Automatic Online News Issue Construction in Web Environment

Canhui Wang, Min Zhang, Shaoping Ma, Liyun Ru

State Key Lab of Intelligent technology & systems,
Tsinghua National Laboratory for Information Science and Technology,
CS&T Department, Tsinghua University, Beijing, 100084, China P.R.

wangcanhui@gmail.com

ABSTRACT

In many cases, rather than a keyword search, people intend to see what is going on through the Internet. Then the integrated comprehensive information on news topics is necessary, which we called news issues, including the background, history, current progress, different opinions and discussions, etc. Traditionally, news issues are manually generated by website editors. It is quite a time-consuming hard work, and hence real-time update is difficult to perform. In this paper, a three-step automatic online algorithm for news issue construction is proposed. The first step is a topic detection process, in which newly appearing stories are clustered into new topic candidates. The second step is a topic tracking process, where those candidates are compared with previous topics, either merged into old ones or generating a new one. In the final step, news issues are constructed by the combination of related topics and updated by the insertion of new topics. An automatic online news issue construction process under practical Web circumstances is simulated to perform news issue construction experiments. F-measure of the best results is either above (topic detection) or close to (topic detection and tracking) 90%. Four news issue construction results are successfully generated in different time granularities: one meets the needs like "what's new", and the other three will answer questions like "what's hot" or "what's going on". Through the proposed algorithm, news issues can be effectively and automatically constructed with real-time update, and lots of human efforts will be released from tedious manual work.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Clustering; H.3.4 [**Systems and Software**]: Performance evaluation

General Terms

Experimentation

Keywords

News issue, Topic detection and tracking, Clustering

1. INTRODUCTION

People often have information needs like "what's new", or "what's hot", or "what's going on". However, it's difficult to form a keyword query and get what they need with the help of search engines in these cases, therefore they may choose to surf news websites. But there still remain problems. Plenty of new contents are generated at all times; various kinds of contents are reported in one website, and a same message is covered in many websites; information in one theme is not well structured together. It will be very helpful if all mess-up information could be gathered, turned into news issues and provided to users. The process has been accomplished manually by news website editors. Two examples have been given in this paper: one is a special report at CNN, concerning about rugby world cup 2007 (http://edition.cnn.com/SPECIALS/2007/news/rugby.worldcup/), and the other talks about the 2008 Presidential Election in the US (http://news.yahoo.com/fc/US/2008_Presidential_Election) at Yahoo.

A news issue contains all topics related to a relatively general theme. Taking the news issue "rugby world cup 2007" for an example, it is linked to the "finals of rugby world cup 2007" topic with a hyperlink. Moreover, a news issue is well organized with all materials concerning some designated theme, such as news stories, images, audios, videos, weblogs and comments etc. Topics mainly last for several days and become valueless in a short time, while news issues often span longer periods and turn into significant records. Users can conveniently be aware of the background, history, current progress, different opinions and discussions etc of the topics through news issues. In a word, news issues have well-arranged Web contents and provide clear materials for reading.

However, creating news issues by hand is a heavy and tedious job. As a result, only the most important major events such as "rugby world cup" and "Presidential Election" are selected by news website editors to generate news issues. Furthermore, it takes so much time that newly appearing information cannot be added immediately to update an existing news issue or create a novel one.

Topic Detection and Tracking (TDT) tasks are intended to structure news stories from newswires and broadcasts into topics [1]. Compared with news issue generation, TDT tasks have similar aims but different results, as a topic is different from a news issue. After long-time studying, the state-of-the-art TDT techniques [2, 3, 4, 5, 6, 7, 8, 13] are still far from satisfying expectation. *Google News* is a kind of practical application that uses the techniques of TDT. In spite of Google's leading position in the Internet industry, Google News Alerts "generates more than 50% false alarms" [2, 14].

More practical algorithms are needed to generate precise news issues instantaneously. There exist the following problems to be investigated: How to bring order to a large amount of Web data automatically according to the themes, precisely and efficiently? How to combine materials from various sources to construct news issues? How to update old news issues and detect new ones instantaneously?

WWW 2008, May 8--12, 2008, Beijing, China.

An automatic online algorithm for news issue construction is proposed in this paper. Information from a lot of news websites, weblog sites and Web forums is gathered and organized as news issues automatically. Newly appearing materials will be fetched soon after issuance and combined to update existing news issues or utilized for a novel one creation.

More details of this algorithm are described as follows. (1) Since a news issue contains several related topics, we cluster news stories into topics, and then cluster topics into news issues. We select a clustering method as our topic detection algorithm. Experiments are made to compare the results of various clustering methods. Parameters in our model are tuned to achieve the best performance. (2) Our topic detection and tracking algorithm firstly clusters newly appearing stories into new topic candidates, and then compares those with previous topics and decides whether the candidates should be used to update old topics or regarded as new ones. (3) News issues are constructed based on the topics in a similar way like our topic and tracking process. News stories added to old topics are added to the corresponding old issues. New topics are clustered into new news issue candidates and the candidates are compared with previous news issues. Old news issues are updated and new ones generated. Data from weblogs and Web forums, images, audios and videos related to the news issues are added finally.

Four results are generated in different time granularities depended on the news issues constructed: the instant result, daily result, weekly result and monthly result. The instant result provides news issues ranked by their updated time and meets the needs like "what's new". The other three provide news issues ranked by their sizes and will answer questions like "what's hot" and "what's going on".

The rest of the paper is organized as follows: Section 2 gives a brief review of related work in Hierarchical Topic Detection. Section 3 presents our topic detection and tracking algorithm. The news issue construction algorithm is proposed in section 4. Section 5 gives the experimental data and evaluation metrics. We describe the experimental results and discuss on them in Section 6, followed by the conclusion and a discussion of future work in Section 7.

