Approach of Information Retrieval with Reference Corpus to Novelty Detection

Ming-Feng Tsai, Ming-Hung Hsu and Hsin-Hsi Chen

Department of Computer Science and Information Engineering
National Taiwan University
Taipei, Taiwan
E-mail: {mftsai, mhhsu}@nlg.csie.ntu.edu.tw; hh_chen@csie.ntu.edu.tw

1 Introduction

According to the results of TREC 2002, we realized the major challenge issue of recognizing relevant sentences is a lack of information used in similarity computation among sentences. In TREC 2003, NTU attempts to find relevant and novel information based on variants of employing information retrieval (IR) system. We call this methodology IR with reference corpus, which can also be considered an information expansion of sentences. A sentence is considered as a query of a reference corpus, and similarity between sentences is measured in terms of the weighting vectors of document lists ranked by IR systems. Basically, we looked for relevant sentences by comparing their results on a certain information retrieval system. Two sentences are regarded as similar if they are related to the similar document lists returned by IR system. In novelty parts, similar analysis is used to compare each relevant sentence with all those that preceded it to find out novelty. An effectively dynamic threshold setting which is based on what percentage of relevant sentences is within a relevant document is presented. In this paper, we paid attention to three points: first, how to use the results of IR system to compare the similarity between sentences; second, how to filter out the redundant sentences; third, how to determine appropriate relevance and novelty threshold.

2 Procedure

The flow of IR with reference corpus is illustrated in Figure 1, which contains an IR system and a reference corpus inside. To begin with, each sentence from the known relevant documents is treated as a query to a certain IR system that retrieves documents from the reference corpus. Then, a sentence can be transformed into a vector that uses each unique document retrieved by the IR system as one dimension and set the relevant weight assigned by the IR system as the weight of each dimension. An IR system, for instance, may retrieve top *m* documents from the reference corpus for a query. Therefore, a sentence can be regarded as a vector of *m* dimensions of weights assigned by the IR system. Finally, similarity metric is applied to measure the similarity between vectors, and the threshold is also applied to the following operations, retrieval or filter. Below we discuss this approach in detail.

2.1 IR System and Reference Corpus

In the experiments, the document sets used in TREC-6 text collection (Voorhees and Harman, 1997) were considered as a reference corpus. It consists of 556,077 documents. Okapi IR system (Robertson, Walker and Beaulieu, 1998) is adopted to experiment this approach. In the initial experiments, Okapi was in the option of *bm25*, and had average precision 0.2181 on TREC-6 text collection.

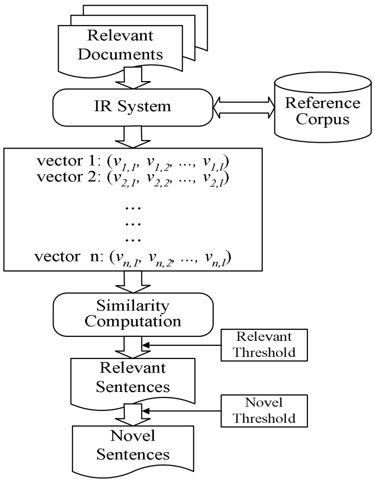
Relevant Documents
IR System
Reference Corpus
vector 1: ($v_{1,1}$, $v_{1,2}$, ..., $v_{1,l}$)
vector 2: ($v_{2,1}$, $v_{2,2}$, ..., $v_{2,l}$)
...
...
...
vector n: ($v_{n,1}$, $v_{n,2}$, ..., $v_{n,l}$)
Similarity Computation
Relevant Threshold
Relevant Sentences
Novel Threshold
Novel Sentences

Figure 1. Flow of IR with Reference Corpus Approach

2.2 Similarity Computation

The cosine similarity computation is considered in our task. The metric is shown as follows.

$$\cos(s_i, s_j) = \frac{\sum_{k=1}^{l} v_{i,k} \times v_{j,k}}{\| s_i \| \cdot \| s_j \|} \quad \textbf{(1)}$$

where s_i is represented as a sentence-vector ($v_{i,1}$, $v_{i,2}$, ..., $v_{i,l}$), l denotes the number of documents retrieved from the reference corpus by IR system; and s_j is another sentence-vector.

2.3 Threshold Setting

We consider what percentage of sentences is relevant within a document. In TREC 2002, Larkey *et al.* showed that about 5% of the sentences contained relevant materials for average topic. We also discovered the percentage of relevant sentences gets less when total number of given sentences is more. Therefore, we used logarithmic regression as follows to simulate the relationship between total number of the given sentences and percentage of the relevant sentences.

