Sentence Level Paraphrase Recognition Based on Different Characteristics Combination

Maoyuan Zhang, Hong Zhang, Deyu Wu, Xiaohang Pan

Central China Normal University, Computer Science Department,
Luoyu Road. 152, Wuhan, China
zhangmyccnu@126.com,qjzhanghong@gmail.com
wdy158@sohu.com,panxiaohang_love@126.com

Abstract. This paper has proposed a novel method based on different characteristics combination to do paraphrase recognition. We employ different measurements to weigh the lexical part and syntactic part due to that the different part of sentence makes distinguishing contribution to the sentence semantic during the task of paraphrase recognition. Our experiment is conducted by parsing the pair sentences of MSRPC first, then followed by adopting differentiated weights to calculate the power of different parts of the sentence.Through this method,we have obtained the outperform precision and average F value result compared with the previous approaches.

Keywords: Pattern Recognition; Sentence Characteristic; Lexical ; Syntactic

1 Introduction

The concept of paraphrasing is generally defined on the basis of the principle of semantic equivalence: A paraphrase is an alternative surface form in the same language expressing the same semantic content as the original form. Paraphrase recognition is aimed to classify the object sentence pair positive paraphrase or not.Paraphrases recognition has attracted global intellectuals devoting their efforts because of its wide range of applications in Query Expansion, Information Extraction, Machine Translation and so forth [1].Consider the following examples, paraphrased from MRSPC sources:
S1: Special, sensitive light sensors pick up the telltale glow, he said.
S2: A sensitive light detector then looks for the telltale blue glow.
Forward methods like surface string similarity and vector model method will fail in recognize this positive paraphrase as they ignored the synonyms of words as well as the structure which represent the meaning of sentence. This paper concentrates on introducing a novel data-driving method to recognize sentence level paraphrase. Sentence's part-of-speech labels as well its parsing tree are being employed to discovery deep semantic relations of sentence pair. Our work is divided into two phases. Firstly, POS-tagging and syntactic analysis are done on the sentence pair to obtain kernel information of the semantic role and parsing

tree structure. Secondly , adopt train data to construct a two hierarchical Model based on kernel information of semantic role and parsing tree structure extracted from the first phase. Then test data is employed to recognize paraphrase relationship of the sentence pair. We do deep semantic mining and construct the hierarchical model to do the sentence level paraphrase recognition which does not employed by previous methods .we take the distinguish into consideration and they indeed achieve encourage result. The rest of the paper is organized as follows. A survey of related works is conducted in Section 2, then the definition of semantic level difference method and its algorithm are proposed in Section 3. Section 4 introduces the experiment details and the progress of the method,with detailed analysis of the result. Finally we conclude the paper in Section 5 with a guideline of the future work.

2 Related Works

Sentence level paraphrase recognition is aimed to judge whether two sentences express the same meaning or not. Previous methods can be classified into followed categories due to their different viewpoints. These are the method based on surface string similarity, operated on syntactic or semantic, Machine Learning Method, method employs co-reference or text entailment,and method combine information from different levels.

Surface string similarity approaches which use string edit distance,number of common words,and combination of several string similarity measures to recognize paraphrase[2].This method just considered the surface feature without semantic information of string so that it can not deal with the synonym in sentences. Vector Space Model for semantics is another popular method combine the vector of single word and its cosine similarity to recognize paraphrase[4,5].Vector Space Model employed bag-of-words model but ignored the syntactic relations to calculate the similarity of the two original sentences.

Part researchers employed syntactic similarity based on syntax level to compute the similarity of the dependency trees to detection paraphrase,this method neglected basic lexical similarity's contribution to the task[6,7,8,9].

There are several approaches use Machine Learning to do the recognize job.They trained a classifier using training data to classify unseen pairs paraphrase or not by examining their inherited features[10,11,12,18]. The point is to select proper restraint of the object sentences. Other researchers had put forward the opinion that co-reference resolution could be employed to solve paraphrase recognition to some extend. more practical work is needed to examine the theory's efficiency [3,13].

