Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/070619

A Topic Detection Approach Through Hierarchical Clustering on Concept Graph

Xiaohui Huang[1,3], *Xiaofeng Zhang*[1,3], *Yunming Ye* [1,3,*], *Shengchun Deng*[2] *and Xutao Li* [1,3]

[1] Shenzhen Graduate School, Harbin Institute of Technology, Shenzhen 518055, China
[2] School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China
[3] Shenzhen Key Laboratory of Internet Information Collaboration, Shenzhen 518055, China

Received: 24 Feb. 2013, Revised: 22 Jun. 2013, Accepted: 23 Jun. 2013
Published online: 1 Nov. 2013

Abstract: Topic detection and tracking (TDT) algorithms have long been developed for the discovery of topics. However, most existing TDT algorithms suffer from paying less attention to: (1) temporal distance between a pair of topics; (2) the mutual effect between highly correlated topic terms. In this paper, we proposed a novel topic detection approach by applying hierarchical clustering on the constructed concept graph (HCCG), which is able to solve aforementioned shortcomings simultaneously. In this approach, the concept is first defined as well as the concept behavior curve. Then, the temporal graph is constructed with concept as vertexes and connected by the edges sharing the same topic terms. By performing hierarchical clustering on this concept graph, the highly correlated concept behavior curves will be grouped together as topics. The proposed approach is evaluated on a number of datasets and the promising experimental results show that our approach is superior to K-means, agglomerative hierarchical clustering algorithm(AGH), and LDA with respects to precision, recall and F-measure. Moreover, the proposed concept behavior curves can be used to track the topic change trend by monitoring on the peak frequency of the concept behavior curves.

Keywords: Topic detection, concept graph, hierarchical clustering, text clustering.

1 Introduction

Topic detection and tracking (TDT) [1–5] is one of the most important techniques to perform knowledge discovery task on the tremendous collection of web pages or microblogs posted by billions of social networks, such as twitter and facebook, users. TDT technique is originally designed to detect emerging topics from streaming broadcast news by calculating similarity between incoming news and existing topics. However, there are two main shortcomings in most of existing TDT algorithms: (1) documents are represented by bag-of-words which is insensitive to contextual sequence of co-occurent terms; (2) only contextual distance between topics is consider, but rarely consider the temporal distance between topics.

In the past, researchers have worked on these issues in an independent manner. In papers [6, 7], authors replaced single keyword in the feature vector with a small set of keywords which helps to capture the specific meaning of the co-occurrent terms. However, the mapping relationship between topics and a set of keywords needs to perform a careful study. Several approaches [8, 9] were proposed to consider the topic's temporal distance. LDA [10] was shown as one of the best topic model and a temporal LDA was proposed in [9] to take into account the temporal distance of documents. In their work, the Beta distribution of the topic over a (normalized) time interval was introduced into the original LDA model. Alternatively, researchers proposed the graph based approaches [11–14] in which the collection of streaming documents were extracted and represented by a graph, then the topic detection was performed by discovering the densely connected terms in the graph. However, the clustering process performed on the document-term graph requires a lot of computational resources.

Inspired by the graph model [11], the approach called as hierarchical clustering on the concept graph (HCCG) was proposed in this paper. We believe that the co-occurrent keywords can uniquely represent certain perspective of a topic, and these perspectives were then defined as concepts of topic. Obviously, concepts from

* Corresponding author e-mail: yeyunming@hit.edu.cn

the same topic would have similar change trends. Therefore, topics could be detected by discovering highly correlated concept change trends in the graph. Such change trends are defined as concept behavior curves in this paper. Then, the main task of the proposed HCCG was to group highly correlated concept behavior curves as the underlying topics. The main contributions of this paper are:

–To the best of our knowledge, it is the first attempt to detect topics by considering topics' temporal distance and represent topics with co-occurrent keywords;
–We defined concept frequency, concept behavior curve as well as the formulation to compute the similarity between concepts;
–We speeded up the AGH by using revised single pass algorithm, and the complexity of the proposed approach is largely reduced which makes the approach a scalable one.

The rest of the paper is organized as follows. Section 2 reviews related works. Problem is formulated in section 3 as well as the definition of concept, concept frequency and concept behavior curve. The construction of concept graph is given in section 4. The proposed HCCG approach is described in section 5. The approach is evaluated in section 6 and section 7 concludes the paper.

2 Related Work

Topic detection and tracking generally involves five tasks: story segmentation, first story detection, story link detection, topic detection and topic tracking [1]. Among these tasks, topic detection (TD) was paid more attention as it can automatically find new topics from the streaming data [15]. To assign incoming documents to existing topics or to create a new topic, the contextual similarity between document and topics were calculated using the cosine distance of the feature vectors [16–18]. However, the single keyword extracted for the feature vector conveys different semantic meanings [19] with the contextual environment changes. To overcome this issue, Zhou et al. proposed a context-sensitive smoothing method [20] which decreased the weight of observed keywords and assigned weights to unobserved terms to increase the discriminative ability of missing keywords. Swan and Allan [4] proposed a statistical model of term occurrence over time with the focus on studying the influence of term frequency at different time slot. External knowledge, such as ontology and Wikipedia, was used by researchers for document clustering [17, 21, 22], which increased the weight of keywords by taking the effect of their synonyms into account.

To explore the temporal characteristics of topic, most of existing works [23, 24] utilized the timestamps of documents in such a way that documents within the same time interval were assigned with higher weights to be grouped into the same topic. Recently, generative probability models [25, 26], such as latent dirichlet allocation model [10], became a main research stream in topic detection. There were many studies on online topic detection based on generative models [27–29]. Alternatively, [11–13] constructed a graph and used the community detection algorithms to detect topics. In their approach, keywords were treated as the vertexes of the graph, and each keyword was assigned to only one topic. As keywords were not necessarily to confine themselves to only one topic, the model performance inevitably deteriorates due to this assumption. Holz and Teresniak [8] argued that keywords can represent the meaning of topic and then they defined keywords' volatility as its' temporal fluctuation in the global contextual environment (i.e., the keyword and its neighboring keywords' fluctuation). However, topic naturally contains more than one keyword which limits the performance of this approach. Inspired by the spirit of this concept graph approach, we, in this paper, proposed a novel topic detection approach to discover the topics through hierarchical clustering on the constructed concept graph.

