Automated Generalization of Phrasal Paraphrases from the Web*

Weigang Li
School of Computer Science and Technology, Box 321, Harbin Institute of Technology, Harbin, P.R. China, 150001
lee@ir.hit.edu.cn

Ting Liu
School of Computer Science and Technology, Box 321, Harbin Institute of Technology, Harbin, P.R. China, 150001
tliu@ir.hit.edu.cn

Yu Zhang
School of Computer Science and Technology, Box 321, Harbin Institute of Technology, Harbin, P.R. China, 150001
zhangyu@ir.hit.edu.cn

Sheng Li
School of Computer Science and Technology, Box 321, Harbin Institute of Technology, Harbin, P.R. China, 150001
lis@ir.hit.edu.cn

Wei He
School of Computer Science and Technology, Box 321, Harbin Institute of Technology, Harbin, P.R. China, 150001
truman@ir.hit.edu.cn

Abstract

Rather than creating and storing thousands of paraphrase examples, paraphrase templates have strong representation capacity and can be used to generate many paraphrase examples. This paper describes a new template representation and generalization method. Combing a semantic dictionary, it uses multiple semantic codes to represent a paraphrase template. Using an existing search engine to extend the word clusters and generalize the examples. We also design three metrics to measure our generalized templates. The experimental results show that the representation method is reasonable and the generalized templates have a higher precision and coverage.

1 Introduction

Paraphrases are alternative ways to convey the same information (Barzilay and McKeown, 2001) and they have been applied in many fields of natural language processing. There are many previous work on paraphrase examples extraction or combining them with some applications such as information retrieval and question answering (Agichtein et al., 2001; Florence et al., 2003; Rinaldi et al., 2003; Tomuro, 2003; Lin and Pantel, 2001;), information extraction (Shinyama et al., 2002; Shinyama and Sekine, 2003), machine translation (Hiroshi et al., 2003; Zhang and Yamamoto, 2003), multi-document (Barzilay et al., 2003).

There is also some other research about paraphrase. (Wu and Zhou, 2003) just extract the synonymy collocation, such as <turn on, OBJ, light> and <switch on, OBJ, light> using both monolingual corpora and bilingual corpora to get an optimal result, but do not generalize them. (Glickman and Dagan, 2003) detects verb paraphrases instances within a single corpus without relying on any priori structure and information. Generation of paraphrase examples was also investigated (Barzilay and Lee, 2003; Quirk et al., 2004).

Rather than creating and storing thousands of paraphrases, paraphrase templates have strong representation capacity and can be used to generate many paraphrase examples. As (Hirst, 2003) said, for each aspect of paraphrase there are two main challenges: representation of knowledge and acquisition of knowledge. Corresponding to the problem of generalization of paraphrase templates, there are also two problems: the first is the representation of paraphrase templates and the second is acquisition of paraphrase templates.

There are several methods about paraphrase templates representation. The first method is using the Part-of-Speech (Barzilay and McKeown, 2001; Daumé and Marcu, 2003; Zhang and Yamamoto, 2003), the second uses name entity as the variable (Shinyama et al., 2002; Shinyama and Sekine, 2003), the third method is similar to the second method which is called the inference rules extraction (Lin and Pantel, 2001).

A paraphrases template is a pair of natural language phrases with variables standing in for certain grammatical constructs in (Daumé and

*: Supported by the Key Project of National Natural Science Foundation of China under Grant No. 60435020

Marcu, 2003). He used Part-of-Speech to represent templates. But for some cases, the POS will be very limited and for some other cases will be over generalized. For example:

在我看来——我觉得　　　　（1）

(*In my view/mind ----I feel*)

The above pair of phrases is a paraphrase, it can be generalized using POS information:

在 [pronoun] 看来

(*In [pronoun] view/mind*)

[pronoun] 觉得

(*[pronoun] feel*)

But for this template many noun words will be excluded. From this point of view, the template representation capacity is limited. But for other examples, the POS information will be over generally. For example:

苹果的价格是多少？

(*What's the price for the apples?*)

苹果多少钱一斤？

(*How much is the apples per Jin?*)

Here, we just generalize one variable "苹果". Then, the template becomes:

[noun] 的价格是多少？

(*What's the price for the [noun]?*)

[noun] 多少钱一斤？

(*How much is the [noun] per Jin?*)

If there is a sentence "笔记本的价格是多少(*What's the price for the notebook?*)", its' paraphrase will be "笔记本多少钱一斤(*How much is the notebook per Jin?*)" according to this template. Obviously, the result is unreasonable.

