Journal of Computational Science xxx (2017) xxx–xxx

Contents lists available at ScienceDirect

Journal of Computational Science

journal homepage: www.elsevier.com/locate/jocs

# A pattern-based topic detection and analysis system on Chinese tweets

Lu Zhang*, Zhiang Wu, Zhan Bu, Ye Jiang, Jie Cao

*Jiangsu Provincial Key Laboratory of E-Business, Nanjing University of Finance and Economics, Nanjing 210003, China*

ARTICLE INFO

*Article history:*
Received 30 November 2016
Received in revised form 24 March 2017
Accepted 23 August 2017
Available online xxx

*Keywords:*
Social media
Topic detection
Evolutionary analysis
Sentimental analysis
Weibo

ABSTRACT

Online social media is able to convey rich and timely information about real-world events. Uncovering events on social media and sensing topics from them can acquire much valuable information, which has attracted significant research effort. However, due to the large scale of data, to detect events or topics in real time is still a challenging problem. In this paper, we propose a Pattern-based Topic Detection and Analysis System (PTDAS) on Weibo, a Twitter-like platform in China. As one of the key components of the whole system, a FP-growth-like algorithm is employed to mine cosine interesting patterns from a set of tweets, and then summarize them as topics. Specially, in order to discover topics in real-time, we parallelize the algorithm on Spark for efficient mining. Along with pattern-based topic detection, we also present some analytic techniques, including both topic evolving analysis and sentimental analysis. Extensive experiments on the real-world data set demonstrate the effectiveness and efficiency of PTDAS.


## 1. Introduction

With a huge number of users involved in, online social media is able to convey rich and timely information about real-life events of all kinds. Compared with the traditional media such as newspapers, web portals, television, etc, information on social media is usually generated more timely and propagated more quickly. Intuitively, uncovering events on social media and sensing topics of these events can motivate a wide rage of applications, such as public opinion monitoring [1], viral marketing [2], early warning for major emergencies (e.g., disease outbreaks and terrorist incidents) [3], and so on. Unsurprisingly, the rapid growth of microblogging services such as Twitter and Weibo brings on the explosion of the amount of short-text messages that gradually evolve as the mainstream form of information released on social media. For instance, Sina Weibo (http://weibo.com/), a Twitter-like platform in China, receives over 100 million tweets every day. Due to the large scale of data that is produced on social media, to detect events in real time is a challenging problem which has attracted significant research effort [4,5]. Moreover, people may further be interested in how some events or topics changing over time, or what is the public sentiment (e.g., positive or negative) towards some events. So, various analytic processing on social events is needed to meet different interests.

Strictly speaking, an event in multimedia contains multiple aspects [6], such as temporal aspect, information aspect, structural aspect, spatial aspect, and so on. Among these aspects, the information aspect refers to the involved topics and it is usually most difficult to be extracted from a stream of documents. Therefore, in this article, we focus on detecting and analyzing time-varying topics from a set of short-texts produced within social media platforms. In the literature, the existing methods for extracting topics of events can fall into two categories [7]: *document-pivot* and *feature-pivot*. The document-pivot methods cluster documents into a number of groups and each group represents a topic [8,9]. Whereas the feature-pivot methods group together terms (i.e., words) according to their cooccurrence patterns and thus a set of keywords represents a topic [10,11]. The famous probabilistic topic models, e.g., Latent Dirichlet Allocation (LDA) [12], can be viewed as one of feature-pivot approaches, since they employ statistical models to extract sets of terms that are representative of the topics occurring in a corpus of documents. However, the probabilistic topic models usually suffer from the computational inefficiency and thus they are not suitable for extracting topics in real-time from large-scale data emerging from ever-growing social media.

Another important feature-pivot methods employ frequent pattern mining to discover simultaneous cooccurrence patterns containing more than two terms [13]. Once the set of frequent patterns has been mined, the remaining task is to pick out relevant

* Corresponding author.
*E-mail address:* luzhang@njue.edu.cn (L. Zhang).

http://dx.doi.org/10.1016/j.jocs.2017.08.016
1877-7503/

yet diverse patterns to represent an event [10]. Except the *support* measure, some interestingness measure (e.g., *lift*) is often used as the extra index for ranking patterns [10]. Similar to the *lift*, many interestingness measures have been proposed to find truly interesting patterns [14]. Patterns with a measure value above some given thresholds are called interesting patterns with respect to that measure (e.g., *lift*). In this work, we propose to utilize cosine interesting patterns for detecting topics, which is embedded as one of key components of the whole system. In order to discover topics in real-time, we present an efficient distributed framework on `Spark` [15] for mining cosine interesting patterns. Moreover, since the number of patterns is often overlapped, we develop an agglomerative hierarchical clustering method to summarize main topics from patterns. Along with mining patterns for topic detection, some analytic techniques are presented, including both topic evolving analysis and sentimental analysis. Our main contributions are summarized as follows:

1 We build PTDAS, a comprehensive system for topic detection and analysis on Chinese Tweets. To our best knowledge, PTDAS is among the early systems serving for pattern-based topic detection. Experiments on real-life dataset have demonstrated that PTDAS can efficiently detect meaningful topics from the Chinese Twitter, i.e. Sina Weibo.
2 We propose to utilize cosine interesting patterns for detecting topics, which could extract more closely-knit terms with better semantics.
3 We design a parallel algorithm for cosine pattern mining to meet the big data and real-time mining challenge.

The reminder of this paper is organized as follows. In Section 2, we describe the framework of the topic detection and analysis system. Section 3 gives the algorithmic details of pattern-based detection and the parallelization on `Spark`, followed by Section 4 introducing the analytic methods of topic evolving analysis and sentimental analysis. Experiment results are reported in Section 5, followed by related work in Section 6. We conclude the paper in Section 7.

## 2. System overview

In this section, we give an overview of our Pattern-based Topic Detection and Analysis System (PTDAS in short). Fig. 1 illustrates the three-tier architecture of PTDAS, with a brief introduction as follows:

### 2.1. Data acquisition and pre-processing

This tier is responsible for collecting real-time data from Weibo, and efficiently storing the huge-volume data for further access. Meanwhile, some pre-processing is also necessary to make the detecting algorithm to be effective and more efficient. To perform these tasks, we first design a distributed crawler based on `Scrapy`,[1] a widely used crawler implemented by Python. The main improvement is that we re-design the job scheduling mechanism in `Scrapy` and employ `Docker`[2] to encapsulate spiders for easier horizontal scaling. The details of this crawler could be referred in literature [16]. Then, since most of the collected data is unstructured data (e.g., texts) with huge volume and shareable requirement, the `Hadoop Distributed File System (HDFS)`[3] is adopted as the basic storage. As the supplement, we also employ `MongoDB`[4] to store the structured data and a little of small-volume unstructured data. At last, unlike English texts in which words have been delimited by white spaces, the collected Chinese tweets must be segmented first for further processing. Many open tools such as `JieBa`[5] and `ICTCLAS`[6] could be used here to perform the word segmentation. After this, the punctuation, stop-words, duplicated words and mentions (i.e., IDs or names of other users) are removed from each tweet. Meanwhile, we also filter out the advertisements, which have no significance to topic detection, by blacklist and a SVM based classifier.

### 2.2. Topic detection and analysis

This tier undertakes the core tasks to detect topics from the large-scale Weibo data, and analyze the evolution and sentiment of the detected topics. Corresponding to these tasks, three modules are contained in this tier as following.

- *Topic detection*. In this module, a cosine interesting pattern (CIP in short) based approach is proposed to detect topics from the collected data. With a *transaction data model* where a tweet corresponds to a transaction containing all terms in the pre-processed tweet, we first design a FP-growth-like algorithm: CP-Miner to mine CIPs. Then, to meet the challenge of real-time mining, we parallelize CP-Miner to fit `Spark`, a memory based lightning-fast distributed computing framework, which gives the enhanced algorithm: CP-Miner*. Furthermore, since many mined patterns are overlapped, which means multiple patterns essentially represent a same topic. We cluster the patterns to summarize main topics, utilizing the method of agglomerative hierarchical clustering. Specially, to obtain meaningful topic clusters, we propose to utilize a so-called *Weighted-Q* measurement as the indicator for clustering dendrogram partitioning. Note that although agglomerative hierarchical clustering has an $O(n^2 \log n)$ time complexity, the topic detecting efficiency is not seriously influenced because the number of mined patterns is usually not very large. The reason is that we generally set a relative high cosine threshold so that only closely-knit terms are considered to form patterns. This significantly decreases the number of mined CIPs.
- *Evolutionary analysis*. To analyze the evolution of topics, we first detect topics independently in each time slot, and then correlate the topics in two successive slots. The correlation between two topics are calculated by cosine distance of the corresponding cluster centroids. Since the correlation between topics is not injective, we propose a greedy algorithm to heuristically correlate topics one to one. Then, the successive correlation from one time-slot to another expresses the evolving of topics, which could uncover important trends combining with some statistic on the topics (e.g., hotness).
- *Sentimental analysis*. To analyze the sentimental polarity of topics, a sentiment-word based method is proposed here. First, we select adjectives, adverbs, nouns and verbs as sentimental features from all terms in patterns. Secondly, we calculate the sentimental semantic orientation (SSO in short) of each features according to unambiguous positive/negative paradigm sets. The PMI (Pointwise Mutual Information) is used here to measure the association between the features and paradigm words. Thirdly, pattern sentiment is calculated by averaging the SSO of terms in the pattern. Finally, we could obtain the topic sentiment by averaging the sentiments of patterns belong to this topic.

[1] https://scrapy.org/.
[2] https://www.docker.com/.
[3] http://hadoop.apache.org/.
[4] https://www.mongodb.com/.
[5] https://github.com/fxsjy/jieba.
[6] http://ictclas.nlpir.org/.