2. RELATED WORK

Topic detection and tracking (TDT) techniques have developed for years [2, 3, 4, 5, 6, 7, 8, 13] but there still lack accurate, efficient and practical solutions. Hierarchical Topic Detection (HTD) task was proposed and evaluated in TDT2004. The best HTD result in TDT2004 was achieved by a sample based approach [16] while Trieschnigg and Kraaij concluded that "the results have too little precision to be really useful" [16]. The first step of the algorithm proposed in [16] is "to take a random sample from the corpus" [16], which we consider impractical to be applied in real Web environment. The other approaches used in HTD were mainly variants of the single pass method, kNN and agglomerative clustering algorithms [17, 18]. There are still many problems existing in HTD such as the hierarchical structure and the mechanism of evaluation [15]. As [18] pointed out, overlapping clusters can get better results.

The major part of our work can be regarded as a two-layer topic detection and tracking algorithm for practical use. We suggest it unnecessary to consider complicated hierarchies in the context of the state-of-the-art TDT techniques. We also have not considered overlapping topics currently for simplicity.

Fung et al. proposed a time-dependent event hierarchy construction algorithm which identifies the features related to a query, extracts the documents highly related to the features, "partitions the extracted documents to form events and organizes them in a hierarchical structure" [19].

Our work is different. We construct news issues online automatically without any queries and recommend the results forwardly to users for reading. Experiments are performed using practical data from Web environment. We simulate the process of a practical system and show results in Web pages.

3. TOPIC DETECTION AND TRACKING BASED ON CLUSTERING

A news issue usually contains several related topics. Similar to TDT, the first stage of news issue construction is to organize information from various websites into topics, and so we borrow its name. We propose our topic detection and tracking system based on clustering, considering the characteristics of news appearance. Newly appearing Web pages are firstly clustered into new topic candidates, which are used to track previous topics. A new topic is generated if the candidate is not combined with any previous topic. Topic models are updated after the operation of combination. The whole topic detection and tracking process is performed online automatically.

3.1 Pre-Processing and Page Representation

Web pages crawled by spiders are analyzed to get the main body. Their titles and contents are extracted along with metadata such as publishing date, category, source, images, audios and videos. We call a Web page's content part (including the title) a story, following the name used in TDT.

We split stories into sentences with some punctuation symbols. Redundant sentences are removed as they bring nothing but noises. And then we tokenize words (word segmentation is performed when dealing with Chinese texts), perform part-of-speech tagging, recognize named entities and remove stopwords. Finally a term vector is created for each story.

Incremental TF-IDF model is widely applied to term weight calculation in TDT [3, 5, 7, 13]. We choose our incremental TF-IWF model to weight terms for its steadier performance in our experiments. *WF* (word frequency) of term *w* at time *t* is calculated as:

$$wf_t(w) = wf_{t-1}(w) + wf_{S_t}(w) \quad (1)$$

where S_t means a set of stories coming at time *t*, and $wf_{St}(w)$ means the number of times term *w* appears in the newly appearing stories. $wf_{t-1}(w)$ represents the number of times term *w* appears before time *t*. A training corpus comprised of a sufficient amount of Web pages is used for the calculation of *WF* initially. As it is showed in formula (1), *WF* is updated dynamically at time *t*.

Then each story *d* coming at time *t* is represented as an *n*-dimension vector, where *n* is the number of distinct terms in story *d*. Each dimension is weighted using incremental TF-IWF model and the vector is normalized so that it is of unit length:

$$weight_t(d,w) = \frac{tf(d,w)\log((W_t+1)/(wf_t(w)+0.5))}{\sqrt{\sum_{w' \in d}(tf(d,w')\log((W_t+1)/(wf_t(w')+0.5)))^2}} \quad (2)$$

where *tf(d, w)* means how many times term *w* appears in story *d* and W_t represents the total number of term appearances before time *t*:

$$W_t = \sum_{t_d \le t} \sum_{w' \in d} tf(d, w') \quad (3)$$

t_d in formula (3) means that story d appears at time t.

Titles are weighted and added to the story representation.

3.2 Similarity Calculation

Cosine similarity is used to calculate the similarity between two stories. For story d and d' at time t, their similarity is calculated as:

$$similarity_t(d, d') = \sum_{w \in d \cap d'} weight_t(d, w) * weight_t(d, w) \quad (4)$$

3.3 Topic Detection Using New Stories

New coming stories are clustered into new topic candidates according to their pair-wise similarities, which is similar to the process of Topic Detection in TDT. The clustering results, called as new topic candidates, are compared with previous topics, and then the results will show if they are really "new" or not. Topics are represented as term vectors defined as the arithmetic average of term vectors of all stories within them. Unlike the single pass and reallocation clustering method in [8], we perform topic detection process within new stories for two reasons. First, the same topic stories are more likely to arrive at one time. Clustering within new stories (without the influence of previous stories) is supposed to ensure the same topic stories are put together at a higher probability. Second, the term vector of a topic is more accurate than that of a single story when representing a topic model, which makes it possible to bring better results at the stage of tracking previous topics.

Many clustering methods including state-of-the-art algorithms are tested with the help of cluto toolkits [9]. UPGMA (unweighted pair group method using arithmetic averages), a traditional agglomerative clustering method, is arranged to perform topic detection process because it achieves the best performance in our experiments. The clustering process is described as follows:

First step, every new story is considered as a cluster that contains only one story and pair-wise similarities between the stories in the new story set are calculated.

