A Close-to-linear Topic Detection Algorithm using Relative Entropy based Relevance Model and Inverted Indices Retrieval

Steve Kansheng Shi
College of Electrical and Electronic Engineering, Shanghai Jiaotong University, 200240 Shanghai
Email:steve@joinvc.com
www.sjtu.edu.cn

Lemin Li
Fellow, Chinese Academy of Engineering, 100088 Beijing
Email:lml@uestc.edu.cn
www.cae.cn

Received 5 September 2011

Accepted 5 July 2012

Abstract

Although timely access to information is becoming increasingly important and gaining such access is no longer a problem, the capacity for humans to assimilate such huge amounts of information is limited. Topic Detection(TD) is then a promising research area that addresses speedy access of desired information. However, ironically, the time complexity of existing TD algorithms themselves is usually *$O(n^3)$ or* up to the *x*-th power of *e*. Linear performance requirement of real world topic detection has not been significantly addressed. This paper reveals a new patented topic detection algorithm called **RMIR** that combines **r**elevance **m**odel with **i**nformation **r**etrieval technique to improve on time efficiency. Relevance Model(RM) is a theoretical extension of statistical language modeling that was developed for the task of document retrieval. To reduce the costs of fetching RM, we reduce the number of comparisons for stories by a query-based approach that makes similar stories exist in the top-k query results. We also build our query based on inverted indices, which have the complexity close to linear. The time cost of rest of operations in the **RMIR** topic detection process is a constant. Hence, the total complexity of **RMIR** topic detection algorithm should be close to linear as shown in experimental results. In addition, **RMIR** also gains better detection rates and robustness by relative entropy based topic model design**.**

Keywords: Topic Detection; Link Topic Detection; Retrospective Event Detection; Information Retrieval; Relevance Models; Inverted Indices

1. Introduction

Although timely access to information is becoming increasingly important in today's knowledge-based economy, gaining such access is no longer a problem because of the widespread availability of broadband in both homes and businesses. Ironically, high-speed connectivity and the explosion in the volume of digitized textual content available online has given rise to a new problem, namely, information overload. Clearly, the capacity for humans to assimilate such vast amounts of information is limited. Topic Detection (TD) has emerged as a promising research area that harnesses the power of modern computing to address this new problem.

A topic is defined as a seminal event or activity, along with all directly related events and activities [2]. Thus, we can infer that a topic consists of events and activities, both of which are defined in greater detail [1]. A Topic Detection and Tracking(TDT) event is defined as something that happens at a specific time and place, along with all the necessary preconditions and unavoidable consequences [1]. Such an event might be a car accident, a meeting, or a court hearing. A TDT activity is a connected series of events with a common focus or purpose that happens in specific places[1]. TD enables the automatic discovery of new topics from a news corpus and the subsequent assignment of news documents to discovered topics[9]. A new topic typically

Published by Atlantis Press
Copyright: the authors

corresponds to a newsworthy incident such as the 2012 US presidential elections.

Moreover, TDT includes several evaluation tasks, each of which explores one aspect of that organization{i.e., splitting a continuous stream of news into stories that are about a single topic (“segmentation"), gathering stories into groups that each discuss a single topic (“detection"), identifying the onset of a new topic in the news (“first story detection"), and exploiting user feedback to monitor a stream of news for additional stories on a specified topic (“tracking"). Topic Detection (TD) is an important sub-task of TDT and mainly consists of clustering news stories as topics.

We believe the performance issue of TD itself is significant to gain timely access to desired information from overloaded digitized textual contents.

Topic detection is essentially text clustering. Its main characteristic is the finer granularity of clusters and such clustering is no longer a spherical shape because the topic drift may occur. In addition, clustering on a larger scale is no longer suitable for memory-based algorithms. In fact, text clustering has two common methods, i.e., partition-based approach and hierarchical based approach. The former method is simple and fast convergent, but its parameter *K* is difficult to be determined. Unfortunately, the topic number of text clustering is uncertain in TD. In addition, it is not suitable for topic drifting cases. Existing TD solutions have higher complexity of $O(n^2)$ or $O(n^3)$ usually [3,4,5,6,7,8,9,10,13,15]. In ref. 14, we can observe its TD performance even went up to the *x*-th power of *e*.

Despite the fact that existing TD solutions play important roles in their applications, they do not explicitly incorporate language model and document retrieval model into their formulations. Based on our researches [16,17,18,19] , we start to view TD from an angle of information retrieval(IR) as there are lots of established high performance IR models and algorithms. Enlightened by achievements in IR field, we adopt Relevance Model(RM) for our language model and Inverted Indices for document retrieval. RM is a theoretical extension of statistical language modeling and applicable in both retrieval and TD[26]. Inverted Indices are well known technique widely adopted in document retrieval and full text search engines[32,33]. By treating news and stories as documents, we can use IR methods to retrieve relevant news or stories and form them into TD’s clusters. This paper hence aims to combine relevance language model with inverted indices retrieval technique. Moreover, by creating relative entropy for RM, we have better topic detection rates while achieving linear TD performance.

The remainder of this paper is organized as follows: In Section 2, we define key concepts and terms and introduce works that are directly related to the ideas presented in this paper. Section 3 describes our novel approach for TD via relative entropy based RM and inverted indices modeling. In Section 4, we demonstrate the superiority of our approach with comparative empirical results for both English and Chinese news. Finally, in Section 5, we present our conclusions.