Please cite this article in press as: L. Zhang, et al., A pattern-based topic detection and analysis system on Chinese tweets, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.08.016

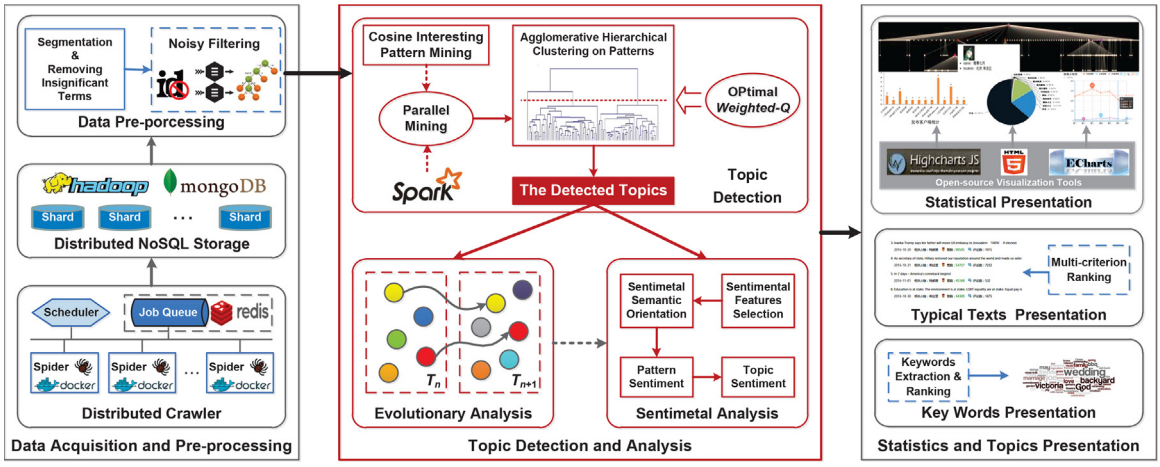



**Fig. 1.** System architecture of PTDAS.

### 2.3. Statistics and topics presentation

This tier provides Web-based visualization for user interactions. Some statistics about the detected topics and raw data, such as retweeting, opinion leaders, sentimental distribution and temporal trend, are presented by `Highcharts`[7] and `Echarts`.[8] Insider the topics, we also present typical tweets and key words to make the semantics of topics more explicit. Taking the tweets containing patterns of one topic as the topic's related tweets, the typical tweets come from the Top-$K$ of the ranked related tweets. In terms of different preferences, the ranking criterions could be *Post Time*, *Popularity*, *Sensitivity* or *Relevancy*, which are defined in literature [17]. The key words are also extracted from the topic related tweets, using the classic word ranking approaches such as TF-IDF [18] or TextRank [19].

## 3. Pattern-based topic detection

As introduced in literature [10], frequent patterns could be used to represent topics. However, to identify real hot topics, it usually employs some interestingness measure (e.g., *lift*) as the extra index for patterns ranking. In the literature, many interestingness measures have been proposed for mining truly interesting patterns [20]. Among them, the cosine similarity gains particular interests. Indeed, cosine similarity has been widely used as a popular proximity measure in text mining [21] and information retrieval [22] to avoid the "curse of dimensionality" [23]. Moreover, it is very simple and has a clear physical meaning the cosine value of the angle of two vectors. Therefore, in this work, we adopt *cosine* as the measurement and propose to utilize cosine interesting pattern [24] (CIP in short) for topic detecting. Restrained by the cosine threshold, the CIP mining could extract more closely-knit terms to form patterns with better semantics. Furthermore, to meet the real-time requirement of detecting, we parallelize the mining algorithm on `Spark`, a distributed memory based computing framework.

### 3.1. Preliminaries

The definition and application of CIP could be seen in literature [24–26]. For the self-contained purpose, we also give the background knowledge of CIP in this section.

Similar with the traditional market basket transaction in association analysis, the tweets to be analyzed construct a *transaction data set* (denoted as $\mathcal{T}$), in which a tweet corresponds to a transaction (denoted as $t$) containing all terms in the pre-processed tweet as its items (denote as $i$). For any set of multi-itemsets $I = \{i_1, i_2, \ldots, i_K\} (K \geqslant 2)$, the cosine similarity of $I$ is defined as:

$$Cos(I) = \frac{Supp(I)}{\sqrt[K]{\prod_{k=1}^{K} Supp(\{i_k\})}}, \tag{1}$$

where $Supp(\cdot)$ is the support of an itemset given by $Supp(I) = SC(I)/|\mathcal{T}|$, and $SC(I) = |\{T_{i_k} | I \in T_{i_k}, T_{i_k} \in \mathcal{T}\}|$ means the support count of $I$. As defined in [24], we formulate the definition of cosine interesting pattern (CIP) as follows.

**Definition 1.** CIP: Given $\tau_s \in [0, 1]$ and $\tau_c \in [0, 1]$ as the minimum thresholds of support and cosine measures, respectively, an itemset $I$ is called a cosine interesting pattern with respect to $\tau_s$ and $\tau_c$, if $Supp(I) \geqslant \tau_s$ and $Cos(I) \geqslant \tau_c$.

As is known, the anti-monotone property (AMP) is an important property for accelerating frequent pattern mining. However, the cosine similarity does not hold the AMP. To guide the pruning in CIP mining, the ordered anti-monotone property (OAMP) is proposed as follows.

**Definition 2.** OAMP: Let $I$ be a universal itemset. A measure $M$ holds the *OAMP*, if $\forall X, Y \subseteq I$, given that *(1)* $X \subseteq Y$, and *(2)* if $Y \setminus X \neq \emptyset$, $\forall i \in X$ and $i' \in Y \setminus X$, $Supp(i) \leqslant Supp(i')$, we have $M(X) \geqslant M(Y)$.

Compare with the well-known AMP, OAMP demands an extra condition that all the items in the difference set $(Y \setminus X)$ must have higher supports than the items in the subset $(X)$. The cosine similarity is proved to satisfy the OAMP [24,26], which lays the foundation for the efficient mining in Section 3.2.

### 3.2. Cosine interesting pattern mining

In this section, we present **CP-Miner**, a FP-growth-like algorithm for CIP mining based on the OAMP of cosine similarity. In general, CP-Miner first builds an FP-tree, a special structure for fast support computation in frequent pattern mining [27], and then searches the FP-tree for cosine patterns using a FP-growth-like [25] procedure: CP-GROWTH. The pseudocodes of CP-Miner are given in Algorithm 1.

**Algorithm 1.** CP-Miner

[7] http://www.highcharts.com/.
[8] http://echarts.baidu.com/.

Please cite this article in press as: L. Zhang, et al., A pattern-based topic detection and analysis system on Chinese tweets, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.08.016

**Input:**
*Tree*: The (conditional) FP-tree, initially built from $\mathcal{T}$
*S*: The suffix pattern corresponding to *Tree*, initially $S \longleftarrow \emptyset$
$\tau_s$: The minimum support threshold
$\tau_c$: The minimum cosine threshold
**Output:**
*F*: The set of CIPs, initially $F \longleftarrow \emptyset$

```
1:  procedure CP-GROWTH(Tree, S, F, τ_s, τ_c)
2:    for each item i_k from bottom to top in Tree's head table do
3:  Generate candidate CIP S' ⟵ S∪{i_k};
4:  if Cos(S') ⩾ τ_c then
5:        F ⟵ F∪S';
6:        Create the conditional FP-tree Tree_S' for S' with τ_s;
7:        if Tree_S' ≠ ∅ then
8:          CP-GROWTH(Tree_S', S', F, τ_s, τ_c);
9:        end if
10:     end if
11:   end for
12:   return F;
13: end procedure
```

Note that to utilize the OAMP, CP-Miner demands the FP-tree be built on transactions with items sorted in a *support-descending* order [25]. Take the toy data in Fig. 2 as an example, given the transaction data set in the upper-left part of the figure, we have $Supp(E) > Supp(D) \geqslant Supp(C) > Supp(B) > Supp(A)$. As a result, we must first sort the items in each transaction from *E* to *A* (*C* and *D* are interchangeable). The FP-tree for CP-Miner can then be built in the bottom-left part of Fig. 2. In this way, CP-GROWTH can traverse itemsets ended by *A* to *E* orderly, no matter in the FP-tree or a conditional FP-tree. To demonstrated the high efficiency of CP-Miner root in the OAMP of cosine measure, we illustrate CP-GROWTH by a toy example as shown in Fig. 2. Let the $\tau_s = 25\%$ and $\tau_c = 0.6$. Assume that we want to mine cosine patterns ended by item *B*. To this end,

(i) CP-GROWTH first generated the conditional FP-tree based on the suffix $S = \{B\}$, as shown in the upper-right part of Fig. 2.
(ii) CP-GROWTH then proceeded to check the cosine value of $\{C, B\}$, which was $Cos(\{C, B\}) = 0.447 < \tau_c$. Based on the OAMP of the cosine measure, CP-GROWTH stopped the sub-tree projection for potential cosine patterns ended by $\{C, B\}$. As shown in the bottom-right of Fig. 2, this means the pruning of the whole sub-tree circled by the red dotted-line.
(iii) CP-GROWTH then jumped to the itemset $\{D, B\}$, the immediate sibling of $\{C, B\}$. Since $Cos(\{D, B\}) = 0.671 > \tau_c$, CP-GROWTH added $\{D, B\}$ into the CIP set *F* and create $\{D, B\}$'s conditional FP-tree. In the recursive call of CP-GROWTH, the cosine value of $\{E, D, B\}$ is examined, which was $Cos(\{E, D, B\}) = 0.254 < \tau_c$. As a result, the itemset $\{E, D, B\}$ is also pruned.
(iv) CP-GROWTH then proceeded to jumped to the immediate sibling of the itemset $\{D, B\}$, i.e., $\{E, B\}$. Since $Cos(\{E, B\}) = 0.756 > \tau_c$, it can be added to the CIP set *F*.

As a result, two itemses $\{D, B\}$ and $\{E, B\}$ will be returned from 8 itemsets. In fact, the CP-Miner could totally obtain 7 multi-item CIPs: $\{E, A\}$, $\{D, B\}$, $\{E, B\}$, $\{D, C\}$, $\{E, D, C\}$, $\{E, C\}$ and $\{E, D\}$ after creating 7 conditional FP-trees only. This case well demonstrates the high efficiency of CP-Miner.

### 3.3. Parallelization on Spark

To further improve the efficiency of CP-Miner, we propose to parallelize it with distributed computing framework, which gives an enhanced algorithm: CP-Miner*. In FP-growth, the FP-tree can be easily split into several smaller trees by projecting the transactions in $\mathcal{T}$. CP-Miner inherits this merit and we employ the so-called *aggressive projection* technique [28] to parallelize CP-Miner. The definition of *aggressive projection* is given as follows.

**Table 1**
Example of aggressive projection.

| Groups | Projected transactions |
|---|---|
| $\{D, E\}$ | $\{D, E\}(5)$, $\{E\}(2)$ |
| $\{B, C\}$ | $\{B, C, D, E\}(2)$, $\{B, D, E\}$, $\{C, D, E\}(2)$, $\{B, E\}$, $\{C\}$ |
| $\{A\}$ | $\{A, B, D, E\}$, $\{A, C, D, E\}$, $\{A, E\}$ |

**Definition 3.** Aggressive projection: Assume the items in headertalbe FList are divided into K crisp groups from top to bottom, i.e., $FList = \beta_1 \cup \beta_2 \cup \ldots \cup \beta_K$, with $\beta_k = \{i_{k_1} \ldots i_{k_r}\}$ consisting of r successive items in FList. Then the $\beta_k$-projected data can be defined as: $\mathcal{T}_{\beta_k} = \{T_{i_p} \cap (\cup_{j=1}^{k} \beta_j) | T_{i_p} \cap \beta_k \neq \emptyset, T_{i_p} \in \mathcal{T}\}$.