Secondly, calculate pair-wise similarities between clusters. The similarity between cluster A and B is calculated as the arithmetic average of pair-wise similarities between the stories in A and the stories in B. As pointed out in [9], the similarity calculated equals to the inner product between topic vectors of cluster A and B, so the topic similarity calculation is the same as the story similarity calculation.

Thirdly, suppose cluster A and B are of the largest similarity θ among all cluster pairs. If θ is larger than designated threshold θ_d, combine cluster A and B into a new cluster, turn back to the second step and continue. Otherwise the clustering process stops and new topic candidates are generated.

3.4 Tracking Previous Topics

New topic candidates are made good use to compare with previous topics to check whether they are actually "new" or not. There are two reasons for not putting new and old stories together to operate the topic detection algorithm and collect all the topics. Firstly, we can easily get new topics during the topic tracking process. Secondly, the amount of previous stories can be very large in practical applications; it's unnecessary to cluster all the stories as a whole dataset.

Since the life cycles of topics are usually short, we select previous topics updated during the past N days. Topics that have not been changed for N days are frozen, like what [8] did in topic detection.

Similarities between every new topic candidate and previous one are calculated. For each new topic candidate t_n, assume the largest similarity θ is the similarity between t_n and the previous topic t_o. Compare θ with designated threshold θ_t ($\theta_t < \theta_d$). If θ is larger, we take as a truth that they belong to the same topic, combine t_n to t_o and update t_o.

Since some previous topics are updated, we scan all topics again and check if there're any updated topics that need to be combined with other topics. This process will be repeated until no topics are combined. New topics are generated and previous ones are updated finally.

4. NEWS ISSUE CONSTRUCTION

News websites are the origin of new topics in Web environment, so topics are generated using all pages from news websites as described in Section 3. We construct news issues with the topics generated, combining information from weblog sites and Web forums. Images, audios and videos extracted from news pages are also applied to make news issues more interesting and attractive to users. The combination of topics and construction of news issues are performed online automatically.

4.1 News Issue Construction Based On Topic Combination

Similar to the topic detection and tracking process described in Section 3, the news issue construction process is to deal with new and previous stories. Previous stories have been put in topics and these topics have been clustered into news issues, which are called previous news issues. New coming stories are clustered firstly, previous topics updated and new topics generated. Since some previous topics have been updated with new stories, previous news issues containing these old topics will also be updated. New topics generated are then clustered into new news issue candidates using the UPGMA algorithm described in Section 3.3. Similarly, a designated threshold θ_{ni} ($\theta_{ni} < \theta_t < \theta_d$) is used to control the clustering process. Finally, new news issue candidates are compared with previous news issues. A decision about whether a candidate is truly new or not is made, using a similar method described in Section 3.4. Previous news issues are updated and new news issues generated.

4.2 Adding Elements to News Issues

News issues contain not only news topics related to a designated theme but also weblogs and comments etc. We are able to collect all these additional data from weblogs and Web forums.

Data from weblogs and Web forums have much more noise than Web news. So we choose to use weblogs of a famous people list and forum articles from some famous forums, with click and comment amounts beyond designated thresholds θ_{click} and $\theta_{comment}$. The publishing dates of weblog and forum data are limited within N_b days after the latest updates of the topics. These weblogs and forum articles are processed, represented as term vectors and compared with all existing topics one by one. The combining similarity threshold is set to be θ_b. We will keep term

vectors of topics and news issues as the same after combination with weblogs or forum articles, for the latter is not as reliable as the former.

Images, audios and videos extracted from news Web pages are selected to add to corresponding news issues. The selection priorities are influenced by the following factors: data source (data from sites of an important site list are preferred); size of data (data of larger size are preferred); publishing date (the latest data are used); data amount (each topic is limited to own one image, one audio and one video at the most).

5. EXPERIMENTAL SETUP

5.1 Dataset and Experimental Setup

Experiments are performed on Chinese datasets constructed from practical Web environment. Datasets of different sizes are used to examine different parts of news issue construction algorithm proposed.

- Dataset1: This dataset contains 350 news pages with 87 topics. These Web pages, published in March and April in 2007, are all about search engine companies. They are selected from Chinese news websites, including news reports and reviews. The maximum topic has 20 stories, while the minimum has only one.
- Dataset2: This dataset contains 953 news pages with 108 topics. These Web pages are news from the sports channel of SOHU (http://sports.sohu.com/), one of China's major websites, on April 22, 2007. The maximum topic has 151 stories, while the minimum has only one.
- Dataset3: This dataset contains 24,872 news pages from sports channels of dozens of Chinese news websites and 1,339 articles from weblogs and Web forums in a designated list, published Sept 26 to Oct 25, 2007 (the latest 30 days when this page is written). All the pages are crawled by spiders.

5.2 Evaluation Metric

Different experiments are made to evaluate the three steps of news issue construction algorithm separately. We treat each topic and news issue as a cluster and select traditional evaluation metrics widely used in Information Retrieval [10] and clustering [11]: Recall, Precision and F-measure. We don't use C_{Det} [20], which is commonly used in TDT, because the conditions of our problem and real TDT tasks are different.

Topics and news issues generated using our algorithms are called clusters, actual topics and news issues called classes, and Recall, Precision are calculated as [11]:

$$\mathrm{Re}call(i,j) = \frac{n_{ij}}{n_i} \quad (5)$$

$$\mathrm{Pr}ecision(i,j) = \frac{n_{ij}}{n_j} \quad (6)$$

where n_{ij} is the number of members of class i in cluster j, n_j is the size of cluster j and n_i is size of class i.

