WUST at NTCIR-10 RITE-2 Task：Multiple Feature Approach to Chinese Textual Entailment

Maofu Liu, Yue Wang, Yan Li, Huijun Hu

1. College of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan 430065, Hubei, P.R.China

2. Hubei Province Key Laboratory of Intelligent Information Processing and Real-time Industrial System, Wuhan 430065, Hubei, P.R.China

liumaofu@wust.edu.cn, 289455785@qq.com, liyan880923@sina.com, huhuijun@wust.edu.cn

ABSTRACT

This paper describes our work in NTCIR-10 on RITE-2 Binary-class (BC) subtask and Multi-class (MC) subtask in Simplified Chinese. We construct the classification model based on support vector machine to recognize semantic inference in Chinese text pair, including entailment and non-entailment for BC subtask and forward entailment, bidirectional entailment, contradiction and independence for MC subtask. In our system, we use multiple features including statistical feature, lexical feature and syntactic feature.

Categories and Subject Descriptors

I.2.7 [Artificial Intelligence]: Natural Language Processing - text analysis.

I.3.1 [Information Storage and Retrieval]: Content Analysis and Indexing - linguistic processing.

General Terms

Experimentation.

Keywords

Textual Entailment, Multiple Textual Featur, SVM-based Classifier

1. Introduction

In NTCIR-10, RITE-2 is an evaluation-based workshop aiming to recognize entailment, paraphrase, and contradiction between sentences, which is a common problem shared widely among researchers of natural language processing and information access. In RITE-2, there are four subtasks, i.e. Binary Class (BC), Multi Class (MC), Entrance Exam and RITE4QA. We just focus on two of them, BC and MC subtasks.

BC subtask means that given a text pair (t1, t2), a system can automatically identify whether text t1 entails or infers hypothesis text t2 or not. Text t1 entailing text t2 means that t1 has the same meaning with t2 while t1 also has more meaning than t2. In other words, the events described in t2 can be inferred from t1. If the events described in t1 can be true, the events in t2 are always true.

For example, the following text pair is an example of Sentence1 entailing Sentence2.

In Simplified Chinese (CS):

Sentence1: 异位性皮肤炎患者则可能会对某些食物或环境中的过敏原过敏。

Sentence2: 异位性皮肤炎患者可能会过敏。

In English(EN):

Sentence1: Patients with atopic dermatitis might be allergic to certain food or allergens in the environment.

Sentence2: Patients with atopic dermatitis may be allergic.

MC subtask is a 4-way labeling task to automatically detect there is one of the four semantic relations, i.e. **Forward**, **Bi-direction**, **Contradiction** and **Independence**, in a given text pair (t1, t2).

Forward entailment means that t1 entails t2 and t2 does not entail t1 and it is a one-way entailment. If it is the case that t1 entails t2 and t2 entails t1, then t1 and t2 are true in exactly the same conditions, and are thus equivalent or paraphrase. In other words, equivalence or paraphrase is the bidirectional entailment and we also call it **Bi-direction**.

For example, in the following text pairs, the pair of Sentence3 and Sentence4 is an example of **Forward** entailment, namely the pair of Sentence5 and Sentence6 is an example of **Bi-directional** entailment.

In Simplified Chinese (CS):

Sentence3: 由朗霍华德执导的《美丽境界》是本届金球奖的最大赢家，获得戏剧类“最佳影片”、“最佳男主角”、“最佳女配角”与“最佳编剧”四项大奖。

Sentence4: 《美丽境界》在本届金球奖中独得戏剧类最佳影片等四项大奖。

Sentence5: 洋基队为保护选手而要求“只投一场且不超过一百球”。

Sentence6: 洋基球队开出“只投一场且不超过一百球”的保护选手的条件。

In English(EN):

Sentence3: The movie "A Beautiful Mind" directed by Ron Howard was the biggest winner of current Golden Globe Awards. It won four drama awards for "Best Film" "Best Actor" "Best Supporting Actress" and "Best Screenplay".

Sentence4: The movie "A Beautiful Mind" won four drama awards such as "Best Film" and so on in current Golden Globe Awards.

Sentence5: The Yankees required that they only cast for a baseball game and no more than 100 goals.

Sentence6: The Yankees made it a condition that they only cast for a baseball game and no more than 100 goals.

