J. Shanghai Jiaotong Univ. (Sci.), 2013, 18(4): 385-391
DOI: 10.1007/s12204-013-1416-z

Thread Labeling for News Event

YAN Ze-hua (闫泽华), *LI Fang** (李　芳)
(Department of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai 200240, China)

Abstract: Automatic thread labeling for news events can help people know different aspects of a news event. In this paper, we present a method to label threads of a news event. We use latent Dirichlet allocation (LDA) topic model to extract news threads from news corpus. Our method first selects the thread words subset then extracts phrases based on co-occurrence calculation. The extracted phrase is then used as a label of a news thread. Experimental results show that about 60% of generated labels visualize the meaningful aspects of a news event. These labels can help people fast to capture many different aspects of a news event.
Key words: news event, topic labeling, latent Dirichlet allocation (LDA)
CLC number: TP 391　　**Document code:** A

0 Introduction

According to the recent survey on the Internet[1] by China Internet Network Information Center (CNNIC), reading news reports is among the most five applications on the Internet. News events happened in the real world, many news reports describe different aspects of the events. For example, when earthquake happens, news reports will first report the event, then report the actions taken by the government, aid from the international world, while reconstruction plan will be reported later. It needs a method to automatically present different aspects of a news event. It then saves a lot of time for people to know many aspects and even the plot of the event. According to Ref. [2], news thread has been defined as below:

Definition 1 (News Thread) A news thread z in an event corpus D is a probability distribution of words $\{p(w|z)_{w\in V}\}$, where V is a vocabulary set. Usually, a list of high probability words is used to represent the semantic theme of a thread. A news thread illustrates an aspect of news event. Threading news event is the process of extracting threads from an event corpus.

Definition 2 (Thread Label) A thread label for a news thread z, is a sequence of words which is representative and captures the latent meaning of thread z. A word, a phrase are all valid labels for threads.

Recently emerged topic models, such as latent Dirichlet allocation (LDA)[3] can extract latent topics from a corpus. They explain a topic with a multinomial word distribution. As it for our work, LDA is used on event-based corpus, and topics in LDA are regarded as threads of a news event. After analyzing LDA results on news corpora, we find that LDA can produce meaningful results for human interpretation.

Although LDA captures the semantic meaning of events, it is not convenient for people to comprehend a thread by a top word list, because people need to figure out its meaning based on these words. The task of thread labeling is to automatically generate a representative label to visualize a thread. All these labels help people fast to know many aspects of the event.

Our method can be illustrated as: ① extract threads from an event corpus by LDA; ② select thread words from LDA result; ③ extract thread label based on thread words.

1 Related Work

Related work is divided into the methods on how to thread news events and the methods on how to label the threads.

1.1 Threading News Events

In general, there are two basic approaches to threading news event.

The first approach is using probabilistic model to thread news events[4-8]. In this approach, thread is regarded as a probability distribution over words. An event model is used to capture the structure of news events and their dependencies in a news event[4-5]. Some features such as temporal locality of stories and time-ordering are utilized to capture events and their dependencies. "Topics over time" (TOT) model

Received date: 2011-07-18
Foundation item: the National Natural Science Foundation of China (No. 60873134)
***E-mail:** fli@sjtu.edu.cn

associates topics with a continuous distribution over time stamps based on LDA[6]. Evolution of events over time stamps is used to describe different threads.

The second approach is using clustering method to thread news events[9-10] or text message streams[11-12]. Thread is regarded as a part of a hierarchy structure[13-14]. According to the pre-selected features based on time, highly related documents are first extracted, and then partitioned them to form events and organized them in a hierarchical structure. Similarly, an algorithm for constructing a hierarchical structure for the features in the text corpus is proposed by using an infinite-state automaton[15]. Key phrase can also be used to thread news events[9,16-17].

1.2 Labeling Threading

There are lots of significant works on topic labeling[18-19]. Many useful features are proposed in the following.

