Automatic Collection of Related Terms from the Web

Satoshi Sato and **Yasuhiro Sasaki**
Graduate School of Informatics
Kyoto University
Sakyo, Kyoto, 606-8501
Japan
sato@i.kyoto-u.ac.jp, sasaki@pine.kuee.kyoto-u.ac.jp

Abstract

This paper proposes a method of collecting a dozen terms that are closely related to a given *seed* term. The proposed method consists of three steps. The first step, compiling corpus step, collects texts that contain the given seed term by using search engines. The second step, automatic term recognition, extracts important terms from the corpus by using Nakagawa's method. These extracted terms become the candidates for the final step. The final step, filtering step, removes inappropriate terms from the candidates based on search engine hits. An evaluation result shows that the precision of the method is 85%.

1 Introduction

This study aims to realize an automatic method of collecting technical terms that are related to a given seed term. In case "natural language processing" is given as a seed term, the method is expected to collect technical terms that are related to natural language processing, such as morphological analysis, parsing, information retrieval, and machine translation. The target application of the method is automatic or semi-automatic compilation of a glossary or technical-term dictionary for a certain domain. Recursive application of the method enables to collect a list of terms that are used in a certain domain: the list becomes a glossary of the domain. A technical-term dictionary can be compiled by adding an explanation for every term in the glossary, which is performed by *term explainer* (Sato, 2001).

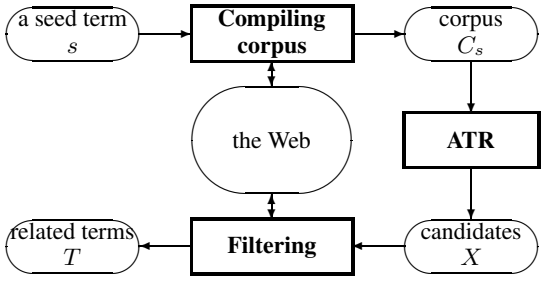

Figure 1: System configuration

Automatic acquisition of technical terms in a certain domain has been studied as automatic term recognition (Kageura and Umino, 1996; Kageura and Koyama, 2000), and the methods require a large corpus that are manually prepared for a target domain. In contrast, our system, which is proposed in this paper, requires only a seed word; from this seed word, the system compiles a corpus from the Web by using search engines and produces a dozen technical terms that are closely related to the seed word.

2 System

Figure 1 shows the configuration of the system. The system consists of three steps: compiling corpus, automatic term recognition (ATR), and filtering. This system is implemented for Japanese language.

2.1 Compiling corpus

The first step, compiling corpus, produces a corpus C_s for a seed term s. In general, compiling corpus is to select the appropriate passages from a document set. We use the Web for the document set and select the passages that describe s for the corpus. The actual procedure of compiling corpus is:

1. Web page collection
   For a given seed term s, the system first makes four queries: "s toha", "s toiu", "s ha", and "s", where toha, ha, and toiu are Japanese functional words that are often used for defining or explaining a term. Then, the system collects the top K (= 100) pages at maximum for each query by using a search engine. If a collected page has a link whose anchor string is s, the system collects the linked page too.
2. Sentence extraction
   The system decomposes each page into sentences, and extracts the sentences that contain the seed term s.

The reason why we use the additional three queries is that they work efficiently for collecting web pages that contain a definition or an explanation of s. We use two search engines, Goo[1] and Infoseek[2]. We send all four queries to Goo but only the query "s" to Infoseek, because Infoseek usually returns the same result for the four queries. A typical corpus size is about 500 sentences.

2.2 Automatic term recognition

The second step, automatic term recognition (ATR), extracts important terms from the compiled corpus. We use Nakagawa's ATR method (Nakagawa, 2000), which works well for Japanese text, with some modifications. The procedure is as follows.

1. Generation of term list
   To make the term list L by extracting every term that is a noun or a compound noun from the compiled corpus.
2. Selection by scoring
   To select the top N (= 30) terms from the list L by using a scoring function.

