Automatic Acquisition of Context-Specific Lexical Paraphrases

Shiqi Zhao, Ting Liu, Xincheng Yuan, Sheng Li, and Yu Zhang
Information Retrieval Laboratory, Harbin Institute of Technology
No. 27 Jiaohua Street, Nangang, Harbin, China, 150006
{zhaosq, tliu, xcyuan, lisheng, zhangyu}@ir-lab.org

Abstract

Lexical paraphrasing aims at acquiring word-level paraphrases. It is critical for many Natural Language Processing (NLP) applications, such as Question Answering (QA), Information Extraction (IE), and Machine Translation (MT). Since the meaning and usage of a word can vary in distinct contexts, different paraphrases should be acquired according to the contexts. However, most of the existing researches focus on constructing paraphrase corpora, in which little contextual constraints for paraphrase application are imposed. This paper presents a method that automatically acquires context-specific lexical paraphrases. In this method, the obtained paraphrases of a word depend on the specific sentence the word occurs in. Two stages are included, i.e. candidate paraphrase extraction and paraphrase validation, both of which are mainly based on web mining. Evaluations are conducted on a news title corpus and the presented method is compared with a paraphrasing method that exploits a Chinese thesaurus of synonyms -- Tongyi Cilin (Extended) (CilinE for short). Results show that the f-measure of our method (0.4852) is significantly higher than that using CilinE (0.1127). In addition, over 85% of the correct paraphrases derived by our method cannot be found in CilinE, which suggests that our method is effective in acquiring out-of-thesaurus paraphrases.

1 Introduction

Paraphrases are alternative ways that convey the same information. Paraphrasing tasks can be classified as lexical, phrase-level, sentence-level, and discourse-level. Lexical paraphrasing aims to acquire paraphrases of words.

Lexical paraphrasing is important in many NLP applications since it is an effective solution to the problem of "word mismatch". For example, in QA, expanding a question by paraphrasing its content words can make it easier to find the answer [Hermjakob et al., 2002]. In IE, words and phrases in the IE patterns should be paraphrased in order to extract the target information expressed in various ways [Shinyama and Sekine, 2003]. In MT, paraphrases of unseen source words can be incorporated into the statistical MT process. Specifically, paraphrases of unseen words can be translated rather than leave the words untranslated [Callison-Burch et al., 2006].

Two broad approaches to lexical paraphrasing have dominated the literature. One approach acquires paraphrases from dictionaries, such as WordNet. Some researchers extract WordNet synonyms as paraphrases [Langkilde and Knight, 1998], while some others use looser definitions [Barzilay and Elhadad, 1997]. In general, the correspondence between paraphrasing and types of lexical relations defined in WordNet is not clear [Barzilay and McKeown, 2001]. In Chinese, CilinE has been exploited for paraphrasing in stead of WordNet [Li et al., 2005].

The other approach collects lexical paraphrases from monolingual or bilingual corpora. Lin [1998] identified words that are similar in meaning by measuring the similarity of the contextual words. Barzilay and McKeown [2001] extracted paraphrases from a corpus of multiple English translations of the same source text. Bannard and Callison-Burch [2005] derived paraphrases using bilingual parallel corpora. Wu and Zhou [2003] extracted lexical paraphrases with multiple resources, including a monolingual dictionary, a bilingual corpus, and a monolingual corpus.

These methods facilitate the acquisition of paraphrases. However, none of them specify the contexts in which the paraphrases can be adapted. This problem is crucial as almost all paraphrases can only be adapted in certain contexts.

Recently, topic adaptation for paraphrasing has been researched. For example, Kaji and Kurohashi [2005] selected lexical paraphrases according to different topics. However, the topics are limited and predefined. Thus, their method cannot paraphrase a word according to any given context.

This paper addresses the problem of context-specific paraphrasing (CSP), which aims at acquiring specific paraphrases according to a given context. In lexical CSP, words are to be paraphrased. Accordingly, a specific context means a sentence in which a word occurs. Specifically, if a word occurs in different sentences, then different paraphrases should be extracted within each sentence.