To illustrate the idea of *aggressive projection*, we take the transaction data in the upper-left of Fig. 2 as an example. Assume *FList* is divided into 3 groups: $\{D, E\}$, $\{B, C\}$ and $\{A\}$. Table 1 shows the projected transactions for each group. Taking $\{B, C\}$-projected transactions as example. All transactions containing *B* or *C* are extracted after removing the items listed behind *B* in *FList*, i,e., *A*. This results in 7 projected transactions, with "$\{B, C, D, E\}(2)$" indicating the two containing $\{B, C, D, E\}$.

Then, we propose to use `Spark` [15], a lightning-fast cluster computing framework, for parallielizing CP-Miner. We encapsulate CP-Miner on Spark as two MapReduce procedure as Fig. 3. In detail, a transaction dataset is horizontally yet uniformly split and loaded into the *resilient distributed datasets* (RDDs) of every computational nodes. Firstly, the first MapReduce procedure is invoked for sorting all of items and generating *FList*. The `map` phase is responsible for counting the support of items on local data, and the `reduce` phase collects local count tuples from all computational nodes and sorts items to produce *FList*. Secondly, the master node uniformly splits *FList* into *K* groups and broadcasts *K* group to a computational node. Thirdly, the second MapReduce procedure is started for projecting the transaction dataset by using the *aggressive projection* technique [28]. In particular, the `map` phase splits every transaction that is stored in the local RDD into *K* groups according to the grouping results on *FList*. Then, all of sub-transactions with the same group ID on different computational nodes are assigned to a new RDD, i.e., a total of *K* new RDDs. Finally, multiple `reduce` phases are started to invoke CP-Miner on each new RDD for cosine pattern mining.

### 3.4. Summarizing topics from patterns

The extracted patterns are often overlapped, i.e., multiple terms are shared among them, which means these patterns are corresponding to the same topic. Hence it is indeed that we could aggregate these patterns as one topic for further analysis.

Here, we develop an agglomerative hierarchical clustering method to summarize main topics from patterns. Generally, there are two important components in agglomerative hierarchical clustering, distance computing and cluster proximity. Assume there are totally *m* different items in the pattern set, we could construct a *m*-dimension feature space $\mathcal{F} = \{f_1, f_2, \ldots, f_m\}$ with $f_n$ corresponding to the *n*-th item. Then, a pattern *P* cloud be modeled as a vector $P = \langle p_1, p_2, \ldots, p_m \rangle$, where $p_i = 1$ when $f_i \in P$ and $p_i = 0$ otherwise. Due to the high-dimension and sparsity of the pattern vectors, we select the cosine distance for distance computing. Given two vectors $P^i$ and $P^j$, the cosine distance between them is

$$Dist(P^i, P^j) = 1 - \frac{\sum_{n=1}^{m} p_n^i p_n^j}{\sqrt[2]{\sum_{n=1}^{m} {p_n^i}^2} \sqrt[2]{\sum_{n=1}^{m} {p_n^j}^2}}, \quad (2)$$

where the equation in the right of minus is the *cosine similarity* of $P^i$ and $P^j$. In practice, we could store the vector as a sparse vector and only consider the non-zero elements when computing distance,

Please cite this article in press as: L. Zhang, et al., A pattern-based topic detection and analysis system on Chinese tweets, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.08.016

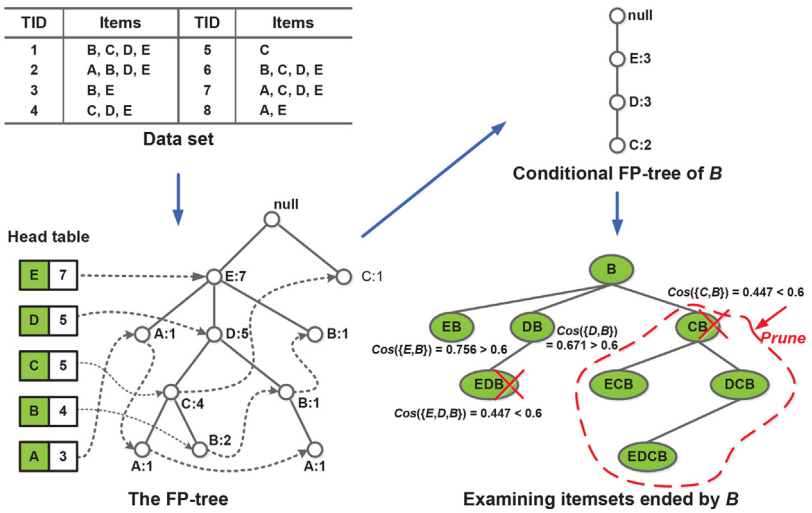



**Fig. 2.** Illustration of CP-GROWTH.

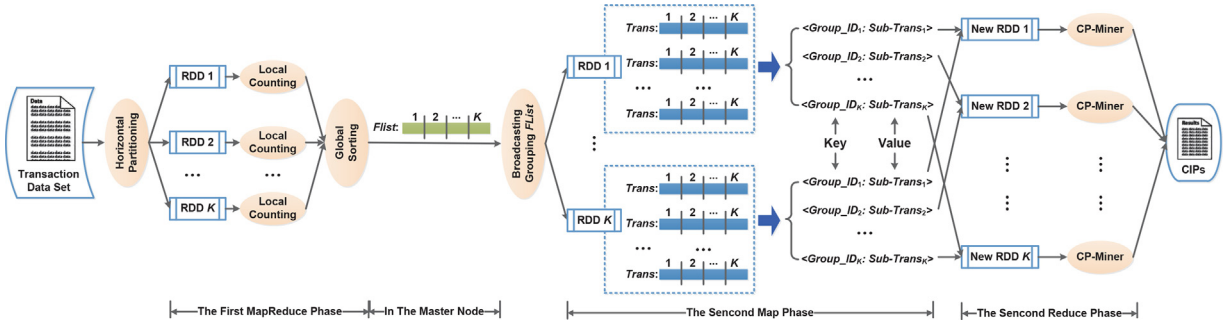



**Fig. 3.** Procedure of CP-Miner*.

which will significantly improve the efficiency. For cluster proximity, we take centroid to represent the cluster, and the object is to minimize the sum of the squared distances of instances from their cluster centroids. Thus, the Ward's method is adopted for cluster proximity, which measures the proximity between two clusters in terms of the increase in the SSE that results from merging the two clusters.

Generally, hierarchical clustering repeatedly merge the clusters until all the instances are aggregated in a single cluster, and gives a dendrogram as the result. However, to obtain meaningful topics, it is crucial to know where to partition the dendrogram. Essentially, topics could be seen as multiple sets that patterns inside a set is more similarly but outside is more differently. This property is obviously similar to the communities in complex networks. Along this line, we could convert the pattern set to a complex network with patterns as vertexes and the cosine similarities between patterns as edges. In this model, the measurements used to evaluate community could be apply to analyze the pattern clusters. Here, we adopt *Modularity* (Also called *Q*) [29], one of the most widely used community measurements, to guide the partition of the dendrogram. However, the original *Modularity* could only be applied in unweighed networks. Thus, we use a weighted version of the *Modularity* [30] defined as following.

$$Weighted - Q = \frac{1}{2n}\sum_{ij}\left[A_{ij} - \frac{k_i k_j}{2n}\right]\delta(c_i, c_j), \tag{3}$$

where $n$ is the number of edges in the network, $A_{ij}$ is the adjacency matrix, $k_i$ and $k_j$ means the degree of vertex $i$ and $j$ respectively. The $\delta$-function $\delta(u, v)$ is 1 if $u = v$ and 0 otherwise. $c_i$ is the community to which vertex $i$ is assigned. In our scenario, this means pattern $i$ is assign to cluster $c_i$. We repeatedly calculate *Weighted-Q* in terms of the dendrogram, and finally take the optimal *Weighted-Q* as the partition line, under which the clusters are corresponding to the summarized topics.

It is worth noting that the time complexity of agglomerative hierarchical clustering is $O(n^2 \log n)$, however, the efficiency of topic detecting is not seriously decreased. The reason is that the number of mined patterns is often not very large, owe to the cosine measurement. Since we generally set a relative high cosine threshold, only closely-knit terms are considered to form patterns. This could significantly decrease the number of mined CIPs, which is also verified in the experimental Section 5.3.

## 4. Topic analysis

After multiple topics are extracted, some analytic techniques are applied on these topics to uncover the evolution information and the underlying sentiment.

### 4.1. Evolutionary analysis on topics

Evolutionary analysis studies how topic changes with time elapsing. Dividing the elapsed time into multiple slots, we first detected topics independently in different time slots. Since there are no specific labels on detected topics, we must correlate the same topics in different slots to analyze how this topic evolves. The topics are represented as clusters of patterns, thus we could correlate two topics according to the distance of the clusters. Here, we use cosine distance of cluster centroids to measure the dis-

Please cite this article in press as: L. Zhang, et al., A pattern-based topic detection and analysis system on Chinese tweets, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.08.016

tance between two clusters. If the distance between two clusters is smaller than a threshold $\theta$, we heuristically correlate the corresponding topics. However, in practice multiple topics in one time slot are often correlated to a same topic in another time slot, and vice versa. To solve this problem, we propose to model the topics and distances as a directed weighted bipartite graph, in which vertexes with only out-degree in one side means topics in a earlier slot, and vertexes with only in-degree in the other side means topics in a later slot, and the weighted edges means the distance between two topics. Note that if the distance between two topics is larger than $\theta$, there is no edges between the corresponding vertexes in the graph. Based on this model, a greedy algorithm CO-TOPIC is proposed to assign the final correlation of topics. If two topics are finally correlated, it indicates that they are essentially the same topic that evolves from one time slot to another. There is also a situation in the algorithm that some topics could not be correlated to any other topics, and this indicates the emerging or fading of the topic. The pseudocodes of CO-TOPIC are given in Algorithm 2.

