行政院國家科學委員會專題研究計畫　期中進度報告

多語句子相關性和新穎性偵測及其應用研究(1/2)

計畫類別：個別型計畫
計畫編號：NSC93-2213-E-002-078-
執行期間：93 年 08 月 01 日至 94 年 10 月 31 日
執行單位：國立臺灣大學資訊工程學系暨研究所

計畫主持人：陳信希

計畫參與人員：蔡銘峰　許名宏

報告類型：精簡報告
報告附件：出席國際會議研究心得報告及發表論文
處理方式：本計畫可公開查詢

中　華　民　國 94 年 12 月 16 日

行政院國家科學委員會補助專題研究計畫 □ 成 果 報 告 ■期中進度報告

（計畫名稱）

多語句子相關性和新穎性偵測及其應用研究(1/2)

計畫類別：■ 個別型計畫　□ 整合型計畫
計畫編號：NSC 93－2213－E－002－078－
執行期間：2004年08月01日至2005年07月31日

計畫主持人：陳信希
共同主持人：
計畫參與人員：蔡銘峰　許名宏

成果報告類型(依經費核定清單規定繳交)：□精簡報告　■完整報告

本成果報告包括以下應繳交之附件：
□赴國外出差或研習心得報告一份
□赴大陸地區出差或研習心得報告一份
□出席國際學術會議心得報告及發表之論文各一份
□國際合作研究計畫國外研究報告書一份

處理方式：除產學合作研究計畫、提升產業技術及人才培育研究計畫、列管計畫及下列情形者外，得立即公開查詢
□涉及專利或其他智慧財產權，□一年□二年後可公開查詢

執行單位：國立臺灣大學資訊工程學系暨研究所

中　華　民　國　九十四　年　五　月　十二　日

Relevant and Novel Detection Using Reference Corpus

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

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

Abstract. This article proposes an information retrieval (IR) with reference corpus approach to identify relevant and novel sentences among documents and applies this method to multilingual relevant sentence detection. The main difficulty in determining the relevance and novelty of sentences is the lack of information within sentence used by similarity computation. Therefore, an information retrieval with reference corpus method is proposed. In this approach, each sentence is regarded as a query to a reference corpus, and similarity between sentences is measured in the weighting vectors of document lists ranked by IR systems. Two sentences are considered similar if they have similar documents lists returned by the IR system. A dynamic threshold setting method is adopted in IR with reference corpus approach. The proposed procedures and conventional IR method are compared. The reference corpus based method with dynamic threshold outperforms direct retrieval fashion. The average F-measure of relevance and novelty detection using Okapi system is 0.212 and 0.207, 57.14% and 58.64% of human performance, respectively. Moreover, we employ this manner to a parallel corpus for multilingual relevance detection. Both sentence-aligned and document-aligned corpora, i.e., Sinorama Corpus and HKSAR corpus, are experimented. The experimental results demonstrate that MRR for similarity computation between multilingual sentences is 0.839 when test data uses larger finer grain parallel corpus of the same domain. Generally speaking, the sentence-vector approach is superior to the term-vector approach when sentence-aligned corpus is employed. The document-vector approach performs better than the term-vector approach if document-aligned corpus is used. In term-vector approach, Log-Chi-Square weighting scheme performs better than Okapi-FN1 weighting scheme. In out experiments, Chinese basis is more suitable than English basis in language issue. The test bed of TREC Novelty Track is translated to evaluate the overall performance. The experimental results show that multilingual relevance detection has 80% of the performance of monolingual relevance detection.

1 Introduction

Precisely retrieving relevant information from a considerable amount of data collection has become increasingly important in an era of information explosion. Current information retrieval (IR) systems only can return documents satisfying users' information needs, they could not precisely locate the relevant sentences. Therefore, users have to go through the whole documents to find the relevant information. Moreover, traditional IR systems do not identify which sentences contain new information. Filtering redundant information out and locating novel information is indispensable for some emerging applications like summarization and question-answering [4].

Some relevance and novelty detection methods on document level have been proposed in Topic Detection and Tracking (TDT) [2]. Link detection relates news stories on the same topic [3] and first story detection tries to identify the first article with a new event. Novelty track in TREC 2002 [5] is the first attempt to locate relevant and new sentences instead of the whole documents containing duplicate and extraneous information. Similarity computation is a fundamental operation for relevance and novelty judgment on both sentence and document levels. However, the amount of information of a sentence that can be used in similarity computation is much fewer than that of a document. Therefore, lack of information within one sentence is the major challenging issue in this topic.

In the previous work, we addressed the problem by word matching and thesaurus expansion if two sentences touched on the same theme in multi-document summarization [4]. Such an approach has been employed to detect the relevance between a topic description and a sentence [8]. The similarity computation can also be performed by an information retrieval system.

Zhang *et al*. [11] employ an Okapi system and a fixed heuristic threshold to retrieve relevant sentences with a topic description. Larkey *et al*. [6] study how many sentences are relevant in different size of documents. Allan *et al.* [1] focus on the novelty detection algorithms and show how the performance of relevant detection affects that of novelty detection. Instead of using an IR system to select relevant sentences directly, an external corpus can be consulted [8]. Both a topic description and a sentence are considered as queries to the reference corpus through an IR system. Two sentences are relevant if similar sets of relevant documents are retrieved.

In this paper, we not only propose the reference corpus based approach to address this problem, but also apply this approach to multilingual relevant sentences detection. This paper shows how to extract relevant sentences from several known relevant documents, and how to determine new sentences from the extracted relevant sentences. The decision about what information is new depends on the order of the occurrence of the information. Section 2 presents relevant detection which includes concept matching approach and IR with reference corpus approach. The effects of different issues, including with/without reference corpus, static/dynamic settings of thresholds, and various IR systems, are compared. Section 3 extends the IR with reference corpus approach to extract novel sentences from relevant sentences. Section 4 presents IR with reference corpus approach to multilingual relevant detection. Section 5 concludes the remarks.

2 Relevant Detection

2.1 Concept Matching Approach

The problem of novelty task is defined as follows:

> *Given a topic description and a sequence of sentences, a novelty detection system should identify which sentences are relevant to the topic description, and which sentences are novel relative to the other sentences under a specific topic*.

The original sequence of sentences is called *given sentences*, and the resulting two lists are called *relevant sentences* and *novel sentences*. The given sentences came from some relevant documents. A novelty task is composed of two major components, i.e., a relevance detector and a novelty detector. The relevance detector receives a sequence of sentences from known relevant documents, and determines which sentences are on topic. Those relevant sentences will be delivered to the novelty detector and the redundant sentences will be filtered out. The remaining sentences are *novel* and *relevant*. In this task, relevant detector is very important because the performance of novel detector heavily depends on its performance.

A relevant detector attempts to identify those sentences containing the relevant information from the known relevant documents. The key issue behind relevance detection is how to measure the similarity of a topic description and the given sentences. Because the basic unit of similarity measure is a sentence instead of the whole document, we should deal with the problem of the lack of information within a sentence during discrimination of relevant and irrelevant sentences. Concept matching approach is proposed for relevance detection. A predicate and its surrounding arguments form a kernel skeleton in a sentence, so that verbs and nouns are important features for similarity computation. In this way, all the given sentences are tagged by using a part-of-speech tagger. After tagging, nouns and verbs are extracted. Then WordNet is applied to find the synonymous terms for concept matching. Noun and verb taxonomies with hyponymy/hypernymy relations are consulted. The similarity of two sentences is in terms of noun-similarity and verb-similarity as follows.

$$noun\_sim(s_1, s_2) = \frac{m}{\sqrt{ab}} \quad (1)$$

$$verb\_sim(s_1, s_2) = \frac{n}{\sqrt{cd}} \quad (2)$$

$$sim(s_1, s_2) = noun\_sim(s_1, s_2) + verb\_sim(s_1, s_2) \quad (3)$$

where s_1 and s_2 denote two sentences, respectively; m and n denote the number of concept matching for nouns and verbs, respectively; a and b are the total number of nouns in s_1 and s_2, respectively; and c and d are the total number of verbs in s_1 and s_2, respectively.