The remainder of the paper is organized as follows: Section 2 introduces the context-specific paraphrasing method in detail. Section 3 describes the experiments and results. Conclusion and future work are presented in Section 4.

2 Method

Two main stages are included in the method: candidate paraphrase extraction and paraphrase validation. For a given sentence S, in the first stage, a set of similar sentences are retrieved from the web using a search engine (Baidu[1] in the experiments). From the similar sentences, candidate paraphrases for words in S are extracted by measuring syntactic similarities. The candidates are filtered using part-of-speech (POS) information. In the second stage, candidate paraphrases are validated using a combined similarity measurement, which integrates co-occurrence similarity, syntactic similarity, and semantic similarity. Both the web and CilinE are exploited in the validation stage.

2.1 Candidate Paraphrase Extraction

2.1.1 Motivation

Candidate paraphrase extraction is based on a web mining method. A similar method has been exploited for paraphrasing answer patterns in QA [Ravichandran and Hovy, 2002]. Using the web as a paraphrasing resource has three advantages compared with conventional resources (monolingual parallel corpora, monolingual unparallel corpora, and bilingual parallel corpora). First, the web is not domain limited. Almost all kinds of topics and contexts can be covered. Second, the scale of the web is extremely large, which makes it feasible to find any specific context on it. In addition, the web is dynamic, which means that new words and concepts can be retrieved from the web.

The method for candidate paraphrase extraction is based on two principles:

Principle 1: Authors on the web create information independently. Thus their "vocabularies" vary greatly [Cui et al., 2002].

This principle means that different people tend to use different words to express the same meaning. In other words, if a concept is widely discussed on the web, then various expressions (lexical paraphrases) will be found in the corresponding web documents. However, a principle for detecting these paraphrases and extracting them from the web is needed. Therefore, the second principle is introduced.

Principle 2: Lexical paraphrases play similar syntactic functions in sentences.

The second principle indicates that paraphrases of a given word w can be derived by extracting words whose syntactic functions are similar with w.

In fact, the principles above have been used in recognizing paraphrases in previous work. Shinyama et al. [2002] acquired paraphrases using different reports on the same event of the same day (based on Principle 1). Lin [1998] clustered similar words by measuring syntactic similarities (based on Principle 2). The rationality of the principles has been verified in their work.

[1] http://www.baidu.com

2.1.2 Procedure of Candidate Paraphrase Extraction

Two steps are included in candidate paraphrase extraction:

Step1: Query S on the web and retrieve similar sentences. Obviously, only similar sentences of the given sentence S need to be considered when extracting candidate paraphrases since if two words are context-specific paraphrases they should occur in identical or at least similar sentences. In this step, S is searched on the web using Baidu. From the retrieved snippets, sentences whose similarities with S exceed a predefined threshold T_{CE} (T_{CE}=0.3 in our case) are retained for further candidate extraction (these sentences are called candidate sentences hereafter). Word overlapping rate (WOR) is used here for computing the similarity between S and any candidate sentence S_C:

$$WOR(S, S_C) = \frac{|S \cap S_C|}{\max(|S|, |S_C|)} \qquad (1)$$

where "$S \cap S_C$" denotes the common words in both S and S_C. "|.|" denotes the number of words.

Step2: Extract candidates according to syntactic similarity. As stated in Principle 2, lexical paraphrases usually play similar syntactic functions in sentences. This is an important clue for candidate extraction. In this step, sentence S and all the candidate sentences obtained in the Step1 are first parsed by a Chinese dependency parser, in which 24 dependency relations (e.g. *SBV, VOB, ATT*…) are defined. Figure 1 depicts the dependency tree of an input sentence.

Sentence:
他(He) 喜欢(likes) 中国(Chinese) 文化(culture) (.)

