# IPOMS: an Internet Public Opinion Monitoring System

Jie Ding, Jungang Xu
School of Information Science and Engineering
Graduate University of Chinese Academy of Sciences
Caixa Postal Beijing 2707
China
{dingjie.gucas, xujungang}@gmail.com

***ABSTRACT:*** *In this paper, an IPOMS (Internet Public Opinion Monitoring System) is proposed. This system can collect web pages with some certain key words from Internet news, topics on forum and BBS, and then cluster these web pages according to different 'event' groups. Furthermore, this system provides the function of automatically tracking the progress of one event. With this system, supervisors can know what is exactly happening and what has happened from different views, which can improve their work efficiency a lot. This system is composed of web crawler, html parser and topic detection and tracking tool. Because of the existence of numerous data in web pages, in order to improve efficiency of Internet public opinion analysis, the technologies of web page cleansing and k-d tree algorithm in topic tracking are adopted.*

**Keywords**: Public opinion, Clustering, Topic tracking, Web page cleansing, k-d tree

**Received:** 18 September 2009, Revised 4 November 2009, Accepted 16 November 2009



## 1. Introduction

Public opinion refers to the society and politics attitude toward the social administrator in certain social space [1]. Public opinion online is called Internet public opinion.

There are three characteristics of Internet public opinion. First, Internet public opinion emerges rapidly, with great influence on society. Second, there are large amount of comments to the relevant news and hot topics. Third, different people may have different opinions on the same event due to their different position, personal quality and breakthrough point.

Owing to the above three points, there are three kinds of requirement of monitoring Internet public opinion. First, New public opinion should be found quickly. Second, the progress and change of public opinions should be tracked. Moreover, history and present public opinion situation should be displayed in various formats, so that the supervisor can analyze, research and judge them.

There are several challenges in monitoring Internet public opinion. Technically, it is difficult to obtain a large number of relevant web pages rapidly at first. It is also difficult to judge the relevant degree of two text sections. Moreover, it is difficult to process the public opinion change quickly, because the web pages are enormous in quantity and processing speed is one bottleneck.

Traditional way that public security department processes Internet public opinion is manual, which results in waste of manpower, limited speed of information processing, relatively narrow goal range to research and judge, single form, slow response speed, hard to find relations among information.

In this paper, one new way of monitoring Internet public opinion is proposed, and the corresponding system is designed and realized.

The remainder of this paper is organized as follows. Section 2 discusses the source and concepts of topic detection and tracking. Section 3 describes the architecture of the system. Section 4 proposes the features of the system. Section 5 discusses system evaluation. Section 6 summarizes current work and outlines future work.

## 2. Topic detection and tracking

Topic Detection and Tracking (TDT), originally introduced by Defense Advanced Research Projects Agency (DARPA), is a research program concerned with organizing a stream of broadcast and print news stories by the events that they discuss

[2]. TDT encompasses several tasks, but one of them requires that a system gather arriving news stories into clusters that correspond to real-world events. That task is known in the community as either 'cluster detection' or just 'detection'.

Topic detection and tracking has been widely studied for years [2][3][7][8]. CMU (Carnegie Mellon University) and Umass (University of Massachusetts) have carried on similar research, and have obtained the positive appraisal [2] [3] [4].

### 2.1 Event detection

Event detection can be defined as 'discovering new or not found event in continuous bunch of news' [4], divided into retrospective detection and online detection.

### 2.2 Event tracking

The purpose of event tracking is to sort out following text in previous event [2]. It is a kind of application with categorized files. CMU adopted kNN classification (k-Nearest Neighbor Classification) to do this, and changed kNN in general M-way into 2- way kNN [3].

### 2.3 Web page cleansing

The purpose of web page cleansing is to improve the efficiency of the entire system by means of removing html tags.

## 3. The architecture of IPOMS

In this research, we improve the previous work on topic detection and tracking, and represent the result in Internet browser. This system mainly includes spider, web page parser and detection and tracking tool. Figure 1 illustrates the system architecture of the IPOMS.