Sentence level paraphrase recognition can be treated as the extended synonym detection for sentence pair. So the principal work is to seek appropriate characteristics and set fit constraints[19] on those characteristics to measure two sentences indicating the approximately same meaning or not. Generally components of sentence are from lexical and syntactic aspects,different lexical parts or syntactic parts do distinguishing contribution on the semantic point.Take this

into account,we combine the POS part with syntactic relationships to meet the approval of the paraphrase standard.

Consider the forward methods' strengths and weaknesses,we utilize machine learning process to treat the lexical and syntactic features of the sentence to do paraphrase recognition and get our improvement. The learning process provide us the chance to know the internal rules of the paraphrase and the proper environment to adjust our method. Our method employ the advantages of machine learning and avoid the unilateral feature of surface string similarity and vector space model approaches.

3 Our Method and Idea

3.1 Approach Overview

As introduced forward,we use machine learning which employ the two different levels of a sentence, one is the lexical level ,the other the syntactic level, which correspond to the POS-phase and Syntactic-phase respectively, to do sentence level paraphrase recognition. Figure 1 describes the whole process of our approach. The POS-phase: Do sentence POS-tag job to obtain the sentence lexical

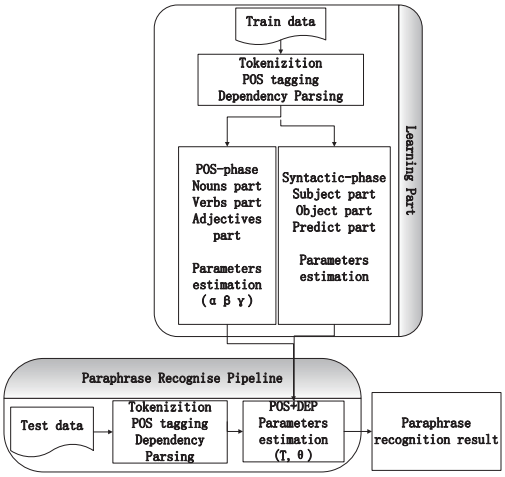

Fig. 1. Overview of our approach.

analysis ,then to calculate the object sentence pair's similarity on lexical level .The main idea is based on the premise that paraphrase sentences consists of the same or similarity concepts, so the Noun part ,the Verb part and the Adjective part of sentence should reach a certain level of similarity.

The Syntactic-phase: Do dependency parsing on the sentence, and obtain the object sentence pair's similarity on syntactic level .The core part of sentence is

consisted of the subject ,object and the predict connects the above two.Similarity of this phase is to extract the subject ,object and the predict part of the object sentence pair and setting proper constraints on them to identify paraphrase.

With the forward two-phase, we can get a hierarchical model consist of POS-phase and the Syntactic-phase to recognize paraphrase.The model can give a similarity score of the sentence pair to compared with the threshold,which is obtained from the training data.Sentence pair will be identified as positive paraphrase if the final result over the threshold,otherwise the negative.

3.2 Calculating Lexical and Syntactic Similarity

As described forward, the lexical level contribution on paraphrase recognition is based on POS-tagging result. The nouns shoulder the responsibility to express the static part of the sentence, the verbs part represent the action, while the adjective part describes how the verbs affects the nouns or the degree of the change. Those three parts take the dominate state to determine the similarity of the sentence pair.Furthermore, those three parts also have different degrees of effect on determine the paraphrase positive or not. So we first employ POS-tagging tools to obtain the part of speech result .Here, We take a sentence pair for example.
Sentence pair(Sentence 1 , Sentence 2)
Sentence 1 :The settling companies would also assign their possible claims against the underwriters to the investor plaintiffs, he added.
Sentence 2 :Under the agreement, the settling companies will also assign their potential claims against the underwriters to the investors, he added.
According to the POS-tagging analysis, we take
Noun parts("NN","NNP","NNS","NNPS"),
Verb parts("VB","VBD","VBG","VBN","VBP","VBZ"),
and Adjective parts("JJ") of the sentence pair and obtain the following Table 1.