Dependency Tree:

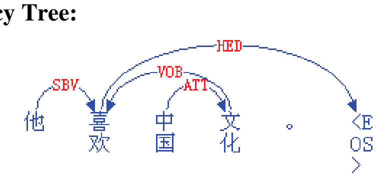

Figure 1: An example of dependency trees

In a dependency tree of a sentence, two words and their dependency relation are represented as a triple. For example, "<他, SBV, 喜欢>" is a triple. The criterion shown in Figure 2 is used for extracting candidate paraphrases:

Given:
- S: original sentence;
- S_C: candidate sentence;
- $DT(S)$: dependency tree of S;
- $DT(S_C)$: dependency tree of S_C;
- $<w_1, rel, w_2>$: a triple in $DT(S)$;
- $<w_1', rel', w_2'>$: a triple in $DT(S_C)$.

Criterion:
- If $rel=rel'$ and $w_2=w_2'$ and $w_1 \neq w_1'$
  then w_1' is extracted as a candidate paraphrase of w_1.
- If $w_1=w_1'$ and $rel=rel'$ and $w_2 \neq w_2'$
  then w_2' is extracted as a candidate paraphrase of w_2.

Figure 2: Criterion for candidate paraphrase extraction

It is obvious that, a word and its paraphrases should have identical POS. Therefore, if the word *w* and its candidate paraphrase *w'* have different POSes, *w'* is filtered.

2.2 Paraphrase Validation

Since the constraints in the candidate extraction stage are quite loose for the sake of recall, there exists much noise in the candidates. The experiments show that over 70% of the candidates are incorrect. Therefore, a paraphrase validation stage is necessary.

Paraphrase validation is one of the key issues in the research of paraphrasing. Hence many methods have been presented. For example, Barzilay and McKeown [2001] recognized phrasal paraphrases using rules automatically extracted from the contexts. However, the method must be based on monolingual parallel corpora. Bannard and Callison-Burch [2005] validated paraphrases by computing the probability that two phrases can be translated to (or from) the same foreign language phrases. Nevertheless, a large bilingual corpus is needed. Lin [1998] identified lexical paraphrases based on Distributional Hypothesis, which computes the similarity of two words' contexts to judge whether they are paraphrases.

The main disadvantage of the above methods is that none of them can determine whether two words are paraphrases within a certain sentence. In other words, they are not context-specific paraphrasing methods. In our work, a novel paraphrase validation method is proposed, in which both web information and a thesaurus are exploited.

2.2.1 Paraphrase Validation Using Web Mining

Generally, when a query is searched using a search engine, the retrieved snippets are related to the query and can be viewed as "description" of the query. Therefore, it can be assumed that if two queries can retrieve similar snippets, then they are similar.

This assumption is used in paraphrase validation. In detail, let *w* be a paraphrased word in sentence *S*, *w'* be a candidate paraphrase of *w* within *S*, and *S'* be a sentence constructed by replacing *w* with *w'* in *S*. If similar snippets are retrieved by searching *S* and *S'*, then we can say that replacing *w* with *w'* does not notably change the meaning of *S*. In other words, *w* and *w'* can be viewed as paraphrases in *S*.

Suppose *Sni*(*S*) and *Sni*(*S'*) are the snippets corresponding to *S* and *S'* respectively. Here, sentences that do not contain *w* (*w'*) are removed from *Sni*(*S*) (*Sni*(*S'*)) to filter noise.

Two similarity measurements are defined to measure the snippet-based similarity between *w* and *w'*, i.e. the Vector Space Model (VSM) similarity (Sim_{VSM}) and the syntactic similarity (Sim_{SYN}).

