Automatic Acquisition of Phrasal Knowledge for English-Chinese Bilingual Information Retrieval

Lee-Feng Chien

Institute of Information Science, Academia Sinica,
Taipei, Taiwan, R.O.C.
E-mail: lfchien@iis.sinica.edu.tw

Abstract

Extraction of phrasal knowledge, such as proper names, domain-specific keyphrases and lexical templates from a domain-specific text collection are significant for developing effective information retrieval systems for the Internet. In this paper, we are going to briefly introduce our ongoing research on automatic phrasal knowledge acquisition for English-Chinese bilingual texts. In addition to the technique of keyphrase extraction which will be described in details, the underlying techniques consist of lexical template extraction, phrase translation extraction and high-order Markov language model construction are introduced in short.

1. Introduction

Capability of phrasal information extraction is crucial in an IR system for the increase of retrieval effectiveness and development of advanced searching techniques, such as automatic term suggestion and cross-language information retrieval [1,2]. Considering the inherent linguistic differences between English and Chinese languages[3] and the demand of high-performance phrasal knowledge acquisition, we are developing an approach in acquisition of English-Chinese bilingual information automatically from Internet resources. The proposed approach is formed as an abstract diagram shown in Fig. 1, in which an IR system is composed of a knowledge acquisition subsystem for extracting phrasal information on-line. The knowledge acquisition subsystem, which consists of keyphrase extraction module, lexical template extraction module, phrase translation extraction and language model construction module, as will be briefly described below, is served to increase the capability of phrasal information extraction, with the input of domain-specific texts in the IR system. At present, although there are several techniques have been successfully developed [4,5,6] at the initial stage, the whole research is still under exploration.

2. Overview of the Developing Techniques

PAT-tree-based High-order Markov Language Model Construction

Statistical N-gram language models are often used in many NLP systems to estimate the probability values or word associations of any word pairs or sequences. For reducing the complexity of model representation, bigram or trigram models are frequently used as an approximation. Of course, this will decrease the power of language modeling. According to our experiments, techniques such as PAT-tree indexing used for recording full-text documents in IR systems can be more efficient in representing high-order N-gram language models, especially for when the training corpus is large and dynamic[5]. The PAT tree actually provides indices to all possible streams of characters or words with an arbitrary length N, where N can be significantly large than 2 or 3, together with the frequency counts for these streams in the network resource databases. Since the content of the searching database can be taken as the corpus to train a domain-specific language model for the database. All of the required statistical parameters for a N-gram model can be extracted directly from the PAT tree. The language model can be easily adapted with the update of the database content and the corresponding PAT tree. The PAT-tree-based language model is efficient in modeling of phrasal information such as lexical patterns and proper names with the change of database contents. The techniques which will be described below actually rely on the basis.

.

Lexical Template Extraction

In addition to the extraction of keyphrases for domain-specific term lexicon construction as in Fig.1, the proposed approach is expected able to identify lexical templates as linguistic cues for extracting significant keyphrases. A lexical template is defined to be a text pattern consisting of separate strings, which frequently occur together and have replaceable keyphrases in the middle. For example, "*approach to .... in...*" in the strings "*approach to* Internet searching *in* Chinese regions" and "*approach to* computer networking *in* application to" is an example of lexical templates. If such a template can be identified, many keyphrases, which are difficult to extract for their low occurrences or with frequently-used composed words, maybe can be treated in certain degree. The technique for such a purpose is still under developing. An initial idea is that, for each data stream with possibility of a keyphrase composed, it will construct a lattice of data streams from a set of top n similar sentences or sentence fragments, which will be retrieved from the searching database. A searching algorithm which can judge whether there are lexical templates constituted in the lattice are developing.