3 Problem Formulation

Given a document set $\mathrm{D} = \{\mathrm{D}^1, \mathrm{D}^2, ..., \mathrm{D}^\mathrm{T}\}$ over time interval T where D^t denotes the document set extracted at time $t (t \in [1, T])$. Let D_i^t denote the i^{th} document in set D^t, $W_i^t = \{w_1, w_2, ..., w_n\}$ denote the set of terms extracted from document D_i^t by applying the typical pre-processing method, Γ_i, Θ_j denote a topic and a concept, respectively.

Without loss of generality, each document D_i^t contains at least one topic Γ_i^t, and the aggregation of such topics forms the topic set of the document set D^t. Documents with the same timestamp tend to have similar topics and a topic consists of at least one keyword or a phrase. Topic can have several subtopics or different aspects. In this paper, we think of these different aspects as concepts. Definitions of concept and its characteristics are given as follows.

Definition 3.1. Concept Θ_s describes different aspects of a topic, and is composed of τ_s terms, i.e.

$$\Theta_s = \{w_1, w_2, ..., w_{\tau_s}\}, where\ \tau_s \geq 2. \quad (1)$$

From the definition, it can be seen that concept is used to describe a topic. The combination of several keywords enables concept a rich semantic meaning. Therefore, at least two keywords are used to represent a concept. To measure the importance of a concept, similar to term-frequency, the concept frequency is defined as the co-occurrence of keywords forming the concept, which is written as:

$$cf_{s,d} = \min(tf_{w_1}, tf_{w_2}, ..., tf_{w_{\tau_s}}) \quad (2)$$

where $cf_{s,d}$ is the frequency of concept Θ_s contained in the document d, w_i is the i^{th} keyword of concept Θ_s and tf_{w_i} is the term frequency of the i^{th} keyword. Based on Equation 2, the concept frequency of a document set D^t can then be defined as

$$cf_{s,t} = \frac{1}{|D^t_{term}|} \sum_{d=1}^{D^t} cf_{s,d} \tag{3}$$

where $cf_{s,t}$ is called as document set concept frequency, the total co-occurrenct keywords over document set D^t is normalized by $|D^t_{term}|$, and $|D^t_{term}|$ is the total number of distinct terms in D^t. Hereinafter, we reuse concept frequency to refer it as document set's concept frequency. With the concept frequency, the complete document set D^t can be represented as

$$D = \begin{pmatrix} cf_{1,1} & cf_{1,2} & \dots & cf_{1,T} \\ cf_{2,1} & cf_{2,2} & \dots & cf_{2,T} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ cf_{S,1} & cf_{S,2} & \dots & cf_{S,T} \end{pmatrix}_{concept \times time} \tag{4}$$

with each row is a concept Θ_s and each column represents time t.

Definition 3.2. Concept Behavior Curve is defined as the changing trend of the concept frequency over time, denoted by B_{Θ_s}. B_{Θ_s} is represented by the concept frequency vector at different time t, written as

$$B_{\Theta_s} = \{\mathrm{cf}_{s,1}, \mathrm{cf}_{s,2}, \dots, \mathrm{cf}_{s,T}\}. \tag{5}$$

Note that if concept Θ_s does not appear in document set D^t, then $cf_{s,t}$ is set to zero. The proposed concept behavior curve depicts the dynamic characteristics of the concept frequency along the timeline. Concept does not necessarily exist in all time interval. If the concept frequency is greater than 0, it means the emergence of the concept. When the concept frequency curve reaches its highest value, then it means the concept becomes very important among all concepts extracted within that time interval. If the concept frequency curve becomes zero, then it means that the concept vanishes at that time. Therefore, similar concepts have similar changing trends, i.e., concept behavior curves, which could be used to detect the topics by finding the clusters of similar curves.

4 Generation of the Concept Graph

4.1 Concept Feature Vector

In this paper, a concept feature vector, instead of "bag-of-words", is proposed to represent a document. On one hand, the dimension of concept feature vector is far lower than that of document term feature vector, which will largely save the computational cost. On the other hand, concept generally has an invariant meaning, whereas terms of "bag-of-words" might represent different meanings if the contextual environment is changed.

Now the concept is the combination of several keywords and its weight is calculated as

$$\phi_{\Theta_s} = \frac{\tau_s * |w_1 \cap w_2 \cap \dots \cap w_{\tau_s}|}{|w_1| + |w_2| + \dots + |w_{\tau_s}|} \tag{6}$$

where $w_1, w_2, \dots, w_{\tau_s}$ are the keywords of the concept Θ_s; $|w_1 \cap w_2 \cap \dots \cap w_{\tau_s}|$ represents the co-occurrence of $w_1, w_2, \dots, w_{\tau_s}$ and $|w_i|$ represents the occurrence of w_i and factor τ_s is introduced to normalize the maximum weight.

As concept generally contains several keywords, the combination of these keywords is quite high, which in turn requires an extremely high computational cost. Thus for simplicity reason, only two keywords per concept are extracted in this paper. Concept Θ_s can be rewrote as $\Theta(w_1, w_2)$.

4.2 Concept Graph

Concept graph G is denoted as $G = \{\{\Theta_s\}, \{\rho(\Theta_i, \Theta_j)\}\}$ with Θ_s as its vertex and $\rho(\Theta_i, \Theta_j)$ is the weighted edge. Edge connects two concepts sharing same keywords, and the weight of edge is calculated as the similarity between two concepts. The concept graph can be constructed on document set D, and thus is able to be used to detect underlying topics. This concept graph can be easily extended to an online version by re-computing the weighted edge of D^{T+1} to track the topic change. One illustrating example of concept graph is given as Figure 2. In this figure, concepts such as "bird_flu" and "chicken_flu" under topic "red(*)" are plotted as vertexes, and an edge is pointing from vertex "bird_flu" to vertex "chicken_flu". The corresponding concept behavior curve is shown in Figure 1. Concept "bird_flu" and "chicken_flu" are plotted in red line as they belong to topic "Red(*)". It can be seen that the red curves have similar changing trends over time. However, it is observed that the red behavior curves differentiate a lot from blue or green behavior curves which represent different concepts from other topics. This property of concept behavior curve can be used to detect topics on the concept graph by discovering densely connected subgraphs formed by the highly correlated concepts.