Sim_{VSM}: In VSM, snippets *Sni*(*S*) and *Sni*(*S'*) are represented as vectors *V*(*S*) and *V*(*S'*). The weight of each word is calculated using a *tf itf* heuristic (Equation 2).

$$tf \cdot itf(t, Sni(S)) = tf(t, Sni(S)) \times \log \frac{\max_{t'}(tf(t', C_{CD}))}{tf(t, C_{CD})} \quad (2)$$

where *tf*(*t, Sni*(*S*)) denotes the term frequency of term *t* in snippets *Sni*(*S*). *tf*(*t*, C_{CD}) is *t*'s term frequency counted on a China Daily Corpus (C_{CD}). max(*tf*(*t'*, C_{CD})) is the largest term frequency obtained on the corpus. Note that the *itf* part in the equation is similar to the *idf* part in *tf idf* heuristic which is widely used in NLP and Information Retrieval (IR) applications. The underlying hypothesis is that the words occur frequently in the whole corpus should be "punished" when weighing the words.

The VSM similarity between *w* and *w'* is calculated as the cosine similarity between *V*(*S*) and *V*(*S'*):

$$Sim_{VSM}(w, w') = \cos(V(S), V(S')) = \frac{V(S) \cdot V(S')}{\|V(S)\| \times \|V(S')\|} \quad (3)$$

where " " denotes the inner product of two vectors. "$\|\cdot\|$" denotes the length of a vector.

Sim_{SYN}: In order to compute the syntactic similarity, *Sni*(*S*) and *Sni*(*S'*) are first parsed using the same dependency parser described above. The syntactic similarity is calculated with the same method as described in [Lin, 1998b], which is rewritten in Equation (4). The similarity is calculated through the surrounding contextual words which have dependency relationships with the investigated words according to the parsing results.

$$Sim_{SYN}(w, w') = \frac{\sum_{(rel,t) \in T(w) \cap T(w')} (I(w, rel, t) + I(w', rel, t))}{\sum_{(rel,t) \in T(w)} I(w, rel, t) + \sum_{(rel,t) \in T(w')} I(w', rel, t)} \quad (4)$$

where *T*(*w*) denotes the set of words which have the dependency relation *rel* with *w*. *I*(*w,rel,t*) is the point-wise mutual information, as defined in Equation (5):

$$I(w, rel, t) = \log \frac{p(w, rel, t)}{p(w \mid rel) p(t \mid rel) p(rel)} \quad (5)$$

Sim_{VSM} and Sim_{SYN} measure the snippet-based similarity of two words from different aspects. Sim_{VSM} can also be called co-occurrence similarity since it measures whether *w* and *w'* co-occur with similar words in the snippets. All words in the snippets are assumed to be independent with each other and no syntactic relations are considered. In contrast, Sim_{SYN} measures whether *w* and *w'* play similar syntactic functions in the snippets. Only the words that have dependency relations with *w* (or *w'*) in the snippets are counted and the dependency relation types are taken into account.

2.2.2 Paraphrase Validation Using CilinE

Beside Sim_{VSM} and Sim_{SYN}, the semantic similarity (Sim_{SEM}) is also investigated for paraphrase validation.

Sim_{SEM}: CilinE is organized as a hierarchy of five levels, in which the first level is the highest and the fifth is the lowest (Figure 3 (a)). Given two words, the lower their common ancestor node is, the more similar their word senses are. Each word in CilinE has a sense code, determining its position in each level of the hierarchy (Figure 3 (b)).

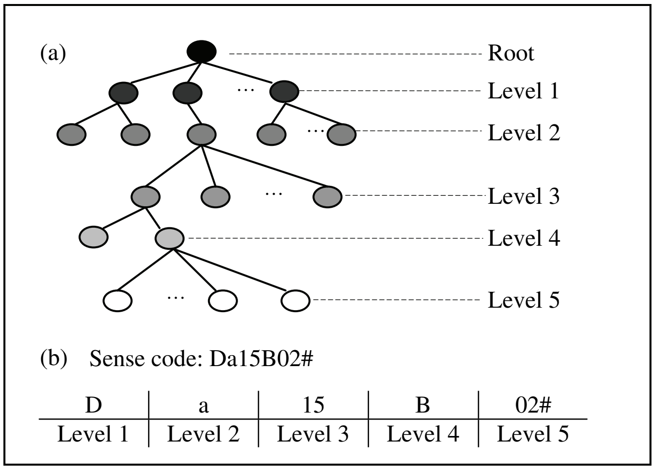

Figure 3: Hierarchy of CilinE and an example of word sense code

For word w and its candidate paraphrase w', ws_i and ws_j' denote the *i-th* sense of w and the *j-th* sense of w' (Note that a word may have more than one word sense). Sim_{SEM} of w and w' is defined in Equation (6):