Phrase Translation Extraction

It is very important for cross-language information retrieval, if the developed techniques are capable of extracting representative phrase translations automatically from comparable bilingual text collection. A preliminary study for such a purpose is undertaking in the presented research. The developing technique is basically a three-step process for English-Chinese comparable texts: similar documents clustering, keyphrase extraction and phrase translation extraction. In the step of document clustering, each English document in the examined comparable text will be first translated into corresponding Chinese text and a feature vector regarding to the transcribed text will be assigned, in order to find similar Chinese documents with similarity estimation in Chinese. At the same time, each Chinese document in the text collection will be also assigned a feature vector with the same method. The clustering processing will be performed for each English document to find all of its similar Chinese documents as the output of the step.

After the similar Chinese documents have been found for each English document[8], the extraction of keyphrases will be performed. For each English document, we extract keywords or keyphrases from its original English text, and extract a set of keyphrases (possible translation equivalents) from its Chinese documents. The used English keyphrase extraction method here is typical, which is mainly based on analysis of POS tagging and compound nouns. But, considering the difficulty of Chinese text segmentation, the adopted Chinese keyphrase extraction is an effective and specially-designed approach [6]. Finally, for each of the extracted English keywords or keyphrases, we will determine translation equivalents in Chinese from the keyphrases extracted from its similar Chinese documents. The method used for determination is based on both linguistic analysis and statistical measurement.

3. PAT-tree-based Keyphrase Extraction

The proposed technique for keyphrase extraction is a statistics-based three-stage approach[6, 7]. In the first step, the above PAT tree is constructed for each database (text collection) or cluster of databases, in which all possible character strings with arbitrary length together with their frequency counts in the text can be retrieved and updated very efficiently, although not every character string needs to be actually stored.

3.1 Filtering of Incomplete Lexical Patterns

In the second step, a mutual-information-based filtering algorithm is applied to filter out the character strings in the PAT trees which are incomplete in semantics. By estimating the

associations among the neighboring characters as well as the component segments within the character strings, this algorithm can efficiently delete many less significant lexical patterns. In this step, those character strings with number of distinct predecessors or successors in the databases less than a predefined threshold are first filtered out, since a string always co-occurring with a small number of characters is very often only a part of a complete string. The basic concept of such a processing is formally defined below:

Let x be a character string to be examined and L be a set of unique left adjacent character strings (LS) of x in the text collection, which may be a single character, character bigram, word etc., depending on the parameters setting in the application. R is a set of right adjacent character strings (RS) of x. First, we define *Left Context Dependency* (LCD) and *Right Context Dependency* (RCD) as follows:

Definition : Left Context Dependency (LCD)

x has *LCD* if $|L| < t1$ or $\text{MAX}_{\alpha} f(\alpha x)/f(x) > t2$, where $\alpha \in L$, $|L|$ means the number of unique left adjacent character strings of x, f(.) is a frequency function, *t1*and *t2* are threshold values.

Definition: Right Context Dependency (RCD)

x has *RCD* if $|R| < t1$ or MAX $f(x\ )/f(x) > t2$, where $\in R$, $|R|$ means the number of unique right adjacent characters of x, f(.) is a frequency function, *t1*and *t2* are threshold values.

The above two metrics are actually used to check if x has complete lexical boundaries, by judging the usage freedom of x according to its contextual information. The basic assumption is that if x has few unique LSs or RSs, or if it frequently occurs together with certain LSs or RSs, it might be incomplete in semantics. According to our observations from experiments, using the above analysis, most incomplete character strings can be filtered out. For instance, the use of the character string "**李登**" can be detected as being very limited by means of RCD checking. In fact in a simple test with a news abstracts database of about 5MB, it was found that "**李登**" appeared 640 times and was always followed by "**輝**", creating the personal name "**李登輝**" (Lee Deng-hui, the president of the R.O.C.). Similarly, the use of the character string "**統府**" is also limited as determined by LCD checking. It was found in the same test that "**統府**" appeared 561 times and was always preceded by "**總**", creating the proper noun "**總統府**" (office of the president). On the contrary, the use of character string "**總統府**" was rather free as determined by both LCD and RCD checking. This string had 36 different left adjacent lexical patterns, 60 right adjacent lexical patterns and occurred a total of 561 times. According to our analysis, a character string which has free usage is almost

rigid in semantics and has complete lexical boundaries, especially when it is a longer string.