2. Formal Representation

Although there are many language tracking and modeling methods based on machine learning, thus far, the vector space model(VSM) has achieved the best results[11].VSM has been successfully applied on such as well-known SMART text retrieval system. There are a number of formal ways of describing relevance feedback, beginning with the notion of an “optimal query” used in the SMART system. The biggest advantage of VSM is to simplify the text as the vector representation by its features and weights.

2.1. *Document Representation*

Contents of the document are expressed by a number of feature items, which generally include the basic linguistic units, such as words or phrases. $Document = D(t_1, t_2, ..., t_n)$, here t_k is a feature item. In a document, each feature item is assigned a weight w_k which denotes the feature item’s degree of importance in the document:

$$D = D(t_1, w_1; t_2, w_2; ...; t_n, w_n) \tag{1}$$

here the weight of t_k is w_k , and $1 \le k \le n$. Given a document $D = D(t_1, w_1; t_2, w_2; ...; t_n, w_n)$, to simplify the analysis, we do not consider the order of t_k in document. $(t_1, t_2, ..., t_n)$ can be viewed as a n dimensional coordinate system, where D is the corresponding coordinate value. So a document can be expressed as a vector of n dimensional vector space. We call expression $D = D(w_1, w_2, ..., w_n)$ as the Vector Space Model of D . The classic weight calculation method is $TF \times IDF$ in statistical methods.

There are many ways to evaluate the significance of a term, ranging from simply identifying its existence to evaluating its distribution level in a document or in a whole corpus in ref.9.20.21. The most common term weighting scheme for processing index terms is $TF \times IDF$, which stands for term frequency—inverse document frequency[11]. $TF \times IDF$ uses the term frequency and inverse document frequency of each feature item to calculate the weight. If tf_{ik} (Term Frequency) represents the number of occurrences of t_k in document D_i , idf_k donates inverse document frequency of t_k , then $TF \times IDF$ is defined as:

$$W_{ik} = tf_{ik} \cdot idf_k \tag{2}$$

here tf_{ik} is a local statistic value which has different values in different documents. idf_k is a global statistic value reflecting a given term's distribution in all data set. IDF 's original definition is as follow:

$$idf_k = \log\left(\frac{N}{n_k}\right) \tag{3}$$

here N represents the number of documents in all data sets, n_k represents the number of t_k that appears in data set. We can see that, the larger idf_k value is, the less the documents which contain the given term. If all documents contain the same given item, idf_k will be 0. In practice, to avoid such a case, equation (3) is improved by equation (4).

$$idf_k = \log\left(\frac{N}{n_k} + constant\right) \tag{4}$$

Generally, constant's value is between 0 and 1, we have the equation (5):

$$idf_k = \log\left(\frac{N}{n_k} + 0.01\right) \tag{5}$$

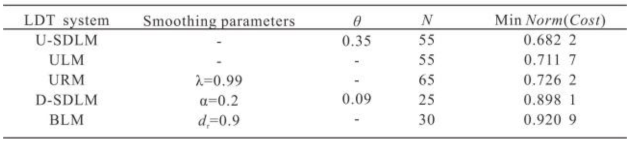
If we take into account the document length on the impact of weights, we have to normalize the feature item weights into the range of [0, 1]:

$$W_{ik} = \frac{tf_{ik} \times \log\left(\frac{N}{n_k} + 0.01\right)}{\sqrt{\sum_{k=1}^{n}\left[(tf_{ik}) \times \log\left(\frac{N}{n_k} + 0.01\right)\right]^2}} \tag{6}$$

2.2. *Unigram Language Model*

VSM's limitations are evident when we set a higher precision or recall rate for TDT work. Based on the traditional similarity model, a sentence vector contains too few words to provide sufficient hints for measuring similarity because a sentence has far fewer key terms to represent the central idea than a paragraph or an entire document[12]. Previous research has demonstrated that it is hard to compare the similarity of sentences when there are insufficient keywords in the sentences being compared, which is a key defeating factor in news event detection[13]. The shortcomings of the simple vector space model approach suggest that there is a need for language model combined with existing metrics. In semantic domain language models (SDLM), by observing the performance records in topic detection for stories in Chinese, U-SDLM (unigram semantic domain language model) achieves the best performance.

Table1 Parameter and minimum normal DET cost



LDT system	Smoothing parameters	θ	N	Min $Norm(Cost)$
U-SDLM	-	0.35	55	0.682 2
ULM	-	-	55	0.711 7
URM	λ=0.99	-	65	0.726 2
D-SDLM	α=0.2	0.09	25	0.898 1
BLM	d_r=0.9	-	30	0.920 9

Table(1) shows the best reported topic detection DET (Detection Error Tradeoff) costs in Chinese stories[24] .

In Unigram model, word is a feature item and the weight of feature item is expressed by the frequency of the word in a topic. The process of topic detection under this model is described here.

1) Topic is defined as $\vec{T} = (f_{T1}, f_{T2}, ..., f_{Tn})$, here $f_{Tj}(1 \le j \le n)$ represents the feature of topic $\vec{T}$;

2) Follow-up story is defined as $\vec{d} = (f_{d1}, f_{d2}, ..., f_{dm})$, here $f_{di}(1 \le i \le m)$ represents the feature of news story $\vec{d}$;

3) Feature Selection is done by following two steps.