The weight of the edge $\rho(\Theta_i, \Theta_j)$ can be calculated using the Pearson correlation coefficient as

$$\rho(\Theta_i, \Theta_j) = \frac{\sum_{t=1}^{T} (cf_{i,t} - \mu_i)(cf_{j,t} - \mu_j)}{\sigma_i \sigma_j} \tag{7}$$

where (μ_i, σ_i), (μ_j, σ_j) represents the mean and variance of the concept frequency Θ_i, Θ_j, respectively. With concepts as vertexes and $\rho(\Theta_i, \Theta_j)$ as the weight of the edge, the concept graph could be extracted from the document set D.

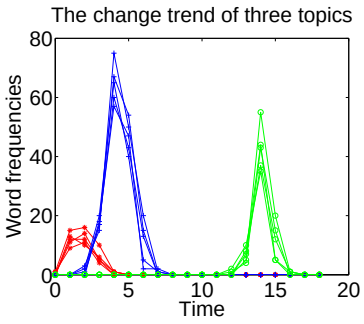


Fig. 1: Example of concept behavior curve. Red(*), blue(+) and green(o) curve represents the topic "Red(*)", "Blue(+)" and respectively.

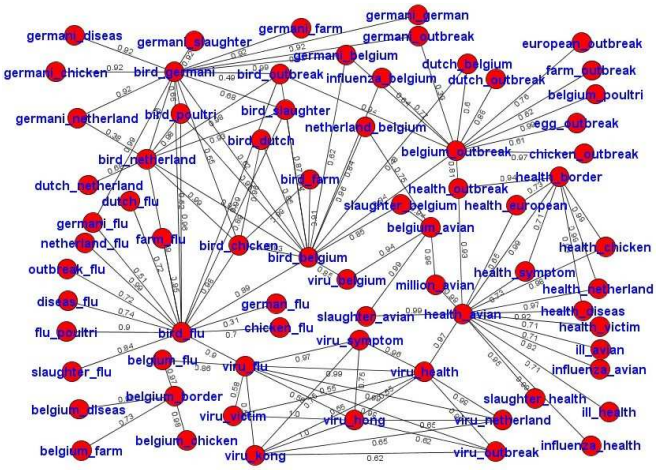


Fig. 2: An illustrating example of concept graph extracted from TDT5.

5 The Hierarchical Clustering Approach on Concept Graph

Given the concept graph, the common practice to find densely-connected subgraphs (topics) is to seek the help of certain graph partition algorithms. However, the graph partition algorithms, such as Girvan-Newman algorithm [30], generally have a higher computational complexity, specially when the document set is huge. Therefore, a revised agglomerative hierarchical clustering algorithm (AGH) is proposed to perform clustering task on the concept graph. The complexity of the original AGH [18] is quite high as it computes the pair-wise distance between items and only merges two nearest data items into one per iteration. Some researchers [31] tries to speed up the original AGH, for example, Murtagh [31] merged more items at each step. In this paper, the single pass algorithm [32] is proposed to perform pre-cluster on concept graph, and then revised AGH algorithm is performed the pre-cluster results

Algorithm 1 Revised single pass algorithm

Input: G={D,E},β
Output: $c_1, c_2, c_3, ..., c_\kappa$
for $i = 1$ **to** $|D|$ **do**
  Initialize Max=1
  for $j = 1$ **to** κ **do**
    if $\Delta(\Theta_i, c_j) > \Delta(\Theta_i, c_{\max})$ **then**
      max=j
    end if
  end for
  if $\Delta(\Theta_i, c_{\max}) > \beta$ **then**
    Add Θ_i to c_{max}
    else create a new cluster for Θ_i, κ++
  end if
end for

5.1 Pre-clustering on concept graph

As mentioned above, the clustering results of the single pass algorithm is used to initialize AGH. First, the SP algorithm is briefly introduced as follows. In the algorithm, similarity between each incoming data item and current clusters is computed. Then the cluster having the maximum similarity value is selected. If the similarity is greater than the pre-defined threshold, then the data item is merged to the current cluster, otherwise, a new cluster is created. However, to apply the SP algorithm, the computation of similarity between concepts and document should be redefined, which is

$$\Delta(\Theta_i, c_j) = \frac{|\Theta_i \cap c_j|}{|c_j|} \tag{8}$$

where $|\Theta_i \cap c_j|$ is the number of the edges connecting Θ_i and cluster c_j. If $\Delta(\Theta_i, c_{\max})$ is greater than threshold β, then Θ_i is merged to c_{max}, otherwise, a new cluster is created for Θ_i. The threshold β is set to control the degree that how Θ_i is similar to the concept cluster c_j, and needs to be carefully tuned to get the best model performance. The revised single pass algorithm is given in Algorithm 1. After applying the SP, concepts are grouped into a number of concept clusters and each cluster formulates a densely connected subgraph. These pre-clustering results are then used as the initialization value for the following AGH.

