

POSTECH at NTCIR-5: Combining Evidences of Multiple Term Extractions for Mono-lingual and Cross-lingual Retrieval in Korean and Japanese

Seung-Hoon Na In-Su Kang Jong-Hyeok Lee
Div. of Electrical and Computer Engineering
Pohang University of Science and Technology (POSTECH)
Advanced Information Technology Research Center (AITrc)
San 31, Hyoja-Dong, Pohang, Republic of Korea, 790-784
{nsh1979, dbaisk, jhlee}@postech.ac.kr

October 20, 2005

Abstract

This paper describes methodologies for NTCIR-5 CLIR involving Korean and Japanese, and reports the official result as well as retrieval results using NTCIR-3 and NTCIR-4 data. We participated in four tasks: K-K and J-J monolingual tracks and K-J and J-K cross-lingual tracks. Unlike English, in Asian languages such as Korean and Japanese term extraction is nontrivial because of segmentation ambiguities. In this regard, we prepared multiple term representations for documents and queries, of which ranked results are merged to generate final ranking. In preliminary experiments using NTCIR-3 and NTCIR-4 data, our model showed the best performances for description queries in Korean and Japanese. In offline results using NTCIR-5 data, our methodology in Korean showed the best performance by achieving 0.5680 for description queries and 0.6159 for others.

Keywords: Information Retrieval, Cross-lingual Information Retrieval, Multiple Evidence Combination, Unsupervised Segmentation, Query Translation, Probabilistic Retrieval Model, Language Modeling Approach

1 Introduction

Unlike English, Chinese and Japanese do not use word delimiters in a normal text. In Korean, no word boundaries exist within *Eojeol*.¹ Thus, word segmentation is nontrivial for the three Asian languages. Compared with Japanese, segmentation problem of Korean is more difficult

¹Eojeol indicates a Korean spacing unit as well as a syntactic unit.

because the basic character unit used in Korean is *Hangul* character not *Hanzi*: the number of different *Hangul* characters is much smaller than that of *Hanzis*.

To avoid word segmentation problem, one can use character n-gram method which produces overlapping n-character strings as index terms. In Korean, the character n-gram method shows stable and robust retrieval performance although it is very simple term extraction method. However, the use of character n-grams has a limitation that they do not make semantically consistent units. Sometimes, the extraction of character n-grams may be dangerous because the method generates a sequence of semantically un-related terms from a given *Eojeol* which may have negative effects on the retrieval performance.

On the other hand, dictionary-based word segmentation can extract semantically consistent units, however, it has the difficulty in segmenting unknown words. Thus, the adaptation of a dictionary is fundamental for higher retrieval performance. However, the hand-driven adaptation of a dictionary is time-consuming. Specially, a dictionary manager may hesitate to decide what is a content word. For example, from “불린함수” (Boolean function), one may extract two content words such as “불린” (Boolean) and “함수” (function), and the other may consider “불린함수” as a single content word. This problem is similar to the phrase extraction problem in English.

To relax such an adaptation problem of dictionary-based word segmentation, we have developed an unsupervised segmentation algorithm without requiring any dictionaries. The algorithm sets a statistical lexicon from a given collection and performs a hybrid segmentation algorithm

based on a rule and statistics on query and documents.

As preliminary experiments, we have performed retrievals using three different term extractions for NTCIR-3 and NTCIR-4 data. Then, from their query-by-query analyses, we have found that the best term extraction scheme is different for each query. This observation makes us build the retrieval system to reflect multiple evidences of different term extractions. For combination of multiple evidences, we used a fusion-based approach which merges retrieval results from multiple representations. We expect that the combination covers some deficits of other extraction methods. For Japanese, we used two term extractions based on Japanese morphological analyzer - COBALT-JK [4] and ChaSen.²

For cross-lingual information retrieval, we use a naive query translation method (NQT) which does not use any word sense disambiguation method based on statistics such as co-occurrence information.

This paper is organized as follows. Section 2 describes an overview of our monolingual retrieval architecture by introducing retrieval model, feedback method, a combination approach and term extraction schemes. In Section 3, we describes cross-lingual retrieval methodologies. Section 4 shows official results and compares them with retrieval results using NTCIR-3 and NTCIR-4 data. Finally, Section 5 provides our conclusion.