In MC subtask, non-entailment contains **Contradiction** and **Independence**. **Contradiction** means that t1 and t2 contradicts, or can not be true at the same time. Independence means that if the text pair (t1, t2) can not be put into any of the three-way entailment, we put it into the **Independence** class. For instance, the relation between Sentence7 and Sentence8 is contradiction, the relation between Sentence9 and Sentence10 is **Independence**.

In Simplified Chinese (CS):

Sentence7: 基特是一位棒球选手。

Sentence8: 基特未曾打过棒球。

Sentence9: 美国二氧化碳排放量目前位居世界第一。
Sentence10: 中国二氧化碳排放量目前位居世界第二。

In English(EN):

Sentence7: Kitt is a baseball player.

Sentence8: Kitt never played baseball.

Sentence9: The carbon dioxide emission in U.S. ranks first in the world currently.

Sentence10: The carbon dioxide emission in China ranks second in the world currently.

Recognizing textual entailment is essentially a classification problem which can be implemented by machine learning methods. In this paper, we use SVM-based classification method and multiple textual features to solve the entailment problem. We totally used 14 features including 8 statistical features, 1 lexical feature and 5 syntactic features.

2. System Description

Our system includes three main modules, i.e. preprocessing, feature extraction and SVM Classifier. Figure 1 illustrates our system architecture in detail.

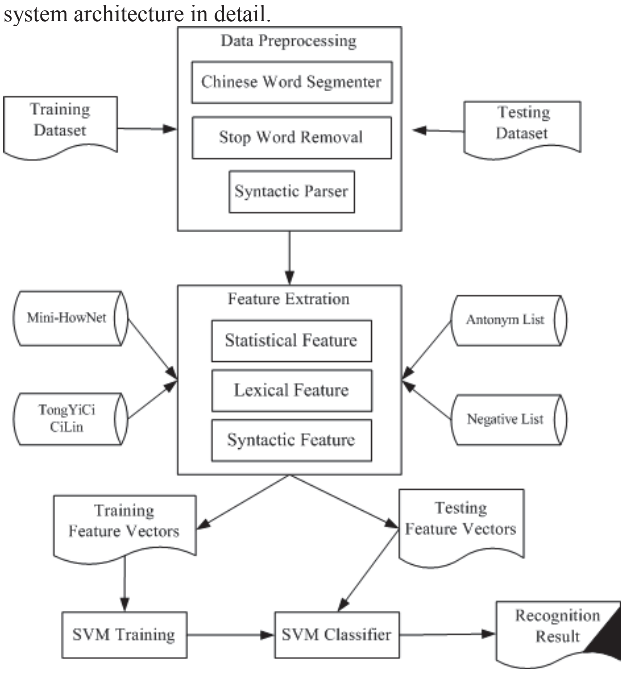


Figure 1. System Architecture

Support vector machine (SVM) is a supervised learning model with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. The following Figure 2 describes the SVM classifier in our system.

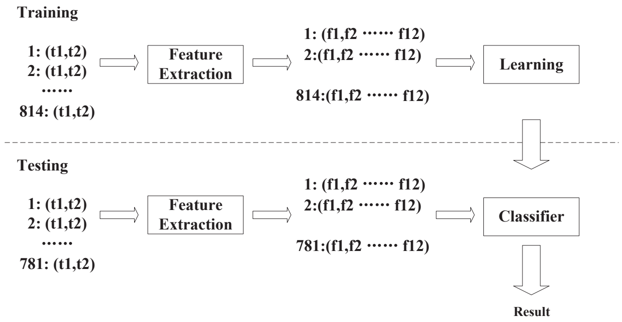


Figure 2. SVM Classifier

2.1 Data preprocessing

In the data preprocessing, the system mainly implements the Chinese word segmentation and removes the stop words according to the stop word list.
As we all know, English word is usually separated by spaces, but Chinese is written without spaces between words in standard. We choose Stanford Chinese word segmenter with PKU standard as the tool to segment the Chinese word. The following example is the preprocessing result of the original training data given by NTCIR-10 after Chinese word segmentation and removing stop words. Our system does not take the POS tagging into accounts.

Example 1:

< pair id "10" label "B" >

<t1> 1998 年 登陆 内地 泰坦尼克号 曾 获得 3.2 亿 元 票房

</t1>

<t2> 1998 年 引进 大陆 泰坦尼克号 票房 3.2 亿 元 </t2>

< pair >

2.2 Feature extraction

In this subsection, we mainly focus on three kinds of features used in our system.