Total 49 topics and 49 sets of given sentences in TREC 2002 Novelty track [5] are applied to evaluate the performance of relevance detector. Precision, recall and F-measure are calculated, and average F-measure is adopted to measure the overall performance.

When the threshold is set to 0.4, the average F-measure of the concept matching approach is 0.125. Besides, a baseline model that randomly selects sentences from the given sentences is also adopted for comparison. The average F-measure of the baseline model was 0.040, and the average F-measure of human judge was 0.371 [5]. The experiments show that the performance of the concept matcher is better than that of the baseline model, but is still far less than that of human being. The outside resource, i.e., WordNet, seems not to be enough to measure the similarity in these experiments. In the following we will consult another external resource – say, a reference corpus.

2.2 IR Approach

2.2.1 IR without Reference Corpus

Even using an IR system, we have two alternatives to select the relevant sentences, i.e., with and without a reference corpus. In the corpus-free approach, the given sentences form a database itself, and a topic is submitted to an IR system to retrieve the similar sentences directly. The resulting sentences ranked and reported by the IR system are called *candidate sentences*. A dynamic percentage of candidates with higher scores will be reported as relevant. The percentage and the relevant thresholds are determined in the similar way as the corpus-based approach. The best F-measures of IR approach without reference corpus are 0.113 and 0.165, respectively.

2.2.2 IR with Reference Corpus

To use a similarity function to measure if a sentence is on topic is similar to the function of an IR system. We use a reference corpus, and regard a topic and a sentence as queries to the reference corpus. An IR system retrieves documents from the reference corpus for these two queries. Each retrieved document is assigned a relevant weight by the IR system. In this way, a topic and a sentence can be in terms of two weighting vectors. Cosine function measures their similarity, and the sentence with similarity score larger than a threshold is selected. The issues behind the IR with reference corpus approach include the reference corpus, the performance of an IR system, the number of documents consulted, the similarity threshold, and the number of relevant sentences extracted.

The reference corpus should be large enough to cover different themes for references. In the experiments, the document sets used in TREC-6 text collection [10] were considered as a reference corpus. It consists of 556,077 documents. Two IR systems, i.e., Smart [9] and Okapi [7], were adopted to measure the effects of the performance of an IR system. In the initial experiments, Smart system with the basic setting (i.e., *tf*idf* scheme without relevance feedback) was employed. It had average precision 0.1459 on the TREC topics 301-350. Okapi was in the option of bm25, and had average precision 0.2181 on the same document set.

How Many Document Reported

How many documents should be reported by an IR system is an important issue for similarity measurement between a topic and a given sentence. Both relevant and irrelevant documents may be reported in the result list. That depends on the IR performance. The effects of the sizes of resulting document lists were investigated. Table 1 summarizes the results of using Smart and Okapi when the threshold is set to 0.1. The first column shows that different number

of documents, i.e., 50, 100, 150, 200, 250, 300, 350, 400, 450, and 500 documents, are returned by Smart and Okapi, respectively.

It shows that smaller result list (e.g., 50 documents) is better than larger result list when Smart system is adopted. This is because the relevant document set is comparatively much smaller than the irrelevant document set for a query, and the irrelevant documents in the two result lists tend to be different. Smaller result list decreases the possibility to incorporate different irrelevant documents, but also decreases the possibility to find out the same relevant documents. Enlarging the result list means the number of the same relevant documents may be increased, but different irrelevant documents are also added. In contrast, the performance of Okapi-based system is increased from reporting 50 documents till 250 documents. After that, the performance starts to decrease. This is because Okapi outperforms Smart. Larger result list (within 250 documents) covers more relevant documents. In the experiments, the best average F-measures, 0.170 and 0.176, were achieved when the sizes of result list were 50 and 250 documents by using Smart and Okapi, respectively.

Table 1. Effects of Size of Returned Documents

Number of consulted documents	Smart-based			Okapi-based		
	Avg. P	Avg. R	Avg. F	Avg. P	Avg. R	Avg. F
50	0.13	0.4	**0.170**	0.15	0.49	0.169
100	0.13	0.43	0.154	0.15	0.48	0.174
150	0.12	0.46	0.144	0.13	0.49	0.174
200	0.11	0.48	0.137	0.14	0.48	**0.176**
250	0.11	0.50	0.137	0.13	0.49	**0.176**
300	0.10	0.51	0.135	0.13	0.49	0.173
350	0.10	0.52	0.130	0.12	0.49	0.170
400	0.10	0.54	0.127	0.12	0.50	0.171
450	0.09	0.54	0.124	0.12	0.50	0.170
500	0.09	0.55	0.120	0.11	0.51	0.169

Threshold Setting

We also made experiments with different thresholds (between 0 and 0.3), and smaller number of returned documents. Figures 1 and 2 show the experimental results. The best F-measures of using Smart and Okapi are 0.175 and 0.182, respectively. Because we did not employ the distribution of similarity scores, the thresholds were "guessed", and the thresholds were fixed in different topics. That is unfair in some cases.

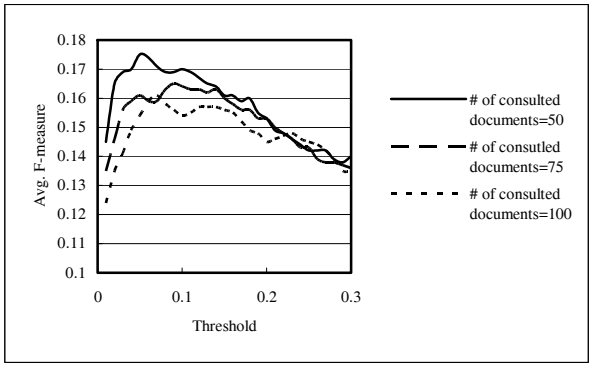

Figure 1. Effects of Fixed Thresholds Using Smart

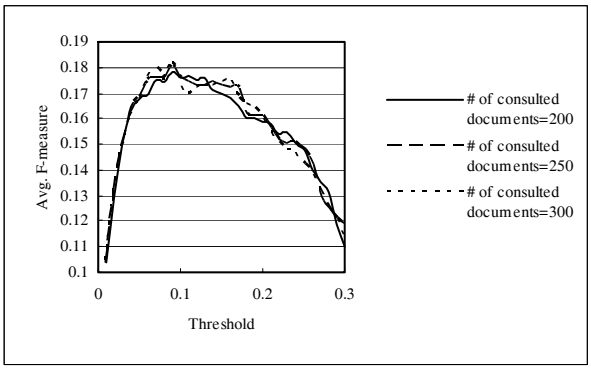

Figure 2. Effects of Fixed Thresholds Using Okapi

A threshold setting model is proposed as follows to deal with this problem. 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 ($1 \le i \le m$) as below, 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 ($TH_{relevance}$) are determined by these percentages.

$$\mu = \frac{\sum_{i=1}^{m} \cos(T, S_i)}{m} \quad (4)$$

$$\sigma = \sqrt{\frac{\sum_{i=1}^{m} (\cos(T, S_i) - \mu)^2}{m}} \quad (5)$$

$$TH_{relevance} = \mu + z\sigma \quad (6)$$

$$\phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} e^{-y^2/2} dy = 1 - n \quad (7)$$

Figure 3 shows that total n (%) of given sentences fall in the gray area are considered as relevant. z is equal to 1.282, 0.84, 0.524, 0.253 and 0 when n is 10%, 20%, 30%, 40%, and 50%, respectively.