2 Monolingual Retrieval

2.1 Overall Architecture

Figure 1 shows the overall architecture of our system for monolingual retrieval in Korean. Basically, the system uses three different term extractions and merges retrieval results from them. The extraction methods are *Character Bi-gram*, *Dictionary-Based Word* and *Collection-Based Segment*. Our intuition is that each extraction method plays discriminative effects on retrieval performance, and can relax the problem of segmentation difficulty. In addition to the combination of term representations, two different retrieval models are combined to optimize the retrieval performance at different retrieval strategies - probabilistic retrieval model [13] and language modeling approach [12]. In pseudo relevance feedback, we use different methods according to the length of query - Model-based feedback [16] for long queries and expansion-based feedback based on likelihood ratio [12] for short queries.

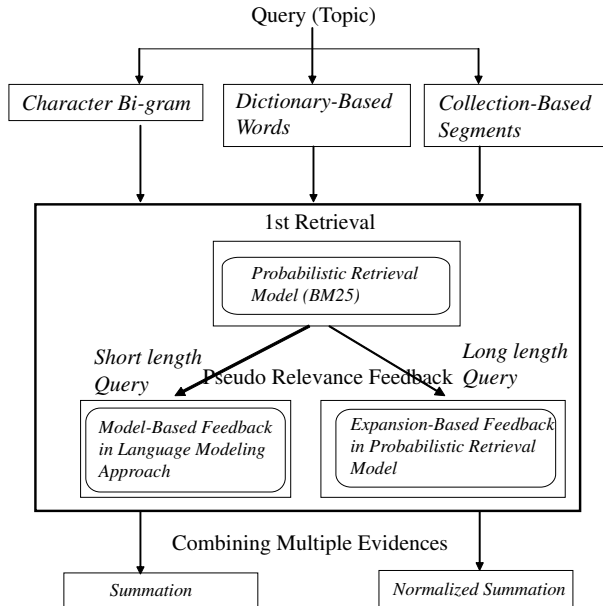


Figure 1: Overall architecture for monolingual retrieval of Korean

2.2 Retrieval Model

The initial retrieval is performed by the BM25 formula of Okapi. Pseudo relevance feedback is executed by using model-based feedback for short queries, and expansion-based feedback for long queries. In pseudo relevance feedback, the use of different strategies according to query length is motivated from our previous research [8]. Okapi's term weighting formula of term t_i in document D_j is as Eq.(1)

$$w_{ij} = w_i' \frac{tf_{ij}}{K + tf_{ij}} \frac{qt f_i}{k_3 + qt f_i} \quad (1)$$

where K is $k_1((1-b) + b \frac{dl_j}{avg dl})$ and tf_{ij} is term frequency of t_i in document D_j . w_i' is based on the Robertson-Sparck Jones weight [14], which is reduced inverse document frequency weight without relevance information ($R = r = 0$) as Eq.(2).

$$w_i' = \log \frac{(r_i + 0.5)/(R_i - r_i + 0.5)}{(n_i - r_i + 0.5)/(N - n_i - R + r_i + 0.5)} \quad (2)$$

where N is the number of documents and R is the number of relevant documents, n_i is the document frequency of t_i and r_i is the frequency of documents to be relevant containing t_i . k_1 , b and k_3 are set to 2.0, 0.75 and ∞ , respectively.

Model-based feedback is performed on top retrieved documents (feedback documents) \mathcal{F} [16]. Query model is estimated by using EM algorithm to maximize likelihood of top-retrieved documents given a mixture model which consists of unknown

²<http://chasen.naist.jp/>

query model θ_Q and background collection language model θ_C . Unlike original Zhai’s approach, we modified the likelihood of feedback documents by reflecting the score of retrieved documents as follows.

$$\mathcal{L} = \sum_i \sum_{d_j \in \mathcal{F}} t f_{ij} rel_j \log \left(\frac{(1 - \lambda)P(t_i|\theta_Q)}{+\lambda P(t_i|\theta_C)} \right) \quad (3)$$

where rel_j is the relevance score of d_j . Given query Q and document model θ_{D_j} , rel_j is formulated as

$$rel_j = \kappa + (1 - \kappa) \frac{\log P(Q|\theta_{D_j})}{\max_j \log P(Q|\theta_{D_j})} \quad (4)$$

where κ is a tuning parameter. In our preliminary experimentation ($\kappa = 0.7$) using NTCIR-3 and NTCIR-4 Korean test sets, the modified likelihood showed slightly better performance with about 1% difference.