Table 1. Noun, Verb, Adjective Parts of the sentence pair.

sentence 1				sentence 2			
Id	Word	Lemma	POS	Id	Word	Lemma	POS
2	settling	settle	VBG	3	agreement	agreement	NN
3	companies	company	NNS	6	settling	settle	VBG
6	assign	assign	VB	7	companies	company	NNS
8	possible	possible	JJ	10	assign	assign	VB
9	claims	claim	NNS	12	potential	potential	JJ
12	underwriters	underwriter	NNS	13	claims	claim	NNS
15	investor	investor	NN	16	underwriters	underwriter	NNS
16	plaintiffs	plaintiff	NNS	19	investors	investor	NNS
19	added	add	VBD	22	added	add	VBD

On lexical level similarity ,We calculate the Noun,Verb,Adjective similarity respectively with Formula (1),(2),(3),then adopt the Noun,Verb,Adjective parts of the sentence pair to calculate the POS similarity.

$$Noun_{sim}\,(S_1,S_2) = \frac{(Noun_{s1}) \cap (Noun_{s2})}{(Noun_{s1}) \cup (Noun_{s2})} \tag{1}$$

$Noun_{s1}$ is composed of the nouns("NN","NNP","NNS","NNPS") of sentence 1, $Noun_{s2}$ of sentence2. We employ WordNet[20] to search the similar word of the nouns to help calculate the common part $Noun_{s1} \cap Noun_{s2}$,then the $Noun_{s1}$,$Noun_{s2}$ union to get the $Noun_{s1} \cup Noun_{s2}$ part.

$$Verb_{sim}\,(S_1,S_2) = \frac{(Verb_{s1}) \cap (Verb_{s2})}{(Verb_{s1}) \cup (Verb_{s2})} \tag{2}$$

Same as the noun parts, Verb parts include "VB","VBD","VBG", "VBN","VBP","VBZ".

$$Adj_{sim}\,(S_1,S_2) = \frac{(Adj_{s1}) \cap (Adj_{s2})}{(Adj_{s1}) \cup (Adj_{s2})} \tag{3}$$

Adjective parts("JJ") also do the same computation as the forward noun parts.

$$Pos_{sim}\,(S_1,S_2) = \alpha \times Noun_{sim} + \beta \times Verb_{sim} + \gamma \times Adj_{sim} \tag{4}$$

Then the POS similarity is derived from the three parts shown with Formula(4), α,β,γ are the coefficients of each part, here To normalization, the result value between 0 and 1 , $\alpha + \beta + \gamma = 1$ .The number of each coefficient will represent the distinguishing power of itself to POS similarity. The value of these parameters will be detailed in the experiments part.

Then we will settle the recognition work combine Dependency phase.A crucial fact about dependency grammars is that the subject and object part play the vital role relative to other parts in determining the meaning of the sentence. The subject part holds the responsibility to illustrate the kernel concept of the sentence, while the object part indicates the variation of the kernel concept. Those two parts consist the backbone of one sentence. Here ,we make use of the dependency relationship to weigh the similarity degree of a sentence pair. we first employ dependency parser to get the dependency relationships.Still take the forward sentence pair(Sentence 1 , Sentence 2) for example. The pair's dependency relationships are shown in Table 2. From the dependency parser result ,we obtain the subject parts(nsubj, nsubjpass, xsubj) and object parts(dobj, iobj) as the kernel part of the sentence which expresses the semantic of the sentence. Consider the two parts' role , we endow them the same proportion in computation dependency similarity. The dependency similarity process is described as follows.Formula(5),(6),(7) give the detail calculation.

$$Subj_{sim}\,(S_1,S_2) = I(subj_{s1}, subj_{s2}) \tag{5}$$

Table 2. Dependency relationships of sentence pair.