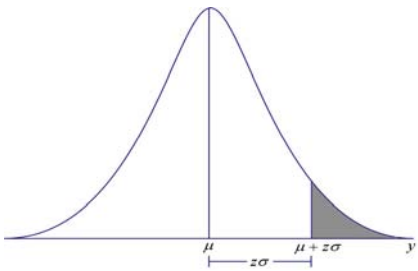

Figure 3. Normal Distribution with Mean μ and Standard Deviation σ

Various settings of n (percentage) were experimented, and the results using Smart and Okapi are listed in Figures 4 and 5, respectively. Smart-based relevance detector achieves better performance when larger percentage of sentences is selected. On the contrary, the larger the percentage is, the worse the performance is, when some critical point is reached using Okapi. The major reason is: Okapi gets better retrieval performance than Smart, so that it pulls the relevant sentences in the front of normal distribution. The best F-measures are 0.190 and 0.206. Using n (%) to determine the thresholds is a dynamic approach, which is better than static threshold approach.

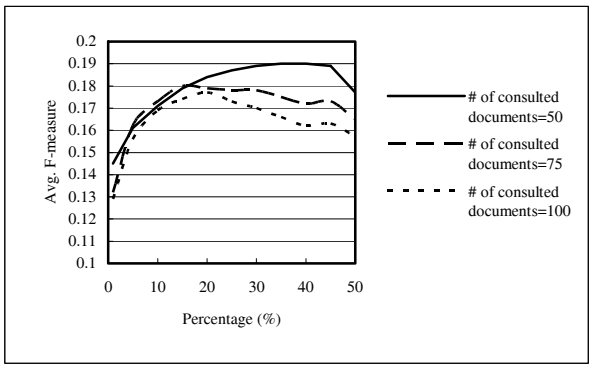

Figure 4. Effects of Fixed Percentages Using Smart

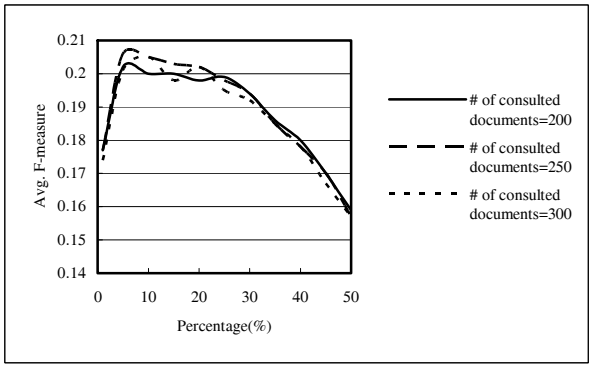

Figure 5. Effects of Fixed Percentages Using Okapi

Even though the above dynamic approach has better performance, it is still "fixed percentage" for all topics. We consider further how to select "good" percentages for individual topics. Larkey *et al*. [6] showed that only 5% of the sentences contained relevant materials for average topic. From their collection statistics [6], we used logarithmic regression as follows to simulate the relationship between total number of the given sentences and number of the relevant sentences. Figure 6 illustrates the trend.

$$n = -2.4938 Ln(x) + 23.157 \qquad (\mathbf{8})$$

where x is total number of given sentences, and n is the suggested percentage.

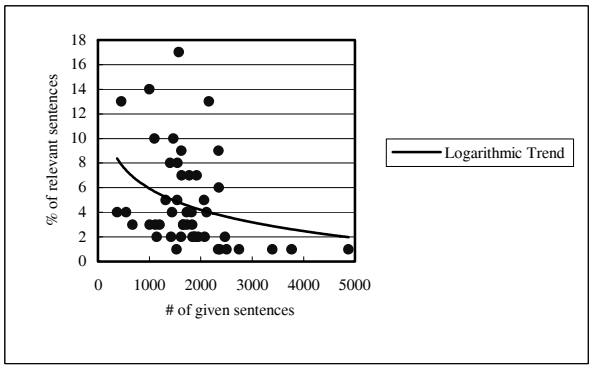

Figure 6. An illustration of Logarithmic Trend

After computing n using Formula (8), we derived z using Formula (7) and finally $TH_{relevance}$ using Formula (6).

Table 2 summarizes the F-measures of using dynamic percentage by Smart and Okapi, respectively. Dynamic percentage is better than fixed percentage. The best performance of dynamic percentage using Smart is 0.191 when the size of consulted documents is set to 50 and logarithmic metric is multiplied by 5, which gets about 1% improvement to the fixed percentage.

The best F-measure of dynamic percentage using Okapi is 0.212, when the size of consulted documents is set to 300 and the original logarithmic metric is employed. It gets about 3% increases to the fixed percentage experiments.

Table 2. Effects of Dynamic Percentage

Number of consulted Documents	Smart-based			Okapi-based		
	50	75	100	200	250	300
Ln-1	0.164	0.167	0.163	0.203	0.208	**0.212**
Ln-2	0.177	0.180	0.176	0.204	0.207	0.205
Ln-3	0.185	0.178	0.176	0.204	0.205	0.205
Ln-4	0.189	0.179	0.174	0.200	0.201	0.198
Ln-5	**0.191**	0.181	0.172	0.194	0.191	0.191

The best performance among these experiments is 0.212, i.e., 57.14% of human performance (0.371). Figure 7 lists the performance of each topic when the number of consulted documents is 300 using Okapi system. Two dotted lines, i.e., one is human performance (0.371) and the other one is baseline performance (0.040), are provided for reference. Performance of our system in 6 topics (358, 364, 365, 368, 397, and 449) is competitive to that of human judge. In contrast, performance in 3 topics (377, 420, and 432) is lower than that of random selection. The average F-measure of the remaining 40 topics are below human performance, but better than that of baseline model.

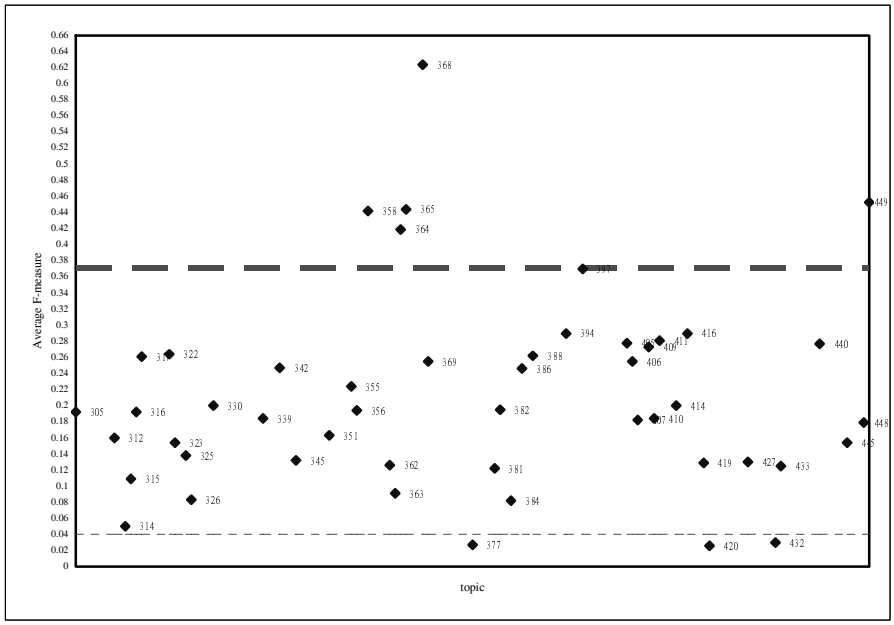

Figure 7. Average F-measure of Relevance Detection for Each Topic

3 Novelty Detection

3.1 IR with Reference Corpus Approach

Novelty detector identifies new information among the sentences extracted by the relevance detector. In other words, novelty detector will filter out the redundant sentences among the relevant sentences. The key issue on the detection of new information is how to differentiate the meaning of sentences accurately. Sentences may contain too less information to distinguish their differences, so that certain information expansion method is required.