The F-measure of cluster j and class i is given by [11]:

$$F(i,j) = \frac{2*\mathrm{Re}call(i,j)*\mathrm{Pr}ecision(i,j)}{\mathrm{Re}call(i,j)+\mathrm{Pr}ecision(i,j)} \quad (7)$$

The F-measure of any class i is defined as the maximum among values calculated with any cluster, and Recall and Precision calculated in such a case are defined as corresponding values of the class. Mark them as $\underline{F_i}$, R_i and P_i.

An overall value for the Recall, Precision and F-measure is calculated by taking the weighted average of all values for the corresponding metric as follows [11]:

$$\mathrm{Re}call = \sum_i \frac{n_i}{n} R_i \quad (8)$$

$$\mathrm{Pr}ecision = \sum_i \frac{n_i}{n} P_i \quad (9)$$

$$F = \sum_i \frac{n_i}{n} F_i \quad (10)$$

Recall and Precision usually contradict each other in Information Retrieval and clustering. If we return the whole news story set as a topic, Recall is 100% and Precision is very low; if we return every story as a topic, Precision is 100% and Recall is very low. In a word, low Precision means the different topic stories are put together wrongly, and low Recall means the same topic stories are not clustered correctly.

6. EXPERIMENTS AND DISCUSSIONS

Experiments are firstly done to evaluate our topic detection algorithm. Parameters are tuned and the results are compared with cluto toolkits [9]. Then we test our topic tracking algorithm. News issue construction results are demonstrated finally.

6.1 Topic Detection Results

We did some experiments to test the topic detection algorithm we had selected firstly, based on the pre-processing process, incremental TF-IWF model and cosine similarity calculation mentioned in Section 3. After that, word frequency (WF), redundant sentence removal (RSR) and title weights are tested and compared in our experiments.

6.1.1 Selection of Clustering Method

Our topic detection algorithm is actually a topic clustering algorithm. We surveyed the cluto toolkits [9] before we select UPGMA as our clustering method. Different clustering methods including state-of-the-art algorithms are tested and compared on Dataset2, with the help of cluto. Since the topic detection process deals with new stories and new stories are coming out all the time, the process runs every few minutes to provide the latest results for the update of news issues instantaneously. There will not appear too many stories in few minutes, so the topic detection algorithm does not have the requirement of dealing with a large amount of stories. Dataset2 contains 953 news stories with 108 topics. The dataset is from practical Web environment and the size is large enough to test our topic detection algorithm.

We compare different clustering algorithms implemented in cluto firstly. Since the amount of target clusters has to be given before clustering with cluto, we fix the number to be 108, which is the actual amount of clusters in Dataset2. Traditional TF-IDF and our incremental TF-IWF weighting models are both tested, and pre-processing is performed as mentioned in Section 3.1. There are many parameters that can be tuned in cluto. We test the algorithms carefully and show the results that are best or produced by outperforming algorithms and criterion functions mentioned in [9]:

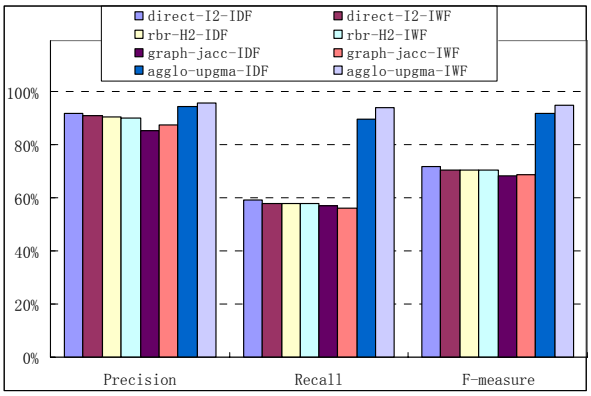

Figure 1. Performance comparison of different clustering methods based on IDF and IWF on Dataset2

The diagrammatic presentations have three parts: the clustering algorithm name, the criterion function name (or the similarity name) and the model name. The first two names are defined in [9] and the model name indicates whether TF-IDF or TF-IWF model is used.

As shown in Figure 1, the best performance is achieved by the "agglo-upgma" algorithm, which is a traditional agglomerative clustering method based on UPGMA. This result does not agree with the conclusion of [11], which finds out that "agglomerative hierarchal clustering performs poorly" and explains that "in many cases, the nearest neighbors of a document are of different classes". We consider that in news story datasets, the nearest neighbors of a story are always of the same class, which may be the reason why UPGMA performs best among all.

The amount of target clusters has to be given before using cluto, which is inconvenient in practical applications. So we add a threshold θ_d when implementing the UPGMA algorithm to decide when the clustering process stops, as mentioned in Section 3. The threshold can be explained as the least similarity between two stories that should be combined together. Many facilities are brought in and the clustering algorithm is controlled easily. Our algorithm achieves best results when θ_d is set as 0.225. Our best result is the same as cluto's best, as shown in Figure 1 ("agglo-upgma-IWF").

6.1.2 *IWF vs IDF*

IWF is used for term weighting instead of IDF in our weighting model. Experiments are made to compare their performance, based on Dataset1 and Dataset2. The result changes when it comes to different threshold θ_d. We draw Precision-Recall curves to show performances as follows:

