Cross-Media Topic Detection Associated With Hot Search Queries

Zhe Xue
University of Chinese Academy of Sciences
Beijing, 100049, China
zhxue@jdl.ac.cn

Guorong Li
University of Chinese Academy of Sciences
Beijing, 100049, China
grli@jdl.ac.cn

Weigang Zhang
School of Computer Science and Technology
Harbin Institute of Technology,
Harbin, 150001, China
wgzhang@jdl.ac.cn

Shuqiang Jiang
Key Lab of Intell. Info. Process.
Inst. Of Comput. Tech., CAS
Beijing, 100049, China
sqjiang@ict.ac.cn

Qingming Huang
University of Chinese Academy of Sciences
Beijing, 100049, China
qmhuang@jdl.ac.cn

ABSTRACT

Although lots of work has been done since NIST proposed the problem of Topic Detection and Tracking (TDT), most of them focus on single media data. Topic detection for cross-media data hasn't been fully investigated. In this paper, we propose an effective method for cross-media topic detection. Unlike traditional topic detection methods that are mainly based on clustering, we consider using hot search queries as guidance to detect topics. Besides, we propose an improved co-clustering method which can be well suited for cross-media data clustering. First, we use queries to detect topics directly, and find the data associated with the topic. Second, we apply our co-clustering method to find the topics existing in the rest of data. Finally, the results obtained by the first two steps are threaded together as topics. Experiments show that our method can effectively detect topics for cross-media data.

Keywords

Topic detection, Cross-media, Hot search queries, Co-clustering

1. INTRODUCTION

As the rapid advancement of Internet and multimedia technology, social media website becomes an important platform for people to access the information they are interested in and share their opinion. Single type of media can no longer meet the users' needs. The same information not only exists in a single media, but also disseminates and integrates between different types of media. Besides, the volume of data from a variety of media grows explosively, which makes it difficult for people to find what they are interested in. If topics can be discovered from the large amounts of data, users can know what is happening and quickly access the information they concern. Topic detection is such an effort to discover topics from a collection of documents and gather together the documents belonging to the same subject. Topic detection from cross-media data will integrate various types of media data and detect topics implied in it.

Although TDT is proposed by NIST in the 1990s [1], so far, most related works are based on clustering method and focus on single media, such as news articles [2, 3, 4] and web videos [5, 6]. Wang et al. [7] adopts a traditional agglomerative clustering method to cluster news stories into topics. Cao et al. [6] first cluster video tags into groups to get small events, and then link these events into topics based on textual and visual similarity. However, because of the lack of efficient clustering guidance information, the size and number of the clusters can't be determined, and the topics can't be detected efficiently. It should be noted that users' queries recorded in the search engine are very useful for topic detection because they can provide strong indications of the real-world topics. Sun et al. [8] propose a query-guided event detection method for news and blogs which is based on the query profile. However, it needs a search engine to obtain relevant documents, and can't be applied to the specified data set for topic detection. The hot search queries are used to refine the results of tag group topic detection in [9]. But if the query-related topics can't be detected by the tag group, these query-related topics will not be detected.

Cross-media data consists of a variety modal of data, and the characteristics of these data are also different. For example, news articles contain abundant textual information while web videos contain rich visual content. Co-clustering is a good method to overcome the problem of sparse and noisy textual features and the distinctive characteristics of multi-modality. In [10], the information-theoretic co-clustering is adopted for video topic detection, but it doesn't consider using other modality information such as visual information. Shao et al. [11] propose a star-structured K-partite Graph based co-clustering for web video topic discovery. Although multi-modal features are used, the clustering result may be effected by the noisy and inaccurate textual information. In fact, some meaningful textual information can be identified by bursty [6] and further used to reduce the influence of noise.