We extend the idea in Section 2.2.2, i.e., employing a reference corpus to select the relevant information, to find the relationship among relevant sentences. Similarly, we use the same

reference corpus and regard each relevant sentence as a query to this corpus. Documents in the corpus are ranked by an IR system, and the documents with higher scores are reported for each relevant sentence. Each retrieved document is assigned a weight, in such a way that a sentence is still represented as a vector. Cosine function measures the similarity of any two sentences. Two sentences are regarded as similar if they are related to the similar document lists.

On the one hand, the cosine value of two sentences indicates that how similar they are. On the other hand, the higher value indicates one sentence is somewhat redundant relative to the other sentence. 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 to deal with the novelty detection problem. 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.

We chose the results from the best relevance detectors mentioned in last section, i.e., Smart-based and Okapi-based systems with average F-measure 0.191 and 0.212, to test these two approaches. The performance of static threshold approach is shown in Figure 8. Okapi-based novelty detector still outperforms Smart-based novelty detector. Besides, it also indicates that more sentences are filtered out when $TH_{novlety}$ is lower. The performance increased as $TH_{novelty}$ increased. Using higher novelty threshold, two sentences should have much higher similarity to pass the threshold if they are similar. The lower the probability two sentences pass the threshold, the higher the probability both sentences are novel. Figure 9 illustrates the results of dynamic threshold approach. When more percentages of sentences are filtered out, the performance of both Smart-based and Okapi-based novelty detectors are decreased.

The best performance among these experiments is 0.207, when the novelty threshold is set to 0.8 statically, and total 300 documents reported by Okapi are consulted. Figure 10 examines the performance of each topic furthermore. Two dotted lines, one for human performance (0.353) and the other one for baseline performance (0.036), are provided for reference. Performance of our approach in 6 topics (i.e., 358, 364, 365, 368, 397, and 449) is competitive to that of human judge. In contrast, performance in 3 topics (377, 420, and 432) is lower than that of the baseline model. The average F-measure of the remaining 40 topics are below human performance, but better than that of baseline model.

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

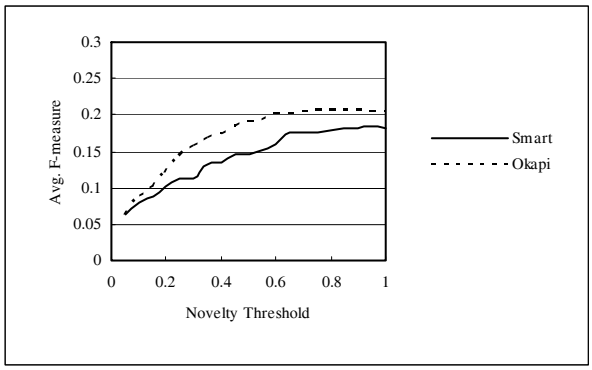

Figure 8. Results of Static Novelty Threshold

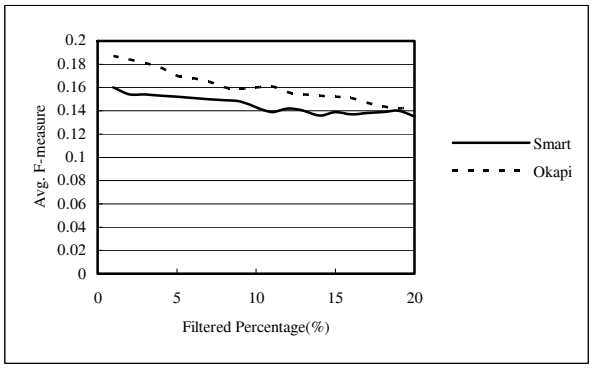

Figure 9. Results of Dynamic Novelty Threshold

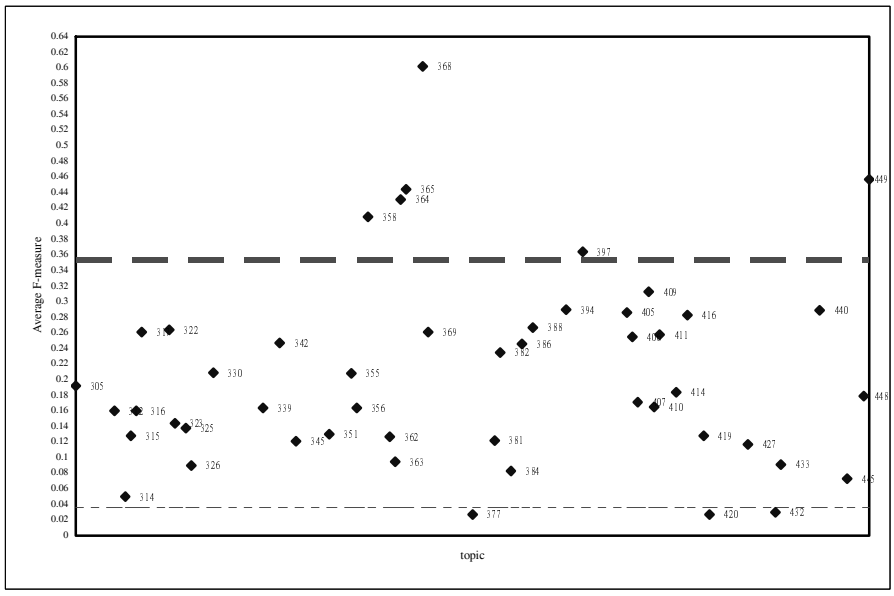

Figure 10. Further Examination of the Best Novelty Detection

In general, the average F-measure of the novelty detector is better than that of the baseline model (i.e., 0.036). However, the performance is still not comparable to the human assessors (i.e., 0.353). It only achieves 58.64% of human performance. The major reason is that the result of relevance detector contains irrelevant sentences, so that novelty detector false identifies that those irrelevant sentences contain new information. As mentioned before, the relevance part is more difficult to be overcome in this task.

We also conducted another set of experiments to evaluate the ideal performance of locating novel sentences. These experiments take correct relevant information as input to novelty detector, so that there are no propagation errors from relevant detectors. Figure 11 shows the results of static novelty threshold approach. The performance was increased when $TH_{novelty}$ was increased. This is because more sentences are filtered out when $TH_{novelty}$ is lower. The ideal

performance of Okapi-based novelty detector is 0.945 and the performance is above 0.912 when threshold is larger than 0.5. Figure 12 shows the results of dynamic novelty threshold approach. The ideal performance of the Okapi-based system is 0.922. The average F-measure dropped quickly, when more percentage of sentences are filtered out.

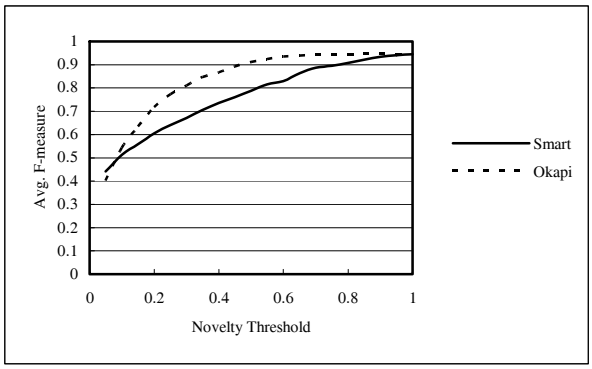

Figure 11. Ideal Performance of Static Novelty Threshold Approach

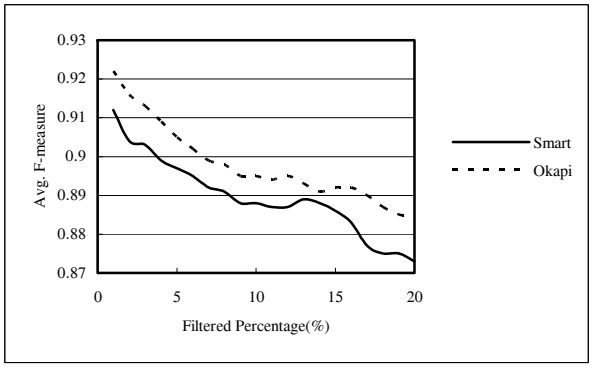

Figure 12. Ideal Performance of Dynamic Novelty Threshold Approach

4 Multilingual Relevant Sentence Detection Using Reference Corpus

The above approaches focus on monolingual relevance sentence detection only. As all know, large scale multilingual data have been disseminated very quickly via Internet. How to extend this work to multilingual information access is very important. Chen, Kuo and Su (2003) explored multilingual multidocument summarization. To measure the similarities between two bilingual sentences is their major concern. Therefore, this section extends the reference corpus approach (Chen, Tsai and Hsu, 2004) to identify relevant sentences in different languages. The computation of the similarities between an English sentence and a Chinese sentence, which is the kernel of multilingual relevant sentence detection, will be studied by referencing sentence-aligned and document-aligned parallel corpora.