2.2.1 Statistical feature

There are eight features being relevant to Statistical feature in our system, word overlap, length difference, Manhattan distance, Euclidean distance, cosine similarity, Jaro-Winkler distance, LCS similarity and same words ratio in shorter text.

(1) Word overlap

This feature considers how many same words existing in text t1 and text t2 for the text pair (t1, t2). This feature is used in our system because we hold the assumption that the more of the same words in the text pair, the higher similarity between the two texts and the pair (t1,t2) is mostly like to express the similar meaning.

$$W_{overlap} = \frac{|Words(t1) \cap Words(t2)|}{|Words(t1) \cup Words(t2)|} \quad (1)$$

Where $Words(t1)$ denotes the set of the words in text t1.

(2) Length difference

This feature concerns the length difference between text t1 and text t2 for the text pair (t1, t2). We use this feature in our system because if text t1 entails text t2, the text t1 mostly contains more information than text t2. And this rule shows that if text t1 entails text t2, the length of text t1 is mostly larger than the length of text t2. And the most important thing is that if the lengths of text t1 and t2 are almost equal, maybe the text pair (t1,t2) holds the bidirectional entailment relationship.s

$$Len_{dif} = length(t1) - length(t2) \tag{2}$$

Where the function $length(t)$ is used to calculate the length of the text.

(3) Manhattan distance

Manhattan distance defines the distance between two points measured along axes at right angle. In a plane with point p1 at (x1, y1) and point p2 at (x2, y2), the Manhattan distance between them is $|x1 - x2| + |y1 - y2|$. Manhattan distance can be used to calculate the similarity between the text pairs.

$$M_{dis}(\vec{t1}, \vec{t2}) = \sum_{i=1}^{n} | t1_i - t2_i | \tag{3}$$

Where $\vec{t1}$ and $\vec{t2}$ are the vectors, using the TF*IDF to compute the value for each element, of the texts t1 and t2 respectively and the parameter n is the dimension of the vector.

(4) Euclidean distance

Euclidean distance also takes the distance between two data points into accounts and can be used to calculate the similarity between the vectors of the texts t1 and t2. The vectors of the text pair in this feature are same as those of Manhattan distance.

$$E_{dis}(\vec{t1}, \vec{t2}) = \sqrt{\sum_{i=1}^{n} (t1_i - t2_i)^2} \tag{4}$$

(5) The cosine similarity

The cosine similarity is used to compute the similarity between the texts t1 and t2 in vector space and the vectors are also the same as the Manhattan distance.

$$Sim_{\cos} = \frac{\sum_{i=1}^{n} t1_i * t2_i}{\sqrt{\sum_{i=1}^{n} t1_i} * \sqrt{\sum_{i=1}^{n} t1_i}} \tag{5}$$

(6) Jaro-Winkler distance

The Jaro–Winkler distance is a measure of similarity between two textual strings. The higher the Jaro-Winkler distance for two textual strings is, the more similar the textual strings are. The Jaro-Winkler distance metric is designed and best suited for short textual strings such as person names.

$$JW_{dis} = \frac{m}{3 * length(t1)} + \frac{m}{3 * length(t2)} + \frac{m - t}{3 * m}$$

$$L_{JW} = \frac{\max(length(t1), length(t2))}{2} - 1 \tag{6}$$

Where m is the number of textual strings that text t1 matching text t2. The concept of matching here means a string appearing in the t1 and t2 at the same time and the position interval is no more than the value of L_{JW}.

(7) LCS similarity

The longest common subsequence (LCS) is to find the longest subsequence common to all sequences in a set of sequences.

$$Sim_{LCS}(t1, t2) = \frac{length(LCS(t1, t2))}{\min(length(t1), length(t2))} \tag{7}$$

Where, LCS(t1,t2) refers to the longest common subsequence between text t1 and text t2.

(8) Same words ratio in shorter sentence

$$Sim(t1, t2) = \frac{Words(\min(t1, t2))|_{similarity=1}}{\min(length(t1), length(t2))} \tag{8}$$

Where $Words(\min(t1, t2))|_{similarity=1}$ refers to the number of words which words similarity equals 1 in the shorter sentence between t1 and t2. Words similarity here is calculated based on CiLin.