(Shinyama et al., 2002) tried to find paraphrases assuming that two sentences sharing many Named Entities and a similar structure are likely to be paraphrases of each other. But just name entities are limited, too. And (Lin and Pantel, 2001) present an unsupervised algorithm for discovering inference rules from text such as "X writes Y" and "X is the author of Y". This generalized method has good ability. But it also has some limited aspect. For example:

[Jack] writes [his homework].

According to the paraphrase template, the target sentence will be transformed into "*[Jack] is the author of [his homework]*". It's obviously that the generated sentence is not standard.

So how to represent paraphrase templates and generalize the paraphrase examples is a very interesting task. In this paper, we present a novel approach to represent paraphrase template with semantic code of words and using an existing search engine to get the paraphrase template.

The remainder of this paper is organized as follows. In the next section, we give the overview of our method. In section 3, we define the representation method in details. Section 4 presents the generalization method. Some experiments and discussions are shown in Section 5. Finally, we draw a conclusion of this method and give some suggestions about future work.

2 Overview of Generalization Method

The origin input of our system is a seed phrasal paraphrase example. And the output is the generalized paraphrase templates from the given examples. The overall architecture of our paraphrase generalization is represented on figure 1.

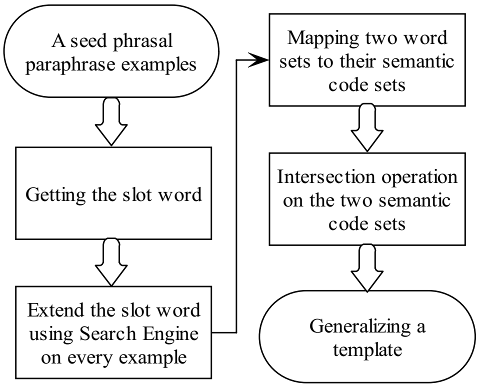
A seed phrasal paraphrase examples
Getting the slot word
Extend the slot word using Search Engine on every example
Mapping two word sets to their semantic code sets
Intersection operation on the two semantic code sets
Generalizing a template

Figure 1: Sketch Map of Paraphrase example Generalization

We also use the example (1) to illustrate the representation. Here a semantic dictionary called "TongYiCiCiLin" (Extension Version)[1] is used. The pair of phrases is a phrasal paraphrase. At first, after preprocessing which includes word segment, POS tagging and word sense disambiguation, we get the slot word in the paraphrase. In this example, the slot word is "我(*I*)". Then we search the web using the context of the slot word. Every phrase in the phrasal pair derives a set of sentences which include the original phrase context. A dependency parser on these sentences is used to extract the corresponding word with the slot word. Two word sets can be obtained through the two sentence sets. Then, we map word sets to their semantic code sets

[1] TongYiCiCiLin (Extended Version) can be downloaded from the website of HIT-IRLab (Http://ir.hit.edu.cn). In the past section, we abbreviate the TongYiCiCiLin (Extended Version) to Cilin (EV)

according to Cilin(EV). Then an intersection operation is conducted on the two sets. We use the intersection set to replace the slot word and generate the final paraphrase template.

In order to verify the validation of the generalized paraphrase template, we also design an automatic algorithm to confirm whether the template is reasonable using the existing search engine.

3 Representation of Template

In the section of introduction, some representation methods of paraphrase template have been introduced. And we proposed a new method using word semantic codes to represent the variable in a template. Before we introduce the representation method, Firstly, we give some general introduction about the semantic dictionary of Cilin(EV).

3.1 TongYiCiCiLin (Extended Version)

Cilin (EV) is derived from original TongYiCiCilin in which word senses are decomposed to 12 large categories, 94 middle categories, 1,428 small categories. Cilin (EV) removes some outdated words and updates many new words. More fine-grained categories are added on the base of original classification system to satisfy the more complex natural language applications. The encoding criterion is shown in the table 1:

Table 1 Encoding table of dictionary

Encoding bit	1	2	3	4	5	6	7	8
Example	D	a	1	5	B	0	2	=
Attribute	Big	Middle	Small		groups	Atom groups		
Layer	1	2	3		4	5		

The encoding bits are arranged from left to right. The first three layers are same with Cilin. The fourth layer is represented by capital letters and the fifth layer is two-bit decimal digit. The last bit is some more detailed information about the atom groups.

3.2 An Example of a Paraphrase Template

For simplicity, we just select one slot word in every paraphrase. And we stipulated that only content word can be slot word. We also use the above paraphrase example (1).