- Stop words are removed,
- According to descending order of word frequency, we take the former *i* words as feature items.

4) In TD research field, **NIST**(National Institute of Standards and Technology) and several universities including Carnegie Mellon University(CMU) have been established benchmarks and corpus for TDT. In this paper, the similarity formula between $\vec{T}$ and $\vec{d}$ is defined as follows by adopting the principle reported by Y．Lo and J．Gauvain of NIST[25]:

$$S(\vec{d},\vec{T})=\frac{1}{L_d}\sum_{w\in d}tf(w,\vec{d})\log\frac{\lambda P(w\,|\,\vec{T})+(1-\lambda)P(w)}{P(w)} \quad (7)$$

here $S(\vec{d},\vec{T})$ is the similarity of $\vec{T}$ and $\vec{d}$. w is the feature item of $\vec{T}$ and $\vec{d}$. $tf(w,\vec{d})$ is the frequency of w in $\vec{d}$. L_d is the whole number of terms in $\vec{d}$. λ is a smooth factor (0, 1) tuned to make the system achieve minimum cost when tracking TDT3 corpus. TDT3 corpus is created by NIST specially to accommodate Chinese news and stories. The smoothing technique is introduced to prevent data sparsity in unigram modeling.

$P(w\,|\,\vec{T})$ is the probability of w in $\vec{T}$.

$$P(w\,|\,\vec{T})=\frac{C(w,\vec{T})}{Nw(\vec{T})}, \quad (8)$$

$C(w,\vec{T})$ is the number of w occurrence in $\vec{T}$, $Nw(\vec{T})$ is the whole number of terms in $\vec{T}$. $P(w)$ is a priori probability of w which is the statistic value in the background corpus.

$$P(w)=\frac{C(w,background)}{N(background)} \quad (9)$$

here $C(w,background)$ is the number of w occurrences in background corpus; $N(background)$ is the whole number of terms in background corpus.

5) According to similarity measurement of NIST, topic detection is then described as the calculation of the similarity between the story and the topic. In other words, if $S(\vec{d},\vec{T})>\theta$, then they are considered as relevant or on-topic, off-topic otherwise.

2.3. *Relevance Model*

Relevance Model[26,27,28,29] establishes a query and then estimates the probability of a document related to this query. RM is basically an extension of unigram language model. Assuming a relevance model R , query Q, word w . $P\left(w|R\right)$ denotes a probability for each word. Hypothesis query and word are samples randomly extracted from the probability distribution. RM estimates the probability $P\left(w|R\right)$ of each word with the query.

The famous probability ranking principle, advocated by Robertson[30], asserts that optimal performance will be achieved if the documents are ranked by the posterior probability that they belong to the relevant class R . Robertson also shows that it is equivalent to rank the documents by the odds of their being observed in the relevant class:

$$P\left(D|R\right)/P\left(D|N\right) \quad (10)$$

Underlying principle of most researches on probabilistic models of Information Retrieval is the probability ranking. Here R represents the class of documents relevant to user's query, and N is the class of non-relevant documents. With w instead of D , we have

$$p\left(D|R\right)=\prod_{w\in D}P\left(w|R\right)\prod_{w\notin D}\left(1-P\left(w|N\right)\right) \quad (11)$$

We suppose a number of query words w is independent with each other, we have:

$$\frac{P\left(D|R\right)}{P\left(D|N\right)}\sim\prod_{w\in D}\frac{P\left(w|R\right)}{P\left(w|N\right)} \quad (12)$$

For $Q=q_1\cdots q_k$, we estimate relevance model R as:

$$P\left(w|R\right)\approx P\left(w|q_1\cdots q_k\right)$$

$$p\left(w|R\right)\approx\frac{P\left(w,q_1\cdots q_k\right)}{p\left(q_1\cdots q_k\right)} \quad (13)$$

$$P\left(w,q_1\cdots q_k\right)=P\left(w\right)\prod_{i=1}^{k}\sum_{M_i\in\mathrm{M}}P\left(M_i|w\right)P\left(q_i|M_i\right) \quad (14)$$

3. Model Design and RMIR Algorithm

3.1. *Description of Model Design*

Kullback Leibler divergence is used to compute Relative Entropy(RE) as relevance measure between topic models to compensate the semantic weakness with similar aim of Ref.31.

$$D(M_1\,||\,M_2)=\sum_{w}P(w\,|\,M_1)\log\frac{P(w\,|\,M_1)}{P(w\,|\,M_2)} \quad (15)$$

M1 and *M2* are the topic models for topic *T1* and *T2* based on RM**.** Two topic models *M1, M2* both contain the word w . The equation (15) shows if the two topic models *M1, M2* have semantic similarity. When value *D* is close to 0, the similarity of two models is high. In order to enhance the robustness of the model, we introduce the Clarity probability[28] for this case when both two models have smaller dissimilarity but they are similar to background corpus. Such a phenomenon is called noise so that it is not a valid topic and should be treated as a noise. Thus, equation (15) becomes the following one:

$$S(M_1 \| M_2) = \sum_{w} P(w \mid M_1) \log \frac{P(w \mid M_2)}{P(w \mid GE)} \quad (16)$$

We use equation (17) in our experiment for more convenience in code design and equation (17) is a conversion of equation (16):

$$D(M_1 \| M_2) = \sum_{w} | P(M_1) - P(M_2) | + (1 - | P(M1) - P(GE) |) \quad (17)$$

3.2. *Fast Inverted Indices*

For each term T, we must store a list of all documents that contain T. This is named as inverted index or inversion table[32,33]. In this paper, we build inverted indices for stories and adopt the method of fast inverted indices with on-line update[33] that allows new on-line story stream to update the dataset in a real time manner. It is reported that the query on such an index can behave in linear performance [32,33] as shown in Fig. 1.