sentence 1	sentence 2
det(companies-3,The-1)	det(agreement-3,the-2)
amod(companies-3,settling-2)	prep-under(added-22,agreement-3)
nsubj(assign-6,companies-3)	det(companies-7,the-5)
aux(assign-6,would-4)	amod(companies-7,settling-6)
advmod(assign-6,also-5)	nsubj(assign-10,companies-7)
ccomp(added-19,assign-6)	aux(assign-10,will-8)
poss(claims-9,their-7)	advmod(assign-10,also-9)
amod(claims-9,possible-8)	parataxis(added-22,assign-10)
dobj(assign-6,claims-9)	poss(claims-13,their-11)
det(underwriters-12,the-11)	amod(claims-13, potential-12)
prep-against(assign-6,underwriters-12)	dobj(assign-10,claims-13)
det(plaintiffs-16,the-14)	det(underwriters-16,the-15)
nn(plaintiffs-16,investor-15)	prep-against(assign-10,underwriters-16)
prep-to(assign-6,plaintiffs-16)	det(investors-19,the-18)
nsubj(added-19,he-18)	prep-to(assign-10,investors-19)
	nsubj(added-19, he-18)

$$Obj_{sim}\,(S_1,S_2) = I(obj_{s1}, obj_{s2}) \tag{6}$$

$$Dep_{sim}\,(S_1,S_2) = 0.5 \times Subj_{sim} + 0.5 \times Obj_{sim} \tag{7}$$

The $Subj_{sim}$ and the Obj_{sim} compose the Dep_{sim}, $Subj_{sim}$ represents the contribution of the subject parts, and the Obj_{sim} delegates the object parts. We employ the Resnik method[14] to measure the Obj_{sim} and the Obj_{sim} shown as follows.

$$\begin{cases} Res_{sim}\,(C_1,C_2) = IC(C_1,C_2) & \\ IC\,(C_1,C_2) = \max[-\log p(c)] & c \in S(C_1,C_2) \\ freq(c) = \sum[count(n)] & n \in words(c) \\ p(c) = \frac{freq(c)}{N} & \end{cases} \tag{8}$$

Resnik method is a measure of semantic similarity in an is-a taxonomy, based on the notion of information content,it gives the similarity score of two concept C_1, C_2, $S(C_1,C_2)$ is the common ancestor nodes. $words(c)$ represents the number of words in c branch, N is number of words of whole tree which the the c in. Here we use the upper formula to compute $Subj_{sim}$ and Obj_{sim}. As referred forward, subject parts include nsubj, nsubjpass, xsubj part, and the information content of $subj_{s1}$,$subj_{s2}$ is computed from the words they contain, $subj_{s1}$ is the subject parts of sentence s1, $subj_{s2}$ is that of s2. For example, nsubj(assign-6,companies-3), nsubj(added-19,he-18) composes the $subj_{s1}$ of s1 , nsubj(assign-10 , companies-7), nsubj(added-22,he-21) creates the $subj_{s2}$. We get the average $IC(assign, assign), IC(companies, companies)$ to express $IC(subj_{s1}, subj_{s2})$. Similarly, the Obj_{sim} is achieved by the same compute process with $Subj_{sim}$. Object parts include iobj, dobj part.

As proposed forward, paraphrase's similarity is based on lexical level similarity and the dependency similarity, which described by POS similarity and DEP similarity. So we get the combination of the two aspects to weigh a sentence pair's similarity. The followed Formula(9) illustrates our combination.

$$PR_{sim}\,(S_1,S_2) = \theta \times Subj_{sim} + (1-\theta) \times Obj_{sim} \tag{9}$$

Where PR similarity represents the sentence pair's final similarity score, θ is the parameter, we can get the best PR_{sim} by adjust θ with train data. From the details of the method described forward ,we give the algorithm in Figure 2.