Pointwise Mutual Information (PMI)[20] is a criterion commonly used in statistical language modeling of word association for different tasks. PMI is measured by gathering term co-occurrence statistics from the web to compare different labeling methods[21], or is used independently over the web and the Wikipedia corpus to re-rank cluster labels[22]. Recently some research uses PMI to calculate the average PMI of topic words according to the Wikipedia in order to find a more meaningful word[23-24].

Inverse Document Frequency (IDF)-like score is used to re-weight the native weights generated by LDA[25]. In this work, authors suppose that the native order of topic keywords produced by LDA may not be ideal for users to understand the semantics of a topic. In LDA, common words are normally ranked high in many topics because they are relevant to all topics. IDF-like score is used to select significant words.

To enhance the interpretation of the model, "turbo topic" was proposed based on analyzing the posterior distribution of the topic structure of a corpus[26]. They performed recursive permutation tests to build up an n-gram phrase to label the LDA result.

Unlike re-ranking LDA results, a probabilistic approach is proposed to automatically label multinomial topic models, which regards the labeling problem as an optimization problem and is merged with term weighting scheme into LDA to improve the result[27-28].

Comparing with a complicated model, re-ranking LDA results could keep statistical simplicity of topic modeling and perform much effectively. In our work, methods are proposed to automatically re-rank LDA results and generate thread label based on the re-ranked results. It keeps effective and makes thread labels easy to extract.

2 Our Methods

2.1 Thread Words Selection

Applying LDA on news corpus can generate news threads represented by a top-10 words list z. Each word w_i in thread z_j has a probability $P^{\mathrm{w}}_{i,j}$ according to term frequency. Each document d_i has a probability $P^{\mathrm{d}}_{i,j}$ over thread z_j.

The task of thread words selection is to select three words from the top-10 term list z as thread word set z'. A thread word should be highly frequent in a thread, discriminative across threads and semantically relevant with others words.

Based on these features, three judges are proposed to assign weight $\xi(w_i)$ for each word w_i in z, and select top-3 words as z'.

Significance Judge Significant words can identify which aspect the thread focuses on and distinguish the thread from others. Common words shared by many threads are helpless to distinguish different threads.

An LDA-version Term Frequency-IDF (TF-IDF) like judge is proposed to evaluate the significance of word w_i in thread j:

$$\mathrm{WR1}_{i,j} = (P^{\mathrm{w}}_{i,j})^{\lambda_1} \left(P^{\mathrm{w}}_{i,j} \Big/ \sum_{k=1}^{K} P^{\mathrm{w}}_{i,k} \right)^{\lambda_2}, \tag{1}$$

where $P^{\mathrm{w}}_{i,j}$ is generated by LDA according to term frequency, K is thread number, and λ_1, λ_2 are control parameters to balance influence of two terms. Specifically, if $\lambda_1 = 1$ and $\lambda_2 = 0$, this score is just identical with original LDA term frequency. In contrast, if $\lambda_1 = 0$ and $\lambda_2 = 1$, it's the same as the technique used in the baseline[25]. We use $\lambda_1 = 1$ and $\lambda_2 = 1$ in our experiment. In this equation, the first term simulates the term frequency and the second one simulates inverse document frequency to identify words that are discriminative for threads.

Thread Coverage Judge An underlying assumption of WR1 is: every thread is equally important. However, threads that cover a significant portion of the corpus content are usually more important than those covering little content. Thread's coverage is introduced to reflect how much contents a thread covered. It is quantified as

$$\mathrm{TW}_j = \frac{1}{M} \sum_{d=1}^{M} P^{\mathrm{d}}_{d,j}, \tag{2}$$

where $P^{\mathrm{d}}_{d,j}$ stands for thread j's probability on report d, and M is the number of news reports.

Combining with thread coverage, the second judge is proposed:

$$\mathrm{WR2}_{i,j} = (P^{\mathrm{w}}_{i,j})^{\lambda_1} \left(P^{\mathrm{w}}_{i,j} \mathrm{TW}_j \Big/ \sum_{k=1}^{K} P^{\mathrm{w}}_{i,k} \mathrm{TW}_k \right)^{\lambda_2}. \tag{3}$$

In the equation above, thread coverage is introduced into IDF-like term. Compared with WR1, terms in a thread covering more content become more important.

