, some hashtags are used by many users and some hashtags are used by few users. By using the GloVe corpus with asymmetric topic LF-LDA, we may get hashtags related to a specific topic, as illustrated in Table 9. Then we rank hashtags according to the number of times they have been used. A hashtag is ranked higher if that hashtag is used by more users. In addition, we classify hashtags according to the number of times they have been used, as follows:

- **Private hashtags** – A private hashtag has long lifetime but low frequency. This class of hashtags is restricted to personal usage, for example, #love, #boring and #mythought. To find private hashtags, we consider lifetime of hashtags and frequency (number of times used) of hashtags.

- **Burst hashtags** – This class of hashtags reflects events happened in the physical world that usually burst in a short time with global discussion in Twitter and then disappear, for example, #20songsthatilike and #superbowl47. This class of hashtags is usually event oriented.
- **General hashtags** – This class of hashtags lies between private hashtags and burst hashtags. This class of hashtags has long lifetime as well as high frequency. This class of hashtags is usually used all days by users, for example, #aries, #capricorn and #goodmorning.

Table 9. Topics and related semantic hashtags.

Topic	Semantic hashtags
Horoscope	#aries, #leo, #sagittarius, #taurus, #virgo, #capricorn, #gemini, #libra, #aquarius, #cancer, #scorpio, #pisces
Music	#np, #nowplaying, #20songsthatilike, #musicfans, #peopleschoice, #music

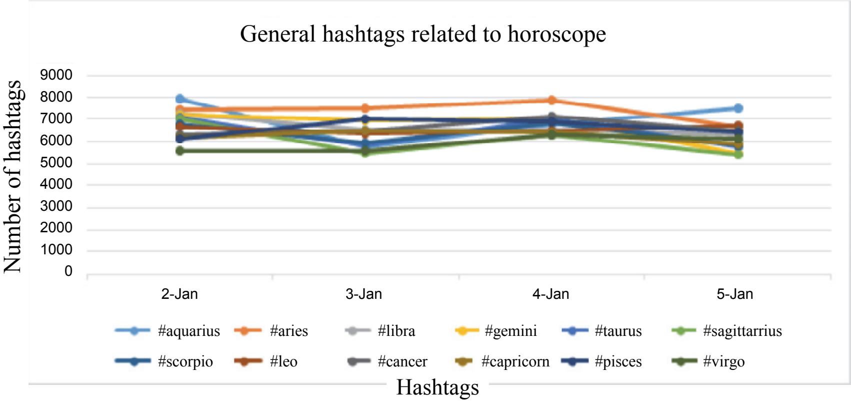


Fig. 14. General hashtags related to horoscope.

Fig. 14 shows general hashtags related to horoscope. The number of hashtags will not increase or decreases considerably in the case of general hashtags and the number of hashtags would remain almost the same during these days. Fig. 15 shows general hashtags and burst hashtags related to music. #20songsthatilike and #musicfans were used more on Jan-03 and Jan-04 of 2013. #20songsthatilike was used around 100 times on Jan-02 and reached around 48000 on Jan-03 and again dropped to below 100 on Jan-05. Therefore, this hashtag is considered as a burst hashtag. #np and #nowplaying remained the same during these days and these hashtags are considered as general hashtags for music.

Fig. 16 (a) shows the edit distance between THLs. We use edit distance to calculate volatility. By calculating volatility, we can find how people react to real world events. As we can see in Fig. 16 (b), volatility is very low on Feb-04-2013. This is because of less difference between THLs on that day and it indicates that people were talking about some particular topic. After examining the hashtags, we found that many top trending hashtags were related to Super Bowl 47. Super Bowl 47 is the 47th championship game of the American professional football that was held on February 04, 2013. More Twitter users were talking about that event; hence, there were not much difference between THLs.