The process of calculating the PR_similarity result

Input: Sentence pairs of train data <s1,s2>
Output: The Similarity score of <s1,s2>
1 Do POS-tagging for the whole train data;
2 for *i*th sentence pair<s1,s2>(1<=*i*<=*n*)
3 get the noun_s1,noun_s2
4 get the verb_s1,verb_s2
5 get the adj_s1,adj_s2
6 Using formula (1)(2)(3)(4) to calculate POS_similarity
7 end for
8 Do denpendency parsing for the whole train data;
9 for *i*th sentence pair<s1,s2>(1<=*i*<=*n*)
10 get the subj_s1,subj_s2
11 get the obj_s1,obj_s2
12 Using formula (5)(6)(7)(8) to calculate DEP_similarity
13 end for
14 Using formula (9) to calculate *PR_similarity*
15 classify <s1,s2> positive paraphrase or not

Fig. 2. The process of calculating the PR similarity result.

4 Experiment and Results

In the experiment we employ Microsoft Research Paraphrase Recognition Corpus[15] to test our method. The MRPRC is made up with 5801 pair sentences which is divided into train data(4077 pair sentences) and test data(1726 pair sentences). The data is unbalanced in that 67words in POS-similarity part and DEP-similarity part both have been expanded with WordNet. The value of the α,β,γ is from 0 to 1 ,and α is bigger than β and γ. We use the train data to adjust the three parameters to achieve the state of art result. Figure 3 describes the variation of the POS_{sim} under different parameters combination, and we get the combination that $\alpha = 0.45$,$\beta = 0.35$,$\gamma = 0.2$ to achieve the best result of POS_{sim}. With the result of the POS_{sim} and Dep_{sim} of the training data, we go on training the θ in PR_{sim} to get the proper threshold to maximize the precision ,and set the $\theta = 0.40$. Figure 4 shows 's effect on precision.

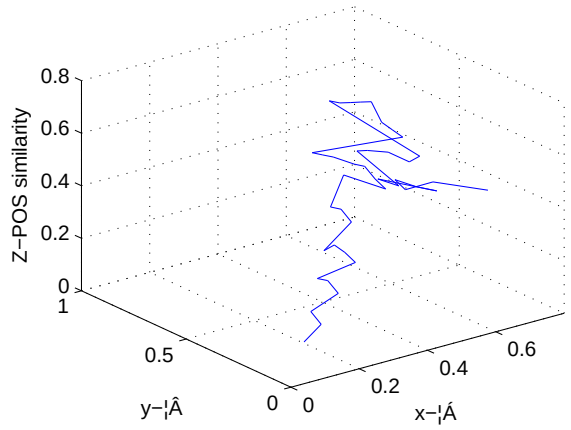

Fig. 3. POS similarity variation on different α,β,γ parameters combination

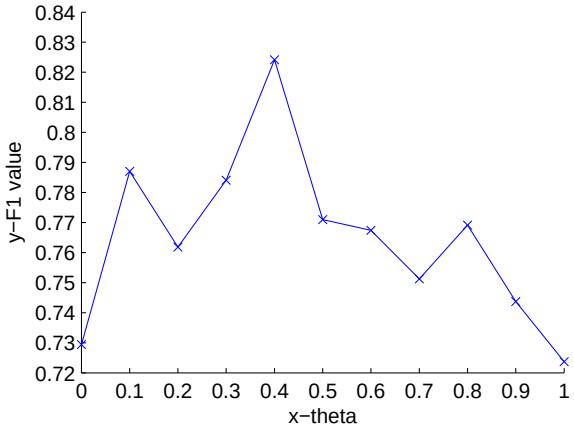

Fig. 4. Precision variation on different θ

The training part of the MSRPC was used to find the classification threshold for the similarity score which maximized accuracy. Strict definition of paraphrase requires the object sentence pair have identical meanings,but the creator of the corpus[11] found that this standard will limit the paraphrase to identical string copies of the each other,so they relaxed the paraphrase from'full bidirectional entailment' to 'mostly bidirectional entailments'. The key message of the guidelines for the annotators of the corpus stated that a paraphrase sentence pairs should describe the same event and contain the same important information .We finally adopted the training data with our method to get the appropriate threshold 0.30.The sentence pair would be recognized as true when the PR similarity of the test data over the threshold,otherwise the false . By setting the parameters,we adopt test data to get the experiment result. Here we use the general evaluation measure: accuracy, precision, recall, and F measure. Those are defined as follows.