PMI Judge PMI is a common approach to evaluate the semantically relevance between words in a pair. It quantifies the discrepancy between the probabilities of their coincidence versus individual distributions.

$$\mathrm{PMI}(w_i, w_j) = \lg \frac{P(w_i, w_j)}{P(w_i)P(w_j)}. \tag{4}$$

The word has a higher PMI and could stand for the thread. Since PMI reflects the "semantic distance" between words in a pair, average PMI could measure how "close" a word is with others. Based on an intuitive assumption that: a word "closer" to the others is more representative for the thread. Average PMI, denoted by $\overline{\mathrm{PMI}}$, is measured for top-10 words.

$$\overline{\mathrm{PMI}}(w_i) = \frac{1}{9} \sum_{i \neq j} \overline{\mathrm{PMI}}(w_i, w_j). \tag{5}$$

Then, the third judge is proposed:

$$\mathrm{WR3}_{i,j} = (P^{\mathrm{w}}_{i,j})^{\lambda_1} \left(P^{\mathrm{w}}_{i,j} \Big/ \sum_{k=1}^{K} P^{\mathrm{w}}_{i,k} \right)^{\lambda_2} \overline{\mathrm{PMI}}(w_i)^{\lambda_3}. \tag{6}$$

We set $\lambda_1 = 1$, $\lambda_2 = 2$, $\lambda_3 = 1$ in our experiment.

2.2 Thread Label Extraction

Compared with unigram, phrases are more meaningful to label threads. In this step, phrases are extracted based on thread words. A candidate phrase (w_i, w_{i+1}) will be extracted from z and at least one of w_i, w_{i+1} belongs to z'. z is the top 10 result of original LDA, and z' is the thread words set.

First, each word's weight $\pi(w_i)$ in z is re-weighted. Words in $z - z'$ and z' are treated differently. The weight of thread words $w_i \in z'$ is calculated according to the following equation:

$$\pi(w_i) = \alpha \xi(w_i) \Big/ \sum_{k=1}^{3} \xi(w_k), \tag{7}$$

where α is the re-weight factor and $\xi(w_i)$ is generated in the thread words selection.

The weight of other words $w_i \in z - z'$ is reweighted according to its semantic distance with thread word. It's calculated as below:

$$\pi(w_i) = (1-\alpha)\overline{\mathrm{PMI}}(w_i) \Big/ \sum_{k=4}^{10} \overline{\mathrm{PMI}}(w_k), \tag{8}$$

$$\overline{\mathrm{PMI}}(w_i) = \frac{1}{3} \sum_{k=1}^{3} \mathrm{PMI}(w_i, w_k). \tag{9}$$

Re-weight factor is set as $\alpha = 0.5$ in our experiment.

Next, a recursive phrase extraction procedure is performed. In this procedure, a candidate phrase is weighted by its components' weight and co-occurrence:

$$\pi(w_i, w_j) = 1.1 \left[\pi(w_i) \frac{n(w_i, w_j)}{n(w_i)} + \pi(w_j) \frac{n(w_i, w_j)}{n(w_j)} \right], \tag{10}$$

where $n(w_i, w_j)$ is the occurrence number of phrase (w_i, w_j). All candidate phrases are recursively tested until there are no more phrases whose weight is larger than the threshold. Threshold is set as $\varepsilon = \pi(w_{10})$ in our experiment. A complete procedure is illustrated as below:

```
set flag = true
while flag do
    extract all candidate phrases (w_i, w_j)
    and count occurrence number n(w_i, w_j)
    for all (w_i, w_j) do
        calculate phrase weight π(w_i, w_j)
    end for
    select (w_i, w_j) with the highest weight
    if π(w_i, w_j) ⩾ ε then
        add (w_i, w_j) into z'
        update π(w_i) and π(w_j) by
        π_new(w_i) = π_old(w_i) [1 − n(w_i, w_j)/n(w_i)]
    else
        flag = false
    end if
end while
select phrase or word with the highest weight in z'
as thread label
```

At each time, a candidate phrase with the highest weight larger than threshold will be added into z'. It can be used to form a longer phrase. For example, phrase "literature prize" may be generated at first iteration, then it may form a new phrase "Nobel literature prize" in later iterations.