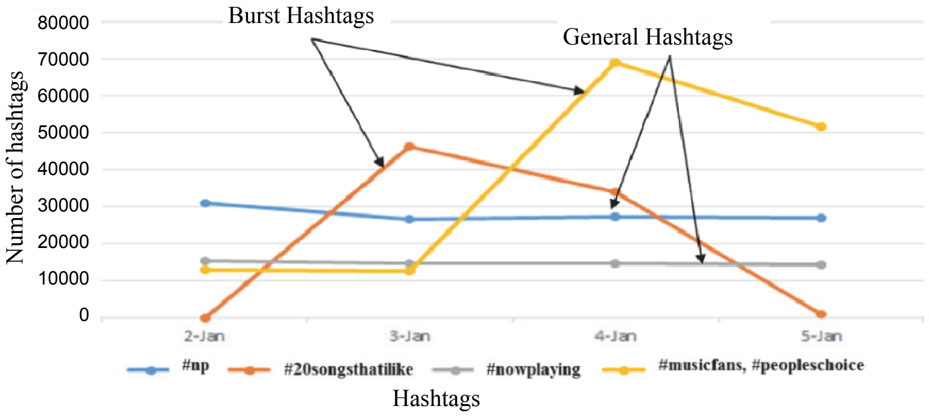
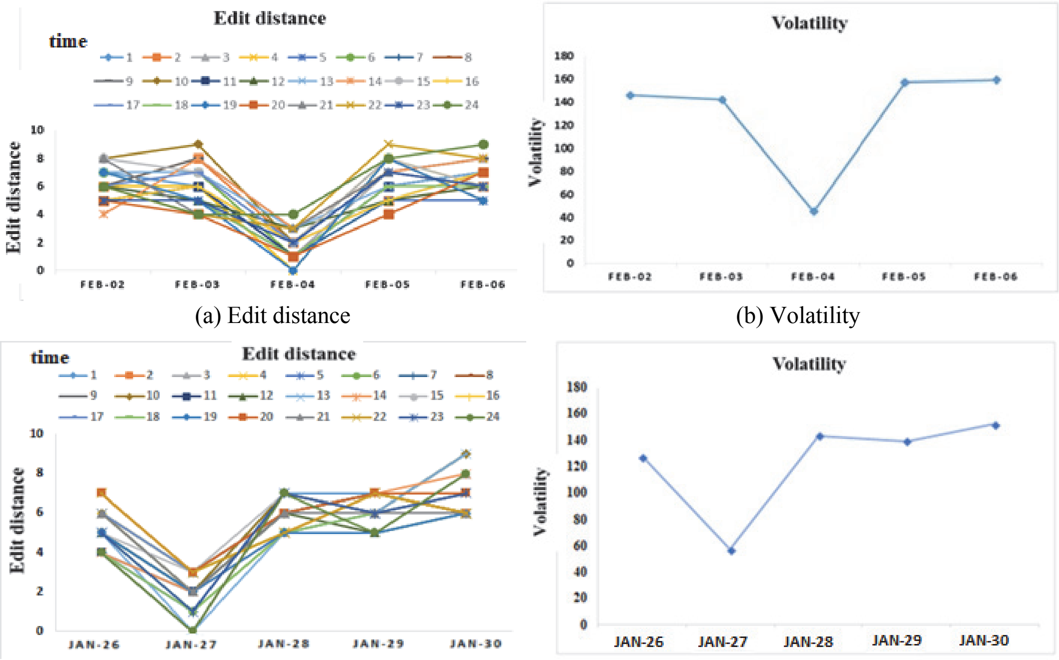


Fig. 15. General hashtags and burst hashtags related to music.



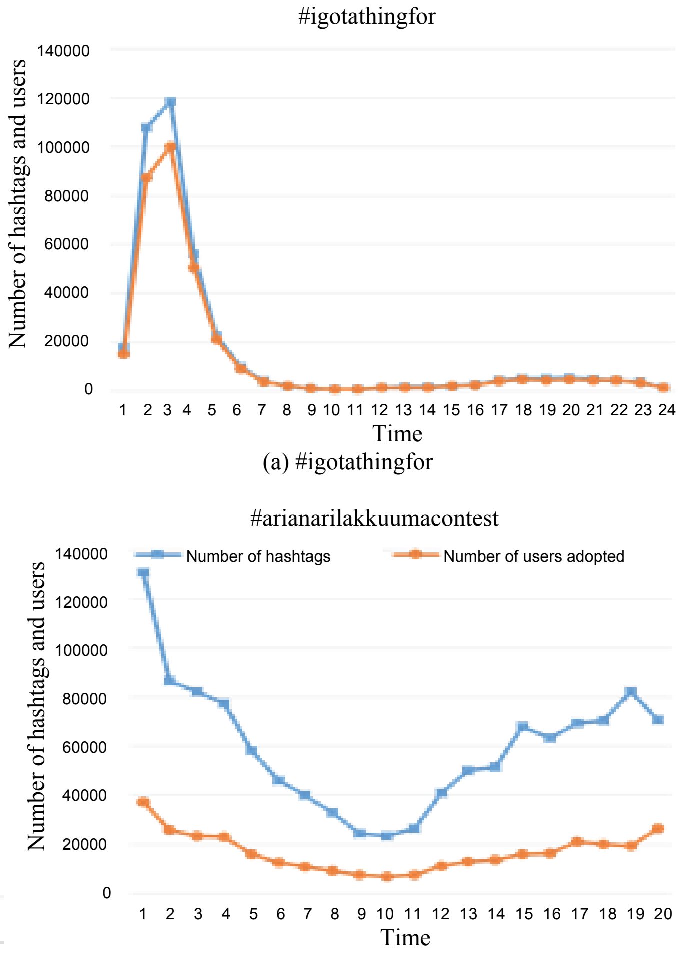
(a) Edit distance (b) Volatility

(c) Edit distance (d) Volatility

Fig. 16. (a) & (c) edit distance, (b) & (d) volatility.