In this paper, we propose an effective approach for cross-media topic detection that not only is based on clustering but also use queries to guide the detection. For clarity, the following

ICIMCS'13, Aug. 17–19, 2013, Huangshan, Anhui, China.

terminologies are defined by us. *Event* is a group of news, videos and other types of cross-media data that related to a story. It is discovered at a time unit. *Topic* is made up of a series of topic related events. *Hot search queries* are the queries that are searched by a large number of times in the search engine. They can be obtained conveniently through the search engine (such as Google and Baidu). Figure 1 shows the framework of the proposed approach (denoted by Q-WCC). First, we preprocess the hot search queries which are collected from search engine to get some effective queries related to topics. To analysis the queries effectively, we use the search engine to obtain some related textual information about the queries. Then this information is used to help query preprocessing as well as query and data matching. Second, these topic-related queries match with cross-media data directly to obtain the events and their related data. The matching are based on textual feature (the news texts and the tags of videos). Third, the remaining data is divided by time unit and the events in each time unit are detected by a weighted co-clustering algorithm. This algorithm can integrate muti-modal features and be applied to the cross-media data effectively. Finally, we thread the detected events together to obtain topics according to the textual and time similarity.

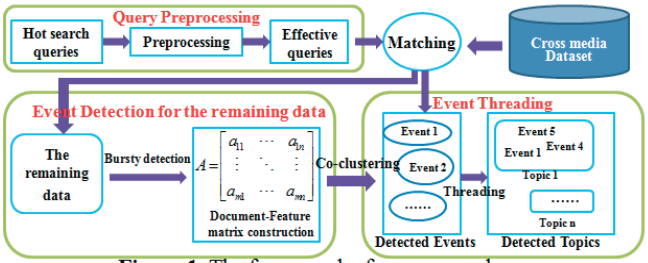

Figure 1. The framework of our approach

The remainder of this paper is organized as follows. In section 2, we present how to preprocess the queries and use it for topic detection. In section 3, we introduce our weighted co-clustering algorithm. Event threading method is briefly described in section 4. Experimental results are provided in section 5 and finally the paper is concluded in section 6.

2. QUERY-GUIDED TOPIC DETECION

Not all search queries are topic-related [8] and different queries may be related to the same topic. So query preprocessing should be conducted before matching them to data. Besides, the queries are lack of enough information for matching. Since the search results by search engine can help enriching the query information [8], we use them (denoted as background information or BI) to expand the query, and the preprocessing and matching are based on this information. The BI is the document collection of the searched results which has only textual information.

2.1 Preprocessing of Queries

The hot search queries are noisy. As many hot search queries are likely to be website names, they should be removed (denoising) before topic detection. Besides, different queries may relate to the same topic. Then the topics that these related queries represent should be treated as one (merging).

2.1.1 *Denoising*

We use BI to analyze the characteristics of the queries. We find that the search results of topic-related queries are similar with each other because most of them are the information related to the same topic. While the search results of noisy queries are different from each other. To measure the similarity between these search results in BI, the average cosine distance is used and defined as:

$$AverSim(q) = \frac{1}{n} \sum_{d_i, d_j \in B_q, i \neq j} \cos(d_i, d_j) \qquad (1)$$

where d_i, d_j are two textual vectors representing two search results in Bq (the BI of query q), n is the total number of calculations. Smaller $AverSim$ means the difference between search results is larger. If the $AverSim$ is smaller than a threshold, this query is treated as noise and can be removed.

2.1.2 *Merging*

If two queries are related to the same topic, they may contain the same term (a query usually consists of several terms). Besides, their content of BI may similar with each other. These are the two criterions that we use to merge similar queries. The term in query is denoted as t. Then the term similarity between two queries q_i, q_j is given by:

$$Sim\_Term(q_i, q_j) = \sum_{t_m \in q_i \cap q_j} t_m \Big/ \Big(\sum_{t_n \in q_i \cup q_j} t_n \Big) \qquad (2)$$

We can see that the greater similarity is, the more same terms that the two queries contain.

Let V denote the vocabulary, and $P(w | Bq_i) = tf(w, Bq_i) \big/ (\sum_{w' \in V} tf(w', Bq_i))$ is the relative term frequency of w in Bq_i, $M = (Bq_i + Bq_j)/2$, $P(w | M)$ is similar to $P(w | Bq_i)$. One common method to compute the distance between two sets of documents is to compute the divergence between the language models of the two sets of documents [17]. The square root of Jensen-Shannon divergence is used as the metric. So the BI similarity between q_i and q_j is defined as:

$$Sim\_BI(q_i, q_j) = 1 - JSD\_sqr(Bq_i, Bq_j) \qquad (3)$$

where

$$JSD\_sqr(Bq_i, Bq_j) = \sqrt{(KL(Bq_i \| M) + KL(Bq_j \| M))/2} \qquad (4)$$

$$KL(Bq_i \| M) = \sum_{w \in Bq_i} P(w | Bq_i) \log_2 \frac{P(w | Bq_i)}{P(w | M)} \qquad (5)$$