3 Experiments

3.1 Experimental Settings

Thread Words Selection and Thread Label Extraction have been tested on two different data sets.

Chinese news corpus is a event based corpus, which contains 147 event sub-corpora, such as "2007 Nobel prize". Each sub-corpus is generated by clustering method. The number of news reports in a sub-corpus varies from 20 to 420. There are totally 31 420 news reports in this Chinese news corpus. Another corpus is Reuter-21578 financial news corpus, which is a general corpus contained many events. Five sub corpora are selected from it, they are "crude", "grain", "interest", "money-fx" and "trade". Each of them contains more than 300 reports which describe many events.

The experiments were run using LDA with 500 iterations of Gibbs sampling. The number of threads were empirically set, 5 for each event sub-corpus of Chinese corpus, 40 for each Reuter sub-corpus.

3.2 Evaluation for Thread Words Selection

There is no golden standard for news thread extraction. Only humans can identify and understand news threads for different news events. The performance of three thread words selection methods is evaluated on news event corpus (Chinese) and general corpus (Reuter).

Baseline One baseline is the re-ranking method KR1 proposed by Ref. [25], which is an IDF-like score:

$$\mathrm{KR1}_{i,j} = P^{\mathrm{w}}_{i,j} \bigg/ \sum_{k=1}^{K} P^{\mathrm{w}}_{i,k}, \tag{11}$$

where $P^{\mathrm{w}}_{i,j}$ is generated by LDA according to term frequency. KR1 simulates inverse document frequency to re-weight the native weights.

LDA top-3 is another baseline in our experiment.

Performance Measures

(1) HumanJudge. For each thread, the top three words generated by our methods and baseline are evaluated by voluntary judgers. Judgers are asked to select the most representative one among them, multi-selection is permitted. The set selected by judger will be tagged as “correct”. We calculate percentage of correct thread word set generated by different methods for comparison.

(2) F_1-measure. Judgers are asked to select best-3 words from LDA top-10 term list for each thread. These best-3 words are regarded as correct words. We calculate precision, recall and evaluate results using F_1-measure:

$$\text{Precision} = \frac{\text{Words selected and correct}}{\text{Total words selected}}, \tag{12}$$

$$\text{Recall} = \frac{\text{Words selected and correct}}{\text{Total words correct}}, \tag{13}$$

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{14}$$

HumanJudge (HJ) results and F_1-measure results are summarized in Table 1. Both measures show that our methods perform much better than the baselines. As mentioned above, thread words should be representative for the thread and discriminative across threads. LDA’s original order ranks those common words with high probability. That’s why LDA’s native result has a low accuracy in Table 1. Similarly, KR1 baseline considers the IDF-like feature to identify words that are discriminative for threads. It ignores words’ probabilities. Our methods combine these two aspects and select significant thread words.

After analyzing content of thread word sets generated by different judges, there are about 80% content of WR1 and WR2 are identical. The reason is that in WR2 differences of $P^{\mathrm{w}}_{i,j}$ over different thread j are much larger than the differences of threads’ coverage TW_j, thread coverage does not contribute the result.

Table 1 HumanJudge and F_1 for thread words selection on two corpora (%)

Measure	Corpus	LDA	Baseline	WR1	WR2	WR3
HJ	Chinese	34.0	36.8	68.9	71	71
	Reuter	27.8	30.2	54.8	54.8	56
F_1	Chinese	42.6	46.6	66.6	67.4	67.5
	Reuter	40.6	40.6	62.6	63.6	63.6

Further, we noticed that person names in political news always appear with other words and have much higher PMI score. PMI score dominates final result and should be weaken. Different categories of events may have different characteristics, the control parameters’ value need to be varied for different categories.