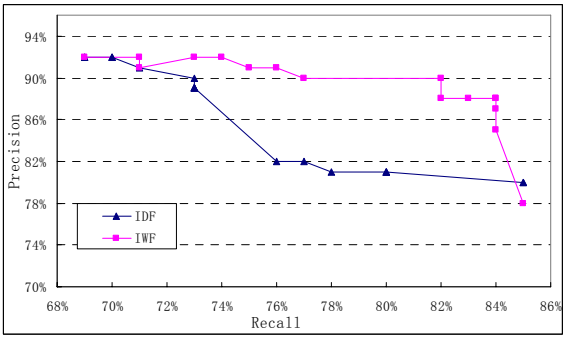

Figure 2. Performance comparison of IWF and IDF on Dataset1

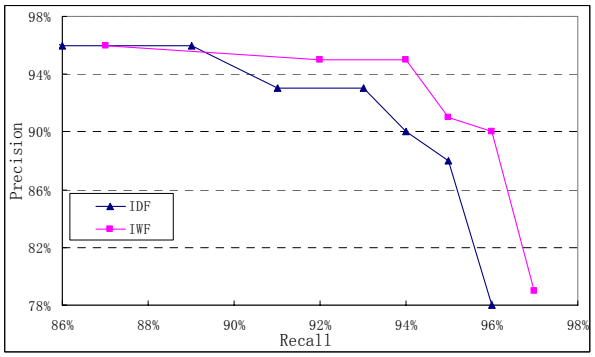

Figure 3. Performance comparison of IWF and IDF on Dataset2

Experiments on Dataset1 and Dataset2 both show that IWF is better than IDF. Through the study of information retrieval heuristics [12], both IWF and IDF satisfy constraints TFC1, TFC2, LNC1 and TF-LNC defined in [12], while IWF behaves more smoothly.

Take TFC1 for example. TFC1 is defined in [12] as: Let $q = \{w\}$ be a query with only one term w. Assume documents d_1 and d_2 are of the same length. If w appears more times in d_1 than in d_2, we judge that d_1 is more similar to q than d_2.

TF-IWF model shows some smoothing effects, compared with TF-IDF model. The TF part is the same in both models, and plays the dominant role in supporting TFC1 (larger TF of the query word w brings larger similarity). IDF remains the same no matter how many times w appears in the documents, while the value of IWF decreases when the frequency of w in document d increases, according to equation (2).

Similar analysis can be done on TFC2, LNC1 and TF-LNC and smoothing effects are found in TF-IWF model through these constraints, which could explain the steadier performance of IWF.

Compare the evaluation results on Dataset1 and Dataset2, and it lays out that better performance is achieved on Dataset2. Dataset2 is a relatively easier dataset for topic detection, for Dataset1 is constructed by hand and some of the stories are deliberately added to confuse our topic detection algorithm.

6.1.3 *Redundant Sentence Removal (RSR)*

News stories are mostly short, simple and clear. Repeating is a good way to show that something is important, and hence keywords to the main body of a news story will be repeated for a few times, not sentences. The most important sentence can be used as the news title and appears in the content once again, but is not usually simply repeated many times. We assume redundant sentences usually don't bring in new information but noises. For instance, one of the news stories in Dataset2 is actually a live broadcast of the draw for the 2007 FIFA Women World Cup in text from the sports channel of SOHU. The sentence "搜狐直播员：" ("SOHU live reporter", separated as a sentence by other contents with a colon), appears 73 times in the news story but in fact has nothing to do with the topic.

Dataset2 is selected to verify our guess. Redundant sentences in the content of every news story are removed. Precision-Recall curves are drawn to show results as Figure 4:

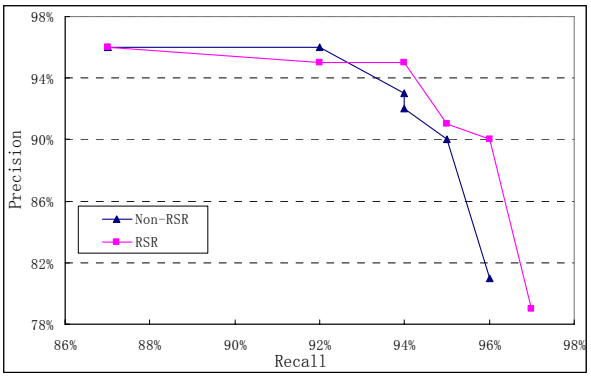

Figure 4. Effect of RSR on Dataset2

Figure 4 shows better performance in most circumstances after removing redundant sentences. It must be pointed out that when Recall increases, which means more stories on the same topics are put together, the RSR method leads to better precision compared with the non-RSR method. Since most news stories do not contain redundant and useless sentences, and some actually contain repeated and important sentences, the improvement is not great, but it's still worthy introducing RSR to our model.

6.1.4 Use of Title and Title Weight

It's natural to add the title of a news story to its representation, but we wonder whether the title is indeed helpful and how large weight should be assigned to it. We observe news pages in Dataset2 and find that:

(1) Some news stories are very short. There are only one or two (even no) sentences in the main body. The main part of this kind of news page is an image. This form of news page is usually to report news flashes. A brief title, an image and short content make up an attractive news report.

(2) Sometimes the title extracted is not related to the topic. The reason may be that the title extraction program fails, or the news page is created using a template and the title is not amended, or a mistake by the news editor.

So we do experiments on Dataset2 to find out the best way to add titles to the representation of news stories. Four conditions are tested: no titles are added (no-title); titles are added to all stories with weight 3 (all-title-3); titles are added to short stories with weight 3 (length shorter than 20, short-title-3); titles are added to short stories with weight 1 (short-title-1). Experimental results are shown as the following figure:

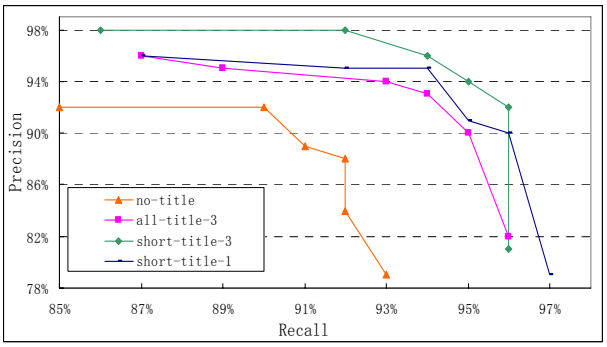

Figure 5. Effect of Title Use and Title Weight on Dataset2

From Figure 5 we can learn that:

(1) The addition of titles help improve the performance;

(2) Adding titles only to short stories achieves better performance than to all stories;

(3) Assigning a relatively larger weight to the title outperforms treating the title as important as the content.