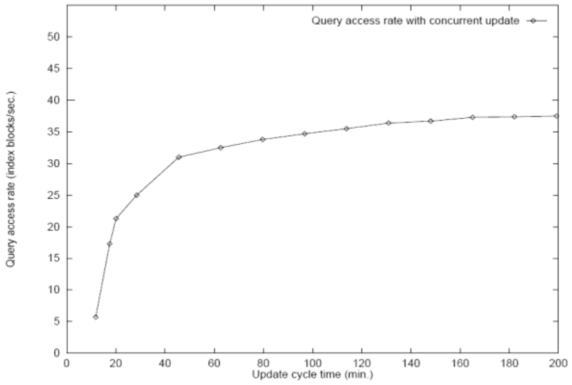


Fig.1 Linear Performance of Fast Inverted Indices

Each pending story should establish its own relevant model. Usually establishing RM is expensive in operation. To avoid this disadvantage, we do it through querying inversion table. Feature item weight is equation (2), i.e., $W_{ik} = tf_{ik} \cdot idf_k$ in our experiment. RM is built on the fly, saving time and space costs while reducing I/O.

3.3. *Blocking Treatment for Retrospective Event Detection（RED）*

Retrospective event detection[34,35] is viewed as a kind of text clustering to detect topic by giving suitable threshold. Text clustering strategy here is single pass clustering. Cluster data stations in the memory by smaller blocks. This leads to relatively small number of comparable objects.

3.4. *Steps of RMIR algorithm*

Our **RMIR** algorithm is carried on according to the following steps.

```
Input: Given a set of stories $\vec{d} = ({}_{d1},{}_{d2},\ldots,{}_{dm})$ that caters for text streams[20];
Output: a set of topics $\vec{T} = ({}_{T1},{}_{T2},\ldots,{}_{Tn})$ ;
Build inverted indices for $\vec{d} = ({}_{d1},{}_{d2},\ldots,{}_{dm})$ based on fast inverted table,
Repeat
   preprocessing of each story
   $d_i = d_i(t_1, w_1; t_2, w_2; \ldots; t_n, w_n)$ ,
   compute weights for each word $W_{ik} = tf_{ik} \cdot idf_k$ ,
   if $d_i$ is the first story
     then
       assign it as the seed topic,
       establish topic model Mi,
       store both the story $d_i$ and topic Mi in memory.
     else
       use feature of $d_i$ to query out the first k stories based on inverted indices,
       from $\vec{d} = ({}_{d1},{}_{d2},\ldots,{}_{dm})$ ,
       build RM for the k stories on the fly in memory, based on
       $D(M_1 \| M_2) = \sum_{w} | P(M_1) - P(M_2) | + (1 - | P(M1) - P(GE) |)$ ,
       compare the k stories,
       select m stories with the lowest value D,
       (here k and m are constants.)
         use
         $S(\vec{d},\vec{T}) = \frac{1}{L_d} \sum_{w \in d} tf(w,\vec{d}) \log \frac{\lambda P(w \mid \vec{T}) + (1-\lambda) P(w)}{P(w)}$
         to calculate similarity of m stories against existing topics,
       record the highest similarity and corresponding topic,
       if the highest similarity $S(\vec{d},\vec{T}) > \theta$ (the preset threshold)
         then
           confirm the story is related with the corresponding topic,
           add the story into the topic,
           update the topic model.
         elseif $S(\vec{d},\vec{T}) < \theta$
           create a new topic seeded by this story,
        add $m$ stories to this new topic,
        update the topic model.
    Until no story comes.
```

4. Experimental Analysis

We implemented a Java-based Hot Topic Extraction cum Business Intelligence system that can be easily deployed on any Java virtual machine (JVM) platform and gathered news reports from standard testbed of NIST's TDT3[40]. Besides, we add on Chinese news from Xinhua News Agency. The experiments tested the viability of the two major components in our work, namely, our new algorithm's TD detection rates and detection performance in both English and Chinese contexts. Detection rates are justified by means of link detection task (LDT) and detection performance is measured by retrospective event detection (RED) .