$$Sim_{SEM}(w,w') = \max_{i,j}\left(\frac{L(F(ws_i, ws_j'))}{L_{Total}}\right) \quad (6)$$

where $L(F(ws_i, ws_j'))$ is the lowest ancestor node that two sense codes have in the hierarchy. L_{Total} is the number of total levels (L_{Total} =5). For w and w', the maximal similarity of their senses is defined as the semantic similarity.

Obviously, Sim_{SEM} only measures the semantic distance between two words, in which no context information is considered. However, it is useful in paraphrase validation as a supplement to the snippet-based similarity. In our future work, a refined semantic similarity measurement [Lin, 1998a] will be investigated.

2.2.3 Linear Combination of Similarities

The three similarities defined above measure the similarity of two words from different sides. In order to integrate the similarities, we get them linearly combined:

$$\begin{aligned} & Sim_{COM}(w,w') \\ & = \alpha * Sim_{VSM}(w,w') + \beta * Sim_{SYN}(w,w') + \gamma * Sim_{SEM}(w,w') \end{aligned} \quad (7)$$

where α , β , and γ are positive and $\alpha+\beta+\gamma=1$. The combined similarity Sim_{COM} is used in paraphrase validation. Detailedly, for word w and its candidate paraphrase w', if $Sim_{COM}(w,w')>T$ (T is a predefined threshold), then the candidate w' will be validated as a true paraphrase. Otherwise, w' will be filtered. α , β , γ and T are estimated using a development set (Section 3.2.1).

3 Evaluation

3.1 Data and Metrics

In order to evaluate the CSP method, a sentence corpus is needed. In our experiments, a corpus of news titles is used as test data. The reasons are two folded. On the one hand, news titles are usually well-formed sentences. On the other hand, in many applications, such as QA, IE, and multi-document summarization, the words and sentences to be paraphrased are usually from news articles. The news titles are from the "important news" section of "Sina news[2]". All titles from March 15, 2006 to April 5, 2006 are downloaded. After removing some duplicated ones, 257 titles are left for constructing the test data. Likewise, another 210 titles from April 6, 2006 to April 30, 2006 are downloaded to form the development data.

The metrics in the experiments are macro-averaged precision, recall, and f-measure. Let $M_1, \ldots, M_T$ be T paraphrasing methods to be compared (in our experiments, the compared methods are M_{CSP}, M_{CilinE}, $\mathrm{M}_{CSP\text{-}CAN}$, $\mathrm{M}_{VSM+SYN}$, $\mathrm{M}_{VSM+SEM}$, and $\mathrm{M}_{SYN+SEM}$, which will be described in the next section), N the number of sentences in test data, nt_i the number of words in the *i-th* sentence that can be paraphrased by method M_t ($1 \leqslant t \leqslant T$), nt_{ij} the number of acquired paraphrases for the *j-th* paraphrased word in the *i-th* sentence using method M_t, mt_{ij} the number of correct paraphrases (judged manually) in the nt_{ij} paraphrases. The precision of method M_t is defined as:

$$precision(M_t) = \sum_{i=1}^{N}\sum_{j=1}^{nt_i}\frac{mt_{ij}}{nt_{ij}} \Big/ \sum_{i=1}^{N} nt_i \quad (8)$$