A dynamic threshold setting model is proposed as follows. Assume normal distribution with mean μ and standard deviation σ is adopted to specify the similarity distribution of the given sentences with a topic. We compute the cosine of a topic vector T and a given sentence vector S_i ($1 \le i \le m$), where m denotes total number of the given sentences. The percentage n denotes that top n percentages of the given sentences will be reported. Similarity thresholds (TH$_{\text{relevance}}$) are determined by these percentages.

$$\mu = \frac{\sum_{i=1}^{m} \cos(T, S_i)}{m} \quad \textbf{(2)}$$

$$\sigma = \sqrt{\frac{\sum_{i=1}^{m} (\cos(T, S_i) - \mu)^2}{m}} \quad \textbf{(3)}$$

$$\text{TH}_{\text{relevance}} = \mu + \text{z}\sigma \quad \textbf{(4)}$$

$$\phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} e^{-y^2/2} dy = 1 - n \quad \textbf{(5)}$$

We first compute the percentage n, and then derive z by Formula (5). Finally, TH$_{\text{relevance}}$ is computed by Formula (4). Therefore, the relevance threshold is determined by the total number of given sentences.

In the novelty part, a threshold of novelty decision, $TH_{novelty}$, determines the degree of redundancy. If the similarity score of two sentences is larger than $TH_{novelty}$, then one of them has to be filtered out depending to their temporal order. In this way, the redundant sentences are filtered out and only the novel sentences are kept. The remaining sentences are the result of the novelty detector. Two algorithms are proposed as follows. Assume there are r relevant sentences, $s_1, s_2, \ldots, s_r$ for topic t.

(1) Static threshold approach

Let T be a set containing novel sentences found up to know. Initially, $T=\{s_1\}$. For each relevant sentence s_i ($2 \le i \le r$), if there exists a sentence in T whose similarity with s_i is larger than a predefined threshold, then s_i is not a novel sentence and is removed; otherwise, s_i is kept in T.

(2) Dynamic threshold approach

Assume s_1 is a novel sentence. Compute the similarities between s_1 and s_i ($2 \le i \le r$). Determine the novelty threshold, $TH_{novelty}$, in the same way as $TH_{relevance}$. Filter out the top n% of sentences with the higher similarities with s_1. Let R be the remaining sentences. If the number of sentences in R is less than 30[1], then regard these sentences as novel sentences and stop. Otherwise, select the first sentence in R, regard it as a novel sentence and repeat the same filtering task.

3 Experiments

3.1 Finding Relevant Sentences

This part is to give the set of 25 relevant documents for each topic and to identify all relevant sentences. We first treated each given sentence as a query to IR system, and then get a vector of document weight assigned by IR system. Next, we applied the cosine function to measure similarity between sentences. In the part of threshold setting, we used the statistics of TREC 2002 novelty track to simulate the relation of total number of given sentences and percentage of relevant sentences. Formula 6 and Figure 2 show the trend. Because some topics may get less percentage, we apply a parameter to multiply the percentage calculated by Formula (6) to retrieve more sentences. Take Ln-4 for example. That means that it multiplies 4 to the calculated percentage.

$$n = -2.4938 Ln(x) + 23.157 \qquad \textbf{(6)}$$

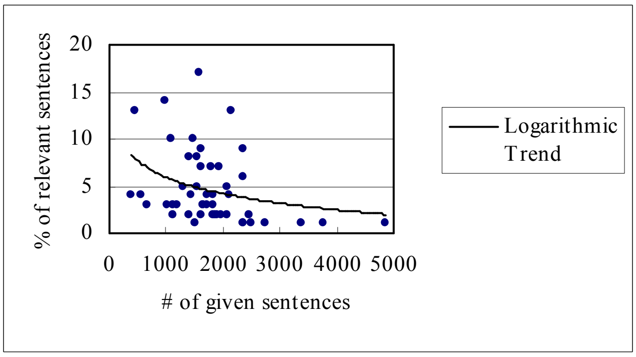

Figure 2. An illustration of Logarithmic Trend

Figure 3 shows the experimental results of relevance detection. These results are totally different to those of last year, because the number of qrels of relevance information is dramatically more than that of last year. Last year, the percentage of relevant sentences within the whole given sentences was about 5%, but this year some topics even has about 50 percent of relevant sentences. Therefore, our average recall gets lower since our relevance threshold is too high. That demonstrates the issue of identifying an appropriate threshold in the novelty detection is very important.