As shown in Fig. 2, a framework of RMIR(Relevance Model with Information Retrieval) TD engine is illustrated to fully utilize the contents relevant to the detected topic outlines, the context and the user's opinions, creativity, personal knowledge and interests. Detected outline shows the brief description of generating a report. Each outline consists of sentences that are represented as a document or paragraphs. Cluster of events and contents are clustered results based on outline, using RMIR engine. The outline and the report evolve while inspired by the clustered events and contents, as users learn more knowledge and able to optimize and decompose the previous outline based on his experience, opinions, interests and creativity, for achieving a better report. In each iteration, the user has a chance to optimize the outline and the new outline is treated as a new guidance for next clustering task. The user lets the system iterate till the outline and relevant contents satisfy his intuition on the background. It seems that the users are able to comparatively quickly change their minds or outlines after gaining some experience with the investigated stimuli from newly detected topics and retrieved contents[36]. Additionally, the applied cluster analysis reveals that the topics are not homogenous in their opinions, and they form new outline having preference structures similar to users' opinions in a dynamic manner. As in the contemporary society, the meanings and values are communicated in great speed and in a variety of ways, changing the nature of their social behaviors and interactions[37]. For gaining business values, the chance discovery[38] process succeeds in systematically integrating a human's knowledge, experience, interest, and intuition on the background context of the problems with the computer process. Hence, the user finally goes to the process of rephrasing the contents and adding introduction, conclusion, etc., for a final BI report with good readability.

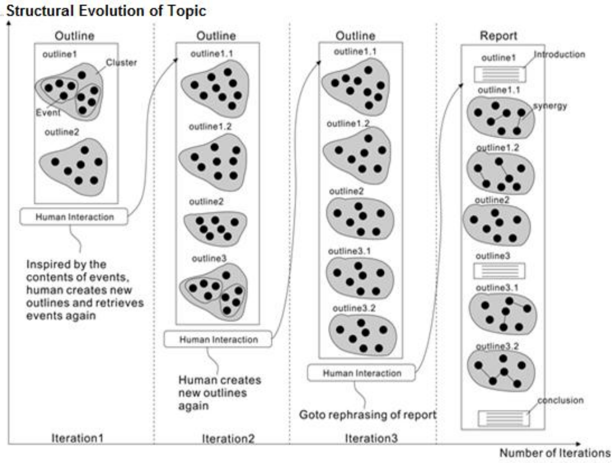


Fig 2. Framework of TD Evolution

We use the standard TDT3 data set of NIST, one of the few news data sets with both class labels and time stamps, released by the TDT community as the testbed. The TDT3 data set includes multilingual news documents collected during the three month period (92 days) of October through December 1998. We use two groups of data in our experiment, learning data with 310 samples with tags and datasets with quantity of 50628 stories. Datasets consist of 38,287 on-line latest stories from Xinhua News Agency and 12,341 stories from Chinese portion of TDT3 corpus. TDT3 consists of 5,153 stories from XIN, 3,817 stories from ZBN and 3,371 stories from VOA_Mandarin. The experiment includes two sub-tasks of TDT that are link detection task (LDT) for justifying detection rates and retrospective event detection (RED) for time complexity. The best results in existing benchmarks and literatures are compared in details.

4.1. *Link Detection Task (LDT)*

310 stories are selected covering 32 topics from Newswire. 20,000 news stories' pairs are selected to observe the experiment results. The performance of **RMIR** is shown as the following DET (detection error

tradeoff) curve. DET is a visualization tool for TDT research. In DET, the vertical axis of two-dimensional coordinates refers to missed detection rate, the horizontal axis refers false alarm rate. The lower the system's false alarm and missed detection are, the better performance it has. The smallest (C_{Det})$_{\text{Norm}}$ of DET curve represents the optimal performance of detection systems.

In fig. 3, bottom left corner curve is the cost curve of this experiment, the smallest performance cost is marked by small circles at 0.04543 which is lower than the best result 0.1047 reported in TDT 2004 benchmark at CMU6[39,40] and shown in table 2. TDT 2004 report is the latest one on public by CMU. In addition, our performance cost is also lower than the best performance of all the algorithms deployed in Chinese documents[18]**.** The best performance cost in Chinese documents is 0.6822 and it also demonstrates the higher difficulty in topic detection task in Chinese. Fig.3 also shows that the missed detection rate is less than 2% when false alarm rate is less than 1%. The upper right corner is a random test performance cost curve.

Table1 is the best LDT results of the research in the Chinese topic detection. As can be seen, this system has higher cost for the upper-right side of the DET curves. Its optimal performance is cost at 0.6822 and far away from the TDT benchmark. This demonstrates the LDT for Chinese topic detection has greater difficulties than that for English ones.

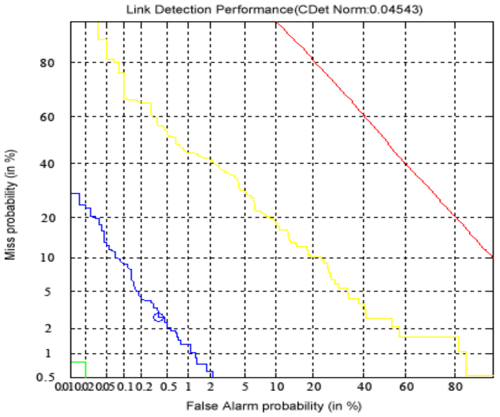
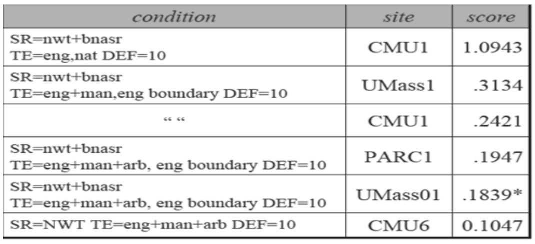