Furthermore, for character strings surviving after the above filtering process, the mutual-information for each string X is then evaluated[Church' 90]:

$$MI(X) = \frac{f(X)}{f(X_s) + f(X_e) - f(X)}$$

Where $MI(X)$ is the mutual information of a target string X, X_s is the longest starting sub-string of X, i.e., the sub-string which is exactly X except that the last character of X is deleted, X_e is the longest ending sub-string of X, i.e., the sub-string which is exactly X except that the first character of X is deleted, and $f(X)$, $f(X_s)$, $f(X_e)$, are the frequency counts of X, X_s, and X_e, in the text respectively. Character stings with the above mutual information below a threshold are still considered to be incomplete and deleted again. In fact, it is worthy to note that the above metrics are easily to be accessed with the PAT trees.

3.2 Extraction of Significant Lexical Patterns

The third step is to determine the finally selected significant lexical patterns as the keyphrases or key phrases. Here an algorithm based on a commonly used word lexicon, a general-domain corpus and a keyphrase selection strategy is performed, under which those lexical patterns appearing frequently and common to many texts, thus not really specific for a given text record will be removed. A parameter based on the normalized inverse document frequency often used in information retrieval is taken as an index here to decide whether a character string is a significant lexical pattern.

3.3 Preliminary Experiments

The above keyphrase extraction approach has been tested extensively and found very powerful in the search of domain-specific significant lexical patterns. In an example test using a total of 5,819 news abstracts related to congress and politics(3.6 MB Chinese characters) to construct PAT trees and extract keyphrases, most of the significant lexical patterns extracted are actually major keyphrases or key phrases, and many of them are in fact domain-specific and can' t be found in a general domain lexicon.

For illustration, the results are shown in Table 1, where "keyphrase length" is the number of characters in an extracted keyphrase. Since keyphrases with different lengths behave differently (for example three-character keyphrases are very often personal names, and four-character or longer keyphrases are very often compound words), the results are shown with the keyphrase length as a special parameter. It can be found that a total of 3,568 two-

character keyphrases were extracted automatically, and that 3,311 out of these 3,568 keyphrases were manually examined and found to be correct keyphrases with complete semantic meanings; thus, a precision rate of 92.80% (3311/3568) was achieved. A similar situation was also observed for longer keyphrases. The precision rate for three-character keyphrases was relatively low because many frequently used single-character words and two-character words are easily combined to produce three-character terms which are not necessarily key elements for information retrieval. Taking all the keyphrases with different lengths into account, a total of 6,082 keyphrases were automatically extracted, and 81.55% of them (4,960 keyphrases) were manually examined and found to be correct. Next, this set of 4,960 correctly extracted keyphrases was further compared with a lexicon of roughly 85,000 commonly used words, which is believed to be large enough for daily use. The last column in Table 1 shows that 1,836 keyphrases among these 4,960 correctly extracted keyphrases were not included in the lexicon. Close examination of these 1,836 keyphrases indicates that most of them are domain-specific such as personal names or proper nouns, which are often very important in information retrieval. This phenomenon is especially significant for keyphrases with three or more characters. As can be found in Table 1, only 552 out of the 3,311 correctly extracted two-character keyphrases were outside of the common-word lexicon, but more than half (385 out of 661 and 899 out of 988) of the correctly extracted keyphrases with three or more characters were outside of the common-word lexicon.

Another experiment was on a Chinese book with a total of about 200,000 Chinese characters. This book had been indexed manually first and a total of 190 keyphrases extracted. With the proposed approach a total of 276 keyphrases were automatically extracted, 38% (precision rate) of which are among the 190 keyphrases extracted manually, and 75% (recall rate) of the manually selected 190 keyphrases have been extracted automatically. Also, though 62% of the automatically extracted keyphrases had not been selected manually, it was found that most of them are actually domain-specific terms with indexing functions. The obtained results show that, if a collection of domain-specific documents can be given, the keyphrases without limitation of string length, including proper names, locations, translated terms, technical terms, abbreviations and even topic terms, can be adatively and effectively extracted using this approach. That is different from conventinal approaches which focus on extraction of specific proper nouns such as human names or unknown words [Chen' 96].