在我看来——我觉得
(*In my view/mind ----I feel*)

Here, we get the slot word "我(*I*)". Through the Word Sense Disambiguation processing, we get its semantic code "Aa02A01=" according to the fifth layer in Cilin(EV). If we just use the semantic code of the slot word, we can get a simple paraphrase template as follows:

在 [Aa02A01=] 看来
(*In [Aa02A01=] view/mind*)
[Aa02A01=] 觉得
(*[Aa02A01=] feel*)

But it is obviously that the template is very limited. Its' representation ability is also limited. So how to extend the ability of a paraphrase template is a challenging work.

3.3 Extending the Template Abstract Ability

According to the feature of Cilin(EV) architecture, we can use the higher layer's semantic code instead of the slot word to generalize the paraphrase template naturally. Of course it's a very simple method to extend the template ability, but it also brings more redundancy of a paraphrase template and it will be proven in the later section.

So we use multiple semantic codes of the different layer instead of only one semantic code of slot word in Cilin (EV). The later experimental results prove this representation has a good performance with a good precision and coverage.

4 Generalizing to Templates

As mentioned above, we can use multiple semantic codes to generalize paraphrase examples. So the problem of how to generalize paraphrase examples is transformed into the problem of how to get the multiple semantic codes set. We proposed a new method which uses the existing search engine to reach the target.

4.1 Getting the Candidate Sentences

After we removed the slot word in the paraphrase examples, two phrasal contexts of the original paraphrase phrases were obtained. Each phrase without slot word is used as a search query for an existing search engine and achieving many sentences which include the query word. For this example, the two queries are "在看来(in…view)" and "觉得(feel)". Each query gets one sentence set respectively. Part of the two result sentence sets are shown in figure 2 and figure 3:

在外资基金经理看来内地市场“风光无限”

在资产阶级官僚看来, 只要资本家赚钱高兴. . .

爱情在外人看来往往有些荒唐的

IT 认证在招聘方看来有多重要?. . .

在布什看来, 民主就是维护美国的利益……

Figure 2. Sentence Set 1

那个星座男孩觉得你很可爱

拥有了, 你才觉得美丽!

大家觉得哪个省最适合旅游 云间城. . .

消费者觉得买不如租…

杰西觉得自己走得累死了_TOM 教育……

Figure 3. Sentence Set 2

From the above two sentence sets, we can find that there is some noisy information in the sentences. In order to extend the correspondent words of the slot word, it is not enough that we just use the position information or POS tagging information of the slot word. Even if we extract these words, many of them can’t be found in the dictionary because they are not simple words. Benefiting from the idea of (Lin and Pantel, 2001), we use a dependency parser to determine the correspondent extended words.

4.2 Dependency Parser

In this paper, we use a dependency parser (Ma et al., 2004) to extract the candidate slot word. For example, the dependency parsing result of the phrase of “在我看来” is shown in figure 4.

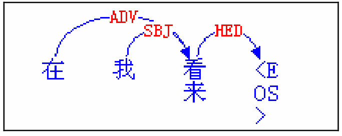

Figure 4. Dependency parsing result

The arcs in the figure represent dependency relationships. The direction of an arc is from the head to the modifier in the relationship. Labels associated with the arcs represent types of dependency relations. Table 2 lists a subset of the dependency relations in the HIT-IRLab dependency parser[2].

Table 2. A subset of the dependency relations

Relation	Description
ATT	定中关系(attribute)
HED	核心(head)
SBJ	主语(subject)
ADV	状中结构(adverbial)
VOB	动宾关系(verb-object)

4.3 Extracting the extended words

We just use a very simple method to get the extended words from the parsed sentences. At first, we record the relations of the original parsed phrasal examples. And then we use these relations to matched similar part in the candidate parsed sentence except slot word. And we omit these unseen relations and content words which don’t appear in the original parsed phrasal examples. Then we can get the extended words.

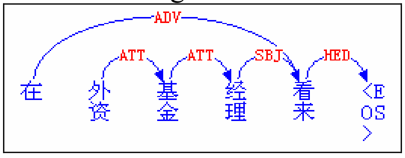

Figure 5. Dependency parsing result

Figure 5 shows the dependency parsing result of the phrase of “在外资基金经理看来”(*In foreign capital fund manager view*). We can easily find that the extended word of the slot word “我”(*I*) is “经理”(*manager*). Two extended word sets can be extracted from two sentence sets. Then we map each word to their semantic code to get two semantic code sets. Intersection operation is conducted on these two semantic code sets to obtain their intersection set. Finally, we use the semantic code set instead of the slot word to generate the paraphrase template.