5.2 Agglomerative hierarchical clustering based on concept graph

The similarity calculation is one of the most important factors affecting the clustering quality of AGH by determining which pair of clusters to be merged together. There are three commonly used linkage criteria for similarity calculation in AGH [18]. The single-link method merges two clusters if items in each cluster have the minimum distance. On the contrary, the complete-link

Algorithm 2 Clustering AGH algorithm

Input: G={D, E}, $\{c_1,c_2,c_3,...,c_\kappa\}$, δ
Output: $c_1, c_2, c_3, ..., c_K$ where $\kappa \geq K$
Initialize MaxSim=0.
repeat
1.Get the two clusters c_i, c_j that has max similarity *MaxSim* with $\Omega_{UPGMA}(c_i, c_j)$
2.Merge c_i, c_j into one cluster.
until $MaxSim < \delta$

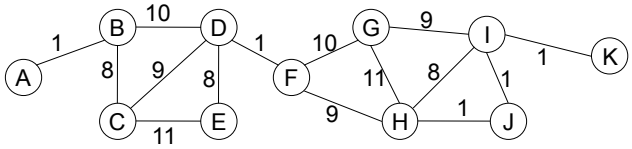


Fig. 3: An example for merging small isolated clusters.

method chooses to merge clusters if items of one cluster are the farthest to the items of the other cluster. Both criteria do not work well in our problem as we need to calculate the similarity of all items in a cluster to avoid merging an item into a wrong cluster. Therefore, we choose the third linkage cirterion, the average linkage method, which computes the average pairwise similarity of the concepts in each cluster, defined as

$$\Omega_{UPGMA}(c_i, c_j) = \frac{\sum_{\Theta_p \in c_i} \sum_{\Theta_q \in c_j} \rho(\Theta_p, \Theta_q)}{|c_i| * |c_j|} \quad (9)$$

where $|c_j|$ represents the number of concepts in cluster c_j. The corresponding revised merging step of the AGH algorithm is given in Algorithm 2.

Initialized by the SP, the number of initial concept clusters of the AGH are far less than that of the AGH initialized by data items, and thus the computation complexity of the revised AGH algorithm could be largely reduced. Detailed complexity analysis is discussed in next section.

After performing the AGH algorithm, there are some isolated concept clusters which can not be merged to any cluster. In fact, these isolated concepts clusters are not physically isolated. Due to the low similarity value, the AGH filters out these concepts as isolated clusters. Therefore, additional adjustment is needed to merge them into correct clusters. The adjustment steps are illustrated in the following example. In Figure 3, the parameter α is set to 3, then total 5 clusters can be extracted which are: A, BCDE, FGHI, K, J. However, there are only 2 actual clusters: ABCDE, FGHIJK. To fix this issue, these small isolated clusters are forced to be merged into the most similar cluster if the number of concepts of isolated clusters is less than a given parameter ε.

5.3 Complexity Analysis

To summarize, the proposed HCCG has three steps: a) pre-clustering step using the revised SP; b) clustering on concept graph using agglomerative hierarchical clustering; c) the adjustment merging step. Assume that m denote the number of concepts, n denote the number of concept clusters. The computational complexity of step a) and step c) is linear one and is written as $O(m)$. As for the complexity of step b), there are two main steps in the algorithm 2. First, it computes the pairwise similarity among all the concept clusters acquired in the Algorithm 1. The complexity of this step is $O(n^2)$. Second, in each merging step, two clusters c_i, c_j with the maximum similarity value are selected and merged together. To further optimize this merging step, a priority queue is created to store the merging list. If c_i is merged with c_j, then both items are removed from the priority queue and the similarity of new cluster c with old clusters is re-computed and re-inserted into the priority queue. By doing so, the complexity of this merging step is $O(n^2 log(n))$. Therefore, the overall computational complexity of the proposed approach is determined by step b, which is $O(n^2 log(n))$. As n is much smaller than m, it indicates that the proposed approach is a scalable one.

6 Experiment

In the experiments, the performance of the proposed approach was extensively evaluated on a number of datasets. Evaluation criteria include precision, recall and F-measure. The benchmark clustering algorithm - K-means, the original AGH, as well as the latent dirichlet allocation (LDA) were chose for the performance comparison. At last, we analyzed how to track the topic trend based on concept behavior curves extracted.

6.1 DataSets and Evaluation Criteria

Seven different datasets were chose for the experiments which are: “TDT5”, “TDT5 subset”, “Reuters”, “Reuters subset1”, “Reuters subset2”,“Blog data” and “TwitterData”. The characteristics of these datasets are reported in Table 1. In this table, the second, third column reports the number of documents and the number of topics in each dataset, respectively. The last two columns reports the number of documents in the largest cluster and the smallest cluster. “TDT5” is extracted from the benchmark dataset LDC2006T18 which contains 250 topics including English, Mandarin and Arabic documents from April 1st, 2003 to Sep 31st, 2003. “TDT5” consists of 607 document from 19 English topics and “TDT5 subset” contains only 8 topics. “Reuters”, “Reuters subset1” and “Reuters subset2” are extracted

Table 1: the characteristics of seven datasets

DataSet	No. of docs	No. of topics	No. of docs in max cluster	No. of docs in min cluster	Time span
TDT5	607	19	81	10	4.1-9.31 2003
TDT5 subset	274	8	81	10	4.1-9.31 2003
Reuters	1626	17	496	10	3.1-10.20 1987
Reuters subset1	293	7	88	9	3.1-10.20 1987
Reuters subset2	380	7	163	9	3.1-10.20 1987
Blog data	455	6	182	7	3.28-3.30 2011
TwitterData	6330	5	1623	1027	6.1-9.10 2011

from the dataset Reuters-21578 and only documents belonging to one topic are extracted. The extracted "Reuters" contains 1626 documents from 17 topics. Seven different topics were extracted from "Reuters" to form the sub dataset "Reuters subset1" and "Reuters subset2". Blog dataset was collected from Mar 28th, 2011 to Mar 31st, 2011 through the Google blog search. "TwitterData" was collected from social media website-www.twitter.com. All the seven datasets were partitioned into different document sets according to its chronological sequence. The document set at time t is abandoned if it contains less than 9 documents.

In the following experiments, precision, recall and F-measure was adopted to evaluate the clustering quality of the proposed approach. Let C_i be the class of dataset labeled manually and K_j be the set of the clusters generated by the clustering algorithms. The precision and recall is calculated as

$$Precision(C_i,K_j)=\frac{n_{i,j}}{|C_i|},Recall(C_i,K_j)=\frac{n_{i,j}}{|K_j|} \quad (10)$$

where $n_{i,j}$ is the number of the documents of class C_i in cluster K_j. The F-measure of the class C and cluster K can be computed as

$$F(C_i,K_j)=\frac{2*P(C_i,K_j)*R(C_i,K_j)}{P(C_i,K_j)+R(C_i,K_j)} \quad (11)$$

The average precision, recall and F-measure of each algorithm will be calculated and compared as the performance evaluation results.