3.3 Evaluation for Thread Label Extraction

Performance measures (HumanJudge) People are asked to judge whether the label could represent the thread or not. A representative label is tagged as correct.

Thread label evaluation is performed individually on WR1, WR2 and WR3. The precision of manually check is showed in Table 2.

Table 2 The precision of correct thread labels

Correct thread label	WR1/%	WR2/%	WR3/%
Chinese corpus	54.8	55.8	59.3
Reuter corpus	45.3	45.3	47.3

The result shows that label extraction based on WR3 performs the best. The reason is that WR3 has the highest precision in thread words selection.

The precision of label extraction is also computed based on correct thread words. The result is shown in Table 3. Thread label extraction performs similar on both corpora. Result based on WR3 is still the best. It indicates that WR3 assign reasonable weight for words which has improved the precision of phrase extraction. Experiments showed that labels for Reuter corpus is usually a single word. A word’s meaning is more general than a phrase, humans are more likely to judge a

Table 3 The precision of thread labels based on correct thread words

Correct thread label	WR1/%	WR2/%	WR3/%
Chinese corpus	79.5	78.6	83.5
Reuter corpus	82.7	82.7	84.5

word label to be correct than a phrase. Therefore, the precision on Reuter corpus is a slightly higher than the result on Chinese corpus in Table 3.

3.4 Comparison with "Turbo Topics"

"Turbo topics" (source code from http://www.cs.princeton.edu/∼blei/topicmodeling.html) is experimented on Reuter corpus for comparison. Tables 4 and 5 show our thread labeling results and "turbo topic" results for Reuter "interest" and "money-fx" corpus. The first column lists the phrases generated by "turbo topics". The second column lists the phrases generated according to the method in Subsection 2.2 while the phrase with bold face is the thread label. The third column is the threads words selected according to the method in Subsection 2.1. The fourth column lists the original LDA top 10 words.

Table 4 Turbo topics vs our method for "interest" (thread number is 5)

Turbo topics	Label extraction	Thread word	LDA result
Lend rate	Prime rate	Rate	Rate, pct, bank
Point cut	Pct	Pct	Say, point, cut, prime
Pct effective	Bank raise	Raise	Lend, raise, april
Dealer say	Federal reserve	Reserve	Say, reserve, fund
Billion mark	Federal fund	Fund	Pct, billion, feed, federal
Federal reserve	Feed fund	Federal	Dlrs, march, money
Band pct	Mln stg	Mln	Mln, bank, stg
London march	Money market	Stg	Pct, market, march, billion
March bank	Billion stg	Money	Money, today, london
Central bank	Official say	Say	Say, bank, year
	Bank say	Bank	Debt, official, meet, credit
	Bank meet	Official	Exchange, new, paris
Reuters author	Analyst say	Say	Say, rate, market
Analyst say	Sterling rise	Rise	Rise, high, cut, analyst
Base rate	Say sterling	Analyst	Bond, dollar, sterling

Table 5 Turbo topics vs our method for "money-fx" (thread number is 5)

Turbo topics	Label extraction	Thread word	LDA result
Industrial nation	Offical say	Say	Say, exchange, paris, rate
Minister kiichi	Paris exchange rate	Paris	Currency, nation, country
Baker say	Say	Official	Tell, economic, official
Billion dlrs	Economist say	Say	Say, pct, rise, year
Government security	Pct rise	Rise	Dls, billion, government
Reuters author	Price	Price	Economist, price, trade
Billion mark	Mln stg	Mln	Bank, mln, stg, pct
Repurchase agreement	Money market	Stg	Market, billion, money
Bank japan	Billion stg	Market	March, today, reserve
Dealer say	Dollar	Dollar	Dollar, say, yen, bank
Tokyo march	Yen	Yen	Dealer, trade, japanese
New york	Dealer say	Dealer	Japan, currency, market
Rate cut	Exchange rate	Rate	Rate, bank, say, exchange
Commercial bank	Central bank	Bank	Mark, foreign, currency
Foreign exchange	Foreign exchange	Exchange	Central, set, bundesbank

Based on the comparison with "turbo topics", some results are concluded as follows.