4.4 Some tricks

Because the precision of the current dependency parser on Chinese is not very high, we just extract a part of the candidate sentences to parse. There are three patterns to segment the long candidate sentences according to position of slot word in paraphrase examples. They are called FRONT, MIDDLE and BACK. Here we use an example to illustrate it as shown in table 3:

Table 3 Examples of sentence segmentation

Pattern	Origin Phrase	Segment examples
FRONT	(SW)觉得	**那个星座男孩觉得**你很可爱…
MIDDLE	在(SW)看来	**在外资基金经理看来**内地市场“风光无限”…

The bold section in the sentence will be extracted to parse. Pattern type can be decided by

[2] More information about the dependency parser can be got from http://ir.hit.edu.cn/cuphelp.htm

the position relation between slot word and context words. And these patterns can reduce the relative error rate of the dependency parser. That is to say, if the original phrase is parsed wrongly, the extracted segments may be parsed wrongly with the similar error. But according to our method, this kind of parser error has little influence on the final extracting result.

5 Experiments and Discussions

5.1 Setting

We extract about 510 valid paraphrase examples from a Chinese paraphrase corpus (Li et al., 2004). For simplicity, we just select those phrasal paraphrase examples which own same word. And we stipulate only content word can be as slot word. We just use four seed phrasal paraphrases as the original paraphrases in this paper. And the generalized paraphrase templates represented by semantic codes of the fifth layer in Cinlin (EV) are also shown in the Table 4:

Table 4: Examples of the generalized template

	Origin Phrases	Generalized Paraphrase templates
1	我觉得	[Aa01A01=,Aa01A05=, Aa01C03=,Aa02A01=, …]觉得
	在我看来	在[Aa01A01=,Aa01A05=, Aa01C03=,Aa02A01=,...]看来
2	太热了	太 [Ac03A01=,Ah04A01=, Ah05A01=,Am03D01@,…]了
	热得很	[Ac03A01=,Ah04A01=, Ah05A01=,Am03D01@,…]得很
3	都烧光了	都[Fb01A01=,Gb07B01=, Hb06A01=,He15B01=,…]光了
	全烧完了	全[Fb01A01=,Gb07B01=, Hb06A01=,He15B01=,…]完了
4	苹果的价格是多少	[Aa03A01=,Ac03A01=, Ba05A10#,Bb02A01=,…] 的价格是多少
	苹果多少钱斤	[Aa03A01=,Ac03A01=,Ba05A10#, Bb02A01=,…]多少钱一斤

5.2 Evaluation on Templates

The goal of the evaluations is to confirm how reasonable this kind of representation method of paraphrase templates is and how well the template is. We evaluated the generalized paraphrase template in three ways. They are listed in the following three categories: 1) Reasonability; 2) Precision; 3) Coverage.

1) Reasonability

The reasonability of a paraphrase template aims to measure the reasonable extent of the presentation method with multiple semantic codes. For example, if we use POS to generalize a paraphrase template, its reasonability is very lower; that is to say, POS is not suitable to represent paraphrase template in some extent.

We use an existing search engine to calculate the reasonability of every paraphrase template. Firstly, we instantiate all paraphrase examples from a template. Then all these examples are as the queries of the search engine. If two phrases in one paraphrase can be matched completely from the search engine, it also means that one or more examples are found on the Web via search engine, we then consider this paraphrase is reasonable. Using this method we can get the approximate evaluation of all the examples. We define two metrics:

Strict_Reasonability = S / N
Loose_Reasonability = (L + S) / N

Where N is the total number of the instantiated examples; S is the number of the paraphrase examples which two phrases in it can be matched all; L is the number of paraphrase examples only one phrase in a paraphrase can be matched.

2) Precision

Every template is correspondent to the examples number with the semantic code of different layer in Cilin (EV) as shown in table 5.

Table 5 Templates and their correspond examples number

Number of Paraphrase templates	Instantiated examples number		
	Cilin3	Cilin4	Cilin5
1	2696	1815	478
2	13032	6354	3011
3	1057	587	177
4	3004	2229	429

From the above table, we can find that every template can instantiate many examples. If manually judging all of these examples will spend plenty of time. So we just sample part of all instantiate examples, 200 paraphrase examples for each template in this paper. For each phrase in a sample paraphrase example, it is as search query to get the first two matched sentences. Evaluators would be asked whether it is semantically okay to replace the query in the sentence by the correspondent phrase in a paraphrase. They were given only two options: Yes or No. If search query have no matched results, we consider that this phrase cannot be replace with its correspondent paraphrase. According to the above regulations, we know that every paraphrase examples correspondent to 4 sentences. If we sample n examples from a template, the precision of a paraphrase template can be calculated by:

Precision = R / (4 * n)

Where, R is the number of sentences which is considered to be correct by the evaluator.