Compared with precision, recall is more difficult to calculate since it is impossible to enumerate all paraphrases that a word has within a context. Therefore, an approximate approach is used to calculate recall of each method. Specifically, for the *j-th* paraphrased word in the *i-th* sentence, all its correct paraphrases acquired by the T methods are put together (with duplication removed). Let n_i be the number of words in the *i-th* sentence that can be paraphrased by at least one method, m_{ij} the total number of correct paraphrases for the *j-th* word. We assume that m_{ij} is the number of paraphrases that the word can really have within this specific sentence. Then the recall of method M_t is defined as:

$$recall(M_t) = \sum_{i=1}^{N}\sum_{j=1}^{nt_i}\frac{mt_{ij}}{m_{ij}} \Big/ \sum_{i=1}^{N} n_i \quad (9)$$

Note that, the recall of a method will be over-estimated using the definition of Equation (9), since some correct paraphrases may be absent. However, it is reasonable to get a set of methods compared in this way.

The f-measure of method M_t is defined as:

$$f-measure(M_t) = \frac{2 \times precision(M_t) \times recall(M_t)}{precision(M_t) + recall(M_t)} \quad (10)$$

3.2 Results and Analysis

3.2.1 Comparison between M_{CSP} and M_{CilinE}

In this section, we compare the CSP method (M_{CSP}) with the method extracting CilinE synonyms as paraphrases (M_{CilinE}).

[2] http://news.sina.com.cn

In M_{CilinE}, word sense disambiguation (WSD) is first conducted. A supervised method based on Bayesian Model is used in the WSD module, which can achieve a precision of 89.67% in our evaluation. Then, all synonyms of a word under the chosen sense are extracted as its paraphrases. In M_{CSP}, the development data is used to determine the parameters. The parameters for getting highest f-measure scores on the development data are selected. As a result, the coefficients α, β and γ in Equation (7) are 0.74, 0.10, and 0.16 respectively. The similarity threshold T is 0.12. The comparison results are shown in Table 1:

Method	Precision	Recall	F-measure
M_{CSP}	0.6380	0.3914	0.4852
M_{CilinE}	0.0630	0.5346	0.1127

Table 1: Comparison between M_{CSP} and M_{CilinE}

It can be seen from Table 1 that the precision of M_{CilinE} is quite low, which shows that most synonyms defined in CilinE are not paraphrases in specific contexts. On the other hand, the recall of M_{CSP} is lower than M_{CilinE}, this is mainly because CilinE can provide some correct paraphrases that are not used in web documents. However, it is found that over 85% of the correct paraphrases derived in M_{CSP} are not synonyms in CilinE. This suggests that M_{CSP} is effective in extracting "new" paraphrases. An example of the derived paraphrases of the two methods is illustrated in Figure 4 (words in bold are manually judged correct paraphrases):

Sentence	巴林 客轮 沉没 48 人 遇难 (Tourist boat sinks off Bahrain, 48 persons died)
Results of M_{CSP}	沉没/sink -- **失事/wreck**; 客轮/tourist boat -- 沉船/sunken ship, **渡轮/ferry boat**, **客船/passenger ship**, **游船/pleasure boat**; 遇难/die -- **死亡/die**;
Results of M_{CilinE}	沉没/sink -- 沉淀/deposit, 沉井/open caisson, 沉陷/subside, 没顶/submerge, 下陷/sag, 陷/sag, 陷落/fall, 陷没/subside; 人/person -- 人士/personage, 人氏/denizen, 人物/personality, 人选/candidate, 士/scholar; 遇难/die -- 被害/be murdered, 落难/be in distress, **蒙难/meet with disaster**, 受害/suffer injury, 死难/die, 遇害/be murdered, 遇险/be in danger, 遭难/suffer, **罹难/die in a disaster**;

Figure 4: Example of the derived paraphrases of M_{CSP} and M_{CilinE}

3.2.2 Evaluation of Validation Methods

In this section, we first evaluate the paraphrase validation method. Therefore, we compare M_{CSP} with the method that only extracts candidate paraphrases as described in Section 2.1 without validation ($M_{CSP\text{-}CAN}$). The comparison results are shown in Table 2.