**Algorithm 2.** CO-TOPIC

**Input:**
$\mathcal{G}$: The weighted bipartite graph of topics
**Output:**
*Evolving*: The evolving topic set, initially *Evolving* $\longleftarrow \emptyset$
*Emerging*: The emerging topic set, initially *Emerging* $\longleftarrow \emptyset$
*Fading*: The fading topic set, initially *Fading* $\longleftarrow \emptyset$

**procedure** CO-TOPIC$\mathcal{G}$
  **while** $\mathcal{G} \neq \emptyset$ **do**
    Selecting the edge $u \rightarrow v$ with the smallest weight;
    *Evolving* $\longleftarrow \{u, v\}$;
    Removing $u$, $v$ and all edges attached to them from $\mathcal{G}$;
    **for** each vertex $s$ with no edges **do**
      **if** $s$ is in the earlier time-slot side **then**
        *Fading* $\longleftarrow \{s\}$;
      **else**
        *Emerging* $\longleftarrow \{s\}$;
      **end if**
      Removing $s$ from $\mathcal{G}$;
    **end for**
  **end while**
**end procedure**

Taking Fig. 4(a) as an example, we construct a directed weighted bipartite graph with $A$, $B$, $C$, $D$ are the topics detected in time slot $T_1$ and $E$, $F$, $G$, $H$ are the topics detected in time slot $T_2$. Firstly, we select the edge with the smallest weight, i.e., $A \rightarrow E$ weighted as 0.11. $\{A, E\}$ is added into the *evolving* set and removed from the graph. The edges attached to $A$ or $E$, i.e., $A \rightarrow E$, $A \rightarrow F$ and $A \rightarrow G$ are also removed. Note that in the modified graph, $F$ is a separated vertex with no edges. Due to $F$ is in the later time-slot side, we add $F$ into *Emerging* set. Secondly, the smallest edge in the modified graph, i.e., $C \rightarrow H$ is selected. We add $\{C, H\}$ into the *evolving* set, then remove them and the attached edges, i.e.,$C \rightarrow G$, $A \rightarrow H$ and $D \rightarrow H$. Thirdly, the smallest edges $B \rightarrow G$ is selected and $\{B, G\}$ is added into the *evolving* set. Removing $B$, $G$ and the attached edges $B \rightarrow G$ and $D \rightarrow G$, the vertex $G$ becomes a separated vertex. Since $G$ is in the earlier time-slot side, we add it into the *Fading* set. At last, we have $\{A, E\}$, $\{C, H\}$ and $\{B, G\}$ are actually the same topic in pairs and evolving from $T_1$ to $T_2$. Meanwhile, we also uncover that $F$ is an emerging topic in $T_2$, and $D$ is faded after $T_1$.

Extending the topic correlation between two time slots, we could uncover the evolution of topics in multiple slots by correlating topics in slot $T_n$ to slot $T_{n+1}$ successively. Fig. 4(b) illustrates the successively correlating with each color indicates an evolving topic. Note that not all the topics are evolving in the whole period. Topics could emerge and fade in arbitrary time slot and even in a single slot. This could also be captured by our correlation algorithm CO-TOPIC.

### 4.2. Sentimental analysis on topics

To analyze the sentimental polarities of detected topics (i.e., Positive or Negative), we propose a sentiment-word based method, which is also used in literature [32] for analyzing sentiment in Tianya (A Chinese Forum). The evaluative character in sentiment of a word is called its sentimental semantic orientation (SSO in short). Positive semantic orientation indicates praise (e.g., "good", "nice") and negative semantic orientation indicates criticism (e.g., "bad", "poor"). Previous researches on sentimental analysis [31,32] show that the underlying sentiment of a short text is mainly decided by those unambiguous sentiment-bearing words. The proposed sentimental analysis method in PTDAS is also along this line: (1) first explores sentiment-bearing words in patterns to construct feature space, (2) then calculate SSO of each individual word, (3) and then average the SSO of the words in each pattern to obtain its sentimental value, 4) and finally accumulates the weighted sentimental values of patterns to determine the sentimental polarity of the topic.

#### 4.2.1. Sentimental features selection

Under the bag-of-words model, the patterns to be analyzed could compose a large set of words that each word could be a feature in sentimental analysis. However, to reduce the dimensions of feature space, only a subset including adjectives, adverbs, nouns, and verbs is considered, in which $n$ high-frequency words are selected as the sentimental features.

#### 4.2.2. Sentimental semantic orientation study

Let $C = \{c_1, c_2, \ldots, c_n\}$ be the feature set, including $n$ sentimental features. We manually selected $n_P$ positive features and $n_N$ negative features with unambiguous sentimental characteristics as positive/negative paradigm sets, denoted as $C_P$ and $C_N$. The SSO of other features, e.g., $S(c_l)$, is calculated from the strength of its association with a set of positive features, minus the strength of its association with a set of negative features.

$$PMI(c_i, c_j) = \log_2 \frac{\sigma(\{c_i, c_j\})}{\sigma(\{c_i\})\sigma(\{c_j\})} \tag{4}$$

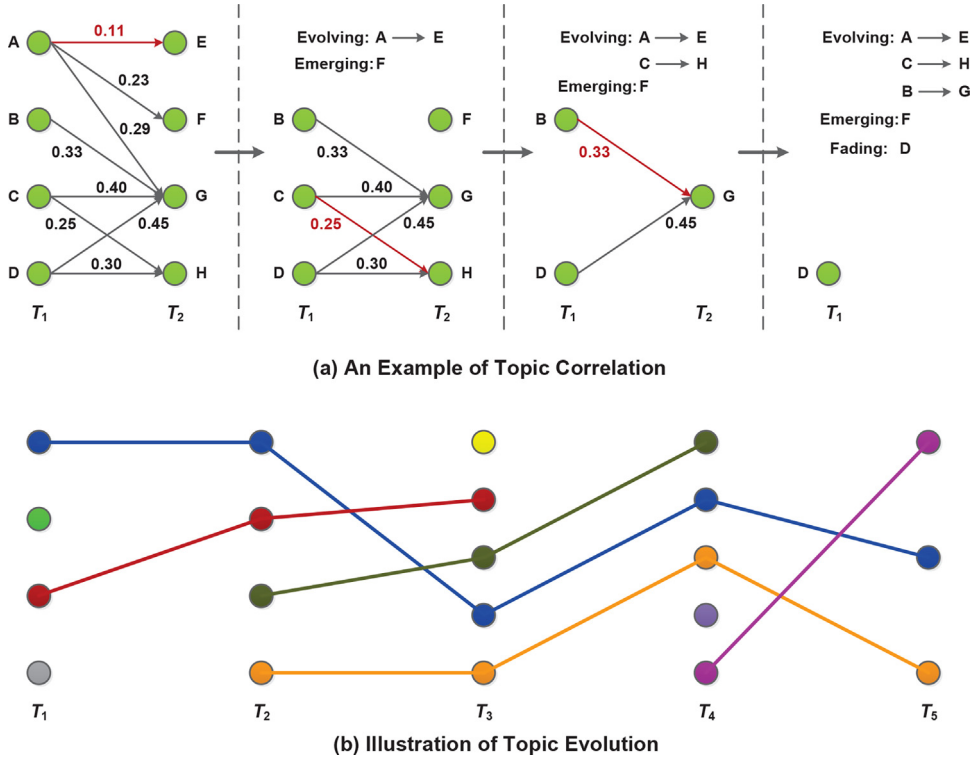
$$S(c_l) = \sum_{c_i \in C_P} PMI(c_l, c_i) - \sum_{c_j \in C_N} PMI(c_l, c_j). \tag{5}$$

Here, $PMI(c_i, c_j)$ is a measure of association between $c_i$ and $c_j$ according to PMI (Pointwise Mutual Information) [33]. $\sigma(\{c_i, c_j\})$ denotes the support for occurrences of patterns where $c_i$ and $c_j$ both appear. If the features are statistically independent, the probability that they co-occur is given by the product $\sigma(\{c_i\})\sigma(\{c_j\})$. The ratio between $\sigma(\{c_i, c_j\})$ and $\sigma(\{c_i\})\sigma(\{c_j\})$ is a measure of the degree of statistical dependence between the features. The log of the ratio corresponds to a form of correlation and the value is positive when the features tend to co-occur. In contrast, the value is negative when the presence of one feature makes it likely that the other feature is absent. Furthermore, to avoid returning negative infinity, we use "Laplasse calibration" which add 0.01 to $\sigma(\{c_i, c_j\})$.

Empirically, the range of $S(c_l)$ is between $-\infty$ and $+\infty$, with a higher value corresponding to more positive polarity. $S(c_l)$ around 0 means that the SSO of $c_l$ is neutral. To eliminate the magnitude difference on those features, a max-min fairness is applied to normalize $S(c_l)$ as following:

$$S'(c_l) = \begin{cases} \dfrac{S(c_l)}{S^M} \cdot 0.5 + 0.5 & S(c_l) \geqslant 0 \\ \dfrac{S(c_l) - S^m}{-S^m} \cdot 0.5 & S(c_l) < 0 \end{cases} \tag{6}$$

Please cite this article in press as: L. Zhang, et al., A pattern-based topic detection and analysis system on Chinese tweets, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.08.016




(a) An Example of Topic Correlation

(b) Illustration of Topic Evolution

**Fig. 4.** Topic correlation and evolution.

where $S^M$ and $S^m$ are the maximum and minimum values for $S(c_l)$ respectively. The normalized SSO, ranging from 0 to 1, represents the degree of "trust" in the form of probability.

#### 4.2.3. Pattern sentimental calculation

Based on the SSO of each word, the sentimental value of pattern $P$ could be calculated as following:

$$E_P = \frac{\sum_{c_l \in P} S'(c_l)}{|P|}, \tag{7}$$

where $c_l$ is one of the sentimental features in pattern $P$, $S'(c_l)$ is the normalized SSO of $c_l$, and $|\cdot|$ returns the number of sentimental features in $P$. Therefore, the range of $E_P$ is also between 0 and 1, with a higher value corresponding to positive sentiment polarity.

#### 4.2.4. Topic sentimental calculation

Since a topic is essentially a set of clustering patterns, thus we could apply weighted average among patterns to calculate topic sentiment value as following:

$$E_T = \frac{\sum_{P \in T} SC(P) \cdot E_P}{\sum_{P \in T} SC(P)}, \tag{8}$$

where $P$ is the pattern belongs to topic $T$, and $SC(P)$ means the support count of pattern $P$. Note that the range of $E_T$ is also between 0 and 1 according to Eq. (8). We finally assign the sentimental polarity of topic $T$ as positive when $E_T > 0.5$, negative when $E_T < 0.5$, and neutral when $E_T = 0.5$, respectively. Furthermore, the value of $E_T$ could expresses the strength of sentimental polarity, which is helpful for sentimental evolution analysis.

## 5. Experiment

In this section, we conduct extensive experiments on Sina Weibo (http://weibo.com/.), a Twitter-like platform in China, to demonstrate the effectiveness of PTDAS in topic detection and analysis.

### 5.1. Experimental setup

#### 5.1.1. Experimental environment

PTDAS is built on a x86 cluster with 12 nodes connected by a Gigabit Ethernet. Each node comes with 4 quad-core E5-2650v2 processors (2.6 GHZ), 128 GB of RAM, 240 GB of SSD disk and 600 GB of SAS disk. 4 out of these 12 nodes are utilized for data acquisition, and totally 24 Docker-encapsulated spiders are implemented for parallel data collecting from the API of Sina Weibo. The rest 8 nodes are used as storage and computational nodes simultaneously. We adopt `HDFS` as the basic storage, the collected data is first split into multiple shards and then stored in `HDFS` distributed across the 8 nodes. For parallel topic detecting, we also implement `Spark` on these 8 nodes. Every computational node can share the shards by accessing `HDFS`. Note that `Spark` is a memory based computation framework, so we only needs to access `HDFS` once to load the data into RDDs, which can significantly decrease the I/O cost of the successive MapReduce procedure.