Some meaningless phrases still exist in "turbo topic" and our method, such as "official say", "bank say" or "dealer say" extracted from the "interest" corpus. Both methods are based on the result of LDA, therefore they can not filter meaningless phrases. Both methods extract some common phrases, such as "federal reserve", "analyst say".

The semantics of phrases in our method is more coherent than those phrases in turbo topics, such as "federal reserve", "federal fund","feed fund" are more coherent than "dealer say", "billion mark","federal reserve" int the Table 4. "Turbo Topic" extends LDA from unigram into n-gram. Our method first select the thread words and then generate phrases by extending these thread words, which keep the semantics of the thread. Table 6 lists two events with their thread labels and reports' titles.

Table 6 Thread labels and news reports for two event corpora

Event corpus	Thread labels	News report titles
2008 USA election	Foreign policy	Barack Obama's Foreign Policy
		Obama plan to visit Iraq and Afghanistan
		Obama's Iraq problem
	Financial crises	Is Obama the Wall Street Candidate?
		Bill v. Barack on Banks
		Obama's plan for financial markets reform
	Senator obama	Senator Obama with USA Troops in Kuwait
		Senator Obama's tobacco control history
		Senator Barack Obama for president
Iran nuclear program	Security Council	Options for the Security Council
		Iran ends cooperation with IAEA
		Iran likely to face Security Council
	Enrich uranium	Iran to Begin Enriching Uranium
		Iran' on brink' of being nuclear nation
		Iran enriches Uranium
	Bush administration	Rice: Iran can have nuclear energy, not arms
		Bush plans strike on Iran's nuclear sites
		Iran details nuclear ambitions

4 Conclusion

In this paper, a method is presented to label threads of news events. Experiments indicate that our method of thread words selection performs significantly better than the baselines. WR3 is proved to be the best among three judges. Based on the selected thread words, our recursive phrase extraction can generate meaningful bigram or trigram phrases as thread labels to visualize the semantic meaning of a news thread from news corpus, the result is better than the phrases generated by "turbo topics". Experiments also show that the method works not only on event-based corpus but also on general corpus.

In the future work, we plan to combine the elements of news event such as "who what when" with our existing methods and improve the precision of label extraction in order to visualize the semantic meaning within a news event.

References

[1] CNNIC. The 28th statistical report on the Internet development in China [R]. Beijing, China: CNNIC, 2011 (in Chinese).

[2] MEI Q, SHEN X, ZHAI C. Automatic labeling of multinomial topic models [C]//*Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. San Jose, California, USA: ACM, 2007: 490-499.

[3] BLEI D M, NG A Y, JORDAN M I, et al. Latent dirichlet allocation [J]. *Journal of Machine Learning Research*, 2003, **3**: 993-1022.

[4] NALLAPATI R, FENG A, PENG F, et al. Event threading within news topics [C]//*Proceedings of the Thirteenth ACM International Conference on Information*

and Knowledge Management. Washington, DC, USA: ACM, 2004: 446-453.

[5] Feng A, Allan J. Finding and linking incidents in news [C]//*Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management*. Lisboa, Portugal: ACM, 2007: 821-830.

[6] Wang X, McCallum A. Topics over time: A non-Markov continuous-time model of topical trends [C]//*Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Philadelphia, PA, USA: ACM, 2006: 424-433.

[7] Mei Q, Liu C, Su H, et al. A probabilistic approach to spatiotemporal theme pattern mining on weblogs [C]//*Proceedings of the 15th International Conference on World Wide Web*. Edinburgh, Scotland: ACM, 2006: 533-542.

[8] Mei Q, Zhai C. A mixture model for contextual text mining [C]//*Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Philadelphia, PA, USA: ACM, 2006: 649-655.

[9] Wang C, Zhang M, Ma S, et al. Automatic online news issue construction in web environment [C]//*Proceeding of the 17th International Conference on World Wide Web*. Beijing, China: ACM, 2008: 457-466.