The total similarity between q_i and q_j is defined as:

$$Sim(q_i, q_j) = \lambda \cdot Sim\_BI(q_i, q_j) + (1 - \lambda) \cdot Sim\_Term(q_i, q_j) \quad (6)$$

If the total similarity is greater than the threshold th_1, the two topics that the two queries represent are merged into one. We set $\lambda = 0.6$ and $th_1 = 0.45$.

2.2 Matching with Data

Since the BI contains some related information of the query, it can enrich and densify the amount of informative text for matching. We use textual features for query and data matching. Based on BI, the similarity between query and video is defined as:

$$Sim\_QV(q_i, v_j) = \theta \cdot Sim\_BIV(q_i, v_j) + (1 - \theta) \cdot Sim\_Term(q_i, v_j) \quad (7)$$

where $Sim\_Term(\cdot)$ is the term similarity between query terms and video tags defined by Equ.(2). $Sim\_BIV(\cdot)$ is the similarity between BI and video, given by:

$$Sim\_BIV(q_i,v_j)=\frac{sim(Bq_i,v_j)-\min\limits_{i,j}\{sim(Bq_i,v_j)\}}{\max\limits_{i,j}\{sim(Bq_i,v_j)\}-\min\limits_{i,j}\{sim(Bq_i,v_j)\}} \quad (8)$$

where $Sim(Bq_i,v_j)=\sum\limits_{w\in Bq_i\cap v_j} TF-IDF(w)$, v_j is the terms of the video tags. As we can see, even there are no common terms between video tags and query terms ($Sim\_Term(\cdot)=0$), they can be matched as long as the BI bridges them ($Sim\_BIV(\cdot)>0$).

The news texts and BI are represented as vectors in vector space model. Cosine distance is used to measure the similarity between query q_i and news n_j :

$$Sim\_QN(q_i,n_j)=\cos(Bq_i,n_j) \quad (9)$$

The news or video whose similarity score is larger than the threshold th_2 is assumed to be matching with the query. Through experiment, we find $\theta=0.55$ and $th_2=0.4$ generate a good result.

3. A WEIGHTED CO-CLUSTERING ALGORITHM FOR CROSS-MEDIA TOPIC DETECTION

Based on [12], we develop a weighted co-clustering algorithm, which can integrate the visual information (NDK) and enhance the effective information. Specifically, let $G=\{D,F,E\}$ denote the undirected bipartite graph, where $D=\{d_1,d_2,\cdots,d_n\}$, $F=\{f_1,f_2,\cdots,f_m\}$ are two sets of vertices and E is the set of edges $\{\{d_i,f_j\}:d_i\in D,f_j\in F\}$. In our method D is the set of documents and F is the set of features (terms and NDKs) they contain. An edge $\{d_i,f_j\}$ exists if the feature f_j occurs in document d_i . The weight matrix A is defined as following:

$$A_{ij}=\begin{cases}\text{TF-IDF}(f_j),\ \text{if}\ \{d_i,f_j\}\in \text{E and} f_j \text{ is a term vertex}\\ \gamma_N,\ \text{if}\ \{d_i,f_j\}\in \text{E and} f_j \text{ is a NDK vertex}\\ 0,\ \text{otherwise}\end{cases} \quad (10)$$

Then we adjust the edge weight to reduce the effects of noise. As burstiness is an important characteristic of the event-related terms [6], more emphasis should be put on the bursty terms. Similar to [6], if there exists a time unit $t*$, the term f_i satisfies the following condition:

$$y_{f_i}(t^*)\ge\mu(y_{f_i}(t))+\alpha\times\sigma(y_{f_i}(t)),\quad t\in[t*-W,t*+W]$$

where $y_{f_i}(t^*)$ is the number of documents with the term f_i at time unit $t*$, μ and σ are the mean and standard deviation of $y_{f_i}(t)$ within the time window $[t*-W,t*+W]$, then it is treated as a bursty term. If the term f_i is bursty, we set $A_{ij}=A_{ij}\times\gamma_T$ for all j , where γ_T is the parameter to enhance the weight and $\gamma_T>1$. We set the time unit to a time span of 3 days, $\alpha=2$ and $W=9$ in our experiment. After obtaining the matrix A , we use the co-clustering algorithm in [12] to cluster the data and detect topics.