#### 5.1.2. Data set

This experiment utilizes a large-scale text corpus collected from Weibo by the crawler. In 2015, a female journalist, Chai Jing, released a documentary named "Under the Dome", which stimulated nation-wide discussions of environment protection in recent China, and lots of interesting topics are involved in these discussions on Weibo. This text data set contains tweets that are concerned with this hot event from 28 February to 6 March, 2015. We first adopted `Jieba`, an open-source Chinese segmentation system, to perform the word segmentation on each tweets, and then removed punctuation, stop-words, duplicated words, mentions and advertisements. We also filtered the pre-processed tweets with less than three words and finally produce a data set with 3,068,369 instances (i.e., pre-processed tweets) and 18,713 different terms. The detailed statistics of the data set could be seen in Table 2, and the "Avg_Len" measures the average terms contained

Please cite this article in press as: L. Zhang, et al., A pattern-based topic detection and analysis system on Chinese tweets, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.08.016

**Table 2**
Data set statistics.

| | Feb.28 | Mar.01 | Mar.02 | Mar.03 | Mar.04 | Mar.05 | Mar.06 |
|---|---|---|---|---|---|---|---|
| #Tweets | 428,167 | 1,048,105 | 573,204 | 370,979 | 293,655 | 191,532 | 162,727 |
| #Terms | 18,701 | 18,578 | 18,275 | 18,052 | 17,878 | 17,471 | 17,380 |
| Avg_Len | 8.94 | 9.02 | 9.71 | 9.73 | 9.13 | 9.28 | 9.07 |

in each instance. Note that with the hotness degrading, the discussion about the documentary, i.e. the number of tweets is also decreased. However, this has little impacts on the accuracy of topic detection. The reason is that the support threshold in cosine pattern mining is pro rata, so even the number of tweets decreases, we can also extract frequent terms that express the hot topics.

### 5.2. Overall performances of topic detection

Here we present the overall performances of topic detection on the data set. By setting the parameter $\tau_c = 0.8$, $\tau_c = 0.02\%$ and $\theta = 0.5$, totally 20 main topics are detected from the data set. According to the semantics, we named each topic and divided them into four categories, i.e., "About the Documentary", "Environmental Protection", "Politics and Economy" and "Criticism and Defense". To show the semantics more clearly, we also extracted multiple key words from the patterns of each topic. The detected topics and their key words are shown in Table 3.

Unsurprisingly, the themes of documentary itself and the environmental protection are the focuses of discussion, however, we still find some interesting things unexpectedly. For example, based on the explosive spreading of this documentary, topic *t2* discusses the new information propagation ways in Internet Age. Moreover, extending from the atmospheric protecting, topics *t8* and *t9* propose to protect our mother river and wild animals. Besides these two obvious topic categories, many underlying topics have been extracted by PTDAS. For example, in category "Politics and Economy", the topics extend environmental problems to politics and economy problems. In this category, topic *t11* mentions that this documentary has hit the interests of some groups (e.g., the petroleum industry group), and worries that it will be banned. Topic *t13* considers the reality to maintain the balance between environment and economy, and emphasizes that the blindly enhancing of environmental protection will be harmful to the economic development and cause unemployment. Topic *t14* takes a perspective from international relations, and imputes the pollution in China to the unfair international division of labor. In the last category, i.e., "Criticism and Defense", we can see that this documentary also causes some denouncements about Chai Jing and conspiracy theories. For example, in topic *t16*, the motivation of this documentary is queried, with an opinion that Chai Jing is actually an agent of foreign forces. They want to advocate privatization, which will make the foreign capitalists easier to control China. Meanwhile, with the denouncements, there are also some advocacies for Chai Jing such as topic *t17*, and some topics (e.g., topic *t18*) also denounce the deep-rooted bad habits of Chinese. Furthermore, from the key words of each topic, we could also find some interesting things. For example, topic *t1* mentions Cui Yongyuan and his documentary about transgenosis, which has various similarities to "Under the Dome" (e.g., the two producers had both worked in CCTV once, and the two documentaries both focus on the livelihood issues with little major media reporting). The netizens thus associate these two documentaries together and treat the producers as the mainstay of our society. We also could find some interesting insider stories from the key words of topic *t16*, e.g., "Ford Foundation" and "Bush Family", which imply that Chai Jing is ordered by some foreign organizations. Although the authenticity is hard to make sure, these key words indeed reveal some interesting aspects of the event.

Besides static analysis, we also present the evolution of the topics during the seven days in Fig. 5. Each bubble in the figure denotes a topic in one day. The size of bubbles express the hotness of topics, which measures by the sum of support counts of the patterns in each topic. We also mark the number of the largest bubble of each topic, which demonstrates the burstiness. For the documentary itself, we can find that some straightforward topics, such as *t1* and *t3* could achieve the highest hotness in their first arising day, and gradually cool down with time elapsing. In contrast, the underlying topics, e.g., *t2* and *t4*, get hot increasingly and last more time than straightforward topics. This is consistent with the feature of focuses changing in real-world. The similar features also exist in

**Table 3**
Detected topics with key words.

| Category | ID | Topic | Key words |
|---|---|---|---|
| About the documentary | *t1* | Deeply Touched by the Responsibility | Responsibility, Touched, Respect, Conscience, Cui Yongyuan, Transgenosis, Mainstay |
| | *t2* | Information Propagation Ways | Internet, Opinion Leaders, Propagation Mode, Paragon, Beyond Television, Milestone |
| | *t3* | Motivation of this Documentary | Daughter, Born, Tumor, Hurt, Resignation, Care, Investigation, Authority, Answer |
| | *t4* | The Epitome of Social Change | Course, Autobiography, Growth, Social Changes, Memo, Personal, Country, Period |
| Environmental protection | *t5* | Protecting Environment for the Future | Earth, Environment, Deterioration, Guardian our Home, Governance, Children, Hope |
| | *t6* | Environmental Deterioration in China | China, Air, Water Source, Crisis, Confused, Governance, Save the Nation, Individual |
| | *t7* | Depicting and Missing Good Environment | White Clouds, Blue Sky, Pigeons, Courtyard, Rain, Irresistible, Facing the Status quo |
| | *t8* | Protection of the Mother River | Mother River, Civilization, Chinese, Revival, Advocacy, Ecology, Saving, Benefit |
| | *t9* | Protection of Wild Animals | Fur, Refused to Eat and Buy, Animals, Habitat, Hunting, Intimidation, Maltreatment |
| | *t10* | Do sth for Environmental Protection | Chimney, Photograph, Inform, Position, Pollution Sources, Protracted War, Participation |
| Politics and economy | *t11* | Hitting the Interests of Some Groups | Discerning Eyes, Interest Groups, Angered, Monopoly Classes, Risk, Published or Not |
| | *t12* | China's Political System | Fraud, Corruption, Despotism, Constitutionalism, Judicial Independence, Supervision |
| | *t13* | Balance between Economy and Environment | Achievements, Life Expectancy, Unemployment, Poor, Development, Confliction, Reality |
| | *t14* | International Relations | Kyoto Protocol, USA, Greenhouse Effect, Climate Change, International Specialization |
| Criticism and defense | *t15* | Denouncing Chai Jing | Children, Smoker, Advanced Maternal Age, Genetic Defect, Villa, High-Emission Cars |
| | *t16* | Denouncing Privatization and Betrayal | Privatization, Traitor, International Capital, Agent, Ford Foundation, Bush Family |
| | *t17* | Defending for Chai Jing | Privacy, Two Things, Professional Ethics, Not Saint, Nationality, Reason, Slanders, Defense |
| | *t18* | Ugly Chinese | Chinese, Devoid of Conscience, Nation's Bad Character, Gossip, Instigation, Insult, Ridicule |
| | *t19* | Querying Authenticity and Scientificity | Tumor, Unrelated, Secondary School Students, Calculation Errors, Common-Sense Errors |
| | *t20* | Defending for Authenticity and Scientificity | Science, Data, Misunderstanding, Icon Errors, Facts, Reasons, logistically, Intellectuals |

Please cite this article in press as: L. Zhang, et al., A pattern-based topic detection and analysis system on Chinese tweets, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.08.016

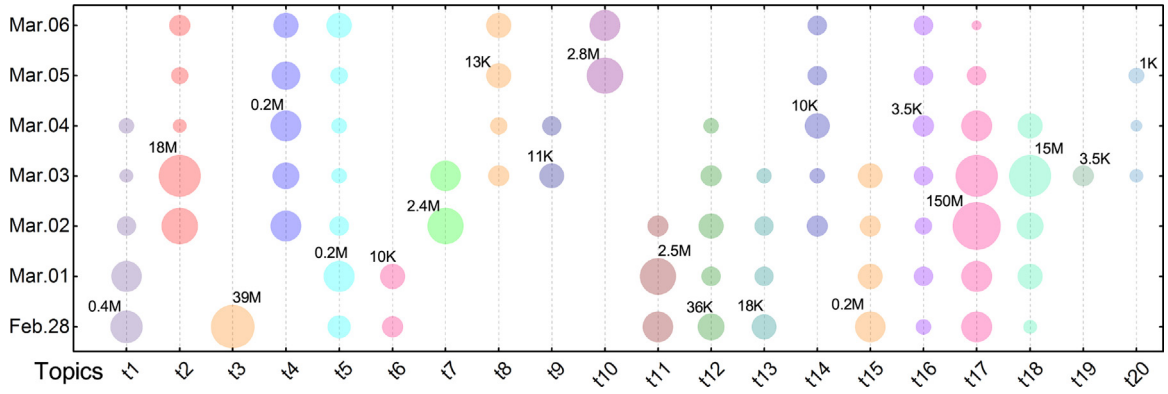



**Fig. 5.** Topic evolution with hotness.

the "Environmental Protection" category. The straightforward topics, e.g., *t5* and *t6*, burst in early period and achieve the highest hotness in the second day, and then gradually cool down. As time goes on, a topic that depicting and missing good environment (*t7*) arises. Then, the discussions extend to the topics of water resources protection (*t8*) and ecosystem protection (*t9*). At last, after several days discussions, a topic arise to call on peoples to do some significant things for environmental protection (*t10*). In the "Politics and Economy" category, we find that the topics discussing domestic things (*t11*, *t12* and *t13*) arise earlier and quickly achieve the highest hotness, and then cool down in general. In contrast, the topic of international relations (*t14*) arises a little later, and get hot increasingly and achieve generally the same hotness of domestic topics at last. In the "Criticism and Defense" category, the topics of denouncement arise earlier. In the early phase, the denouncements mainly focus on personal attack (*t15*), but then gradually turn to conspiracy theories (*t16*) about privatization advocator and traitor, which lasts the whole period of the seven days. With the denouncements, the defending topic (*t17*) is also continues seven days and achieves the highest hotness in the third day. Note that after the early personal attack, a reasonable querying topic (*t19*) arises in the forth day, which queries the documentary from scientificity and data authenticity. Meanwhile, the defending topic (*t20*) towards these querying arise in the same day and continues three days with increasingly trend.