4.1 Multilingual Relevant Detection Using Reference Corpus

In the reference corpus approach, the reference corpus should be large to cover different themes for references. In monolingual relevant sentence detection, we consider TREC-6 text collection as a reference corpus. Two IR systems, i.e., Smart and Okapi, were adopted to measure the effects of the performance of an IR system. The experimental results show that Okapi-based relevance detector outperforms Smart-based one. Therefore, Okapi system is adopted in the latter experiments.

We modify Okapi-Pack[2] from City University (London) to support Chinese information retrieval in the following way. A Chinese word-segmentation system is used for finding word

[2] http://www.soi.city.ac.uk/~andym/OKAPI-PACK/

boundaries. Unknown words may be segmented into a sequence of single Chinese characters. While indexing, Okapi will merge continuous single characters into a word and treat it as an index term. We build a single-character word list to avoid merging a single-character word into an unknown word. Chinese stop word list is not adopted.

We adopted NTCIR3 Chinese test collection (Chen, *et al*., 2003) to evaluate the performance of Chinese Okapi system (called *C-Okapi* hereafter). Table 3 summarizes the performance of C-Okapi comparing to the results of the participants in NTCIR3 (Chen, *et al*., 2003). The first column denotes different query construction methods, where T, C, D, and N denote topic, concept, description, and narrative, respectively. The 2nd-4th columns, i.e., AVG, MAX, and MIN, denote the average, the maximum, and the minimum performance, respectively. C-Okapi outperforms or competes with the maximum one in T and C methods, and is above the average in the other two query construction methods. In the later experiments, we will adopt Okapi and C-Okapi for bilingual relevance detection.

Table 3. Performance of C-Okapi

Topic Fields	AVG	MAX	MIN	C-Okapi
C	0.2605	0.2929	0.2403	0.2822
T	0.2467	0.2467	0.2467	0.2777
TC	0.3109	0.3780	0.2389	0.3138
TDNC	0.3161	0.4165	0.0862	0.3160

4.2 Similarity Computation between Multilingual Sentences

In monolingual relevant detection, we consult a monolingual corpus to determine the similarity between two sentences in the same language. When this approach is extended to deal with multilingual relevance detection, a parallel corpus is used instead. This corpus may be document-aligned or sentence-aligned. Figure 13 shows the overall procedure. English and Chinese sentences, which are regarded as queries to a parallel corpus, are sent to Okapi and C-Okapi, respectively. Total *R* English and Chinese documents/sentences[3] accompanying with the relevance weights are retrieved for English and Chinese queries. Because the corpus is aligned, the returned document (or sentence) IDs are comparable. Cosine function is used to compute the similarity, and thus the degree of relevance.

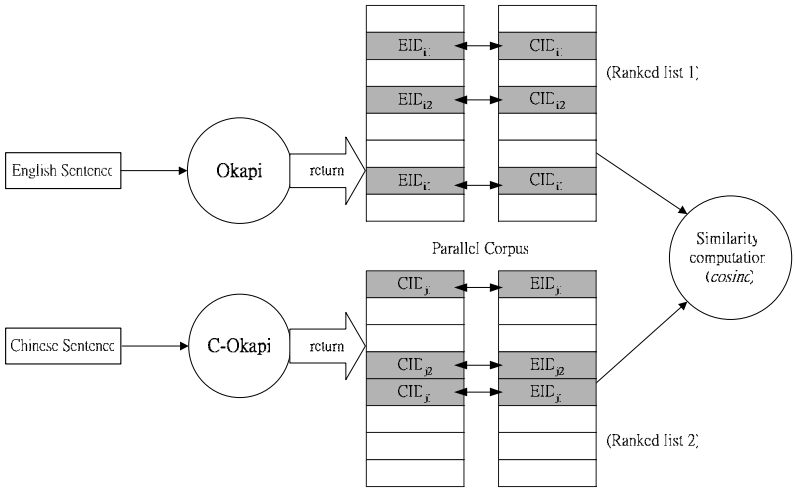

Figure. 13. Document-Vector/Sentence-Vector Approach

In the above diagram, two sentences are considered as relevant if they have similar behaviors on the results returned by IR systems. The results may be ranked list of documents or sentences depending on the aligning granularity of the parallel corpus. Besides the

[3] The word documents/sentences denotes either document-aligned or sentence-aligned corpus is used.

document-vector/sentence-vector approach shown in Figure 13, the two vectors used in similarity computation may be in terms of relevant terms. This idea follows the corpus-based approach to query translation (Davis and Dunning, 1995) in cross language information retrieval (CLIR). In CLIR, a query in language *A* is submitted to an *A*-*B* parallel corpus. An IR system for language *A* selects the relevant documents in *A*. The documents in language *B* are also reported at the same time. The target query is composed of terms selected from the relevant documents in *B*, and finally submitted to IR system for *B* language.

The above procedure is considered as *translation* in CLIR. Now, the idea is extended and plays the roles of both *translation* and *information expansion*. Figure 14 shows the overall flow. Similarly, an English sentence and a Chinese sentence, which will be determined relevancy, are sent to the two IR systems. *R* most relevant documents/sentences in two languages are returned. Instead of using the retrieval results directly, we select *K* most representative terms from the resulting documents/sentences. The two sets of *K* terms form two vectors, so that this approach is called *term-vector approach* later. Cosine function determines the degree of relevance between the English and the Chinese sentences.

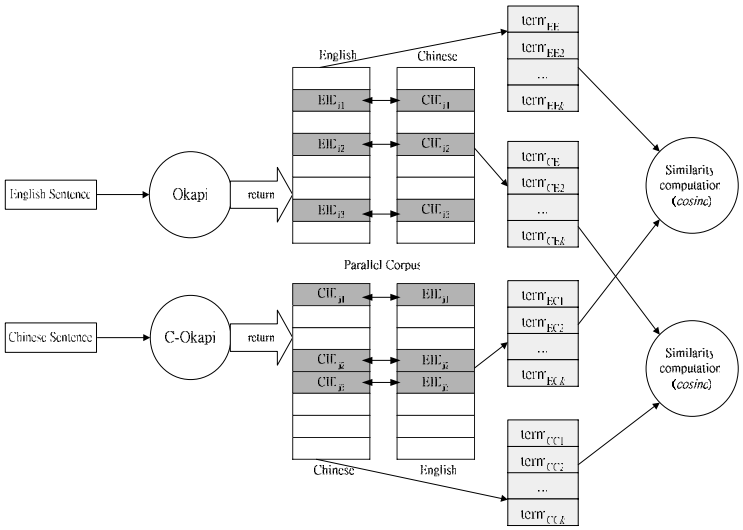

Figure. 14. Term-Vector Approach

Because the *R* most relevant documents/sentences are in two languages, we can consider either English or Chinese documents/sentences as a basis. In other words, if we select Chinese, then we map ranked list 1 (i.e., English results) into Chinese correspondent through the document-aligned/sentence-aligned chains. Similarly, ranked list 2 (i.e., Chinese results) may be mapped into English correspondent when English part is selected as a basis. Now we consider how to select the *K* most representative terms. Two alternatives shown below are adopted.