So adding titles to the representation of short stories with weight 3 is our chosen method of using titles.

6.2 Topic Detection and Tracking Results

Dataset2 is used to test our topic detection and tracking algorithm. The news stories of Dataset2 are from the sports channel of SOHU (http://sports.sohu.com/) on April 22, 2007. We divide the dataset into 24 parts. Each part contains news stories that appear in an hour of the day. For instance, the first part contains 30 news stories, which appear from 00:00 to 01:00 of the day. A topic tracking process is simulated as follows:

News stories appearing from 0:01 to 01:00 are firstly clustered into topics as the topic detection process does. Then news stories appearing from 01:01 to 02:00 are clustered in the same way and the results are used to perform the topic tracking process with existing topics. The topic detection and tracking process is repeated 23 times until all the 23 parts (except the first part which is only processed by the topic detection algorithm) of the dataset have been processed.

We evaluate all topics we've got once one more part of the dataset is processed. The topic detection threshold θ_d is set as 0.225 and the topic tracking threshold θ_t is set as 0.125. The results at 24 hours of the day are shown as the following figure:

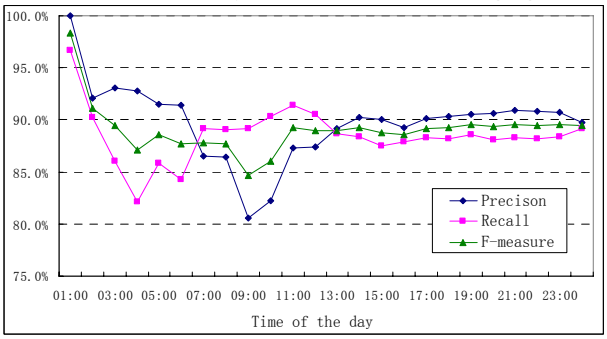

Figure 6. Performance of Topic Detection and Tracking at 24 hours of the day on Dataset2

From Figure 6 we find out that the performance of the topic detection and tracking process is good as a whole. The final result of the day which is the result at 24:00: P=89.7%, R=89.2%, F=89.4%. Compare this result with the best result achieved in the topic detection process which takes Dataset2 as a whole and perform the clustering method: P=95.6%, R=93.7%, F=94.6%. The loss of performance is only about 5%. The amount of previous stories can be very large in practical applications, and hence it is impossible to cluster all the stories as a whole dataset. So the topic tracking algorithm meets the practical need and keeps the clustering performance in general.

As we can see in Figure 6, the performance is not so good at 09:00 and 10:00. We will explain this through the statistics of new stories and topics appearing at each hour of the day, as it is shown in Figure 7:

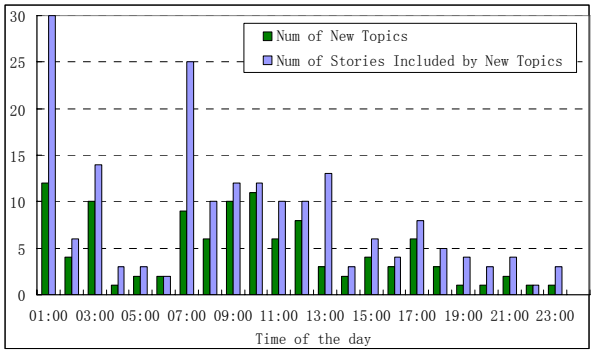

Figure 7. Number of New Topics and Number of Stories Included by New Topics at 24 hours of the day on Dataset2

A new topic means that the first story of the topic appears. The number of stories that make up new topics appearing in each hour of the day is compared with the number of new topics, as demonstrated in Figure 7. We can see that many new topics appear at 09:00 and 10:00 but the number of news stories that support the new topics at 09:00 and 10:00 is not so large, which means almost every story appearing at 09:00 and 10:00 is expected to be returned as a new topic itself. This may be difficult and our algorithms fail with a relatively bad performance.

6.3 News Issue Construction Results

News issue construction experiments are done as Section 4 describes. Firstly, we perform the topic detection and tracking process with Web pages from news websites and then cluster the topics into news issues. Secondly, data from weblogs, Web forums are selected and added. Images, audios and videos are finally added to corresponding news issues, following some rules.

Dataset3 is used in practical applications. All the pages are crawled from sports channels of news websites by spiders. We perform an experiment on Dataset3 to simulate automatic online news issue construction. Newly crawled Web pages are used to do topic detection. The result is used to update the previous topics and news issues, or create new topics and news issues. Images, audios and videos from crawled news pages, data from weblogs and Web forums in a designated list are selected and added. The process is repeated every 5 minutes and always provides the latest result. Since the dataset spans 30 days, the experiment costs a month to complete. Finally the news issues constructed are sorted, ranked and shown. The thresholds are set as: θ_d =0.225, θ_t = 0.125, N=10, θ_{ni} =0.1, θ_{click} =5000, $\theta_{comment}$ =50, N_b=3, θ_b =0.1.

Four results, generated in different time granularities, are provided to users in the final result page:

(1) Instant result. News issues updated in the latest few hours.
(2) Daily result. News issues updated in the latest 24 hours.
(3) Weekly result. News issues updated in the latest 7 days.
(4) Monthly result. News issues updated in the latest 30 days.