6.2 Comparison Results

As aforementioned, "TDT5" and "TDT5 subset" were partitioned into 19 time intervals, each of which covers 10 days; "Reuters", "Reuters subset1" and "Reuters subset2" were partitioned into 16 time intervals, each of which covered 15 days; and "Blog data" was partitioned into 3 time intervals, each covers only one day. The standard pre-processing step was applied to all documents. Terms with its term frequency (TF) lower than 3 were filtered out from "Reuters", "Reuters subset1", "Reuters subset2" and "Blog data", and those with their TF lower than 8 were filtered out from the regular document set "TDT5" and "TDT5 subset". "TwitterData" was partitioned into 10 time intervals, each of which contains 10 days. Terms with their document frequency higher than 70% were removed. After a careful preliminary study, the threshold of the concept weight is set to 0.6 for the rest experiments.

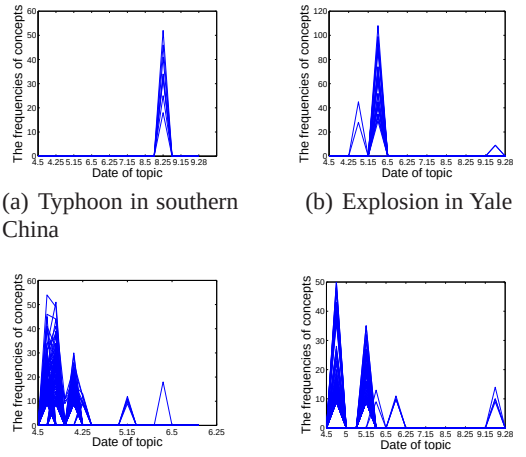
Table 2: The average precision, recall and Feasure of seven datasets

name	algorithm	precision	recall	F-measure
TDT5	K-Means	0.3668	0.3642	0.3184
	AGH	0.5906	0.3241	0.2653
	LDA	0.8772	0.8354	0.8243
	HCCG	**0.9923**	**0.9877**	**0.9897**
TDT5 subset	K-Means	0.9159	0.8928	0.8717
	AGH	0.9159	0.9030	0.8795
	LDA	0.8194	0.9424	0.8455
	HCCG	**0.9969**	**0.9926**	**0.9946**
Reuters	K-Means	0.5109	0.4068	0.3849
	AGH	**0.6966**	0.3221	0.1849
	LDA	0.3959	**0.6307**	0.3975
	HCCG	0.5331	0.5211	**0.4456**
Reuters subset1	K-Means	0.6970	0.7645	**0.7162**
	AGH	**0.7572**	0.5342	0.4555
	LDA	0.6300	0.7718	0.6407
	HCCG	0.7010	**0.7759**	0.7100
Reuters subset2	K-Means	0.5698	0.6290	0.5321
	AGH	0.6338	0.3957	0.2747
	LDA	0.5448	**0.7709**	0.5361
	HCCG	**0.6513**	0.7562	**0.6160**
Blog data	K-Means	0.9410	0.8569	0.8347
	AGH	0.9132	0.7356	0.7810
	LDA	0.6309	0.8379	0.6393
	HCCG	**0.9749**	**0.8967**	**0.9322**
Twitter Data	K-Means	0.7419	**0.7488**	0.7384
	AGH	**0.9993**	0.2059	0.3401
	LDA	0.4145	0.4654	0.4250
	HCCG	0.9392	0.6905	**0.7756**

The comparison results of the average precision, recall and F-measure with the baseline algorithms: K-means, AGH and LDA are reported in Table 2. As for the F-measure, it is well noticed that the performance of the proposed HCCG is superior to the rest three algorithms in all six datasets and is close to the best one in the dataset "Reuters subset1" which is 0.71 vs. 0.7162. For all three criteria, it can be seen that the proposed HCCG acquires the highest value in dataset "TDT5", "TDT5 subset" and "Blog data". From the F-measure results on "TDT5", the model performance of the HCCG is higher than the K-means, AGH and LDA by 211%, 273% and 20%, respectively. Similar observations could be found for precision and recall in "TDT5". It is observed that the proposed HCCG is 46% and 80% higher than that of the LDA in the dataset "Blog data" and

TwitterData, respectively. The possible reason is that the LDA might not work very well in the short context dataset.

For the precision comparison, performance of the proposed HCCG exceeds both K-means and LDA, and is lower than that of the AGH in "Reuters" and "Reuters subset1". The reason is that the HCCG is performed on top of the pre-clustering results of the SP algorithm, however, the pre-clustering results are affected by the threshold of the weight. If some concepts are wrongly merged during the pre-clustering steps, then the performance of the HCCG also degraded. To summarize, with respects to the three criteria, we can conclude that the proposed HCCG is superior to the rest three baseline algorithms. Moreover, the HCCG can achieve a much higher model performance especially on a high dimensional dataset.



(a) Typhoon in southern China

(b) Explosion in Yale

(c) Bird flu outbreak in Belgium

(d) Ten USS Cole attack suspects escape

Fig. 4: The changing trends of four topics on TDT5

6.3 Comparison of Keywords Extracted

To further demonstrate whether the keywords of the topics extracted are reasonable or not, the comparison between the HCCG and the LDA was performed on three datasets: "TDT5", "Reuters", "Blog data" and "TwitterData", and the results were reported in Table 3, Table 4, Table 5 and Table 6, respectively. In these tables, only the top k frequent terms of the LDA and the top k frequent concepts of the HCCG were recorded for the comparison.

From topic "Typhoon in southern China" in Table 3, the top k frequent terms extracted by the LDA included "necessari", "surviv" , "krovanh", "southern" and etc.; whereas the top k frequent concepts extracted by the HCCG include "hainan_krovanh", "typhoon_krovanh", "china_south" and etc. It is obvious that the term such as "necessari" is not relevant to topic "Typhoon in southern China" and the keyword "surviv" could also be used to describe topic "Explosion at Yale" and thus is not a discriminative term for current topic. It is noticed that the LDA extracted many such keywords. However, most of frequent concepts extracted by the HCCG are highly related to the topic. The same observations could be found for the rest topics in Table 3, Table 4 and Table 5. From these observed results, we can conclude the concept keyword extracted by the proposed HCCG is more accurate than the keyword extracted by the LDA algorithm.