(1) Okapi-FN1

An intuitive weighting scheme is the weighting function of IR system. The weighting function of a term *t* in Okapi is as follows.

$$W(t) = \log \frac{(r+0.5)(N-R-n+r+0.5)}{(R-r+0.5)(n-r+0.5)} \qquad (\mathbf{9})$$

where N is total number of documents/sentences in the reference parallel corpus, R is the number of relevant documents/sentences to a query, n is the number of documents/sentences in which term t occurs, and r is the number of relevant documents/sentences in which term t occurs.

In our experiments, top *R* documents in the ranked list are parsed and the total occurrences *r* of a term *t* in the *R* documents are counted. Terms with the top *K* weights are employed for similarity computation.

(2) Log-Chi-Square

The *Chi-Square* test is used to find the terms highly relevant to the returned documents/sentences. At the same time, Chi-Square test is also considered as a basis for weighting. A 2×2 contingency table shown in Table 4 is conducted for Chi-Square test.

Table 4. A Contingency Table for Chi-Square test

	Relevant documents/sentences	Non-relevant documents/sentences
Term *t* occurs	A=*r*	B=*n* – *r*
Term t not occur	C=R – r	D=N – R – (n – r)

The meanings of *N*, *n*, *R*, and *r* are the same as those described in Okapi-FN1. The formula for Chi-Square test is shown as follows.

$$\chi^2 = \frac{N(AD - BC)^2}{(A+B)(A+C)(B+D)(C+D)} \qquad (\mathbf{10})$$

For the value of χ^2 could be very large (even larger than 10^6), we take logarithm of χ^2 as the weight of a term to avoid the cosine value between two vectors to be dominated by some few terms. This operation is similar to smoothing and drops the scale of weights.

Summing up, two alternatives, i.e., vectors in terms of resulting documents/sentences (Figure 13), and vectors in terms of representative terms (Figure 14), may be considered for similarity computation in multilingual relevance detection. In the latter case, either English part or Chinese part may be considered as a basis, and each has two possible weight schemes, i.e., Okapi-FN1 and Log-Chi-Square. Thus, four possible combinations are conducted in total for the latter experiments.

4.3 Experiment Material and Evaluation Method

Two Chinese-English aligned corpora are referenced in our experiments. One is Sinorama corpus[4], and the other one is HKSAR Corpus[5]. Sinorama consists of documents published by Sinorama magazine within 1976-2001. This magazine, which is famous for her superior Chinese-English contrast, recorded Taiwan society's various dimensions of evolvements and changes. HKSAR collects news articles released by the Information Services Department of Hong Kong Special Administrative Region (HKSAR) of the People's Republic of China. The following compares these two corpora from corpus scale, aligning granularity, average length, and so on.

Sinorama is a "sentence-aligned" parallel corpus, consisting of 50,249 pairs of Chinese and English sentences. We randomly select 500 Chinese-English pairs as test data to simulate multilingual relevance sentence detection. The remaining 49,749 pairs are considered as a parallel reference corpus. They are indexed separately as two monolingual databases, in which a Chinese sentence or an English sentence is regarded as a "small document". The average length of Chinese sentences in the reference corpus is 151 bytes and that of English sentences is 254 bytes. The average length of Chinese and English test sentences is 146 and 251 bytes, respectively.

HKSAR corpus contains 18,147 pairs of aligned Chinese-English documents released by HKSAR from July 1, 1997 to April 30th, 2000. Similarly, we index all articles in the same language as a monolingual database. The average document length is 1,570 bytes in Chinese

[4] http://rocling.iis.sinica.edu.tw/ROCLING/corpus98/mag.htm
[5] http://wave.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2000T46

and 2,193 bytes in English. The test data used in experiments are the same sentences pairs as Sinorama.

At first, we develop an evaluation method and a set of experiments to measure the kernel operation of relevance detection only, i.e., the similarity computation between Chinese and English sentences, in Section 4.4. Then, we measure the overall performance of multilingual relevance detection in Section 4.5. As mentioned, total 500 pairs of Chinese-English sentences are randomly selected from Sinorama corpus. They are denoted as: $<C_1, E_1>$, $<C_2, E_2>$, …, $<C_{500}, E_{500}>$, where C_i and E_i stand for Chinese and English sentences, respectively. Among the 500 Chinese sentences C_1, …, C_{500}, C_i is the most relevant to E_i. In other words, when we compute the similarities of all combinations consisting of one Chinese and one English sentences, C_i should be the most similar to E_i for $1 \leqq i \leqq 500$ ideally. Let $Sim(i, j)$ be the similarity function between C_i and E_j. A match function $RM(i, j)$ is defined as follows.

$$RM(i, j) = |\{k \mid Sim(i,k) > Sim(i, j),\ 1 \le k \le 500\}| + 1 \qquad (\mathbf{11})$$

The match function assigns a rank to each combination. The perfect case is $RM(i, i)=1$. We call it a *perfect match* later. We also relax the case. If $RM(i, i)$ is no larger than a threshold, we consider the result of matching is "good". In our experiments, the threshold is set to 10. That is, we postulate that the first 2% of matching pairs will cover the correct matching. Consulting the evaluation method in question answering track of TREC, we adopt **MRR** (mean reciprocal rank) score to measure the performance of our methods. Let $S(i)$ be the evaluation score for a topic i (Chinese sentence). MRR is summation of $S(i)$.

$$S(i) = \begin{cases} 1/RM(i,i) & \text{if } RM(i,i) \le 10 \\ 0 & \text{if } RM(i,i) > 10 \end{cases} \qquad (\mathbf{12})$$

$$MRR = \frac{1}{500}\sum_{i=1}^{500} S(i) \qquad (\mathbf{13})$$

4.4 Results and Discussion

4.4.1 Using Sinorama Corpus

Sentence-Vector Approach

Table 5 shows the experimental result of sentence-vector approach along with Sinorama corpus. Row "$RM(i, i)=1$" denotes how many topics get a "perfect match" and row "$RM(i, i) \leqq 5$" denotes how many topics get a correct match in the first 5 ranks. For example, 77.40% of test data are perfect match if 200 sentences are consulted by Okapi and C-Okapi, i.e., 200 sentences are returned for reference. In this case, the MRR is 0.839, which is the best in this experiment. When the number of returned sentences increases from 50 to 200, MRR score also increases. Then MRR score goes down until the number of returned sentences reaches about 600. After that, MRR score rises again and reaches to a stable state, i.e., 0.82-0.83.

Table 5. Performance of verus Number of Returned Sentences in Sinorama

Total returned sentences	50	100	200	400	600	1000	1500	2000	2500	3000
RM(i,i)=1	339	371	387	360	188	379	379	383	380	379
RM(i,i)≦5	402	431	460	464	442	463	461	459	457	455
RM(i,i)≦10	416	449	472	478	447	478	480	478	477	476
RM(i,i)>10	84	51	28	22	23	22	20	22	23	24
MRR	0.732	0.794	**0.839**	0.812	**0.586**	0.833	0.828	0.832	0.827	0.823

Figure 15 captures the performance change. Analyzing the result, we find there may be two degrees of relevancy of small documents (i.e., sentences) in the corpus to a query. Documents

with high relevance are easily retrieved with ranks smaller than 200. When the rank increases larger than 200, lowly relevant documents are retrieved with more non-relevant documents. That introduces noise for similarity computation. The influence reaches to the worst between ranks 500 and 600, and then goes down since the weights of vector elements are decreased. The other reason may be that some returned sentences in both vectors are complementary when smaller number of sentences is consulted, and the complementary parts show up when more sentences are consulted.