### 5.3. Efficiency analysis

To analyze the efficiency of topic detection, we select LDA [12], the widely used probabilistic topic model, for comparison. For fairness, we using a distributed-version LDA which is implemented in Spark MLlib (Spark-LDA in short). Both the numbers of RDDs used in CP-Miner* and Spark-LDA are set to 64. We set the parameter $\tau_c = 0.8$ and $\tau_s = 0.02\%$ in CP-Miner*. We also set Spark-LDA to detect the same number of topics to PTDAS in each day. As can be seen from Fig. 6, PTDAS takes shorter time than Spark-LDA for topic detection in each day. Note that to detect topic in PTDAS, the execution time consists of both mining time and clustering time, where the later is mainly determined by the number of mined patterns. As shown in Fig. 6, clustering consumes acceptable time when setting a relative high cosine threshold 0.8. In some days, such as Mar.01, Mar.04, Mar.05 and Mar.06, the clustering time could even be ignored since the number of mined patterns is very small (shown in Fig. 7, the number is less than 3000). We also could find that the gap between PTDAS and Spark-LDA is the largest on Mar.01 whereas the gaps tend to be smaller on Mar.04-Mar.06. The reason is that the number of tweets on Mar.01 is the largest (1,048,105) whereas it tend to be smaller on Mar.04–06. The consuming time of Spark-LDA increases more quickly when the tweets number

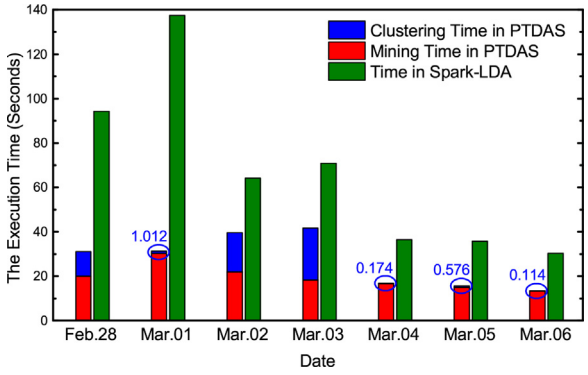



**Fig. 6.** Comparison on the efficiency with LDA.

increasing. However, the time increases slowly in PTDAS because of the sophisticated cosine pattern mining algorithm and its parallel implement. This also demonstrates the efficient of PTDAS when suffer massive data. In general, PTDAS could detect topics from hundreds of thousands of tweets in only tens of seconds. This order of magnitude in topic detecting could be seen as real-time detecting in practice.

We further study the impact of cosine interesting pattern on efficiency. By setting the parameter $\tau_c = 0$, cosine interesting pattern mining is degraded to frequent pattern mining (FPM in short), and we compare the performance of topic detection based on CP-Miner* and FPM. Without the restriction of cosine threshold, more loosely-knit terms are extracted as patterns, which leads to serious increasing of mining time and patterns number. Thus, we have to set a changeable parameter of $\tau_s$ in FPM to confine the mining time around 10 min. Fixing the parameters $\tau_c = 0.8$ and $\tau_s = 0.02\%$ in CP-Miner*, we take an overall comparison between CP-Miner* and FPM from five aspects, including support parameters, mining times, clustering times, number of patterns and number of topics. As shown in Fig. 7, the CP-Miner* based method performs better with smaller support parameter, shorter mining time and clustering time, fewer patterns, however, more topics. The reason is that constrained by cosine threshold, only closely-knit terms are extracted, which limits the number of patterns. Therefore we could set smaller support parameter to detect more topics with relative lower support. Meanwhile, the fewer patterns bring on shorter time in clustering, which is crucial for real-time detecting.

### 5.4. Sentimental analysis

In this section, we present experimental results on sentimental analysis. Since the sentiment of topics are calculated by the related patterns, we first evaluate the effectiveness of pattern sen-

Please cite this article in press as: L. Zhang, et al., A pattern-based topic detection and analysis system on Chinese tweets, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.08.016

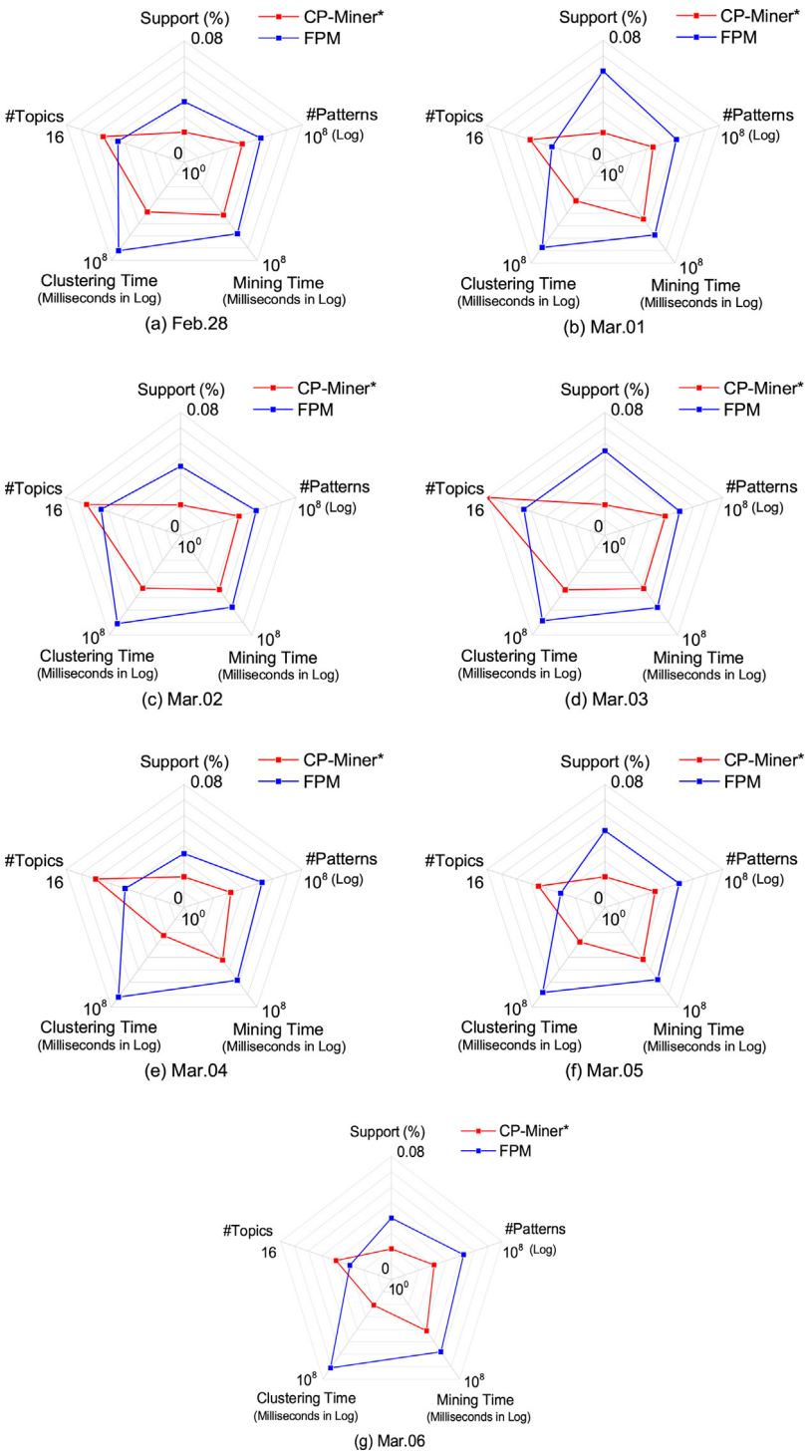



**Fig. 7.** Overall comparison between CP-Miner* and FPM.

timent calculation, and then present the sentiment of each topic with quantitative value.

Since the patterns have no ground-truth, to evaluate the sentimental calculation method, an indirect evaluation method is employed here. First, the quantitative sentimental value of each pattern is calculated using Eq. (7), and then ranging the patterns according to their values to form a descending sequence. Empirically, if our sentimental analysis method is effective, it must rank the highly likely positive patterns at the top and highly likely negative ones at the bottom. Finally, we use a supervised learning to classify likely positive patterns and likely negative ones, if the classification result is good, the pattern ranking based on the sentimental value is efficient. In what following, we consider the patterns from the top $x\%$ sequence to be positive (+) class while patterns from the bottom $x\%$ of the sequence to be negative (−) class. Here, we report for $x = 5\%$, 10%, and 15%, in which we induce three machine learning methods (SVM, J48, and Naive Bayes) and present 10-fold cross validation results with Precision ($P$), Recall ($R$) and F1-Score ($F1$) as metrics. We evaluate the effectiveness of pattern sentiment calculation in each day respectively. As shown in Table 4, all the classifiers perform well on the three test sets, and SVM and J48 slightly outperform Naive Bayes. As $x$ increases, a

Please cite this article in press as: L. Zhang, et al., A pattern-based topic detection and analysis system on Chinese tweets, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.08.016

**Table 4**
Indirect evaluation of sentiment calculation on patterns.

| Date | x% | SVM | | | J48 | | | Naive Bayes | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | P | R | F1 | P | R | F1 | P | R | F1 |
| Feb.28 | 5% | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| | 10% | 0.99 | 0.99 | 0.99 | 1.00 | 1.00 | 1.00 | 0.95 | 0.93 | 0.94 |
| | 15% | 0.96 | 0.95 | 0.96 | 0.98 | 0.97 | 0.98 | 0.93 | 0.92 | 0.93 |
| Mar.01 | 5% | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| | 10% | 0.99 | 0.98 | 0.98 | 1.00 | 1.00 | 1.00 | 0.94 | 0.93 | 0.94 |
| | 15% | 0.95 | 0.95 | 0.95 | 0.98 | 0.97 | 0.97 | 0.93 | 0.92 | 0.92 |
| Mar.02 | 5% | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| | 10% | 0.99 | 0.99 | 0.99 | 1.00 | 1.00 | 1.00 | 0.95 | 0.93 | 0.94 |
| | 15% | 0.96 | 0.96 | 0.96 | 0.98 | 0.98 | 0.98 | 0.93 | 0.93 | 0.93 |
| Mar.03 | 5% | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| | 10% | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0.97 | 0.96 | 0.96 |
| | 15% | 0.97 | 0.96 | 0.96 | 0.99 | 0.98 | 0.98 | 0.94 | 0.93 | 0.94 |
| Mar.04 | 5% | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| | 10% | 0.98 | 0.97 | 0.98 | 1.00 | 0.99 | 0.99 | 0.94 | 0.93 | 0.93 |
| | 15% | 0.94 | 0.94 | 0.94 | 0.97 | 0.96 | 0.96 | 0.92 | 0.91 | 0.92 |
| Mar.05 | 5% | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| | 10% | 0.99 | 0.98 | 0.98 | 1.00 | 0.99 | 1.00 | 0.94 | 0.93 | 0.94 |
| | 15% | 0.95 | 0.94 | 0.95 | 0.97 | 0.97 | 0.97 | 0.92 | 0.92 | 0.92 |
| Mar.06 | 5% | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| | 10% | 0.98 | 0.97 | 0.97 | 0.99 | 0.99 | 0.99 | 0.93 | 0.93 | 0.93 |
| | 15% | 0.94 | 0.93 | 0.93 | 0.97 | 0.96 | 0.96 | 0.92 | 0.91 | 0.92 |

monotonic degradation in classification performance is also manifest. This observation implies that if the top (positive) and bottom (negative) patterns get closer, the classifiers are hard to distinguish, which is within the bounds of reason.