The explanation for the good performance of the proposed HCCG is that the concept keywords of a topic often appear together, and thus get a higher concept weight. For instance, some sentences of topic "Typhoon" were found like "Typhoon Krovanh has killed two people and injured eight in southern China" and "Typhoon Krovanh leaves southern island city submerged". In these sentences, term "Typhoon" and "Krovanh" appeared together with a higher frequency, therefore, the concept "Typhoon_Krovanh" can be easily extracted. Terms "city" and "Typhoon" also appeared together but with a lower frequency which were then filtered out from the concept cluster. However, the LDA does not possess such ability to generate a pair of keywords. In this experiments, we extracted concepts which contain only two keywords. However, some concepts may not be denoted by only two keywords, i.e. it is not accurate for some concepts only represented by only two keywords. For example, "Japan nuclear power" contains more concrete semantic information than "nuclear power". Accordingly, three keywords are more appropriate to describe the topic.

6.4 Behavior Curve Analysis

In addition to detect topics, the proposed HCCG is able to track the topic change trend with the help of behavior curves. The change trends of four topics on datasets "TDT5", "Reuters", "Blog data" and "TwitterData" are plotted in Figure 4, Figure 5, Figure 6 and 7 respectively. Each curve in the subfigure denotes a concept behavior curve and the change of the curve indicates the change of concept.

From these figures, the following observations could be found. First, there exist topics which happen only once on the dataset. For instance, in Figure 4(a), Figure 5(a) and 5(d), the concept behavior curves only appear during a short period of time. This type of topic is incidental topic which does not repeat. Second, some topics are periodical, seen in Figure 4(c), 4(d) and Figure 5(b), 5(c). The corresponding behavior curve begins with the burst of the concept keywords, and the concept behavior curve gradually vanishes within a short time, then it re-bursts after some time with the similar curve shape but with a decreased peak value. This observation indicates that this

Table 3: the comparison of keywords of four topics on TDT5

Topics	Typhoon in southern China	Explosion at Yale	Bird Flu outbreak in Belgium	Ten USS Cole attack suspects escape
LDA	enditem,island,krovanh,region water,warn,strand,necessari south,surviv,southern,speed separ,typhoon,china,forc	fbraisi,threat,law,georg messag,damag,ceremoni room,connecticut,alert student,classroom	case,ban,outbreak,author Health,flu,poultry,border diseas,farm,bird,Belgium netherland,danger,chicken	kill,laden,secur,yemen uss,arrest,port,osama prison,escap,aden,cole terror,explos,yemeni
HCCG	hainan_krovanh,china_south china_hainan,typhoon_provinc krovanh_provinc,hainan_provinc china_krovanh,typhoon_krovanh	fbi_student,explos_law law_school,yale_student explos_classroom,bomb_law bomb_yale,explos_student	viru_belgium,infec_victim belgium_flu,influenz_victim viru_symptom,health_avian belgium_chicken,viru_flu	escap_quso,uss_laden terror_yemeni,yemeni_uss ashcroft_badawi,uss_cole yemen_cole,yemeni_aden

Table 4: the comparison of keywords of four topics on Reuters

Topics	cocoa	Money/Foreign Exchange	trade	veg-oil
LDA	stock,council,cocoa,rules, member,pact,prices,friday meeting,price,delegates agreement,market,buffer	west,finance,monetary,paris meeting,german,baker,dollar exchange,ministers,currency minister,germany,markets,	japan,japanese,officials,trade united,ministry,cut,nakasone minister,pact,states,official dispute,industry,april,action	vegetable,deadlock,bloc farm,june,agriculture,fats diplomats,european,oils germany,ec,community
HCCG	cocoa_buffer,deleg_consum stock_buffer,buffer_deleg buffer_consum,stock_cocoa council_deleg,cocoa_icco	german_west,exchang_west baker_rate,baker_treasuri rate_currenc,financ_minist financ_german,rate_german	japan_japanes,japan_u.s trade_u.s,commun_european disput_trade,trade_sourc current_trade,japan_trade	veget_britain,marin_cereal diplomat_summit,ec_farm ec_fat,german_summit veget_portug,commiss_fat

Table 5: the comparison of keywords of four topics on Blog data

Topics	Amazon	Artifical Leaf	Libya war	Japan nuclear power explosion
LDA	music,amazon,service,apple google,locker,launched,today digital,services,upload,called users,store,play,announced	leaf,harvick,club,auto artificial,world,mit,race year,kevin,sunday,brain energy,fontana,speedway	obama,libyan,company yesterday,time,war,lot military,american,libya night,monday,president	japan,water,radiation,reactor news,reactors,found,workers times,high,officials,highly safety,radioactive,quake,level
HCCG	amazon_music,cloud_music player_music,cloud_player player_amazon,music_servic servic_launch,cloud_amazon	mit_tata,artifici_develop artifici_leaf,practic_artifici professor_daniel,leaf_mit claim_daniel,leaf_scientist	libyan_obama,presid_libya presid_obama,libyan_presid defend_speech,speech_obama libyan_militari,libya_obama	japan_nuclear,japan_reactor plant_japan,nuclear_fukushima tepco_electr,power_fukushima power_electr,fukushima_daiichi

Table 6: The comparison of keywords of four topics on TwitterData

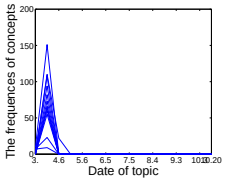
Topics	DailyGalaxy	GuardianTravel	HealthLive	SportArsenal
LDA	life song team world nasa cancer wilshere milky coachella space fabregas nasri massive djourou lp news mystery mars weekend	tour travel top music london announce day release great review big film holiday week summer year record hear single	health universe care black earth win uk today night years free future fans play hospital hotel human found star	arsenal wenger full goal match time game latest season van report league ai city back cup persie barcelona drug
HCCG	500_daily,blocks_comets largest_power,environment_nasa animals_fossil,event_fireball caves_crystal,detects_structure	round_travel,mexico_traditional aran_islands,islands_solomon greetings_holiday,caribbean_latest reports_thomas,honeymoon_kate	nurses_strike,california_carcinoma hospital_issue,governor_marijuana hazard_healthy,living_scans action_foods,novel_treatment	derby_influence,Arsenal_defender future_partner,footballer_lionel footballer_world,fifa_snoods draw_shots,breakthrough_campaign

kind of topic is a periodical one, and could be tracked by identifying the peak frequency of the curve.