Similarly, Fig. 16 (c) shows the edit distance between THLs. Fig. 16 (d) shows that volatility is minimum on Jan-27, 2013. This is because of less difference between THLs on that day and it indicates that people were talking about some particular topic. After examining the hashtags, we found that many top trending hashtags were related to 2013 Pro Bowl. 2013 Pro Bowl was the National Football League's sixty-third annual all-star game that was held on January 27, 2013. More Twitter users were talking about that event; hence, there were not much difference between THLs.

Fig. 17 shows the difference between two trending hashtags with respect to number of users who used these hashtags. In Fig. 17 (a), #igotathingfor is one of the popular hashtags on Jan-14-2013 and in Fig. 17 (b), #arianarilakkuumacontest is one of the popular hashtags on Jan-04-2013. If we compare the number of hashtags used and the number of users in these two hashtags, we can clearly see the difference between them. In the case of #igotathingfor, the number of hashtags is almost equal to the number of users and we can say that this is a popular hashtag. As to #arianarilakkuumacontest, we can see the difference between the number of hashtags and the number of users is larger. That is, the latter hashtag is considered as less popular than the former one.



(a) #igotathingfor

(b) #arianarilakkuumacontest

Fig. 17. Difference between two popular trending hashtags.

5. CONCLUSIONS

In the era of big data, we are capable to track, observe, analyze and predict the spread of information by utilizing large amounts of digital data. In this paper, we have presented how we can derive topics in social media like Twitter using a corpus. Compared to intJNMF, a representative related work, our corpus-based topic derivation (CTD) performs better in terms of purity and F-measure because of using asymmetric topic LF-LDA with a corpus. The purity (F-measure) of our proposed CTD increases from 5.26% (27.81%) to 11.32% (34.28%) for 20 to 100 topics. We have also presented a timestamp-based popular hashtag prediction (TPHP) to predict popular hashtags. We use edit distance to find the difference between consecutive THLs. By using the edit distance, we can calculate volatility to find how people react to real world events. Compared to Hybrid+, a representative related work, the mean average precision (MAP) of our TPHP approach increases by 19.45% (week-day), 15.08% (week-week) and 16.95% (month-week). Our experiment results show that using timestamp we can achieve better MAP by creating THLs for every 60 minutes.

6. FUTURE WORK

We have used a Twitter corpus in our CTD approach. It would be interesting to see whether other corpuses like a Wikipedia corpus or any other social media like Instagram or Facebook corpuses will help improve the purity of their resulting topics. In addition, it will be useful to consider ELMo [26] vectors to help solve the OOV hashtags problem if the number of OOV hashtags is large. This also helps in word sense disambiguation and performance improvement relative to GloVe across a variety of NLP tasks. We have used trending hashtags lists (THLs) as a feature to increase the MAP while predicting popular hashtags in our TPHP. It would be interesting to explore what other properties such as URLs and user locations of Twitter could be used for enhancements.

ACKNOWLEDGEMENTS

The support by the Ministry of Science and Technology (MOST) under Grants MOST105-2221-E-009-091-MY3 and MOST107-2622-8-009-020 is gratefully acknowledged.

REFERENCES

1. J. Lehmann, B. Gonçalves, J. J. Ramasco, and C. Cattuto, "Dynamical classes of collective attention in twitter," in *Proceedings of the 21st ACM International Conference on World Wide Web*, 2012, pp. 251-260.
2. R. Nugroho, J. Yang, Y. Zhong, C. Paris, and S. Nepal, "Deriving topics in twitter by exploiting tweet interactions," in *Proceedings of IEEE International Congress on Big Data*, 2015, pp. 87-94.
3. IBM, "Big data, information integration, data warehousing, master data management, lifecycle management and data security," http://www-01.ibm.com, 2016.