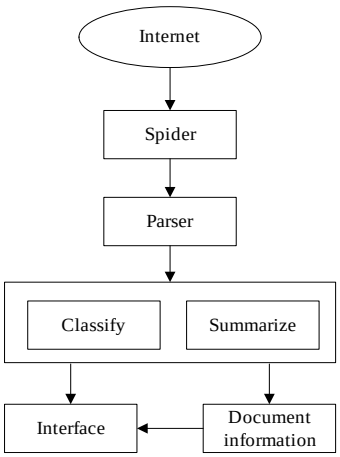


Figure 1. The architecture of IPOMS

### 3.1 Spider

We adopt the network spider to crawl web pages. Most TDT systems are designed for analyzing news stories, which are very "clean". This system can collect web pages from several news sites and big forums.

### 3.2 Web page parser

The collected web pages are in many formats, such as html, shtml, php, which influence the quality of outcome and the efficiency of the entire system, the collected web pages should be cleaned immediately. The main function of the web page parser is to remove 'noise' in web page, leaving web page link, head, title, time, text and first-level title.

We adopt the method of DOM (Document Object Model) tree [4] to get link, title, time, first-level title and text, combining with the method of Embley [5]. We adopt web page parser to construct a DOM tree, set the sub-tree number with maximum covered area number, search the tree with depth first search method, and write the detected text in document. If the leaf node is null, no content is written, else if the leaf node is '\ n', blank is written, which means it has relation with the previous text. When the text under a text node is searched, a tag is written in document, which means the partitioned tag with another section. The first-level title is written in document, under the head, with a "*" as a tag. The tracking and sectioning of text area are shown in Figure 2.

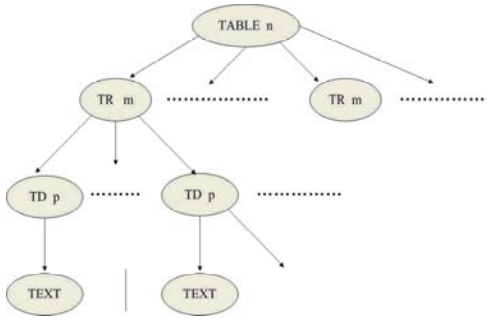


Figure 2. Tracking and sectioning of the text area

### 3.3 Topic detection and tracking tool

#### 3.3.1 Sensitive word list and degree

According to the sensitive words that public opinion experts provide, we build one sensitive word list, and then set the degree for each sensitive word (according to its importance, sensitive degree), and some of them are set as keywords.

#### 3.3.2 Document vector model

With TF-IDF (Term Frequency-Inverse Document Frequency) method, we transfer text into vectors. Calculate formula is shown in formula 1:

$$w_{ij} = tf_{ij} \times idf_i \tag{1}$$

$w_{ij}$ stands for the weight of word i in file j. $tf_{ij}$ stands for the frequency of word i in file j. $idf_i$ stands for the reciprocal of the file frequency of word i.

To strengthen the importance of one keyword in critical location, we enhance the weight of keywords that appear in head, title, first-level title of that web page.

#### 3.3.3 Cluster

We adopt the method CMU proposed to cluster the vectors [2]. First we calculate the similarity degree between every two vectors. We adopt cosine formula to calculate similarity degree:

$$sim(x,c) = \frac{\sum_{j=1}^{M} w_{jx} \times w_{jc}}{\sqrt{(\sum_{j=1}^{M} w_{jx}^2) \times (\sum_{j=1}^{M} w_{jc}^2)}} \tag{2}$$

$sim(x,c)$ stands for the similarity degree between the vector $x$ which comes from a new text and a cluster $c$, $w_{jx}$ stands for the weight of word $j$ in cluster $c$, M stands for total amount of words in the vector space.

We adopt the k-means method to cluster the vectors [5] [6].

#### 3.3.4 Topic detection

Each cluster is viewed as an event, the average weight of which is then calculated. Firstly, we calculate the similarity degree between the vector coming from a new webpage and the existing average vector.