[10] Xu R, Peng W, Xu J, et al. On-line new event detection using time window strategy [C]//*The Proceeding of International Conference on Machine Learning and Cybernetics (ICMLC)*. Guilin, China: IEEE, 2011: 1932-1937.

[11] Shen D, Yang Q, Sun J, et al. Thread detection in dynamic text message streams [C]//*Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Seattle, WA, USA: ACM, 2006: 35-42.

[12] Kim J, Candan K, Dönderler M. Topic segmentation of message hierarchies for indexing and navigation support [C]//*Proceedings of the 14th International Conference on World Wide Web*. Chiba, Japan: ACM, 2005: 322–331.

[13] Fung G P C, Yu J X, Liu H, et al. Time-dependent event hierarchy construction [C]//*Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. San Jose, California, USA: ACM, 2007: 300-309.

[14] Trieschnigg D, Kraaij W. Hierarchical topic detection in large digital news archives [C]//*Proceedings of the 5th Dutch Belgian Information Retrieval Workshop*. Utrecht, The Netherlands: University of Twente, 2005: 55-62.

[15] Kleinberg J. Bursty and hierarchical structure in streams [J]. *Data Mining and Knowledge Discovery*, 2003, **7**(4): 373-397.

[16] Turney P. Coherent keyphrase extraction via web mining [C]//*Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI-03)*. Acapulco, Mexico: Morgan Kaufmann Publishers, 2003.

[17] Ong T, Chen H, Sung W, et al. Newsmap: A knowledge map for online news [J]. *Decision Support Systems*, 2005, **39**(4): 583-597.

[18] Chang J, Boyd-Graber J, Gerrish S, et al. Reading tea leaves: How humans interpret topic models [C]//*Proceedings of the 23rd Annual Conference on Neural Information Processing Systems*. Vancouver, British Columbia, Canada: Curran Associates Inc., 2009.

[19] Pantel P, Ravichandran D. Automatically labeling semantic classes [C]//*Proceedings of HLT/NAACL*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2004: 321-328.

[20] Yang Y, Pedersen J. A comparative study on feature selection in text categorization [C]//*Proceedings of the Fourteenth International Conference on Machine Learning (ICML'97)*. Palo Alto, California, USA: AAAI Press, 1997: 412-420.

[21] Gabrilovich E, Markovitch S. Overcoming the brittleness bottleneck using wikipedia: Enhancing text categorization with encyclopedic knowledge [C]//*Proceedings of the 21st National Conference on Artificial Intelligence*. Palo Alto, California, USA: AAAI Press, 2006: 1301-1306.

[22] Carmel D, Roitman H, Zwerdling N. Enhancing cluster labeling using wikipedia [C]//*Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. Boston, Massachusetts, USA: ACM, 2009: 139-146.

[23] Lau J H, Newman D, Karimi S, et al. Best topic word selection for topic labelling [C]//*Coling 2010: Posters*. Beijing, China: Coling 2010 Organizing Committee, 2010: 605-613.

[24] Lau J, Grieser K, Newman D, et al. Automatic labelling of topic models [C]//*Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011: 1536-1545.

[25] Song Y, Pan S, Liu S, et al. Topic and keyword re-ranking for LDA-based topic modeling [C]//*Proceedings of the 18th ACM Conference on Information and Knowledge Management*. New York, NY, USA: ACM, 2009: 1757-1760.

[26] Blei D, Lafferty J. Visualizing topics with multiword expressions [EB/OL]. (2009-07-06) [2011-07-07]. http://arxiv.org/abs/0907.1013.

[27] Wilson A T, Chew P A. Term weighting schemes for latent dirichlet allocation [C]//*Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2010: 465-473.

[28] Wang X, McCallum A, Wei X. Topical n-grams: Phrase and topic discovery, with an application to information retrieval [C]//*Seventh IEEE International Conference on Data Mining*. Omaha, NE, USA: IEEE, 2007: 697-702.