4. EVENT THREADING

The detected events by the query-matching or co-clustering may relate to the same topic. These events are similar in textual content, and also very close in time. We define the similarity between two events as follows:

$$Sim\_event(e_i,e_j)=Sim\_content(e_i,e_j)\times Sim\_time(e_i,e_j) \quad (11)$$

We use only the textual information in each event and Cosine distance is used as content similarity ($Sim\_content$). The time similarity is given by:

$$Sim\_time(e_i,e_j)=e^{-\beta\cdot\frac{dt(e_i,e_j)}{T}} \quad (12)$$

where $dt(e_i,e_j)=\begin{cases}|t(e_i)-t(e_j)| & if\ |t(e_i)-t(e_j)|\le T\\ 0 & else\end{cases}$, $t(e_i)$ is the average time of documents in event e_i . Based on Equ.(11), the events will be threaded when the similarity is greater than the threshold th_3 . We set $T=6$, $\beta=0.4$ and th_3=0.47 . Finally, the events threaded together are treated as a topic.

5. EXPERIMENTS

To test the proposed method, we construct a cross-media dataset, which includes more than 9100 news articles and 6000 web videos for one month, say from 1 May to 31 May, 2012. The news and video data are crawled from Sina news [13] and YouKu [14], respectively. The content of the news is constituted by the title and body, while the textual content of web videos is obtained from the titles and tags. For Chinese's characteristic, we use a natural language processing (NLP) tool [15] to parse the content first and then filter out the stop words. The 430 ground-truth topics of the dataset are manually labeled by 4 assessors.

Baidu is a major search engine in China, and it records the daily news hot search queries [16]. The hot search queries are obtained from Baidu in this experiment. We collect the queries from the corresponding period of the dataset and obtain a total of 620 queries. The queries are also parsed by the NLP tool to get the terms for preprocessing and matching.

To evaluate the results quantitatively, we use Precision, Recall and F-measure to assess the performance,

$$P(S_D,S_G)=\frac{|S_D\cap S_G|}{|S_D|}\qquad R(S_D,S_G)=\frac{|S_D\cap S_G|}{|S_G|} \quad (13)$$

$$F(S_D,S_G)=\frac{2\times P(S_D,S_G)\times R(S_D,S_G)}{P(S_D,S_G)+R(S_D,S_G)} \quad (14)$$

where SD, SG are the sets of detected and ground-truth data respectively. If the F-measure of a detected topic is more than 0.5, we consider it as a correctly detected topic.

5.1 Effectiveness of Weighted Co-clustering

To evaluate the performance of our weighted co-clustering (WCC) on single media data, we compare it with the method in [9] (TG). As TG is proposed for web video topic detection, we use all the video data in the data set as the test data. Moreover, to evaluate the effect of weight adjustment and fuse visual information, we also implement the original co-clustering (OCC) and the WCC without visual information (WCC-V). We set $\gamma_T = 10$ and $\gamma_N = 5$. All the detected topics are sorted by F-measure and the average precision, recall and F-measure of the top-20 topics are calculated for evaluation which is the same as in [9]. The comparison result is shown in Table 1. We can see that WCC-V outperforms OCC, which proves that our clustering algorithm is more effective. Through integrating visual information, WCC obtains better result, indicating that visual information is very useful for topic detection.

Table 1. The comparison of the 4 methods

Method	Precision	Recall	F-measure
TG	0.923	0.848	0.869
OCC	0.889	0.799	0.816
WCC-V	0.885	0.868	0.858
WCC	0.900	0.897	**0.881**

Furthermore, another experiment is conducted to show that our co-clustering method can effectively integrate the information of cross-media data and make up for the deficiency of the single media data. The WCC method is tested on video data set, news data set and the total data set. In the correctly detected topics, topic containing videos is denoted by VT and topic containing news is denoted by NT. The number of these topics is compared in Table 2. The second and third column illustrate that some topics which can't be detected by single media data are detected by integrating another media data. When testing on the total data set, since some detected topics contain both videos and news, the total number is less than the summation of the number of VT and that of NT, as shown in the fourth row.