The instant result is ranked by the time when news issues are updated and meets the needs like “what’s new”. The other three are ranked by news issues’ sizes (number of news stories) and will satisfy people who wonder “what’s hot” and “what’s going on”. Figure 8 is part of the screenshot of the instant result page in the afternoon on Oct 26, 2007. It shows the top three news issues, talking about “the sixth City Games in China”, “matches of Phoenix Suns, NBA” and “matches of Milwaukee Bucks, NBA”, respectively:

Figure 8. Top 3 Instant Results of News Issue Construction[1]

Figure 9 shows the top tress news issues of the monthly result, talking about “Domanski as the head coach of the Chinese National Women's football team”, “sports lotteries of China” and “Yi Jianlian in Milwaukee Bucks, NBA”:

Figure 9. Top 3 Monthly Results of News Issue Construction[1]

We use titles of the latest news as the anchor texts of the hyperlinks linking to news issues and topics currently. Snippets of the latest news are shown as the summaries of news issues. In this way, the up-to-date status of a news issue will be viewed. Users will read more about the background, history, different opinions and discussions etc through a hyperlink like “*204* related topics”. It is worth noting that images of related news are added automatically and shown in Figure 8 and 9, and an article from the weblog of Sun Wen, a famous female football star, is added to the “Domanski” issue.

[1] The results of Figure 8 and 9 have been translated into English at APPENDIX.

We have got 8,259 topics and 742 new issues totally. It's difficult and time consuming to give an overall evaluation. We have just had assessors view the results and their feedbacks are summarized as follows:

(1) Results of the topic detection and tracking process do not seem as good as what we get in Section 6.1 and 6.2, for the situation here is much more complicated than dealing with sports news from one website one day.

(2) News issue construction results look better. News stories in the same news issue are mostly about the same theme, which indicates Precision will be good; some of the news issues should be combined as one, which indicates Recall will be worse than Precision.

(3) Images are usually correctly added. Audios and videos are very few. Data from weblogs and Web forums cause much more errors than news stories.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we propose a novel automatic online algorithm for news issue construction, through which news issues can be effectively and automatically constructed with real-time update, and lots of human efforts will be released from tedious manual work.

Our algorithm includes three steps. The first step is a topic detection process, in which newly appearing stories are clustered into new topic candidates. The second step is a topic tracking process, which compares those candidates with previous topics, and then merges the candidates into old topics or creates new ones. In the final step, news issues are constructed by the combination of related topics and updated by the insertion of new topics. Experimental results under practical Web circumstances indicate that Web contents are well arranged and instantaneously updated by our algorithm.

In the future, we hope to adjust the news issue rankings with page view information of news pages. We also plan to do user opinion mining on the news issue construction results.

8. REFERENCES

[1] http://www.nist.gov/speech/tests/tdt/index.htm

[2] Q. He, K. Chang, and E.-P. Lim. A model for Anticipatory Event Detection. In ER, pages 168–181, 2006.

[3] Y. Yang, T. Pierce, and J. Carbonell. A Study of Retrospective and On-line Event Detection. In Proceedings of the 21st Annual International ACM SIGIR Conference, Melbourne, Australia. ACM Press. 1998, 28-36.

[4] N. Stokes and J. Carthy. Combining Semantic and Syntactic Document Classifiers to Improve First Story Detection. In Proceedings of the 24th Annual International ACM SIGIR Conference, New Orleans. ACM Press. 2001, 424-425.

[5] B. Thorsten, C. Francine, and F. Ayman. A System for New Event Detection. In Proceedings of the 26th Annual International ACM SIGIR Conference, New York, NY, USA. ACM Press. 2003, 330-337.

[6] G. Kumaran and J. Allan. Text Classification and Named Entities for New Event Detection. In Proceedings of the 27th Annual International ACM SIGIR Conference, Sheffield, UK, ACM Press. 2004, 297-304.

[7] K. Zhang, J. Li, and G. Wu. New Event Detection Based on Indexing-tree and Named Entity. In Proceedings of the 30th Annual International ACM SIGIR Conference, Amsterdam, the Netherlands. ACM Press. 2007, 215-222.

[8] M. Spitters, W. Kraaij. TNO at TDT2001: Language Model-Based Topic Detection. Topic Detection and Tracking Workshop Report, 2001.

[9] Y. Zhao and G. Karypis. Criterion Functions for Document Clustering. Technical Report, 2005.

[10] C. J. van Rijsbergen, Information Retrieval, Buttersworth, London, second edition, 1989.

[11] M. Steinbach, G. Karypis and V Kumar. A Comparison of Document Clustering Techniques. KDD Workshop on Text Mining, 2000.

[12] H. Fang, T. Tao, C. Zhai. A Formal Study of Information Retrieval Heuristics. In Proceedings of the 27th Annual International ACM SIGIR Conference, Sheffield, UK, ACM Press. 2004, 49-56.

[13] J. Allan, R. Papka, and V. Lavrenko. On-line new event detection and tracking. In Proceedings of the 21st Annual International ACM SIGIR Conference, Melbourne, Australia. ACM Press. 1998, 37-45.

[14] Q. He, K. Chang, and E.-P. Lim. Analyzing Feature Trajectories for Event Detection. In Proceedings of the 30th Annual International ACM SIGIR Conference, Amsterdam, the Netherlands. ACM Press. 2007, 207-214.

[15] Overview of the TDT 2004 Evaluation and Results, http://www.nist.gov/speech/tests/tdt/tdt2004/papers/NIST-TDT2004.ppt

[16] D. Trieschnigg and W. Kraaij. TNO Hierarchical topic detection report at TDT 2004. Topic Detection and Tracking Workshop Report, 2004.