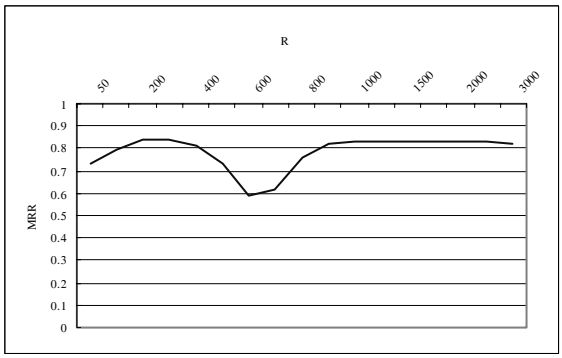

Figure. 15. MRR score vs. Number of Returned Sentences

Term-Vector Approach

Figures 16, 17, 18 and 19 show the results of term-vector approach, where terms in either English or Chinese are used, and two weighting schemes, i.e., Okapi-FN1 and Log-Chi-Square, are applied. The *x* axis represents *k*, the number of terms used for similarity computation. The *y* axis denotes the MRR score.

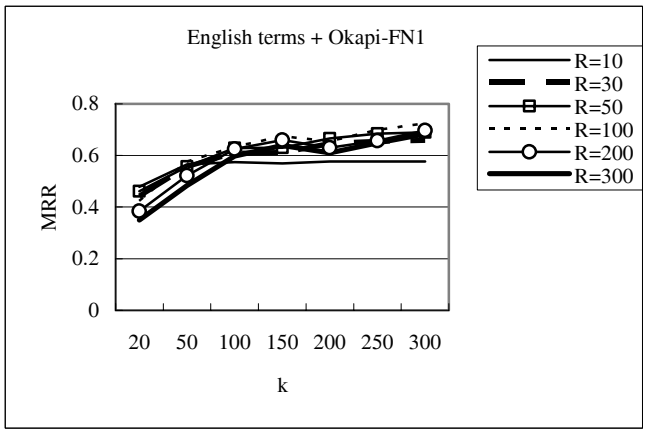

Figure. 16. English Terms plus Okapi-FN1 Weighting Scheme

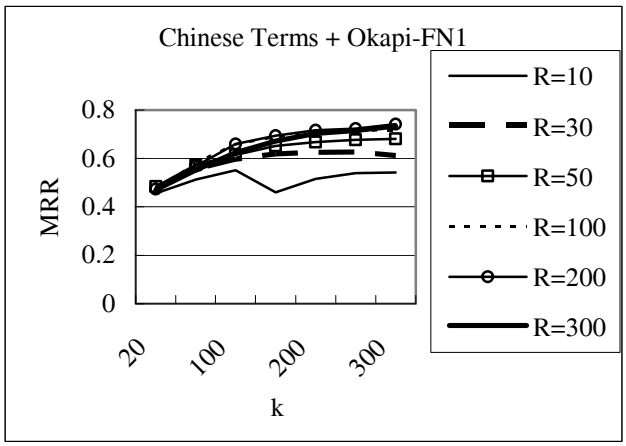

Figure. 17. Chinese Terms plus Okapi-FN1 Weighting Scheme

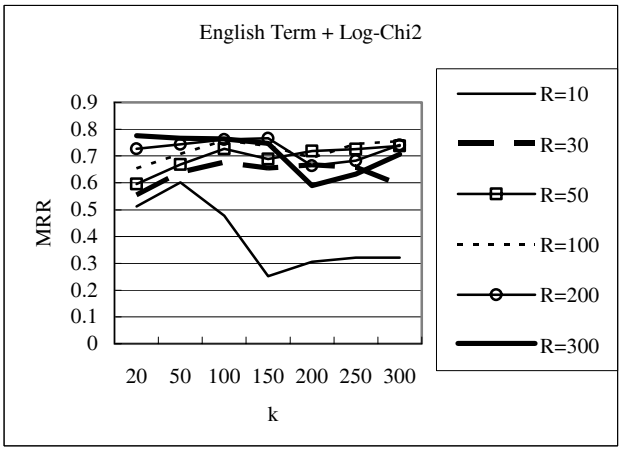

Figure. 18. English Terms plus Log-Chi-Square Weighting Scheme

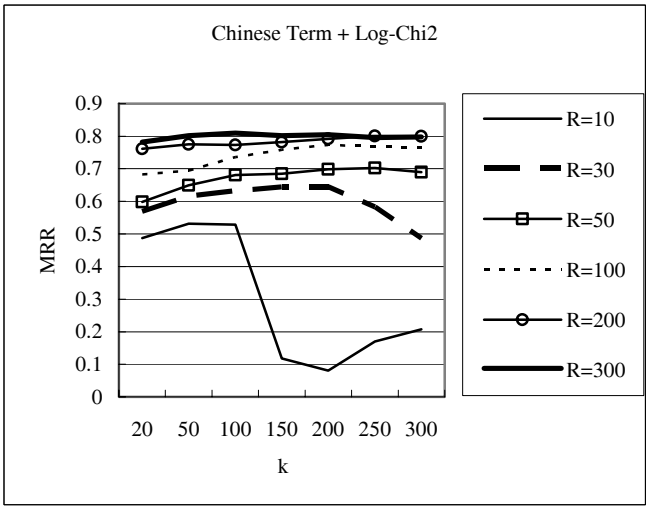

Figure. 19. Chinese Terms plus Log-Chi-Square Weighting Scheme

Several interesting conclusions can be made after the factors of language and weighting schemes are considered. Performances of Figures 16 and 17 are inferior to those of Figures 18 and 19. It shows that Log-Chi-Square weighting scheme is more suitable for term-vector approach than Okapi-FN1 weighting scheme. It meets our expectation that Log-Chi-Square weighting scheme properly captures concepts embedded in resulting sentences returned by Okapi and C-Okapi. Performances of the runs of smaller *k* (=50) in the four figures show that

Log-Chi-Square scheme will assign higher weights to terms which are truly relevant to the sentence (query).

Observing the differences between Figures 16 and 17, and between Figures 18 and 19, we can find that the trends of performances using terms in different languages are dissimilar. Using English terms as vector elements, the performance trend shows undulation as we saw in Figure 14, though the drops are smaller in Figures 16 and 18. On the other hand, performance trend of using Chinese terms as vector elements is monotonously increasing with *k* when *R* is greater than 30. It may indicate that English suffers from more noises, such as word sense ambiguity, than Chinese.

The best performance, near 0.81, appears in the case "*R*=300" of Figure 19, i.e., take Chinese as a basis and Log-Chi-Square formula. It is lower than the best performance 0.84 in sentence-vector approach. The whole performance of term-vector approach is also inferior to sentence-vector approach.

4.4.2 Using HKSAR Corpus

Document-Vector Approach

Figure 8 shows the results of the application of the document-vector approach on HKSAR corpus, which is a document-aligned Chinese-English corpus. The best one has only 30% of the performance shown in Figure 3. The result shows the influence of corpus domain on reference corpus approach. Since the 500 pairs of test sentences are randomly selected from Sinorama corpus, the domain of test sentences and the reference databases are the same, i.e., content focused on major events and construction in Taiwan from 1976-2001. In contrast, the HKSAR corpus contains the news issued by HKSAR within 1997-2000. The test sentences and the reference corpus are totally different in domain of concepts so that there are rarely relevant documents in HKSAR. That introduces much more noises than useful information in ranked list. Besides the domain issue, the small size of HKSAR corpus causes bad effect in retrieval too.