4. R. Nugroho, Y. Zhong, J. Yang, C. Paris, and S. Nepal, “Matrix inter-joint factorization-a new approach for topic derivation in twitter,” in *Proceedings of IEEE International Congress on Big Data*, 2015, pp. 79-86.
5. Z. Ma, W. Dou, X. Wang, and S. Akella, “Tag-latent Dirichlet allocation: Understanding hashtags and their relationships,” in *Proceedings of IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technologies*, Vol. 1, 2013, pp. 260-267.
6. Wikipedia, “Edit distance,” https://en.wikipedia.org/wiki/Edit_distance, 2016.
7. Twitter.com, “Twitter,” https://twitter.com/, 2016.
8. I. Weber, V. R. K. Garimella, and A. Teka, “Political hashtag trends,” in *Proceedings of European Conference on Information Retrieval*, 2013, pp. 857-860.
9. W. Feng and J. Wang, “We can learn your #hashtags: Connecting tweets to explicit topics, in *Proceedings of IEEE 30th International Conference on Data Engineering*, 2014, pp. 856-867.
10. L. Yang, T. Sun, M. Zhang, and Q. Mei, “We know what @you #tag: does the dual role affect hashtag adoption?” in *Proceedings of the 21st ACM International Conference on World Wide Web*, 2012, pp. 261-270.
11. Y.-T. Lu, S.-I. Yu, T.-C. Chang, and J. Y.-J. Hsu, “A content-based method to enhance tag recommendation,” in *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, Vol. 9, 2009, pp. 2064-2069.
12. F. Sebastiani, “Machine learning in automated text categorization,” *ACM Computing Surveys*, Vol. 34, 2002, pp. 1-47.
13. Carl.cs.indiana.edu, “NaN | Web-accessible data repository for NaN Group,” http://carl.cs.indiana.edu/data/#topic2014, 2016.
14. J. Pennington, R. Socher, and C. D. Manning, “GloVe: Global vectors for word representation,” in *Proceedings of Conference on Empirical Methods in Natural Language Processing*, Vol. 14, 2014, pp. 1532-1543.
15. D. M. Romero, C. Tan, and J. Ugander, “On the interplay between social and topical structure,” in arXiv preprint arXiv: 1112.1115, 2011.
16. C. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*, Cambridge University Press, Cambridge, Vol. 1, 2008, pp. 155-156 & pp. 356-357.
17. D. Q. Nguyen, R. Billingsley, L. Du, and M. Johnson, “Improving topic models with latent feature word representations,” *Transactions of Association for Computational Linguistics*, Vol. 3, 2015, pp. 299-313.
18. D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent Dirichlet allocation,” *Journal of Machine Learning Research*, Vol. 3, 2003, pp. 993-1022.
19. Internetlivestats.com, “Twitter usage statistics – Internet live stats,” http://www.inter livestats.com/twitter-statistics, 2016.
20. Wikipedia, “Twitter,” https://en.wikipedia.org/wiki/Twitter, 2016.
21. R. Salakhutdinov and G. Hinton, “Replicated Softmax: An undirected topic model,” *Advances in Neural Information Processing Systems*, Vol. 22, 2009, pp. 1607-1614.
22. N. Srivastava, R. Salakhutdinov, and G. Hinton, “Modeling documents with a deep Boltzmann machine,” in *Proceedings of the 29th Conference on Uncertainty in Artificial Intelligence*, 2013, pp. 616-624.
23. https://en.oxforddictionaries.com/definition/volatility.
24. S. Seo, J. Kim, and L. Choi, “Semantic hashtag relation classification using co-occ-

urrence word information," in *Proceedings of the 9th International Conference on Ubiquitous and Future Networks*, 2017, pp. 860-862.

25. S. Maity *et al.*, "WASSUP? LOL: Characterizing out-of-vocabulary words in Twitter," in *Proceedings of the 19th ACM Conference on Computer Supported Cooperative Work and Social Computing Companion*, 2016, pp. 341-344.
26. M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," https://arxiv.org/abs/1802.05365, 2018.

Sharath Kumar B R (尚庫柏) received the B.E. degree in Computer Science and Engineering from the Visvesvaraya Technological University, India, in 2010 and the M.S. degree in Computer Science from National Chiao Tung University, Taiwan, in 2016. His research interests include big data analytics and machine learning.

Kuochen Wang (王國禎) received the B.S. degree in Control Engineering from National Chiao Tung University, Taiwan, in 1978 and the M.S. & Ph.D. degrees in Electrical Engineering from the University of Arizona in 1986 and 1991, respectively. He is currently a Professor in the Department of Computer Science, National Chiao Tung University. He was the Chair of the Department from August 2013 to July 2016. He was the Director of the Institute of Computer Science and Engineering/Institute of Network Engineering, National Chiao Tung University from August 2009 to July 2011. He was the Acting/Deputy Director of the Computer and Network Center at this university from June 2007 to July 2009. He was a Visiting Scholar in the Department of Electrical Engineering, University of Washington from July 2001 to February 2002. From 1980 to 1984, he was a Senior Engineer at the Directorate General of Telecommunications in Taiwan. He served in the army as a Second Lieutenant Communication Platoon Leader from 1978 to 1980. His research interests include internet of things, fog/cloud computing, big data analytics/machine learning, and SDN/NFV/5G.

Shi-Min Shen (沈士閔) received the B.S. degree in Computer Science and Information Engineering from the National Cheng Kung University, Taiwan, in 2014 and the M.S. degree in Computer Science from National Chiao Tung University, Taiwan, in 2017. His research interests include software-defined networking and big data analytics.