Considering the importance of an event diminishing with the elapsing time, and the same key word may have a completely different meaning, we adopt the time span calculation method. The calculation formula is shown as below:

$$score(x) = 1 - \max_{c_i \in window}\{(1 - \frac{k}{m}) \times sim(\vec{x}, \vec{c_i})\} \tag{3}$$

In this formula, x stands for a new document vector, $c_i$ stands for the number $i$ cluster in the time area, $i$ stands for the total amount of vectors in the vector space, $k$ stands for the number of added document vectors coming between $x$ and the latest one. If the outcome score is greater than the default value, the new file is viewed as one new topic.

Figure 3. shows topic detection flow.

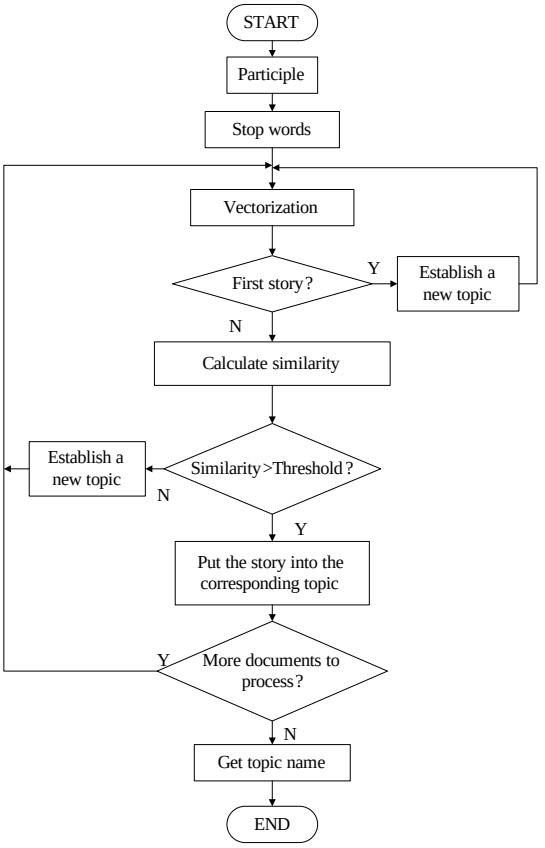


Figure 3. Topic detection flow

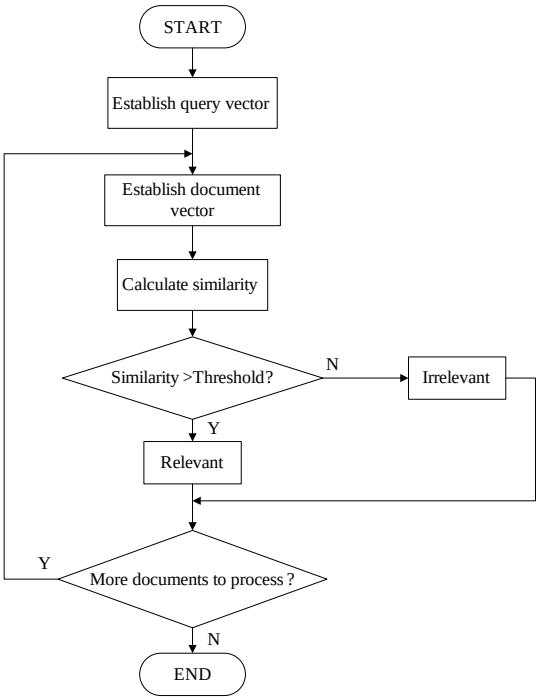


Figure 4. Topic tracking flow

### 3.3.5 Topic tracking

If the similarity degree between the vector coming from a new web page and one of the existing average vectors is lower than the default value, the new vector is viewed as the part of the existing topic.

This is a category process, and we adopt the k-d tree method to do this [7]. The time complexity is more competitive than kNN method [7].

Figure 4 shows topic tracking flow.

## 4. The features of IPOMS

IPOMS has four main features as follows.

1. It can grab web pages from web news sites and forums, which expands the range of monitoring.
2. In the step of parsing web page, it can transfer web pages into different formatted text files, and get web links, head, title, time and first-level title out of the text files. This step can improve the efficiency and accuracy of processing text.
3. It adopts the algorithm of k-d tree instead of kNN, which improves the efficiency of the system a lot.