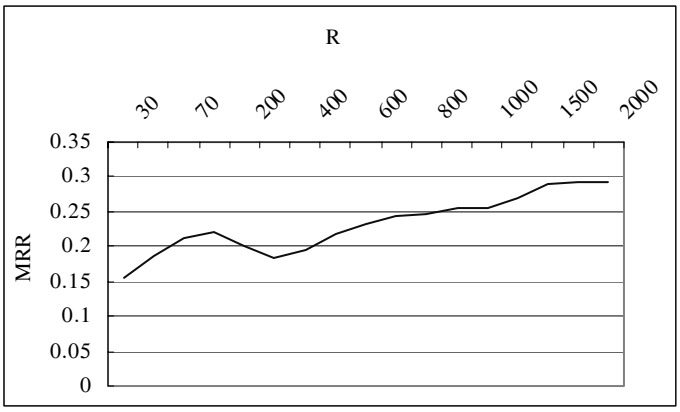

Figure. 20. Document-Vector Approach with HKSAR

Term-Vector Approach

Figures 21 and 22 show the results of term vector approach on HKSAR, using Log-Chi-Square weighting scheme. Chinese-term-based approach (Figure 22) is more robust than English-term-based approach (Figure 21). However, their performance does not compete with that of document-vector approach. As HKSAR is a "document-aligned" parallel corpus, it is more difficult to select terms suitable for information expansion. Thus, the performance goes down from Figure 20 to Figure 21 and Figure 10. The performance drop is more obvious than that between Figure 15 and Figure 19.

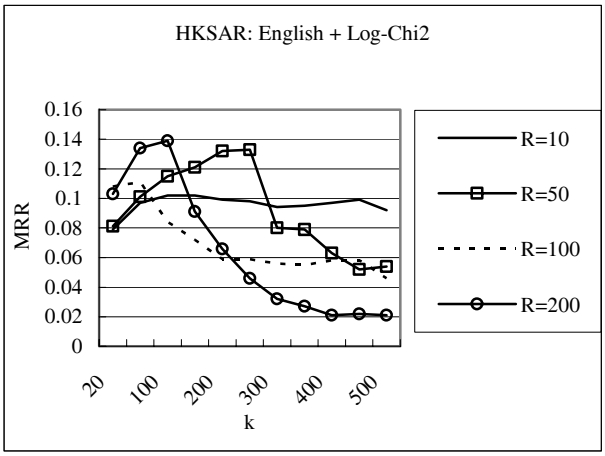

Figure. 21. English Terms plus Log-Chi-Square Weighting Scheme with HKSAR

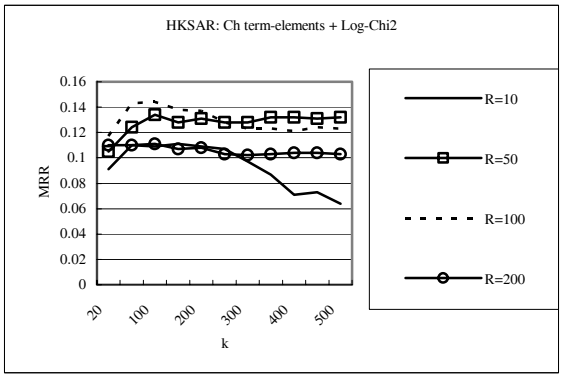

Figure. 22. Chinese Terms plus Log-Chi-Square Weighting Scheme with HKSAR

4.4 Experiments of Multilingual Relevance Detection

Besides evaluating the similarity computation, we also employ the test data in TREC 2002 Novelty track to evaluate the whole multilingual relevance detection. The test data includes 49 topics, each of which is given a set of sentences to evaluate the performance of relevance detector (Harman, 2002). All of these topics and sentences are in English. For multilingual relevant sentence detection, all topics are manually translated into Chinese. Each translated topic (in Chinese) and the corresponding given set of sentences (in English) are sent as queries to C-Okapi and Okapi respectively, so that we can compute similarity between each topic and each sentence in the given set using document-vector or term-vector approach.

Chen, Tsai, and Hsu (2004) use logarithmic regression to simulate the relationship between total number of the given sentences and number of the relevant sentences, in TREC 2002 Novelty track. We adopt the similar approach. A dynamic percentage of sentences most similar to topic *t* in the given set will be reported as relevant. According to the assessment of TREC 2002 Novelty track, we can compute precision, recall, and F-measure for each topic. Figure 23 shows the performance, i.e., average F-measure of 49 topics, using Sinorama and HKSAR as reference corpora, respectively. Sentence-vector approach (Section 4.4.1.1) and document-vector approach (Section 4.4.2.1) are adopted.

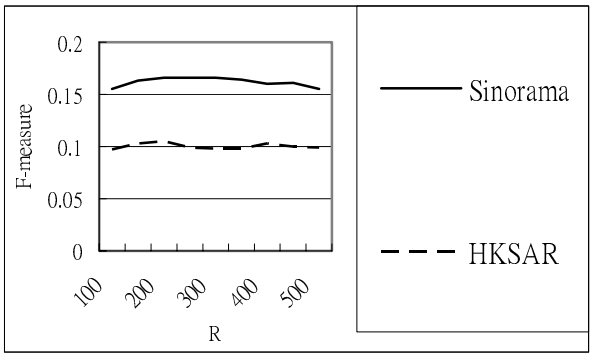

Figure. 23. Document/sentence-vector approach using TREC 2002 topics

Apparently, using Sinorama as a reference corpus outperforms using HKSAR. This result is consistent with the evaluation in similarity computation. Chen, Tsai, and Hsu (2004) used TREC6 text collection, which consists of 556,077 documents, as reference corpus. The best performance using Sinorama for multilingual relevance detection is about 80% of monolingual relevance detection (i.e., 0.212), and using HKSAR is about 50%.

5 Conclusion and Remarks

This paper proposed concept matching and IR approaches to identify sentences that are novel and redundant as well as relevant and irrelevant. Although the method of matching noun and verb keywords and the related expansion achieved average F-measure 0.125, which is better than the baseline performance (i.e., 0.040), words in sentences are still not enough for the relevance detection. We presented an information expansion using a reference corpus to deal with this problem. We postulated that if two sentences have the similar meaning, then their behavior on information retrieval to the reference corpus is similar. Logarithmic regression approximates how many percentages of sentences are relevant for each topic. This value determines an offset from mean in normal distribution and thus the similarity threshold. That forms a rigid procedure instead of heuristics to determine the needed parameters. The experiment results show that Okapi-based relevant detector with dynamic threshold setting, which depend on topics and given sentences, are better than the other approaches. The best average F-measure of relevance detector is 0.212, which is 57.14% of human performance (0.373). When the idea was extended to novelty detector, the average F-measure is 0.207, which is 58.64% of human performance (0.353). The effects of the IR systems, e.g., query construction and relevance feedback, will be investigated. Besides, the deep syntactic and semantic analysis of sentences to distinguish relevant and novel sentences will be explored.

We also consider the kernel operation in multilingual relevant sentence detection. A parallel reference corpus approach is adopted. The issues of aligning granularity, the corpus domain, the corpus size, the language basis, and the term selection strategy are addressed. In the intensive experiments, the best MRR (0.839) is achieved when the test data and the reference corpus come from the same domain, the finer-grained alignment (i.e., sentence alignment), and the larger corpus are adopted. In that case, 77.40% of test data are ranked 1.

Generally speaking, the sentence-vector approach is superior to the term-vector approach when sentence-aligned corpus is employed. The document-vector approach is better than the term-vector approach if document-aligned corpus is used. In term-vector approach, Log-Chi-Square weighting scheme is better than Okapi-FN1 weighting scheme. Considering the language issue, Chinese basis is more suitable to English basis in our experiments. It shows that performance trends may depend on the characteristics of different languages. Comparing the monolingual and multilingual relevance detection, the latter has 80% performance of the former. It shows that IR with reference corpus approach is adapted easily.

Reference

1. 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
2. Allan, J., Carbonnell, J., and Yamron, J.: Topic Detection and Tracking: Event-Based Information Organization. Kluwer (2002)
3. 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
4. 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)
5. 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)
6. 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)
7. 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.
8. 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)
9. Salton, G., and Buckley, C.: Term Weighting Approaches in Automatic Text Retrieval. In Information Processing and Management. Vol. 5, No. 24, pp. 513-523.
10. 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)
11. 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)