For the scoring function of a term x, we use the following function, which is multiplying Nakagawa's Imp_1 by a frequency factor $F(x, L)^\alpha$.

$$score(x, L) = Imp_1(x, L) \times F(x, L)^\alpha$$

$$F(x, L) = \begin{cases} 1 & \textbf{if } x \text{ is a single noun} \\ \text{"frequency of } x \text{ in } L\text{"} & \textbf{otherwise} \end{cases}$$

[1] www.goo.ne.jp
[2] www.infoseek.co.jp

While Nakagawa's Imp_1 does not consider term frequency, this function does: α is a parameter that controls how strongly the frequency is considered. We use $\alpha = 0.5$ in experiments.

The result of automatic term recognition for "自然言語処理 (natural language processing)" is shown in the column *candidate* in Table 1.

2.3 Filtering

The filtering step is necessary because the obtained candidates are noisy due to the small corpus size. This step consists of two tests: technical-term test and relation test.

2.3.1 Technical-term test

The technical-term test removes the terms that do not satisfy conditions of technical terms. We employ the following four conditions that a technical term should satisfy.

1. The term is sometimes or frequently used in a certain domain.
2. The term is not a general term.
3. There is a definition or explanation of the term.
4. There are several technical terms that are related to the term.

We have implemented the checking program of the first two conditions in the system: the third condition can be checked by integrating the system with *term explainer* (Sato, 2001), which produces a definition or explanation of a given term; the fourth condition can be checked by using the system recursively.

There are several choices for implementing the checking program. Our choice is to use the Web via a search engine. A search engine returns a number, *hit*, which is an estimated number of pages that satisfy a given query. In case the query is a term, its *hit* is the number of pages that contain the term on the Web. We use the following notation.

$$H(x) = \text{"the number of pages that contain the term } x\text{"}$$

The number $H(x)$ can be used as an estimated frequency of the term x on the Web, i.e., on the hugest set of documents. Based on this number, we can infer whether a term is a technical term or not: in case the number is very small, the term is not a

Table 1: Result for "natural language processing"

candidate	Tech.	Rel.
自然言語処理 (natural langauge processing; NLP)	-	-
自然言語処理技術 (NLP technology)	√	√
自然言語処理システム (NLP system)	√	√
自然言語処理研究 (NLP research)		
自然言語処理学 (NLP study)	√	√
処理 (processing)		
テキスト処理 (text processing)	√	
研究開発 (research and development)		
情報処理学会 (Information Processing Society of Japan; IPSJ)	√	√
意味処理 (semantic processing)	√	√
音声処理 (speech processing)	√	
音声情報処理 (speech information processing)	√	√
情報処理 (information processing)		
自然言語処理分野 (NLP domain)		
研究分野 (research field)	√	√*
構文解析 (parsing)	√	√
情報検索 (information retrieval)	√	√
自然言語処理研究会 (SIGNLP)	√	√
音声認識 (speech recognition)	√	√
機械翻訳 (machine translation)	√	√
形態素解析 (morphological analysis)	√	√
情報処理システム (information processing system)	√	
研究 (research)		
意味解析 (semantic analysis)	√	√
自然言語処理学講座 (chair of NLP)	√	√*
自然言語処理シンポジウム (NLP symposium)		
応用システム (application system)	√	
知識情報処理 (knowledge information processing)	√	√
言語 (language)		
情報 (information)		

technical term because it does not satisfy the first condition; in case the number is large enough, the term is probably a general term so that it is not a technical term. Two parameters, Min and Max, are necessary here. We have decided that we use search engine Goo for $H(x)$, and determined $Min = 100$ and $Max = 100,000$, based on preliminary experiments.

In summary, our technical-term test is:

If $100 \leq H(x) \leq 100,000$
then x is a technical term.

2.3.2 Relation test

The relation test removes the terms that are not closely related to the seed term from the candidates. Our conditions of "x is closely related to s" is: (1) x is a broader or narrower term of s; or (2) relation degree between x and s is high enough, i.e., above a given threshold.

The candidate terms can be classified from the viewpoint of term composition. Under a given seed term, we introduce the following five types for classification.

Type 0 the given seed term s: e.g., 自然 言語 処理 (natural language processing)
Type 1 a term that contains s: e.g., 自然 言語 処理 システム (natural language processing system)
Type 2 a term that is a subsequence of s: e.g., 自然 言語 (natural language)
Type 3 a term that contains at least a component of s: e.g., 言語 解析 (language analysis)
Type 4 others: e.g., 構文 解析 (parsing)

The reason why we introduce these types is that the following rules are true with a few exception: (1) A type-1 term is a narrower term of the seed term s; (2) A type-2 term is a broader term of the seed term s. We assume that these rules are always true: they are used to determine whether x is a broader or narrower term of s.