Based on the sentimental values of patterns, the sentiment polarity of each topic could be calculated using Eq. (8). We present the sentiment polarity of each topic in Table 5, with the evolution of quantitative sentimental values from Feb.28 to Mar.06. The experimental results are generally consistent with our intuition according to the sematics of each topic. The only one exception is topic *t9* that proposing to protect wild animals, which should be positive in our intuition but negative in the experiment. This is due to that the topic contains lots of counter-examples, including wearing fur, hunting or maltreating animals, and so on. For the sentimental evolution, most topics maintain a smooth sentimental value sequence in their lifecycle with little jitter. However, three topics (*t1*, *t11*, *t15*) trend to neutral, either from positive or negative. For topic *t1*, the reason is that the denouncements make people re-examine the documentary. For topic *t11*, the strength of negativeness decreasing because that the documentary is not banned in fact and the worry disappears. For topic *t15*, the reason is that topic *t18* denounces this behaviour and forces it more reasonable.

## 6. Related work

This work focuses on detecting topics from texts produced within social media platforms (e.g., Weibo), with evolving and sentimental analysis on the detected topics.

In the literature, existing methods for topic detecting can be classified into two main categories, in terms of the representation of

**Table 5**
Sentimental analysis and evolution on topics.

| ID | Topic | Sentimental polarity | Sentimental evolution | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | Feb.28 | Mar.01 | Mar.02 | Mar.03 | Mar.04 | Mar.05 | Mar.06 |
| ***t1*** | **Deeply Touched by the Responsibility** | **(+)** | **0.98** | **0.90** | **0.88** | **0.83** | **0.76** | **N/A** | **N/A** |
| *t2* | Information Propagation Ways | (+) | N/A | N/A | 0.62 | 0.64 | 0.66 | 0.63 | 0.67 |
| *t3* | Motivation of this Documentary | (−) | 0.33 | N/A | N/A | N/A | N/A | N/A | N/A |
| *t4* | The Epitome of Social Change | (+) | N/A | N/A | 0.58 | 0.60 | 0.61 | 0.59 | 0.56 |
| *t5* | Protecting Environment for the Future | (+) | 0.88 | 0.86 | 0.86 | 0.89 | 0.90 | 0.88 | 0.87 |
| *t6* | Environmental Deterioration in China | (−) | 0.11 | 0.09 | N/A | N/A | N/A | N/A | N/A |
| *t7* | Depicting and Missing Good Environment | (+) | N/A | N/A | 0.97 | 0.97 | N/A | N/A | N/A |
| *t8* | Protection of the Mother River | (+) | N/A | N/A | N/A | 0.84 | 0.82 | 0.88 | 0.84 |
| *t9* | Protection of Wild Animals | (−) | N/A | N/A | N/A | 0.33 | 0.35 | N/A | N/A |
| *t10* | Do sth for Environmental Protection | (+) | N/A | N/A | N/A | N/A | N/A | 0.92 | 0.92 |
| ***t11*** | **Hitting the Interests of Some Groups** | **(−)** | **0.22** | **0.31** | **0.44** | **N/A** | **N/A** | **N/A** | **N/A** |
| *t12* | China's Political System | (−) | 0.29 | 0.32 | 0.29 | 0.30 | 0.33 | N/A | N/A |
| *t13* | Balance between Economy and Environment | (−) | 0.44 | 0.42 | 0.40 | 0.44 | N/A | N/A | N/A |
| *t14* | International Relations | (−) | N/A | N/A | 0.38 | 0.37 | 0.36 | 0.36 | 0.39 |
| ***t15*** | **Denouncing Chaijing** | **(−)** | **0.11** | **0.22** | **0.31** | **0.36** | **N/A** | **N/A** | **N/A** |
| *t16* | Denouncing Privatization and Betrayal | (−) | 0.12 | 0.09 | 0.11 | 0.13 | 0.10 | 0.08 | 0.09 |
| *t17* | Defending for Chaijing | (+) | 0.77 | 0.78 | 0.74 | 0.76 | 0.75 | 0.78 | 0.73 |
| *t18* | Ugly Chinese | (−) | 0.23 | 0.25 | 0.25 | 0.27 | 0.29 | N/A | N/A |
| *t19* | Querying Authenticity and Scientificity | (−) | N/A | N/A | N/A | 0.44 | N/A | N/A | N/A |
| *t20* | Defending for Authenticity and Scientificity | (+) | N/A | N/A | N/A | 0.56 | 0.61 | 0.60 | N/A |

The symbol (+) denotes Positive and (−) denotes Negative.

Please cite this article in press as: L. Zhang, et al., A pattern-based topic detection and analysis system on Chinese tweets, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.08.016

topics [7]. The first category, referred to as *document-pivot* method, clusters documents into a number of groups and each group represents a topic [8,9]. In these approaches, documents are clustered by leveraging some similarity metric between them. Generally, textual similarities (e.g., *tf-idf*) is the mostly used metrics [8,34,35], however, some dimensions other than text can also be used to improve the clustering quality, such as temporal proximity [9]. The other category, referred to as *feature-pivot* method, groups together terms according to their cooccurrence patterns and thus a set of keywords represents a topic [36,10]. The topic models in NLP, such as LDA [12], can be reviewed as feature-pivot approaches since they extract sets of terms to represent topics. Besides, some approaches trying to capture topics through the detection of keyword burstiness [37,38,36], which identify bursty terms first and then cluster them together to produce topic definitions. Another important type of methods in feature-pivot category are pattern-based approaches [10,13], which discover simultaneous cooccuurrence patterns containing more than two terms and then pick out relevant yet diverse patterns to represent topics. The topic detection method in PTDAS is also a pattern-based approach, which employs a more effective measurement *cosine* instead of traditional measurements such as *support* and *lift*. Based on these topic detection algorithms, various systems have been designed and implemented for extracting topics from social media, such as TweetMotif [34], TweetStand [9], TopicSketch [39], TweetXplorer [40] and TEDAS [41]. All of these systems are end-to-end tools which (1) contain integrated functions for data acquisition and pre-processing; (2) perform topic detection and analysis; (3) incorporate techniques for meaningful visualization of the results. Most of them are using Twitter as the input data stream and designed for English, however, little effort is done in the Chinese scenario, as the processing and analysis on Chinese text are more specific and complex than English. Moreover, most of the existing systems concern about the accuracy. However, how to improve the efficiency by means of the distributed computation frameworks, e.g. `Spark`, receives little attention.

Existing work relevant to the topic evolution mainly based on dynamic topic models [42,43] and evolutionary clustering [44–46]. A dynamic topic model like dynamic LDA [42] allows the documents to be streaming or time-stamped, and mathematically models the topic generation process with more parameters. The evolutionary clustering processes temporal data to produce a sequence of clusters, where each cluster in the sequence is similar to the cluster at the previous time step, and should accurately reflect the data arriving. When applied to topic tracking, the clusters sequence could be used to represent the evolution of topics [47,48]. However, both these two categories of methods perform on the document-level and need to consider the historical information, thus suffer from serious challenge in efficiency. The sentimental analysis on text is also extensively studied [49,32]. The mainstream methods calculate the sentiment of texts based on the semantic orientation of words, which is usually obtained from domain-dependent corpus in the early phase [50–52], and later combined with the semantic association from some corpus dictionaries (e.g., `WordNet` and `HowNet`) [53]. Besides, some work model sentimental analysis as two-class classification problem and utilize classifiers to determine sentimental polarities [54].

To sum up, while many research efforts exist for topic detecting, evolving and sentimental analysis, further study is still needed for high efficiency detecting and analyzing in an integrated framework. Our work attempts to fill this crucial void by conducting a system that utilizes cosine interesting patterns to detect topics, and further analyze the evolution and sentiment on these pattern-represented topics.

## 7. Conclusion

This paper presents a system called PTDAS to detect and analyze topics from Sina Weibo, a Twitter-like platform in China. The key component of PTDAS is the pattern-based method for topic detecting, which employs a FP-growth-like algorithm CP-Miner to extract cosine interesting patterns, and further summarizes them into topics by hierarchical clustering. Specially, to meet the efficiency challenges for real-time detecting, we parallelize CP-Miner for distributed computing on `Spark`. Along with mining patterns for topic detection, some analytic techniques including both topic evolving analysis and sentimental analysis are also proposed, which could bring better understanding on detected topics. Experiments on real-world data set demonstrate the effectiveness and efficiency of PTDAS. In the future work, we will extend the crawler to collect data from more platforms and apply PTDAS to multiple social media, including Facebook, Wechat and so on.

## Acknowledgments

This research was partially supported by the National Key Research and Development Program of China under Grant 2016YFB1000901, the National Natural Science Foundation of China under Grants 61672433, 91646204, 61502222, 71571093 and 71372188, the National Center for International Joint Research on E-Business Information Processing under Grant 2013B01035, the Industry Projects in Jiangsu S&T Pillar Program under Grant BE2014141, the Surface Projects of Natural Science Research in Jiangsu Provincial Colleges and Universities under Grants 15KJB520012 and 15KJB520011, and the Pre-Research Project of Nanjing University of Finance and Economics under Grant YYJ201415.