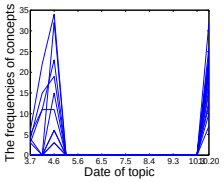
Third, different concepts have different concept behavior curves even in the same time interval, as shown in Figure 6. It is obvious that the curves of topics “Amazon”, “Artificial Leaf”, “Libya War” and “Japan Nuclear power explosion” do not resemble each other, which indicates that the concept curve could also be used to discover topics. These observations conclude that the concept behavior curves could be able to detect the topics as well as to track the topic change trend by monitoring on the curve’s peak value and the peak frequency. Fourth, the topics are always active topics but might have a sudden burst in a point-wise manner which is shown in Figure 7.

7 Conclusion

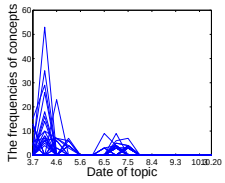
In this paper, a novel approach is proposed for detecting topics based on the hierarchical clustering on the concept graph. The proposed approach first defines concept, concept frequency as well as the concept behavior curves. Then, the concept graph was generated based on the concepts extracted from the documents set. A revised single pass algorithm is adopted to perform pre-clustering task on the original huge volume of documents to initialize the following AGH tree. The pre-clustering step could largely reduce the overall computational cost as well as improve the overall model performance. The results of the hierarchical clustering step are adjusted to re-assign the isolated concept clusters to existing concept clusters. The proposed approach is evaluated on a number of datasets. When compared with standard Kmeans, AGH, LDA, the proposed approach is superior to these algorithms in terms of the precision, recall and



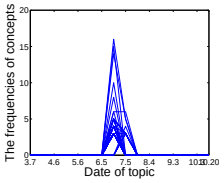
(a) cocoa



(b) Money/Foreign Exchange

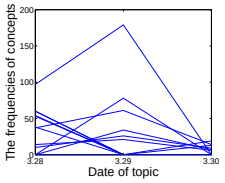


(c) trade

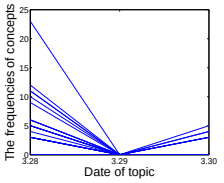


(d) veg-oil

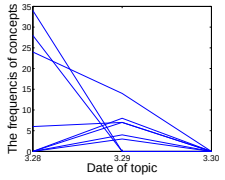
Fig. 5: The changing trends of four topics on Reuters



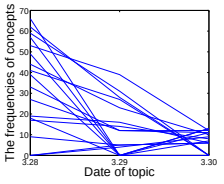
(a) Amazon



(b) Artifical Leaf



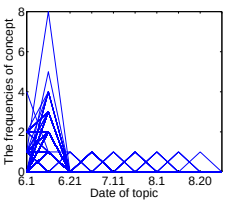
(c) Libya War



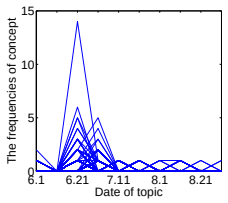
(d) Janpan Nuclear power explosion

Fig. 6: The changing trends of four topics on Blog data

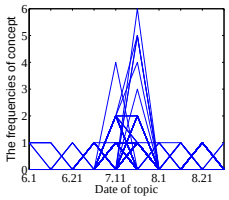
F-measure. Moreover, the proposed approach could extract more meaningful and accurate keywords than the LDA does. The concept behavior curves are able to track the topic change trends by monitoring on the peak value of the concept curves. In addition to these merits, the proposed approach can largely reduced the computational cost which indicates that it is applicable to a large-scale topic detection application. In the future, we will relax the limitation on the number of the concept keywords and perform a more profound study on how to reduce the complexity when the combination of the concepts keywords are huge.



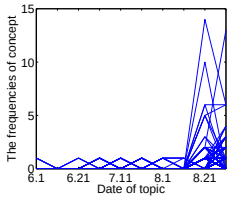
(a) DailyGalaxy



(b) GuardianTravel



(c) HealthLive



(d) SportArsenal

Fig. 7: The changing trend of four topics on TwitterData

Acknowledgement

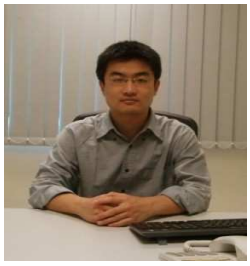
The authors are grateful to the anonymous referees for a careful checking of the details and for helpful comments that improved this paper. This research was supported in part by NSFC under Grant No.61073195 and 61073051.

References

[1] J. Allan, Introduction to topic detection and tracking, Topic detection and tracking: Event-based information organization, 1-16 (2002).
[2] L. Zhang, M. Jiang, D. Farid, and M. Hossain, Intelligent Facial Emotion Recognition and Semantic-based Topic Detection for a Humanoid Robot, Expert Systems with Applications, (2013).
[3] Y. Hattori and A. Nadamoto, Tip information from social media based on topic detection, International Journal of Web Information Systems, **9**, 83-94 (2013).
[4] R. Swan and J. Allan, Automatic generation of overview timelines, in Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval, 49-56 (2000).
[5] C. Lin, Y. He and R. Everson, and S. Ruger, Weakly supervised joint sentiment-topic detection from text, IEEE Transactions on Knowledge and Data Engineering, **24**, 1134-1145 (2012).
[6] F. Smadja, Retrieving collocations from text: Xtract, Computational linguistics, **9**, 43-177 (1993).
[7] X. Zhou, X. Hu, X. Zhang, X. Lin, and I. Song, Context-sensitive semantic smoothing for the language modeling approach to genomic IR, in Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, 170-177 (2006).
[8] F. Holz, and S. Teresniak, Towards Automatic Detection and Tracking of Topic Change, Computational Linguistics and Intelligent Text Processing, 327-339 (2010).