Table 2. The number of topics detected by WCC

Topic Test set	Number of VTs	Number of NTs	Total number
Video set	40	0	40
News set	0	83	83
Total data set	**46**	**90**	124

5.2 Query-guided Topic Detection

To show the effectiveness of query preprocessing approach and the significance of hot search queries in topic detection, we conduct the Q-WCC method and this method that is without query preprocessing (UQ-WCC) on the total data set. We compare the performance of WCC, UQ-WCC and Q-WCC. The average Precision, Recall, F-measure and total number of correctly detected topics are shown in Table 3. We can see that the Q-WCC is more accurate than WCC, and it can detect more topics that can't be detected only through clustering. The preprocessing for queries can also improve the detection performance by enhancing the effectiveness of queries.

Table 3. The comparison of WCC, UQ-WCC and Q-WCC

Method	Precision	Recall	F-measure	Topic Number
WCC	0. 647	0.799	0.634	124
UQ-WCC	0.766	0.756	0.717	183
Q-WCC	0.764	0.772	**0.725**	183

6. CONCLUSION

In this paper, we propose an effective method for cross-media topic detection which is not only based on clustering, but also associated with hot search queries. The events are first detected by queries directly, and then the WCC is used for event detection in the remaining data. Finally, topics are obtained by event threading. Moreover, an effective method for query preprocessing and matching is proposed. Thus we can use the queries as guidance for topic detection. We also modify the original co-clustering algorithm to solve our problem better. Experimental results demonstrate that queries play an important role for topic detection and the weighted co-clustering could be well suited for cross-media data clustering.

7. REFERENCES

[1] LDC, "TDT3 evaluation specification version 2.7." 1999.

[2] [Q. He, K. Chang, and E.P. Lim, "Analyzing feature trajectories for event detection," in *ACM SIGIR Conference*, 2007.

[3] Q. Mei and C. Zhai, "Discovering evolutionary theme patterns from text: an exploration of temporal text mining," in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2005.

[4] J. Allan, J. Carbonell, G. Doddington, J. Yamron, and Y. Yang, "Topic detection and tracking pilot study: Final report." In *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, 1998.

[5] L. Liu, L. Sun, Y. Rui, Y. Shi, and S. Yang, "Web video topic discovery and tracking via bipartite graph reinforcement model," in *International World Wide Web Conference*, 2008.

[6] J. Cao, C.W. Ngo, Y.D. Zhang, and J.T. Li, "Tracking web video topics: discovery, visualization and monitoring." *IEEE Transactions on Circuits and Systems for Video Technology*, 21(12): 1835-1846, 2011.

[7] C.H. Wang, M. Zhang, S.P. Ma, and L.Y. Ru, "Automatic online news issue construction in web environment," in *International World Wide Web Conference*, 2008.

[8] A. X. Sun, and M. S. Hu, "Query-guided event detection from news and blog streams," *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 41(5): 834-839, 2011.

[9] T.L. Chen, C.X. Liu, Q.M. Huang, "An Effective Multi-Clue Fusion Approach for Web Video Topic Detection," In *ACM Multimedia*, 2012.

[10] I.S. Dhillon, I.S. et al., "Information theoretic co-lustering," in *Proc. 9th ACM SIGKDD'03*, pp. 89-98.

[11] J. Shao, S. Ma, W. M. Lu, and Y. T. Zhuang, "A unified framework for web video topic discovery and visualization," *Pattern Recognition Letters*, 33(4): 410-419, 2012.

[12] I.S. Dhillon, et al., "Co-clustering documents and words using bipartite spectral graph partitioning," in *Proc. 7th ACM KDD'01*, pp. 269-274.

[13] DOI=http://news.sina.com.cn/

[14] DOI=http://www.youku.com/

[15] DOI=http://ictclas.org/ictclas_download.aspx

[16] DOI=http://hot.news.baidu.com/

[17] D. Carmel, E. Yom-Tov, A. Darlow, and D. Pelleg, "What makes a query difficult?" in *Proc. SIGIR*, Seattle, WA, 2006.