$$\begin{cases} accuracy = \frac{TP+TN}{TP+TN+FP+FN} \\ precision = \frac{TP}{TP+FP} \\ recall = \frac{TP}{TP+FN} \\ F = \frac{2\times precision\times recall}{precision+recall} \end{cases}$$

where TP are true positives, TN are true negatives,FN are false negatives and FP are false positives.

Table 3. Noun, Verb, Adjective Parts of the sentence pair.

Method	accuracy	precision	recall	F-measure
POSDEP	**72.5**	**76.2**	**90.7**	**82.8**
Malakasiotis	**76.2**	79.4	86.8	82.9
Das and Smith	76.1	**79.6**	86.1	82.9
Wan	75.6	77.0	90.0	**83.0**
Finch	75.0	76.6	89.8	82.7
Qiu	72.0	72.5	**93.4**	81.6
Zhang	71.9	74.3	88.2	80.7
Mihalcea	71.5	72.3	92.5	81.2
Vector-based	65.4	71.6	79.5	75.3
random	51.3	68.3	50.0	57.8

Results of semantic similarity of paraphrase based on MSRPC corpus are shown in Table 3.Our method(POS+DEP) result lays on top line, and previous approach results are down the Table 3. The distinguish level similarity approach outperforms both baselines for all three of the similarity measures used in these experiments. It can also be seen that, our result outperforms the previously reported methods in terms of precision and the accuracy is on the average level. From the Table 3,we find that the accuracy part is lower than the recent previous method ,we suppose the followed factors may have negative effects on it.As we adopted Stanford POS Tagger to do part of speech work, and Stanford Parser to

obtain the dependency relationships. The best resulting accuracy for the tagger is 96.86% overall, and 86.91% on previously unseen words [16],meanwhile the Parser owns the precision 86.32%[17].Because our work based on POS-tagging and Dependency parsing, those two procedures' precision can definitely have effect on the performance of paraphrase recognition, in the future ,we will take those factors into account to improve our current method.

5 Conclusion and Future Work

This paper proposed a novel hierarchical model based on lexical and syntactic level to do paraphrase recognition. We used two aspects, the POS-tagging and syntactical structure to construct the model and obtain the effective results. The POS-tagging gives the lexical level semantic similarity and syntactical structure shows the work of dependency similarity on paraphrase recognition, each aspect use WordNet to do semantic expansion in finding similarity. The outcome of evaluation experiments shows that this method outperforms the previous similar approaches. Our semantic level difference method in paraphrase recognition indicates the following viewpoints: (1)The Nouns ,Verbs and adjectives shoulder the primary responsibility in lexical level semantic level paraphrase recognition. Nouns describes the entities and verbs illustrate the variation of the entities, adjectives describe the degree of the variation.(2) The subject parts and the object parts in dependency structure of a sentence pair , play vital role in sentence pair paraphrase recognition.(3) Both lexical level semantic and syntactic structure have respective effect on the paraphrase recognition, we adopt linear combination of the two to do paraphrase recognition is valid.The two-hierarchical model method can be apply to the task such as sentence translation, query expansion, question answering etc. Future work will be focus on deep mining of paraphrase constrains besides those put forward in this paper, and solutions will be researched to avoid the problem described in the experiment analysis part.

Acknowledgments.

This work was supported by the National Natural Science Foundation of China (No. 61003192), the Major Research Plan of National Natural Science Foundation of China (No. 90920005).