[1] A sample size of at least 30 has been found to be adequate for normal distribution

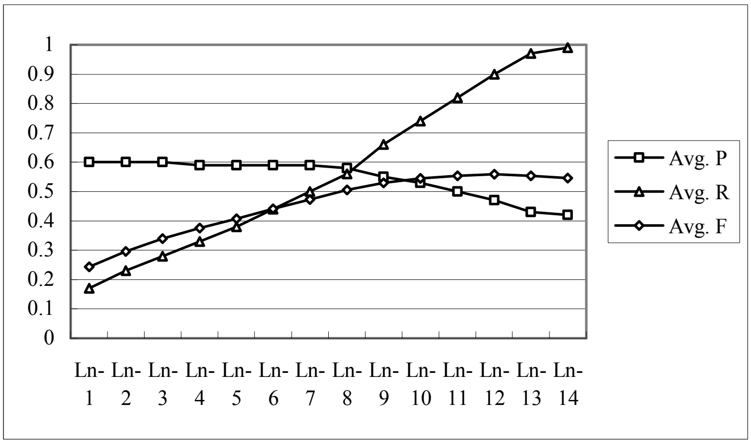

Figure 3. Experimental Results of Relevance Detection

3.2 Finding Novel Sentences

This part is to identify sentences that include new information among the relevant sentences. In other words, this part will filter out the redundant sentences. The key issue of finding novel sentences is how to differentiate the meaning of sentences accurately. We extend the idea, i.e., employing IR with reference corpus approach to expand a sentence, to find novel sentences. We experiment two novelty threshold setting algorithms, i.e., static and dynamic settings. In order to test this model, we use the perfect relevance results to experiment. And the number of consulted documents retrieved by IR system is set to 300.

Figure 4 demonstrates the results of finding novelty with static threshold setting. When novelty threshold is 1, it does not filter out any sentences. The performance gets better as the novelty threshold is higher. Figure 5 shows the results of finding novelty with dynamic threshold setting. The result reveals that the more percentage filtered, the worse the performance is. From these results, the performance will be better if we filter out fewer sentences. Therefore, we set the novelty threshold higher in the submitted runs to achieve better performance.

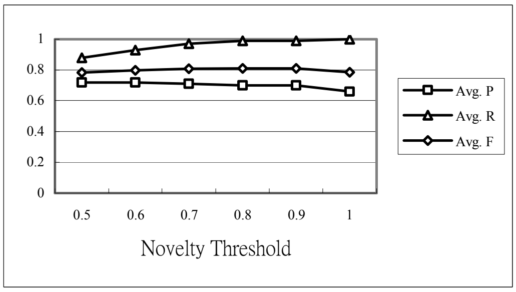

Figure 4. Experimental Results of Novelty Detection with Static Threshold Setting

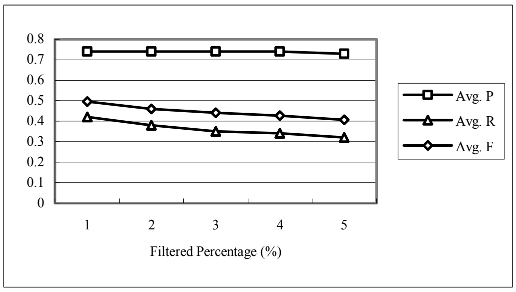

Figure 5. Experimental Results of Novelty Detection with Dynamic Threshold Setting

4 Runs Submitted

4.1 Task1 & Task3

Table 1 and 2 show the runs we submitted in task 1 and 3 of novelty detection, where the number of consulted documents is set to 300 , the dynamic relevance threshold uses Ln-1, and NTU11, NTU12, NTU13 and NTU14 uses topic description and narrative. In the novelty part of task 1, all runs use the static threshold setting where NTU11, NTU13 and NTU15 are set to 0.8; NTU12 and NTU14 are set to 0.9. In the task3, we use Ln-2, and Ln-3 functions to retrieve more relevant sentences.