Fig. 3. DET for 310 Stories Covering 32 Topics as Testing Dataset

Table 2 Historical Scores Reported by CMU6



condition	*site*	*score*

SR=nwt+bnasr
TE=eng,nat DEF=10
	CMU1	1.0943

SR=nwt+bnasr
TE=eng+man,eng boundary DEF=10
	UMass1	.3134

" "	CMU1	.2421

SR=nwt+bnasr
TE=eng+man+arb, eng boundary DEF=10
	PARC1	.1947

SR=nwt+bnasr
TE=eng+man+arb, eng boundary DEF=10
	UMass01	.1839*

SR=NWT TE=eng+man+arb DEF=10	CMU6	0.1047

4.2. *TDT Evaluation Metrics*

In order to compare different systems, TDT conference developed a set of evaluation standards. Each participating system performance is expressed a weighting sum of the results of measurement by omission rate and false alarm rate, known as the detection error cost C_{Det} Its equation is as follows.

$$C_{Det} = C_{Miss} \cdot P_{Miss} \cdot P_{target} + C_{FA} \cdot P_{FA} \cdot P_{non-target} \qquad (18)$$

C_{Miss} and C_{FA} are the costs of a missed detection and a false alarm. P_{Miss} and P_{FA} are the probabilities of a missed detection and a false alarm. P_{target} is a priori probability of finding a target. $P_{non\text{-}target}$ = 1－P_{target}。C_{Miss} = 1、C_{FA} = 0.1 and P_{target} = 0.02 are all the default values as coefficients to tune the proportion of missed detection and false alarm in the results of evaluation. Detection cost is usually normalized between 0 and 1 as follows.

$$(C_{Det})_{Norm} = C_{Det} \Big/ \min\{C_{Miss} \cdot P_{t\arg et}, C_{FA} \cdot P_{non-t\arg et}\} \qquad (19)$$

Generally, (C_{Det})$_{\text{Norm}}$ is as an evaluation scores of system performance.

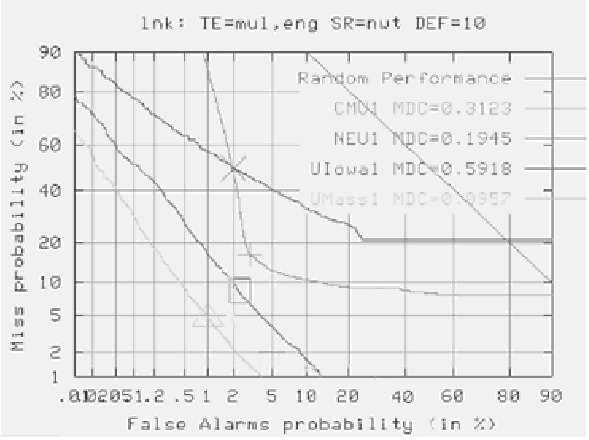


Fig. 4. LDT benchmark including that of Carnegie Mellon University. Our results are better than that of TDT 2003.

In Fig 4, TDT2003 LDT benchmark of NIST is shown. While the false alarm rate is 1%, the missed detection rate is 5%. Our results are better than that of TDT 2003.

4.3. *Retrospective Event Detection*

Retrospective event detection (RED) has to consider not only the system performance but also the effectiveness. RED of **RMIR** is divided into the following steps: 1) data preparation (cutting words); 2) feature selection; 3) retrospective event detection; 4) retrospective event iteration. Based on detecting on 300,600,900,3000,5000,10000,20000,30000,40000,50000 stories, the time costs are shown in Fig. 5.

From fig. 5, we can see that the RMIR algorithm's time complexity is almost linear, that is consistent with our theoretical models and analysis.

Practical deployment of our algorithm in real world patented system in both English and Chinese for municipal government and large enterprises for business intelligence. Because the maximum time consumed for real time topic detection system for all bilingual stories is always less than a constant time, it meets the practical demands of users in the field of business intelligence, community question answering (CQA) [41] , social link management[42,43], learning for personal environment or R&D activities[44,45,46] and hence granted as a distinguished patent.

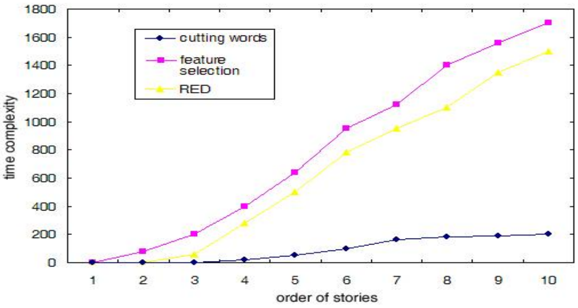


Fig. 5. Close-to-linear Time Costs of RMIR (Action of cutting words is required for stories in Chinese)

5. Conclusion

Although timely access to information is becoming increasingly important and gaining such access is no longer a problem, the capacity for humans to assimilate such huge amounts of information is limited. TD is then a promising research area that addresses speedy access of desired information. However, ironically, the time complexity of existing topic detection solutions themselves is usually $O(n^3)$ *or* up to the *x*-th power of *e*. Linear performance requirement of real world topic detection has not been seriously addressed in literatures. Based on relevance model selection, our RMIR algorithm obtains close-to-linear time complexity and good robustness for stories both in English and Chinese. Even in the dynamic cases of real time feeding of on line stories, RMIR still performs well. To reduce the costs of fetching RM, we reduce the number of comparisons for stories by a query-based approach that makes similar stories exist in the top-k query results. Since we design the query based on inverted table, its complexity is linear or better than linear[33] when quantity of stories increases. The time cost of rest of operations in the RMIR topic detection process is a constant. Hence, in theory, the total time complexity of RMIR topic detection should be close to linear. Experimental results prove that the time complexity of RMIR algorithm is consistent with our theory with the sacrifice of extra space needs for inverted indices. Meanwhile, RMIR achieves better detection rates by comparing with other historical results due to relative entropy based topic model design.