References

1. Clough, P., et al.: MEasuring TExt Reuse. Proceedings of the 40th Anniversary Meeting for the Association for Computational Linguistics, pp. 152-159. Pennsylvania, PA(2002).
2. Barzilay, R., L. Lee: learn to paraphrase, An Unsupervised Approach Using Multiple-Sequence Alignment. Proceedings of HLT-NAACL,pp. 16-23.(2003)

3. Malakasiotis, P. I. Androutsopoulos: Learning Textual Entailment using SVMs and String Similarity Measures. Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing, 45th Annual Meeting of the Association for Computational Linguistics , pp. 42-47. Prague, Czech Republic(2007)
4. Fernando, S., M. Stevenson: A Semantic Similarity Approach to Paraphrase Detection. Computational Linguistics,(2008)
5. Erk, K., S. Pado: Paraphrase assessment in structured vector space Exploring parameters and datasets, Proceeding of the 2nd European Conference on Computational Learning Theory, pp. 57-65.Athens,Greece(2009)
6. Wan, S., et al.: using dependency-based features to take the para-farce out of paraphrase. Proceedings of the 2006 Australasian Language Technology Workshop, pp. 131-138(2006)
7. Qiu, L., M. Kan , T. Chua: Paraphrase recognition via dissimilarity significance classification. In: EMNLP 2006 Association for Computational Linguistics, pp. 18-26.Sydney,(2006)
8. Socher, R., et al.: Dynamic Pooling and Unfolding Recursive Auto encoders for Paraphrase Detection. In : Conference of Neural Information Processing Systems Foundation(2011)
9. Lintean, M. , V. Rus: Paraphrase Identification Using Weighted Dependencies and Word Semantics. Proceedings of the Twenty-Second International FLAIRS Conference. Association for the Advancement of Artificial Intelligence: Sundial Beach and Golf Resort,pp. 260-265. Sanibel Island, Florida, USA(2009)
10. Pang, B., K. Knight , D. Marcu: syntax-based alignment of multiple translations,Extracting Paraphrases and Generating New Sentences. Proceedings of HLT-NAACL, 2003: p. 102-109.
11. Dolan, W.B. , C. Brockett: Automatically Constructing a Corpus of Sentential Paraphrases. Proceeding of the 3rd International Workshop on Paraphrase, pp. 9-16. Jeju island,Korea(2005)
12. Zhang, Y. , J. Patrick: Paraphrase identification by text canonicalization. Proceedings of the Australasian Language Technology Workshop,pp. 160-166. Sydney,Australia(2005)
13. Recasens, M. , M. Vila: On Paraphrase and Coreference. Computational Linguistics,pp. 639-647. 36(4)(2010)
14. Philip Resnik: Using Information Content to Evaluate Semantic Similarity in a Taxonomy. In International Joint Conference on AI, pp. 448-453.(1995)
15. Dolan, B., C. Quirk , C. Brockett: Unsupervised Construction of Large Paraphrase Corpora,Exploiting Massively Parallel News Sources. Proceeding of the 20th international conference on Computational Linguistics, pp. 350-356. Geneva Switzerland (2004)
16. Toutanova, K. , C.D. Manning: Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger. Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora,pp. 63-70.(2000)
17. Klein, D.,C.D. Manning: Accurate Unlexicalized Parsing. Proceedings of the 41st Meeting of the Association for Computational Linguistics,pp. 423-430.(2003)
18. Malakasiotis, P.: Paraphrase Recognition Using Machine Learning to Combine Similarity Measures. Proceedings of the ACL-IJCNLP 2009 Student Research Workshop, pp. 27-35. Suntec,Singapore(2009)
19. Callison-Burch, C.: Syntactic Constraints on Paraphrases Extracted from Parallel Corpora. Proceeding EMNLP'08 Proceedings of the conference on Empirical Methods in Natural Language Processing,pp. 196-205. Stroudsburg,PA,USA(2008)

20. Fellbaum,C.(1998).WordNet:An Electronic Lexical Database.MIT Press