## References

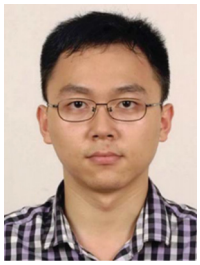
[1] B. O'Connor, R. Balasubramanyan, B.R. Routledge, N.A. Smith, From tweets to polls: linking text sentiment to public opinion time series, ICWSM 11 (122–129) (2010) 1–2.
[2] D.M. Scott, The New Rules of Marketing and PR: How to Use Social Media, Online Video, Mobile Applications, Blogs, News Releases, and Viral Marketing to Reach Buyers Directly, John Wiley & Sons, 2015.
[3] L. Carter, J.B. Thatcher, R. Wright, Social media and emergency management: exploring state and local tweets, in: 47th Hawaii International Conference on System Sciences, IEEE, 2014, pp. 1968–1977.
[4] X. Liu, M. Wang, B. Huet, Event analysis in social multimedia: a survey, Front. Comput. Sci. 10 (3) (2016) 433–446.
[5] Z. Wang, L. Shou, K. Chen, G. Chen, S. Mehrotra, On summarization and timeline generation for evolutionary tweet streams, IEEE Trans. Knowl. Data Eng. 27 (5) (2015) 1301–1315.
[6] U. Westermann, R. Jain, Toward a common event model for multimedia applications, IEEE MultiMed. 14 (1) (2007) 19–29.
[7] F. Atefeh, W. Khreich, A survey of techniques for event detection in twitter, Comput. Intell. 31 (1) (2015) 132–164.
[8] S. Phuvipadawat, T. Murata, Breaking news detection and tracking in twitter, in: IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT), vol. 3, IEEE, 2010, pp. 120–123.
[9] J. Sankaranarayanan, H. Samet, B.E. Teitler, M.D. Lieberman, J. Sperling, Twitterstand: news in tweets, in: Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, ACM, 2009, pp. 42–51.
[10] L.M. Aiello, G. Petkos, C.J. Martín, D. Corney, S. Papadopoulos, R. Skraba, A. Göker, I. Kompatsiaris, A. Jaimes, Sensing trending topics in twitter, IEEE Trans. Multimed. 15 (6) (2013) 1268–1282.
[11] G.P.C. Fung, J.X. Yu, P.S. Yu, H. Lu, Parameter free bursty events detection in text streams, Proceedings of the 31st International Conference on Very Large Data Bases, VLDB Endowment (2005) 181–192.
[12] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent Dirichlet allocation, J. Mach. Learn. Res. 3 (2003) 993–1022.
[13] G. Petkos, S. Papadopoulos, L.M. Aiello, R. Skraba, Y. Kompatsiaris, A soft frequent pattern mining approach for textual topic detection, in: 4th International Conference on Web Intelligence, Mining and Semantics (WIMS 14), WIMS '14, Thessaloniki, Greece, June 2–4, 2014, 2014, pp. 25:1–25:10.
[14] L. Geng, H.J. Hamilton, Interestingness measures for data mining: a survey, ACM Comput. Surv. 38 (3) (2006).

Please cite this article in press as: L. Zhang, et al., A pattern-based topic detection and analysis system on Chinese tweets, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.08.016

[15] M. Zaharia, M. Chowdhury, T. Das, A. Dave, J. Ma, M. McCauley, M.J. Franklin, S. Shenker, I. Stoica, Resilient distributed datasets: a fault-tolerant abstraction for in-memory cluster computing, in: Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation, USENIX Association, 2012, pp. 2.
[16] L. Zhang, Z. Bu, Z. Wu, J. Cao, Dgwc: distributed and generic web crawler for online information extraction, in: International Conference on Behavioral, Economic and Socio-cultural Computing (BESC), IEEE, 2016, pp. 1–6.
[17] L. Zhang, X. Wang, Z. Bu, J. Cao, Z. Wu, Online public opinion system: design and applications, in: The Fourth International Conference on Advanced Cloud and Big Data, IEEE, 2016, pp. 75–80.
[18] G. Salton, C. Buckley, Term-weighting approaches in automatic text retrieval, Inf. Process. Manag. 24 (5) (1988) 513–523.
[19] R. Mihalcea, P. Tarau, Textrank: Bringing Order into Texts, Association for Computational Linguistics, 2004.
[20] L. Geng, H.J. Hamilton, Interestingness measures for data mining: a survey, ACM Comput. Surv. 38 (3) (2006) 9.
[21] Y. Zhao, G. Karypis, Criterion functions for document clustering: experiments and analysis, Tech. rep., Citeseer, 2001.
[22] R.J. Bayardo, Y. Ma, R. Srikant, Scaling up all pairs similarity search, in: Proceedings of the 16th International Conference on World Wide Web, ACM, 2007, pp. 131–140.
[23] R. Bellmann, Dynamic Programming, Princeton University Press, 1957.
[24] J. Wu, S. Zhu, H. Liu, G. Xia, Cosine interesting pattern discovery, Inf. Sci. 184 (1) (2012) 176–195.
[25] J. Cao, Z. Wu, J. Wu, Scaling up cosine interesting pattern discovery: a depth-first method, Inf. Sci. 266 (2014) 31–46.
[26] Z. Wu, J. Cao, J. Wu, Y. Wang, C. Liu, Detecting genuine communities from large-scale social networks: a pattern-based method, Comput. J. (2013) bxt113.
[27] J. Han, J. Pei, Y. Yin, Mining frequent patterns without candidate generation, in: ACM Sigmod Record, vol. 29, ACM, 2000, pp. 1–12.
[28] G. Grahne, J. Zhu, Mining frequent itemsets from secondary memory, Proc. of ICDM (2004) 91–98.
[29] M.E. Newman, M. Girvan, Finding and evaluating community structure in networks, Phys. Rev. E 69 (2) (2004) 026113.
[30] M.E. Newman, Analysis of weighted networks, Phys. Rev. E 70 (5) (2004) 056131.
[31] P.D. Turney, M.L. Littman, Measuring praise and criticism: inference of semantic orientation from association, ACM Trans. Inf. Syst. 21 (4) (2003) 315–346.
[32] Z. Bu, H. Li, J. Cao, Z. Wu, L. Zhang, Game theory based emotional evolution analysis for Chinese online reviews, Knowl. Based Syst. 103 (2016) 60–72.
[33] P.D. Turney, Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews, in: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics, 2002, pp. 417–424.
[34] B. O'Connor, M. Krieger, D. Ahn, Tweetmotif: exploratory search and topic summarization for twitter, in: ICWSM, 2010.
[35] H. Becker, M. Naaman, L. Gravano, Beyond trending topics: real-world event identification on twitter, ICWSM, 11 (2011) 438–441.
[36] J. Lehmann, B. Gonçalves, J.J. Ramasco, C. Cattuto, Dynamical classes of collective attention in twitter, in: Proceedings of the 21st International Conference on World Wide Web, ACM, 2012, pp. 251–260.
[37] D.A. Shamma, L. Kennedy, E.F. Churchill, Peaks and persistence: modeling the shape of microblog conversations, in: Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work, ACM, 2011, pp. 355–358.
[38] J. Yang, J. Leskovec, Patterns of temporal variation in online media, in: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, ACM, 2011, pp. 177–186.
[39] W. Xie, F. Zhu, J. Jiang, E.-P. Lim, K. Wang, Topicsketch: real-time bursty topic detection from twitter, IEEE Trans. Knowl. Data Eng. 28 (8) (2016) 2216–2229.
[40] F. Morstatter, S. Kumar, H. Liu, R. Maciejewski, Understanding twitter data with tweetxplorer, Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM (2013) 1482–1485.
[41] R. Li, K.H. Lei, R. Khadiwala, K.C.-C. Chang, Tedas: a twitter-based event detection and analysis system, in: IEEE 28th International Conference on Data Engineering (ICDE), IEEE, 2012, pp. 1273–1276.
[42] D.M. Blei, J.D. Lafferty, Dynamic topic models, in: Proceedings of the 23rd International Conference on Machine Learning, ACM, 2006, pp. 113–120.
[43] C. Wang, D. Blei, D. Heckerman, Continuous time dynamic topic models, 2012 arXiv:1206.3298.
[44] D. Chakrabarti, R. Kumar, A. Tomkins, Evolutionary clustering, in: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2006, pp. 554–560.
[45] M.C. Naldi, R.J.G.B. Campello, Comparison of distributed evolutionary k-means clustering algorithms, Neurocomputing 163 (2015) 78–93.
[46] A. Mukhopadhyay, U. Maulik, S. Bandyopadhyay, A survey of multiobjective evolutionary clustering, ACM Comput. Surv. 47 (4) (2015) 61.
[47] J. Zhang, Y. Song, C. Zhang, S. Liu, Evolutionary hierarchical dirichlet processes for multiple correlated time-varying corpora, in: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2010, pp. 1079–1088.
[48] W. Cui, S. Liu, L. Tan, C. Shi, Y. Song, Z. Gao, H. Qu, X. Tong, Textflow: towards better understanding of evolving topics in text, IEEE Trans. Vis. Comput. Graph. 17 (12) (2011) 2412–2421.
[49] Z. Zhai, H. Xu, P. Jia, An empirical study of unsupervised sentiment classification of Chinese reviews, Tsinghua Sci. Technol. 15 (6) (2010) 702–708.
[50] V. Hatzivassiloglou, K.R. McKeown, Predicting the semantic orientation of adjectives, in: Proceedings of the Eighth Conference on European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, 1997, pp. 174–181.
[51] E. Riloff, J. Wiebe, Learning extraction patterns for subjective expressions, in: Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2003, pp. 105–112.
[52] S. Poria, E. Cambria, G. Winterstein, G.-B. Huang, Sentic patterns: dependency-based rules for concept-level sentiment analysis, Knowl. Based Syst. 69 (2014) 45–63.
[53] J. Kamps, M. Marx, R.J. Mokken, M. de Rijke, Words with Attitude, Citeseer, 2001.
[54] B. Liu, M. Hu, J. Cheng, Opinion observer: analyzing and comparing opinions on the web, in: Proceedings of the 14th International Conference on World Wide Web, ACM, 2005, pp. 342–351.

**Lu Zhang** received the Ph.D. degree in computer science from Southeast University, Nanjing, China, in 2012. He is currently a Lecturer with the Jiangsu Provincial Key Laboratory of E-Business, Nanjing University of Finance and Economics, Nanjing. His current research interests include data mining, social network analysis, and recommender systems.

**Zhiang Wu** received the Ph.D. degree in computer science from Southeast University, Nanjing, China, in 2009. He is currently an Associate Professor with the Jiangsu Provincial Key Laboratory of E-Business, Nanjing University of Finance and Economics, Nanjing. His current research interests include distributed computing, social network analysis, and data mining.



**Zhan Bu** received the Ph.D. degree in computer science from Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2013. He is currently an Associate Professor with the Jiangsu Provincial Key Laboratory of E-Business, Nanjing University of Finance and Economics, Nanjing. His current research interests include distributed computing, social network analysis, and data mining.

**Ye Jiang** received the Ph.D degree in computer science from the Nan Jing University of Science and Technology, in 2012. He is currently an Associate Professor with the Jiangsu Provincial Key Laboratory of E-Business, Nanjing University of Finance and Economics, Nanjing. His current research interests include data mining, pattern recognition, and machine learning.

**Jie Cao** received the Ph.D degree in computer science from the Nan Jing University of Science and Technology, in 2002. He is currently an Professor with the Jiangsu Provincial Key Laboratory of E-Business, Nanjing University of Finance and Economics, Nanjing. His current research interests include data mining, social network analysis, and machine learning

Please cite this article in press as: L. Zhang, et al., A pattern-based topic detection and analysis system on Chinese tweets, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.08.016