Method	Precision	Recall	F-measure
M_{CSP}	0.6380	0.3914	0.4852
$M_{CSP\text{-}CAN}$	0.2822	0.5026	0.3615

Table 2: Comparison between M_{CSP} and $M_{CSP\text{-}CAN}$

It can be found that M_{CSP} outperforms $M_{CSP\text{-}CAN}$ greatly in precision, which indicates that the validation method is effective in filtering incorrect candidates. At the same time, recall decreases after validation, which suggests that some correct paraphrases are filtered by mistake. Nevertheless, the increase in F-measure demonstrates the effectiveness of paraphrase validation.

As mentioned before, three kinds of similarities are combined during paraphrase validation in M_{CSP}. The contribution of each one should be evaluated. Therefore we compare M_{CSP} with $M_{VSM+SYN}$ (combining Sim_{VSM} and Sim_{SYN}), $M_{VSM+SEM}$ (combining Sim_{VSM} and Sim_{SEM}), and $M_{SYN+SEM}$ (combining Sim_{SYN} and Sim_{SEM}).

For the three methods to be compared, the coefficients α, β and γ in Equation (7) and the similarity threshold T are also estimated using the development data (Table 3). The comparison results are shown in Table 4.

Method	α	β	γ	T
$M_{VSM+SYN}$	0.12	0.88	-	0.10
$M_{VSM+SEM}$	0.44	-	0.56	0.26
$M_{SYN+SEM}$	-	0.86	0.14	0.06

Table 3: Parameters for the methods to be compared

Method	Precision	Recall	F-measure
M_{CSP}	0.6380	0.3914	0.4852
$M_{VSM+SYN}$	0.4587	0.4059	0.4307
$M_{VSM+SEM}$	0.5036	0.3800	0.4332
$M_{SYN+SEM}$	0.5194	0.4028	0.4537

Table 4: Evaluation of each similarity measurement

We can find from Table 4 that eliminating each similarity in the paraphrase validation can produce a notable degradation in precision (drop 28.10%, 21.07%, and 18.59%, respectively) and f-measure (drop 11.23%, 10.72%, and 6.49%, respectively), while the impact on recall is slight.

The comparison results suggest that each of the similarity measurements is useful in filtering incorrect candidates and the combination of all the three similarities can achieve the best performance.

We believe that three major factors should be taken into consideration in paraphrase validation, that is, whether two words co-occur with similar contextual words, whether they play similar syntactic functions in sentences, and whether their semantic distance is small. The combined similarity in Equation (7) integrates all these factors. Hence the f-measure is significantly enhanced.

3.2.3 Error Analysis

False positives[3] are analyzed after experiments. It is found that nearly 84% of the false positives are due to the reason that non-paraphrases occur in similar contexts. For example, in sentence "丁俊晖 无缘 斯诺克 决赛 (Ding Junhui failed to enter the final of the snooker match)", "决赛 (final)" is paraphrased into "正选赛(main draw match)",

[3] False positives are word pairs recognized as paraphrases, but actually are not.

since these two words have occurred in very similar sentences from the reports about Ding's two different matches. In order to solve this problem, we believe that, some new features should be used when representing contexts, such as Named Entities (NE).

Another 10% mistakes are owing to CilinE, because it assigns high semantic similarities to some non-paraphrase word pairs (such as "清华(Tsinghua university)" and "人大(Renmin University)") during paraphrase validation.

The other 6% false positives are due to mistakes of the preprocessing modules, including word segmentation, POS tagging, and syntactic parsing.

4 Conclusion

This paper proposes a web mining method to automatically acquire context-specific lexical paraphrases. There are three main contributions. First, this work focuses on the problem of context-specific paraphrasing, which is very important but has seldom been addressed before. Second, a novel two-stage method is presented, which uses the web as resource instead of monolingual or bilingual corpora used in conventional work. Third, three similarity measurements are investigated and combined for paraphrase validation.

Experiments are carried out on a news title corpus and the results show that our method can achieve a precision of 0.6380, which is dramatically higher than the method extracting paraphrases from CilinE (0.0630). In addition, over 85% of the paraphrases derived by M_{CSP} cannot be extracted from CilinE, which suggests that the web is an eligible resource for acquiring context-specific paraphrases and the presented method is effective. Results also show that all the similarity measurements introduced in paraphrase validation make notable contribution to filtering incorrect candidates.