To measure the relation degree, we use conditional probabilities, which are calculated from search engine hits.

$$P(s|x) = \frac{H(s \wedge x)}{H(x)}$$
$$P(x|s) = \frac{H(s \wedge x)}{H(s)}$$

where

$H(s \wedge x)$ = "the number of pages that contain both s and x"

One of two probabilities is equal to or greater than a given threshold Z, the system decides that x is *closely related* to s. We use $Z = 0.05$ as the threshold.

In summary, our relation test is:

If x is type-1 or type-2; or
$P(s|x) \geq 0.05$ or $P(x|s) \geq 0.05$
then x is closely related to s.

The result of the filtering step for "自然言語処理 (natural language processing)" is in Table 1; a

Table 2: Experimental Result

domain	Evaluation I			Evaluation II					
	correct	incorrect	total	S	F	A	C	R	total
natural language processing	101 (93%)	8 (7%)	109	6	3	14	11	8	43
Japanese language	71 (81%)	17(19%)	88	7	0	19	5	1	32
information technology	113 (88%)	15 (12%)	128	10	5	27	13	0	55
current topics	106 (91%)	10 (9%)	116	2	0	13	19	5	39
persons in Japanese history	128 (76%)	41 (24%)	169	18	0	23	1	0	42
Total	519 (85%)	91(15%)	610	43	8	96	49	14	210

check mark '$\surd$' indidates that the term passed the test. Twenty terms out of the thrity candidate terms passed the first techinical-term test (Tech.) and sixteen terms out of the twenty terms passed the second relation test (Rel.). The final result includes two inappropriate terms, which are indicated by '*'.

3 Experiments and Disucssion

First, we examined the precision of the system. We prepared fifty seed terms in total: ten terms for each of five genres; natural language processing, Japanese language, information technology, current topics, and persons in Japanese history. From these fifty terms, the system collected 610 terms in total; the average number of output terms per input is 12.2 terms. We checked whether each of the 610 terms is a correct related term of the original seed term by hand. The result is shown in the left half (Evaluation I) of Table 2. In this evaluation, 519 terms out of 610 terms were correct: the precision is 85%. From this high value, we conclude that the system can be used as a tool that helps us compile a glossary.

Second, we tried to examine the recall of the system. It is impossible to calculate the actual recall value, because the ideal output is not clear and cannot be defined. To estimate the recall, we first prepared three to five *target* terms that should be collected from each seed word, and then checked whether each of the target terms was included in the system output. We counted the number of target terms in the following five cases. The right half (Evaluation II) in Table 2 shows the result.

S: the target term was collected by the system.
F: the target term was removed in the filtering step.
A: the target term existed in the compiled corpus, but was not extracted by automatic term extraction.
C: the target term existed in the collected web pages, but did not exist in the compiled corpus.
R: the target term did not exist on the collected web pages.

Only 43 terms (20%) out of 210 terms were collected by the system. This low recall primarily comes from the failure of automatic term recognition (case A in the above classification). Improvement of this step is necessary.

We also examined whether each of the 210 target terms passes the filtering step. The result was that 133 (63%) terms passed; 44 terms did not satisfy the condition $H(x) \geq 100$; 15 terms did not satisfy the condition $H(x) \leq 100,000$; and 18 terms did not pass the relation test. These experimental results suggest that the ATR step may be replaced with a simple and exhaustive term collector from a corpus. We have a plan to examine this possibility next.

References

Kyo Kageura and Teruo Koyama. 2000. Special issue: Japanese term extraction. *Terminolgy*, 6(2).

Kyo Kageura and Bin Umino. 1996. Methods of automatic term recognition: A review. *Terminology*, 3(2):259–289.

Hiroshi Nakagawa. 2000. Automatic term recognition based on statistics of compound nouns. *Terminology*, 6(2):195–210.

Satoshi Sato. 2001. Automated editing of hypertext résumé from the world wide web. In *Proceedings of 2001 Symposium on Applications and the Internet (SAINT 2001)*, pages 15–22.