Expansion-based feedback has only been dealt with heuristically in a given retrieval model. The original query is usually literally expanded by adding additional terms to it based on some criterion. Our criterion is Ponte’s likelihood ratio [12] as follows.

$$Score(t_i) = \sum_{d_j \in \mathcal{F}} \log \left(\frac{P(t_i|\theta_{D_j})}{P(t_i|\theta_C)} \right) \quad (5)$$

After adding terms into original query, all of them are entered as an input to probabilistic retrieval model without re-weighting.

2.3 Term Extraction

For Korean, we prepared three different methods for term extraction as follows.

Character Bi-gram *Character Bi-gram* is the well-known term extraction method for Asian languages such as Korean, Japanese and Chinese [6]. Character bi-gram consists of two consequent Korean characters (*Emjeols* in Korean). Special characters such as numeric and English characters are pre-extracted. For example, for Eojeol ‘배아줄기세포’ (embryonic stem cell), terms of ‘배아’ (embryonic), ‘아줄’ (non-sense syllables), ‘줄기’ (stem), ‘기세포’ (spirit) and ‘세포’ (cell) are extracted.

Dictionary-Based Word *Dictionary-Based Word* is produced by applying our Korean morphological analyzer. Our morphological analyzer selects content nouns and numerical words by using compound-noun segmentation based on longest-matching rule [3]. The size of dictionary is about 230,000 nouns, and its entries contains most Korean words and modern foreign words.

Collection-Based Segment *Collection-Based Segments* are extracted by applying unsupervised segmentation algorithm without dictionary. This problem is related to automatic lexicon construction [1, 15, 10]. In information retrieval, unsupervised method is motivated from the fact that there are many unknown words in a given test collection, thus, the segmentation performance for the given corpus is not acceptable without hard-tuning to the domain of collection. By using unsupervised method, unknown terms can be automatically learned based on collection statistics. As a result, we can expect that segmentation accuracy will be improved. Our unsupervised method is different from incremental approaches [1, 15] and iterative approaches [10]. Our method basically employs global search, but does not attempt to learn the statistical dictionary.³ Instead, we focus on pruning unhelpful segmentation candidates over the search space based on simple principle. The unsupervised segmentation algorithm will be described in the next sub-section.

For Japanese, we prepared two methods for term extractions. One method is based on Japanese morphological analyzer of COBAL-T-JK, and another method is based on Chasen. In Japanese, we did not apply unsupervised segmentation.

2.4 Unsupervised Segmentation Method

Let us assume that we have a raw corpus \mathcal{C} and we want to segment an n-character string $T = c_1 \dots c_n$ (c_i is the i -th character). As an alternative notation for $c_1 \dots c_n$, we use c_{1n} . First, we create the statistical dictionary D that is a set of all-length character n-grams of each string in \mathcal{C} . In order to find the most likely segmentation candidate S^* of T , we should calculate Eq.(6), where k -th segmentation candidate is represented as $S_k = s_1 \dots s_{m(k)}$ (s_i is the i -th segment which belongs to D , and $m(k)$ is the index of the last segment of S_k , and $m(k) \leq n$). Note that a segment covers one or more contiguous characters in T . We interpret $P(S_k)$ as the probability that T is decomposed into a sequence of $s_1, s_2, \dots, s_{m(k)}$.

$$S^* = \operatorname{argmax}_{S_k = s_1 \dots s_{m(k)}} P(S_k) \quad (6)$$

The calculation of $P(S_k)$ is simplified to Eq.(7) by assuming the independence between segments which has been adopted by most unsupervised segmentation methods.

³Global search considers all possible segmentation candidates to select the most likely one