## 5. System evaluation

### 5.1. Evaluation criterion

The $(C_{Det})_{Norm}$ evaluation metric which is widely used in TDT methods is used to evaluate the performance of our system. System performance, the miss probability and false alarm probability of the topic i (i =1, 2, …, tn, tn is the number of topics) are defined as follows:

$$\text{Miss}_i = \frac{\text{undetected stories about topic i}}{\text{total stories about topic i}} \tag{4}$$

$$\text{FA}_i = \frac{\text{false stories detected about topic i}}{\text{total false stories about topic i}} \tag{5}$$

The average miss probability, average false alarm probability and $(C_{Det})_{Norm}$ are shown as below:

$$P_{Miss} = \sum_i Miss_i / tn \tag{6}$$

$$P_{FA} = \sum_i FA_i / tn \tag{7}$$

$$(C_{Det})_{Norm} = \frac{C_{Miss} \cdot P_{Miss} \cdot P_{T\arg e} + C_{FA} \cdot P_{FA} \cdot P_{-T\arg e}}{\min(C_{Miss} \cdot P_{T\arg e} + C_{FA} \cdot P_{-T\arg e}} \tag{8}$$

$C_{Miss}$ and $C_{FA}$ are the cost of miss and false. $P_{Target}$ is the prior probability of miss and false of target topic, $P_{\text{-}Target} = 1 - P_{Target}$. $C_{Miss}$, $C_{FA}$ and $P_{Target}$ are preseted, and the values vary in different methods. In this paper, these values are 1.0, 0.1 and 0.02.

### 5.2 Experiments

This research adopts the data from sogou lab, which includes 13,560 Chinese reports from October 1, 2007 to March 30, 2008. We consider the first 1,000 stories and corresponding topics as the training linguistic data, and consider the remaining 12,560 stories as test linguistic data, and these stories belong to 20 topics.

The miss probability is 0.3550, the false alarm probability is 0.0097, and $(C_{Det})_{Norm}$ is 0.4012.

(1) Spider

Figure 5 shows the output of spider.

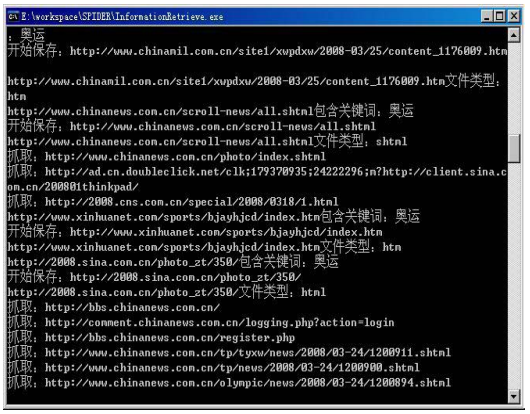


Figure 5. The output of spider

(2) Html parser

Figure 6 shows the output of html parser. The results include link, title, first-level title, and text.

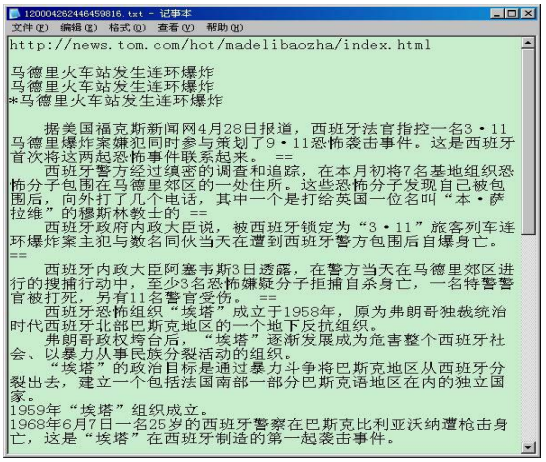


Figure 6. Parsed text

(3) Vectors

Figure 7 shows the vectors.