RMIR has been successfully deployed in Topic Detection and Tracking System of Business Intelligence for National Incubation Center and granted a distinguished patent[47] under the patent no. of 200810063295.1.

Acknowledgements

This research is supported by National Natural Science Foundation under the grant numbers 60970107 and 61073150. Topic tracking application project for Business Intelligence (www.wasuo.com) is sponsored by National Incubating Center. We would like to express our sincere thanks to all of them.

References

1. TDT 2004: Annotation Manual Version 1.2, http://www.nist.gov/speech/tests/tdt/, Aug. 2004.
2. The 2004 Topic Detection and Tracking (TDT '04) Task Definition and Evaluation, ,http: //www.nist.gov/ speech/tests/tdt/, 2004.
3. Y. Yang, T. Pierce, and J. Carbonell, "A Study of Retrospective and On-Line Event Detection," *Proc. ACM SIGIR '98*, 1998.

4. J. Allan, V. Lavrenko, and H. Jin, “First Story Detection in TDT Is Hard,” *Proc. Ninth Int’l Conf. Information and Knowledge Management,* 2000.
5. N. Stokes and J. Carthy, “Combining Semantic and Syntactic Document Classifiers to Improve First Story Detection,” *Proc.ACM SIGIR ’01*, 2001.
6. Y. Yang, J. Zhang, J. Carbonell, and C. Jin, “Topic-Conditioned Novelty Detection,” Proc. ACM SIGKDD ’02, 2002.
7. T. Brants, F. Chen, and A. Farahat, “A System for New Event Detection,” *Proc. ACM SIGIR ’03*, 2003.
8. G. Kumaran and J. Allan, “Text Classification and Named Entities for New Event Detection,” *Proc. ACM SIGIR ’04*, 2004.
9. K.K. Bun and M. Ishizuka, “Topic Extraction from News Archive Using TF□PDF Algorithm,” *Proc. Third Int’l Conf. Web Information Systems Eng. (WISE ’02)*, pp. 73-82, 2002.
10. Kuan-Yu Chen, Luesak Luesukprasert, and Seng-cho T. Chou, “Hot Topic Extraction Based on Timeline Analysis and Multidimensional Sentence Modeling”, *IEEE Transactions on Knowledge and Data Engineering*, 19(8), (2007) .1016-1025.
11. G. Salton and C.S. Yang, “On the Specification of Term Values in Automatic Indexing,” *J. Documentation*, pp. 351-372, 1973.
12. N. Okazaki, Y. Matsuo, N. Matsumura, and M. Ishizuka, “ Activation with Refined Similarity Measure,” *Proc. 16th Int’l Florida Artificial Intelligence Research Soc. Conf. (FLAIRS ’03),* pp. 407-411, 2003.
13. H.L. Chieu and Y.K. Lee, “Query Based Event Extraction Along a Timeline,” Proc. 27th Ann. *Int’l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR ’04)*, pp. 425-432, 2004.
14. Qi He, Kuiyu Chang, Ee-Peng Lim, Arindam Banerjee, Keep It Simple with Time: A Re-examination of Probabilistic Topic Detection Models, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 32(10) (2010) 1795 – 1808.
15. Luo Weihua, Liu Qun, Chen XueQi. Development and Analysis of Technology of Topic Detection and Tracking[A]. *Proceedings of The 7th national conference on computational linguistics* (JSCL-2003) (Beijing University Press, 2003), pp.560-566.
16. Kansheng Shi, Naitong Zhang, Lemin Li, et al. Efficient text classification method based on improved term reduction and term weighting, *Journal of China Universities of Post and Communications*, 18(2011) 131-135.
17. Kansheng Shi, Lemin Li, Haitao Liu, et al. A Linguistic Feature Based K-means Text Clustering Method, In *Proceedings of IEEE Cloud Computing and Intelligent Systems*, (2011) 108-112.
18. Kansheng Shi, Lemin Li, Haitao Liu, et al. Improved GA-based Document Clustering Algorithm, *Proceedings of IEEE Broadband and Multimedia Communications*, (2011) 675-679.
19. Kansheng Shi, Lemin Li, Naitong Zhang, et al. An improved KNN text classification algorithm based on density, In *Proceedings of IEEE Cloud Computing and Intelligent Systems* (2011) 113-117.
20. T. Hisamitsu and J.I. Tsujii, “Measuring Term Representativeness,” *Proc. 19th Int’l Conf. Computational Linguistics (COLING ’02)*, vol. 1, pp. 320-326, 2002.
21. T. Hisamitsu and Y. Niwa, “A Measure of Term Representativeness Based on the Number of Co-Occurring Salient Words,” *Proc. 19th Int’l Conf. Computational Linguistics (COLING ’02)*, vol. 1, pp. 1-7, 2002.
22. G. Salton, A. Wong, and C.S. Yang, A vector space model for information retrieval, *Communications of the ACM*, 18(11) (1975) 613–620.
23. G. Salton, *Automatic Information Organization and Retrieval*, McGraw-Hill, 1968.
24. Hong Yu, Zhang Yu, Fan Ji-Li, etc. Chinese Topic Link Detection Based on Semantic Domain Language Model[J]. Journal of Software, 19(9)(2008) 2265- 2275.
25. Y．Lo and J．Gauvain，The LIMSI Topic Tracking System for TDT2001, In *Topic Detection and Tracking Workshop*, Gaithersburg, MD, (2001), National Institute of Standards and Technology.
26. V Lavrenko, J Allan, E DeGuzman. Relevance Models for Topic Detection and Tracking[C]. In *Proceedings of the Human Language Technology Conference*. (2002) 104–110.
27. Changki Lee,Gary Geunbae Lee, Myunggil Jang, Dependency structure language model for topic detection and tracking, *Information Processing and Management* (43) (2007) 1249–1259.
28. W. B. Croft, S. Cronen-Townsend, and V. Lavrenko. Relevance feedback and personalization: A language modeling perspective[C]. In *Proceedings of the DELOS-NSF Workshop on Personalization and Recommender Systems in Digital Libraries*, (2001) 49-54.
29. V. Lavrenko and W. B. Croft. Relevance-based language models. In *Proceedings of ACM SIGIR Conference on Research in Information Retrieval*, (2001) 267-275.
30. S. E. Robertson. The Probability Ranking Principle in IR, *Morgan Kaufmann Publishers,* Inc., San Francisco, California (1997) 281-286.
31. C. Lee, G. Geunbae Lee and M. Jang, Dependency structure language model for topic detection and tracking, Information Processing and Management 43 (5) (2007) 1249–1259.
32. Anthony Tomasic, Hector Garcia Molina, Performance of Inverted Indices in Distributed Text Document Retrieval Systems, *Stanford University Technical Report STAN-CS-92-1434*, (1992).
33. Charles L.A. Clarke, Gordon V. Cormack, Forbes J. Burkowski, Dept. of Computer Science, University of Waterloo, Canada N2L 3G1, *Technical Report CS 94-40,*(1994).