References

1. Lisa Ballesteros and W. Bruce Croft, Phrasal Translation and Query Expansion Techniques for Cross-Language Information Retrieval, ACM SIGIR' 97, 84-91.
2. Douglas W. Oard, "Cross-language Text Retrieval Research in the USA," April 1997. (http://www.area.pi.cnr.it/ErcimDL/third-DELOS-workshop/Oard/oard-delos/paper.html)

3. K. L. Kwok, “Evaluation of an English-Chinese Cross-Lingual Retrieval Experiment”. Working Notes on AAAI Spring Symposium on Cross-Language Text and Speech Retrieval, Stanford University, 1997
4. Lee-Feng Chien, ‘ Fast and Quasi-Natural Language Search for Gigabytes of Chinese Texts“, ACM SIGIR’ 95.
5. Lee-Feng Chien ,Sung-Chien Lin, et al., “Internet Chinese Information Retrieval Using Unconstrained Mandarin Speech Queries Based on A Client-Server Architecture and A PAT-tree-based Language Model”, Proceedings of the 1997 IEEE International Conference on Acoustics, Speech and Signal Processing, German, pp. 1155-1158 (ICASSP‘ 97).
6. Lee-Feng Chien. PAT-tree-based Keyword Extraction for Chinese Information Retrieval”, ACM SIGIR’ 97, 50-59.
7. Lee-Feng Chien , PAT-Tree-Based Adaptive Keyphrase Extraction for Intelligent Chinese Information Retrieval , accepted and to appear on special issue on “Information Retrieval with Asian Languages”, Information Processing and Management ,1998.
8. Chien-Kang Huang, Yen-Chen Oyang and Lee-Feng Chien, Cross-Language Similar Document Retrieval from Comparable English-Chinese Texts, submitted to the 3rd Workshop on Information Retrieval with Asian Languages(IRAL’ 98), May, 1998.

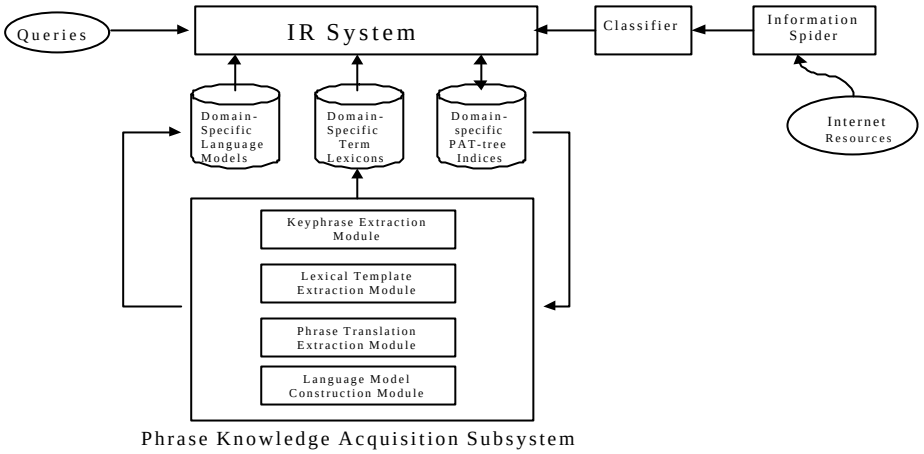

Fig. 1 A abstract diagram showing the proposed approach.

Keyphrase Length	Total Number of Extracted Keyphrases	Number of Correct Keyphrases Extracted	Precision	No. of Correct Keyphrases outside Dictionary
2	3,568	3,311	92.80%	552
3	1,130	661	58.50%	385
4	999	687	68.77%	598
5	207	150	72.46%	150
>=6	178	151	84.83%	151
Total	6,082	4,960	(Avg.)81.55%	1,836

Table 1. The results for keyphrase extraction from the test database.