Table 1. Task 1 Submitted Results

	Relevant Detection			Novelty Detection		
	Avg P	Avg R	Avg F	Avg P	Avg R	Avg F
NTU11	0.59	0.16	0.225	0.43	0.15	0.197
NTU12	0.59	0.16	0.225	0.43	0.15	0.200
NTU13	0.58	0.16	0.223	0.43	0.15	0.195
NTU14	0.58	0.16	0.223	0.42	0.15	0.197
NTU15	0.57	0.14	0.209	0.42	0.14	0.180

Table 2. Task 3 Submitted Results

	Relevant Detection			Novelty Detection		
	Avg P	Avg R	Avg F	Avg P	Avg R	Avg F
NTU31	0.56	0.20	0.266	0.39	0.19	0.217
NTU32	0.57	0.25	0.301	0.39	0.23	0.241
NTU33	0.58	0.22	0.287	0.40	0.21	0.236
NTU34	0.58	0.27	0.330	0.40	0.26	0.270
NTU35	0.57	0.22	0.287	0.39	0.21	0.240

4.2 Task2 & Task4

Tables 3 and 4 show the results of task 2 and 4 of novelty track. We use two novelty algorithms to find novelty sentences. In task 2, NTU21, NTU22 and NTU23 use static threshold; NTU24 and NTU25 use globe threshold setting. In task 4, NTU41, NTU42 and NTU43 also use static threshold; NTU44 and NTU45 use dynamic threshold.

Table 3. Task 2 Submitted Results

	Novelty Detection		
	Avg P	Avg R	Avg F
NTU21	0.71	0.98	0.812
NTU22	0.70	0.99	0.811
NTU23	0.70	0.99	0.812
NTU24	0.74	0.42	0.495
NTU25	0.74	0.42	0.501

Table 4. Task 4 Submitted Results

	Novelty Detection		
	Avg P	Avg R	Avg F
NTU41	0.67	0.98	0.785
NTU42	0.67	0.99	0.784
NTU43	0.67	0.99	0.784
NTU44	0.68	0.46	0.507
NTU45	0.68	0.47	0.509

References

Allan, J., Wade, C., and Bolivar, A.: Retrieval and Novelty Detection at the Sentence Level. In Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Toronto, Canada, July 28–August 01, 2003. ACM (2003) 314-321

Allan, J., Carbonnell, J., and Yamron, J.: Topic Detection and Tracking: Event-Based Information Organization. Kluwer (2002)

Chen, H.H., and Ku, L.W.: An NLP & IR Approach to Topic Detection. In Topic Detection and Tracking: Event-Based Information Organization, James Allan, Jaime Carbonnell, Jonathan Yamron (Editors). Kluwer (2002) 243-264

Chen, H.H., Kuo, J.J., Huang, S.J., Lin, C.J., and Wung, H.-C.: A Summarization System for Chinese News from Multiple Sources. In Journal of American Society for Information Science and Technology. (2003)

Chen, H.H., Tsai, M.F. and Hsu, M.H.: Identification of Relevant Novel Sentences Using Reference Corpus. In Proceedings of the 26th European Conference on Information Retrieval. University of Sunderland, U.K., 5th-7th April 2004. ECIR (2004)

Harman, D.: Overview of the TREC 2002 Novelty Trec. In Proceedings of the Eleventh Text REtrieval Conference. NIST Special Publication: SP 500-251, Gaithersburg, Maryland, November 19-22, 2002. TREC (2002)

Larkey, L. S. et al.: UMass at TREC2002: Cross Language and Novelty Tracks. In Proceedings of the Eleventh Text REtrieval Conference. Gaithersburg, NIST Special Publication: SP 500-251, Gaithersburg, Maryland, November 19-22, 2002. TREC (2002)

Robertson, S.E., Walker, S., and Beaulieu, M.: Okapi at TREC-7: Automatic ad hoc, Filtering, VLC and Interactive. In Proceedings of the Seventh Text REtrieval Conference, Gaithersburg, NIST Special Publication: SP 500-242, Gaithersburg, Maryland, November 9-11, 1998. TREC 7 253-264.

Tsai, M.F., and Chen, H.H.: Some Similarity Computation Methods in Novelty Detection. In Proceedings of the Eleventh Text REtrieval Conference. Gaithersburg, NIST Special Publication: SP 500-251, Gaithersburg, Maryland, November 19-22, 2002. TREC (2002)

Voorhees, E.M., Harman, D.K. (Eds.) Proceedings of the Sixth Text Retrieval Conference. NIST Special Publication: SP 500-240, Gaithersburg, Maryland, November 19-21, (1997)

Zhang, M. et al.: THU at TREC2002: Novelty, Web and Filtering. In Proceedings of the Eleventh Text REtrieval Conference, NIST Special Publication: SP 500-251, Gaithersburg, Maryland, November 19-22, 2002. TREC (2002)