The main disadvantage of this method is that its time complexity is high, as snippets must be downloaded in the validation of each candidate paraphrase. This may make the method impractical in some applications, such as IR and QA. In the future work, we will construct a context-specific paraphrase corpus using the CSP method, which contains not only paraphrase pairs, but also contexts in which the paraphrases can be adapted. Context-specific paraphrases can be extracted directly from the corpus in practice.

Besides, the CSP method will be extended to the acquisition of phrase-level paraphrases. In addition, this method will be tested on normal sentences other than news titles.

Acknowledgments

This research was supported by National Natural Science Foundation of China (60575042, 60503072, 60675034). We thank Wanxiang Che for his valuable advice and comments.

Reference

[Bannard and Callison-Burch, 2005] Colin Bannard and Chris Callison-Burch. Paraphrasing with Bilingual Parallel Corpora. In *Proceedings of ACL*, 2005.

[Barzilay and Elhadad, 1997] Regina Barzilay and Michael Elhadad. Using Lexical Chains for Text Summarization. In *Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization*, 1997.

[Barzilay and McKeown, 2001] Regina Barzilay and Kathleen R. McKeown. Extracting Paraphrases from a Parallel Corpus. In *Proceedings of the ACL/EACL*, 2001.

[Callison-Burch et al., 2006] Chris Callison-Burch, Philipp Koehn, and Miles Osborne. Improved Statistical Machine Translation Using Paraphrases. In *Proceedings of NAACL*, 2006.

[Cui et al., 2002] Hang Cui, Ji-Rong Wen, Jian-Yun Nie, and Wei-Ying Ma. Probabilistic Query Expansion Using Query Logs. In *Proceedings of WWW*, 2002.

[Hermjakob et al., 2002] Ulf Hermjakob, Abdessamad Echihabi, and Daniel Marcu. Natural Language Based Reformulation Resource and Web Exploitation for Question Answering. In *Proceedings of the TREC-2002 Conference*, 2002.

[Kaji and Kurohashi, 2005] Nobuhiro Kaji and Sadao Kurohashi. Lexical Choice via Topic Adaptation for Paraphrasing Written Language to Spoken Language. In *Proceedings of IJCNLP*, 2005.

[Langkilde and Knight, 1998] Irene Langkilde and Kevin Knight. Generation that Exploits Corpus-based Statistical Knowledge. In *Proceedings of the COLING/ACL*, 1998.

[Li et al., 2005] Weigang Li, Ting Liu, Yu Zhang, Sheng Li, and Wei He. Automated Generalization of Phrasal Paraphrases from the Web. In *Proceedings of IWP*, 2005.

[Lin, 1998a] Dekang Lin. An Information-Theoretic Definition of Similarity. In *Proceedings of ICML*, 1998.

[Lin, 1998b] Dekang Lin. Optimizing Synonym Extraction Using Monolingual and Bilingual Resources. In *Proceedings of COLING/ACL*, 1998.

[Ravichandran and Hovy, 2002] Deepak Ravichandran and Eduard Hovy. Learning Surface Text Patterns for a Question Answering System. In *Proceedings of ACL*, 2002.

[Shinyama and Sekine, 2003] Yusuke Shinyama and Satoshi Sekine. Paraphrase Acquisition for Information Extraction. In *Proceedings of IWP*, 2003.

[Shinyama et al., 2002] Yusuke Shinyama, Satoshi Sekine, Kiyoshi Sudo, and Ralph Grishman. Automatic Paraphrase Acquisition from News Articles. In *Proceedings of HLT*, 2002.

[Wu and Zhou, 2003] Hua Wu and Ming Zhou. Optimizing Synonym Extraction Using Monolingual and Bilingual Resources. In *Proceedings of IWP*, 2003.