[9] X. Wang, and A. McCallum, Topics over time: a non-Markov continuous-time model of topical trends, in Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, 424-433 (2006).
[10] D. M. Blei, A. Y. Ng, and M. I. Jordan, Latent dirichlet allocation, The Journal of Machine Learning Research, **3**, 993-1022 (2003).
[11] S. E. Garza, and R. F. Brena, Graph Local Clustering for Topic Detection in Web Collections, Web Congress, Latin American, **0**, 207-213 (2009).
[12] H. Sayyadi, M. Hurst, and A. Maykov, Event detection and tracking in social streams, in Proceedings of the International Conference on Weblogs and Social Media, (2009).
[13] C. Wartena, R. and Brussee, Topic detection by clustering keywords, in Proceedings of the 19th International Conference on Database and Expert Systems Application, 54-58 (2008).
[14] C. Zhou, H. Chen, and J. Tao, GRAPH: A Domain Ontology-driven Semantic Graph Auto Extraction System, Applied Mathematics & Information Sciences, **5-2S**, 9S-16S (2011).
[15] Q. He, K. Chang, E. Lim, and A. Banerjee, Keep it simple with time: A Re-examination of probabilistic topic detection models, IEEE Transactions on Pattern Analysis and Machine Intelligence, **32**, 1795-1808 (2010).
[16] A. Banerjee, and J. Ghosh, Frequency-sensitive competitive learning for scalable balanced clustering on high-dimensional hyperspheres, IEEE Transactions on Neural Networks, **15**, 702-719 (2004).
[17] X. Hu, X. Zhang, C. Lu, E. Park, and X. Zhou, Exploiting Wikipedia as external knowledge for document clustering, in Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, 389-396 (2009).
[18] Y. Zhao, and G. Karypis, and U. Fayyad, Hierarchical clustering algorithms for document datasets, Data Mining and Knowledge Discovery, **10**, 141-168 (2005).
[19] H. S. AL-Obaidy, and A. A. Heela, Annotation: An Approach for Building Semantic Web Library, Applied Mathematics & Information Sciences, **6**, 133-143 (2012).
[20] X. Zhou, X. Zhang, and X. Hu, Semantic smoothing of document models for agglomerative clustering, in Proceedings of the 20th International Joint Conferences on Artificial Intelligence, 6-12 (2007).
[21] G. Spanakis, G. Siolas, and A. Stafylopatis, Exploiting Wikipedia Knowledge for Conceptual Hierarchical Clustering of Documents, The Computer Journal,**55**, 299-312 (2012).
[22] A. I. Aggour, F. E. Attounsi, Fuzzy Topological Properties on Fuzzy Function Spaces, Applied Mathematics & Information Sciences Letters, **1**, 1-5 (2013).
[23] Q. Mei, and C. Zhai, Discovering evolutionary theme patterns from text: an exploration of temporal text mining, in Proceedings of the 7th ACM SIGKDD international conference on Knowledge discovery in data mining, 198-207 (2005).
[24] M. BAL, Rough Sets Theory as Symbolic Data Mining Method: An Application on Complete Decision Table, Information Sciences Letters, **2**, 35-47 (2013).
[25] A. Banerjee, I. S. Dhillon, J. Ghosh, and S. Sra, Clustering on the Unit Hypersphere using von Mises-Fisher Distributions, Journal of Machine Learning Research, **6**, 1345-1382 (2005).
[26] M. Yamamoto, and K. Sadamitsu, Dirichlet mixtures in text modeling, Technical Report of Department of Computer Science, 1-13 (2005).
[27] L. AlSumait, D. Barbara, and C. Domeniconi, On-Line LDA: Adaptive Topic Models for Mining Text Streams with Applications to Topic Detection and Tracking, in Proceedings of the 8th IEEE International Conference on Data Mining, 3-12 (2008).
[28] X. Chen, X. Hu, T. Lim, X. Shen, E. Park, and G. Rosen, Exploiting the Functional and Taxonomic Structure of Genomic Data by Probabilistic Topic Modeling, IEEE/ACM Transactions on Computational Biology and Bioinformatics, **9**, 980-991 (2012).
[29] M. D. Hoffman, D. M. Blei, and F. Bach, Online learning for latent dirichlet allocation, Advances in Neural Information Processing Systems, **23**, 856-864 (2010).
[30] M. E. J. Newman, and M. Girvan, Finding and evaluating community structure in networks, Physical review E, **69**, 26113 (2004).
[31] F. Murtagh, A survey of recent advances in hierarchical clustering algorithms, The Computer Journal, **26**, 354-359 (1983).
[32] E. Iosif, Unsupervised Web Name Disambiguation Using Semantic Similarity and Single-Pass Clustering, Artificial Intelligence: Theories, Models and Applications, 133-141 (2010).

Xiaohui Huang is a Ph.D student in the Shenzhen Graduate School, Harbin Institute of Technology, China. His research interests are in the areas of data mining, topic detection and clustering algorithm.



Xiaofeng Zhang has received PhD degree from Department of Computer Science, Hong Kong Baptist University, and is now an assistant professor at Shenzhen Graduate School, Harbin Institute of Technology. His research interests include data mining, machine learning and artificial intelligence.

Yunming Ye received the Ph.D. degree in Computer Science from Shanghai Jiao Tong University. He is now a professor in the Shenzhen Graduate School, Harbin Institute of Technology. His research interests include data mining, text mining, and ensemble learning algorithms.

Shengchun Deng is a professor in School of Computer Science and Engineering at Harbin Institute of Technology, Harbin, China. His research interests include computer integrated manufacturing system, supply chain management, business intelligence, and data mining applications.

Xutao Li received the B.S. and M.S. degrees in Computer Science from Lanzhou University and Harbin Institute of Technology in China in 2007 and 2009, respectively. He is currently working towards the Ph.D. degree in the Department of Computer Science, Harbin Institute of Technology. His research interests include data mining, machine learning, graph mining and social network analysis.