[17] M.-Q. Yu, W.-H. Luo, Z.-T. Zhou and S. Bai. ICT's Approaches to HTD and Tracking at TDT2004. Topic Detection and Tracking Workshop Report, 2004.

[18] M. Connell, A. Feng, G. Kumaran, and et al. UMass at TDT 2004. Topic Detection and Tracking Workshop Report, 2004.

[19] G.P.C. Fung, J.X. Yu, H. Liu and P.S. Yu. Time-Dependent Event Hierarchy Construction. In Proceedings of KDD-2007, pages 300-309, California, USA, 2007.

[20] The 2001 TDT task definition and evaluation plan, http://www.nist.gov/speech/tests/tdt/tdt2001/evalplan.htm.

APPENDIX

Translation for Figure 8

Instant Rank Daily Rank Weekly Rank Monthly Rank

News with image: Jiao Liuyang, champion of women 200 meter butterfly stroke of the 6th City Games

13:49:00 Oct 26, 2007 from SINA

Jiao Liuyang, an athlete from Harbin, left the competition terrain after the match on Oct 26. She won a golden medal with a 2'08''18 result in the final of women 200 meter butterfly stroke of the 6th City Games held in Wuhan. Photo by Sha dati, a reporter from Xinhua news agency.

News with image: Xie Zhi in the final of men 200 meter butterfly stroke of the 6th City Games –SOHU- (3 related news)

Results of the final of men 200 meter butterfly stroke of the 6th City Games –SOHU- (1 relate news)

News with image: Lai Zhongjian in the final of men 200 meter butterfly stroke of the 6th City Games –SOHU- (1 related news)

4 related topics>>

Amare Stoudemire is back and Grant Hill is leading, Suns defeat Nuggets: 116-113

13:06:00 Oct 26, 2007 from TOM

News from TOM sports: Suns play at home with Nuggets, and win 116-113 on Oct 26, Beijing time. Amare Stoudemire is back from an operation and plays as main force. Suns use the complete main force, and Grant Hill substitutes Boris Diaw as the main force.

Amare Stoudemire is back and Suns defeat Nuggets: 116-113 –ENORTH- (1 related news)

Live records of the first period, Nuggets vs Suns: 32-22 –ENORTH- (1 relate news)

"AI combination" unable to win for Nuggets, Amare Stoudemire helps Suns defeat Nuggets by 3 points –SPORTS.CN- (1 related news)

5 related topics>>

News with image: [Preseason of NBA] Bucks defeated by Bulls, Dan Gadzuric is defending

12:33:00 Oct 26, 2007 from SOHU

News from SOHU sports: The preseason of NBA is continuing at 8:30 on Oct 26, Beijing time. Milwaukee Bucks was defeated away by Chicago Bulls and have now lost two straight games. Jianlian Yi played for 21 minutes, made 2 of 7 shots and 3 of 4 penalty shots, won 7 marks and 3 rebounds, and made 2 turnovers and 3 personal fouls.

Images: 81-97, Bucks defeated away by Bulls –SPORTS.PEOPLE.COM.CN- (4 relate news)

[News with image] Jianlian Yi in the match between Bucks and Bulls –TOM- (4 related news)

6 related topics>>

Translation for Figure 9

Instant Rank Daily Rank Weekly Rank Monthly Rank

Dust settled: Domanski will leave her post, and times of the first foreign head coach of women football team has ended

07:45:00 Oct 24, 2007 from SOHU

News from SOHU (reported by Wang Zhanrong and Dai bin): Dust is settled about Domanski's contract extension. The Chinese Football Association (CFA) held a news conference and proclaimed that Domanski, the head coach of China women football team, would leave her post, without a renewal of a contract with CFA.

The CFA feels sorry for Domanski's leave –QQ- (10 related news)

Yang Yimin admitted that the CFA contacted Domanski forwardly –SOHU- (10 relate news)

The Asian Football Confederation pays attention to Domanski's leave and the former head coach of France team is hopeful to be the successor –ENORTH- (10 related news)

34 related topics>>

Weblogs

I don't know what to say about Domanski's leave, and I feel numb about frequent changes of head coach. –Weblog of Sun Wen-

[Song Yu's talks about lotteries] Prediction and analysis on the 2007284th welfare lotteries

14:50:00 Oct 19, 2007 from SOHU

News from SOHU sports: Firstly, analyze the results of the previous stage. Recall the results recently. (Some of the contents are omitted for the difficulty of translation)

Analysis of the 7289th stage from HAOCW.COM –SOHU- (20 related news)

Analysis of the 7291st stage from PAOKOO.COM –SOHU- (11 relate news)

Aggregate analysis of the 291st stage by Bai Jin –SOHU- (11 related news)

29 related topics>>

News with image: Yi Jianlian acts preeminently in Milwaukee Bucks

01:29:00 Oct 19, 2007 from SOHU

News from SOHU sports: Milwaukee Bucks continue their training in the San Francisco training center, on Oct 18, Milwaukee local time (the wee hours of Oct 19, Beijing time). One of the great moments on the training site is shown. (Photo by Zhang Yi, from Milwaukee) (Editor in charge: Wang Hailu)…

Exclusive images: Yi Jianlian gives rejections frequently during the training in Bucks –SOHU- (10 relate news)

Exclusive images: Milwaukee Bucks are preparing for the match with Minnesota Timberwolves –SOHU- (7 relate news)

Report at the end of the 3rd period: Yi Jianlian got 6 marks and 2 rebounds and Bucks fell behind Nuggets –SPORTS.PEOPLE.COM.CN - (6 related news)

26 related topics>>