34. Ramadan, Q.H.; Mohd, M.; A review of retrospective news event detection, *2011 International Conference on Semantic Technology and Information Retrieval (STAIR)*, (2011) 209 – 214.
35. Zhiwei Li, Bin Wang, Mingjing Li, Wei-Ying Ma. A probabilistic model for retrospective news event detection. *In Proceedings of SIGIR'2005*. (2005) 106~113.
36. A. Brossard, M. Abed and C. Kolski, Taking context into account in conceptual models using a Model Driven Engineering approach, *Information and Software Technology* 53 (12) (2011) 1349–1369.
37. R. Michalski, Examining users' preferences towards vertical graphical toolbars in simple search and point tasks, Computers in Human Behavior 27(6) (2011) 2308–2321.
38. Y. Maeno and Y. Ohsawa, Human–Computer Interactive Annealing for Discovering Invisible Dark Events, IEEE Transactions on Industrial Electronics, 54(2) (2007) 1184-1192.
39. Loulwah AlSumait, Daniel Barbar ́a, Carlotta Domeniconi, On-Line LDA: Adaptive Topic Models for Mining Text Streams with Applications to Topic Detection and Tracking, Department of Computer Science, George Mason University, *Eighth IEEE International Conference on Data Mining,* (2008) 1550-4786.
40. http://www.itl.nist.gov/iad/mig/tests/tdt/2004/workshop.html.
41. Zhongfeng Zhang, Qiudan Li, QuestionHolic: Hot topic discovery and trend analysis in community question answering systems, *Expert Systems with Applications* (38) (2011) 6848–6855.
42. García-Crespo, A., Colomo-Palacios, R., Gómez-Berbís, J.M., & Ruiz-Mezcua, B., SEMO: a framework for customer social networks analysis based on semantics. *Journal of Information Technology*, 25 (2) (2010) 178-188.
43. García-Crespo, A., Colomo-Palacios, R., Gómez-Berbís, J.M., & García-Sánchez, F.. SOLAR: Social Link Advanced Recommendation System. *Future Generation Computer Systems*, 26 (3), (2010)374-380.
44. Francisco José García Peñalvo, Miguel Ángel Conde González, Marc Alier Forment, María José Casany Guerrero: Opening Learning Management Systems to Personal Learning Environments. *J. UCS* 17(9) (2011) 1222-1240.
45. Colomo-Palacios, R., García-Crespo, Á., Soto-Acosta, P., Ruano-Mayoral, M., & Jiménez-López, D.. A case analysis of semantic technologies for R&D intermediation information management. International *Journal of Information Management*, 30(5) (2010) 465-469.
46. Juan García, Francisco José García Peñalvo, Roberto Therón, Patricia Ordóñez de Pablos: Usability Evaluation of a Visual Modelling Tool for OWL Ontologies. *J. UCS* 17(9) (2011) 1299-1313.
47. Kansheng Shi, Zhangzu Shi, Computer Aided Topic Based Method for Business Intelligence Reporting and Knowledge Base, Granted Patent under the no. of 200810063295.1, 2011.

Copyright: the authors