3) Coverage

Evaluating directly the coverage of a paraphrase template is difficult because humans can't enumerate all the words to be suitable to the template. We use an approximate method to get the coverage of a template. At first we use another search engine to get candidate sentences with similar method for generalization of a paraphrase template. From these retrieved sentences we can get many different words with the known generalized words because more than 85% of search results from different search engine are different. Evaluators extract every sentence which can be replaced with the correspondent phrase in a paraphrase and the new sentences retain the origin meaning. We know each sentence is correspondent to a word. Then we define two metrics:

Surface_Coverage = M / NS
Semantic_Coverage =
Map(K) / (Map(NS-M) + Map(K))

Where, NS is the number of all manually tagged right words, M is the number of words which can be instantiated from a paraphrase template, K is the number of all the words that generalized the template at the front. Map(X) is the total word number of the word clusters which derived from X word in the semantic dictionary of Cilin(EV).

5.3 Result

In order to exhibit the merit of our method, we conduct four groups of experiment. They are POS-Tag, Cilin3, Cilin4 and Ciln5, respectively. Especially, we just randomly select 400 words to satisfy the POS information.

Table 6: Experiment Results

	Reasonability (%)		Coverage (%)		Preci-sion (%)
	St_R	Lo_R	Su_C	Se_C	
POS	10.50	17.00	90.00	----	11.75
Cilin3	45.57	84.50	27.55	38.71	45.75
Cilin4	46.89	84.54	23.87	44.48	64.13
Cilin5	46.24	83.12	20.39	39.47	69.88

Every value in table 6 is a average value of four values correspondent to four templates. From the table we can find that the reasonability of the Cilin-based representation template changes little, and that of POS-based representation is very lower. We find that the longer original phrases are, the lower the coverage of the generalized template is. Although the average coverage of generalized template is relatively low, we can draw a conclusion that using multiple semantic codes to generalize phrasal paraphrase examples is reasonable.

The column of the coverage shows that the coverage rates of Cilin-based templates are all not more than 50%. And the POS-based template has a very high coverage rate. And we know that the extended information is not enough only depending on one search engine. We will combine several different search engines with together to solve this problem in the future work.

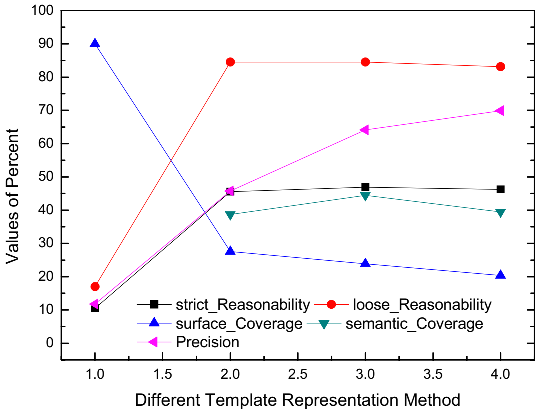

Figure 6. Experimental Results

The numbers from one to four on the X-axis are correspondent to POS, Cilin3, Cilin4 and Cilin5 in figure 6. We can see the features clearly of different representation methods of template from the figure 6. We can find that

Cilin5-based template has the highest precision, but its coverage is lower. And Cilin3-based template has opposite feature. This is because that one semantic code of Cilin3 includes more words than that of Cilin5. At the same time, more words bring more redundant information. And Cilin4-based template has a good tradeoff between coverage and precision. So we conclude that the semantic code of fourth layer in Cilin (EV) is more suitable to represent paraphrase template.

Some additional information can be extracted from the generalized template. Such as, the collocation information between the slot word and the context words can be extract. For example, in the fourth template, we can get the information about which words can be collocated with "斤(*Jin*)".

Although this kind of representation of paraphrase template has a good performance, it is weak for those words or structures that don't exist in dictionary. Also, this method is not suitable to the named entities representation.