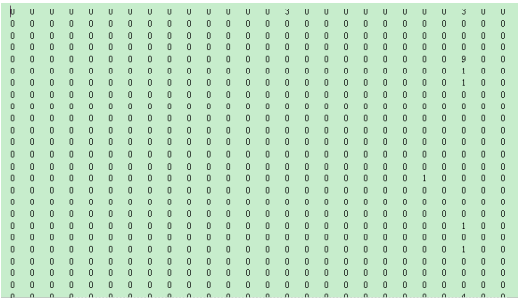

Figure 7. Vectors

(4) Clusters

Figure 8 shows the output of clusters.

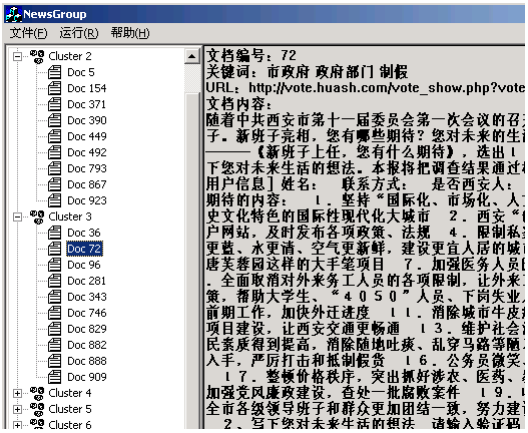


Figure 8. Clusters

## 6. Conclusions

In this paper, we adopt the method of topic detection and tracking, and propose a system which can efficiently monitor the public opinion. This system extends the range of monitoring web, improves the efficiency with web page parser and k-d tree algorithm. The results show that the system can help the supervisor track the progress of hot topics.

## References


[1] Laihua, W., Yi, L (2005). An Overview of China 2004 Public Sentiment Research. Xinhua Digest, 2005, 18, pages 133-134.

[2] James, A., Papka, R., Lavrenko, V (1998). On-line New Event Detection and Tracking. In: Proc. of the 21st International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 37-45. Melbourne, Australia, 1998.

[3] Yiming, Y., Jaime, C., Ralf, D.B., Thomas, P., Brian, T.A., Xin, L (1999). Learning Approaches for Detecting and Tracking News Events. IEEE Intelligent System, 1999, 14(4), pages 32-43.

[4] James, A., Jaime, C., George, D., Jonathan, Y., Yiming, Y (1998). Topic Detection and Tracking Pilot Study: Final Report. In: Proc. of the DARPA Broadcast News Transcription and Understanding Workshop, pages 194-218. Virginia, USA, 1998.

[5] MacQueen, J (1967). Some Methods for Classification and Analysis of Multivariate Observations. In: Proc. of the 5th Berkeley Symposium on Mathematical Statistics and Probability, pages 281-297. Berkeley, California, USA, 1967.

[6] Huajun, Z., Qicai, H., Zheng, CH., Wenying, M., Jinwen, M (2004). Learning to Cluster Web Search Results. In: Proc. of the 27th International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 210-217. Sheffield, United Kingdom, 2004.

[7] Jon, B (1990). K-d Trees for Semi-dynamic Point Sets. In: Proc. of the 6th Annual Symposium on Computational Geometry, pages 187-197. Berkley, California, USA, 1990.

[8] Kuo, ZH., Juan, Z., Ligang, W (2007). New Event Detection Based on Indexing-tree and Named Entity. In: Proc. of the 30th International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 215-222. Amsterdam, Netherlands, 2007.

**Authors Biographies**

**Jie Ding** obtained his B.Sc. degree in Electronic Engineering from Beijing University of Aeronautics & Astronautics in Beijing (China) in 2005 and his M.Sc. degree in Computer Science from Graduate University of Chinese Academy of Sciences in Beijing (China) in 2009. His research interests include topic detection and tracking, information retrieval.

**Jungang Xu** obtained his Ph.D. degree in Computer Science from Graduate University of Chinese Academy of Sciences in Beijing (China) in 2003. He is an Associate Professor in School of Information Science and Engineering, Graduate University of Chinese Academy of Sciences since 2006. His research interests include topic detection and tracking, information retrieval, data management and data mining.