6 Conclusion

In this paper, a novel method for automated generalization of paraphrase examples is proposed. This method is not dependent on the traditional limited texts instead it is based on the richness of the Web. It uses the multiple semantic codes to generalize a paraphrase example combing a semantic dictionary (Cilin (EV)). The experimental results proved that this representation method is reasonable and the generalized templates have a good precision and coverage.

But this is just the beginning of the paraphrase examples generalization. And we simplify the problem in some aspects, such as we limited the number of the slot word in a paraphrase example, and we stipulate only the same word can be slot word. Also, we find that our templates are weak for those words or structures that don't exist in dictionary. Some methods in information extraction about named entities generalization can be used for reference in the future. Moreover, how to combine the semantic code with other representation forms together is also an interesting work.

References

[1] Chris Quirk, Chris Brockett, and William Dolan. Monolingual Machine Translation for Paraphrase Generation. editors, Dekang Lin and Dekai Wu, In Proceedings of EMNLP 2004, Barcelona, pages 142-149

[2] Dekang Lin and Patrick Pantel. 2001. Discovery of Inference Rules for Question Answering. Natural Language Engineering 7(4):343-360

[3] Dekang Lin and Patrick Pantel. Discovery of inference rules for question answering. Natural Language Engineering, 1, 2001.

[4] E. Agichtein, S. Lawrence, and L. Gravano. Learning search engine specific query transformations for question answering. In Proceedings of the 10th International World-Wide Web Conference (WWW10), 2001

[5] Fabio Rinaldi, James Dowdall, Kaarel Kaljurand, Michael Hess, Diego Molla. 2003. Exploiting Paraphrases in a Question Answering System. The Second International Workshop on Paraphrasing: Paraphrase Acquisition and Applications

[6] Florence Duclaye France. Learning paraphrases to improve a question-answering system. In EACL Natural Language Processing for Question Answering, 2003

[7] Graeme Hirst. Paraphrasing Paraphrased. In Proceedings of the Second International Workshop on Paraphrasing, 2003

[8] Hal Daumé III and Daniel Marcu. Acquiring paraphrase templates from document/abstract pairs. In NL Seminar in ISI, 2003

[9] Hua Wu, Ming Zhou. Optimizing Synonym Extraction Using Monolingual and Bilingual Resources. In Proceedings of the Second International Workshop on Paraphrasing, 2003

[10] Hua Wu, Ming Zhou. Synonymous Collocation Extraction Using Translation Information. In Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics, 2003

[11] Jinshan Ma, Yu Zhang, Ting Liu, and Sheng Li. A Statistical Dependency Parser of Chinese under Small Training Data. Workshop: Beyond shallow analyses - Formalisms and statistical modeling for deep analyses, IJCNLP-04, 4 2004.

[12] Noriko Tomuro. 2003. Interrogative Reformulation Patterns and Acquisition of Question Paraphrases. The Second International Workshop on Paraphrasing: Paraphrase Acquisition and Applications

[13] Oren Glickman and Ido Dagan. Identifying lexical paraphrases from a single corpus: A case study for verbs. In Proceedings of Recent Advantages in Natural Language Processing, September 2003

[14] Regina Barzilay and Kathleen McKeown. Extracting paraphrases from a parallel corpus. In Proceedings of the ACL/EACL, Toulouse, 2001

[15] Regina Barzilay and Lillian Lee. Learning to Paraphrase: An Unsupervised Approach Using Multiple-Sequence Alignment. In Proceedings of HLT-NAACL 2003, pages 16-23

[16] Regina Barzilay, Noemie Elhadad, Kathleen R. McKeown. 2003. Inferring Strategies for Sentence Ordering in Multidocument News Summarization. The Second International Workshop on Paraphrasing: Paraphrase Acquisition and Applications

[17] Weigang Li, Ting Liu, Sheng Li. Combining Sentence Length with Location Information to Align Monolingual Parallel Texts. AIRS, 2004, pages 71-77

[18] Yusuke Shinyama and Satoshi Sekine. Paraphrase acquisition for information extraction. editors, Kentaro Inui and Ulf Hermjakob, In Proceedings of the Second International Workshop on Paraphrasing, 2003, pages 65-71

[19] Yusuke Shinyama, Satoshi Sekine, Kiyoshi Sudo, and Ralph Grishman. Automatic paraphrase acquisition from news articles, In Proceedings of Human Language Technology Conference (HLT2002), San Diego, USA, Mar. 15, 2002

[20] Zhang Yujie, Kazuhide Yamamoto. Automatic Paraphrasing of Chinese Utterances. Journal of Chinese Information Processing. Vol. 117 No. 16: 31-38(Chinese)