2.2.2 Syntactic feature

Compared with RITE-1, this feature is newly added to improve the system. It's effect is significant. In the text pair (t1, t2), we suppose that the syntactic structures of text t1 and text t2 has higher similarity, text t1 and text t2 mostly have higher probability to express the similar meaning. Figure 3 shows the syntax trees, using the Stanford Chinese parser [CHANG et. al. 2009], of the text pair in following Example 2.

Example 2:

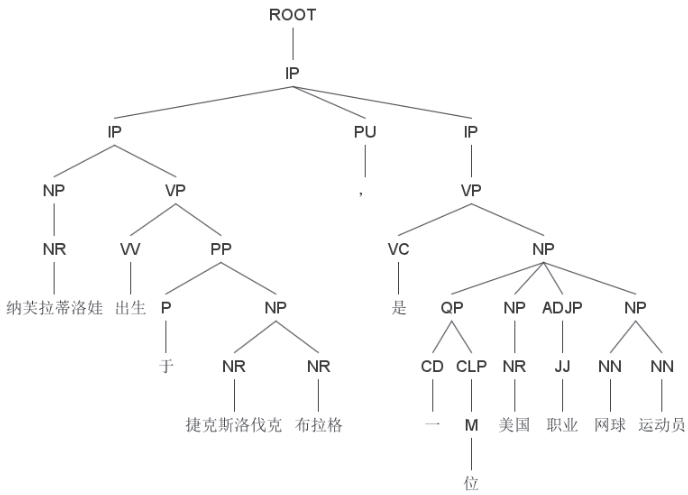
In Simplified Chinese (CS):

Sentence11:纳芙拉蒂洛娃出生于捷克斯洛伐克布拉格，是一位美国职业网球运动员。

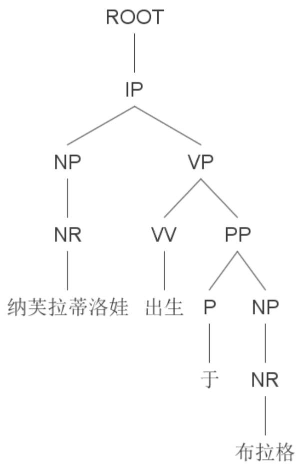
Sentence12: 纳芙拉蒂洛娃出生于布拉格。

In English (EN):

Sentence11: Navratilova was born in Prague, Czechoslovakia, is an American professional tennis player.
Sentence12: Navratilova was born in Prague.



Sentence11



Sentence12

Figure 3. Syntax trees examples for the text pair

We can easily find that the grammatical structure of two syntax trees in Figure 3 is very similar, especial the syntax tress of the Sentence12 to the left branch of the syntax tree of the Sentence11.

According to the syntax tree, we can get the grammatical dependency relationship between the text elements. The following Example 3 is the grammatical dependency relation triples for the text pair in Example 3.

Example 3:

Sentence13: prep(出生-2, 于-3), dep(布拉格-5, 捷克斯洛伐克-4), dep(于-3, 布拉格-5), dep(美国-10, 是-7), num(美国-10, 一-8), dep(一-8, 位-9), root(ROOT-0, 美国-10), amod(美国-10, 职业-11),nn(运动员-13, 网球-12), dep(美国-10, 运动员-13)

Sentence14: root(ROOT-0, 出生-2), prep(出生-2, 于-3), dep(于-3, 布拉格-4)

We calculate the syntactic feature based on these dependency relation triples of the text pair and by the following formula.

$$Sim_{syntree} = \frac{\sum_{p_{t2} \in S_{t2}} \max_{p_{t1} \in S_{t1}} \{sim_p(p_{t1}, p_{t2})\}}{|S_{t2}|} \quad (9)$$

Where the S_{t1} and S_{t2} represent the sets of dependency relation triples of text t1 and t2 in the text pair (t1,t2) respectively, the P_{t1} and P_{t2} denoting dependency relation triples in the S_{t1} and S_{t2} sets, and the $sim_p(p_{t1}, p_{t2})$ is used to compute the similarity between P_{t1} and P_{t2}.

$$sim_p(p_{t1}, p_{t2}) = \frac{1}{2}(\max\{sim_w(w_1, w_1) + sim_w(w_2, w_2), \\ sim_w(w_1, w_2) + sim_w(w_2, w_1)\})$$

$$sim_w(w_1, w_2) = \begin{cases} 1 & w_1 = w_2 \\ 0 & otherwise \end{cases} \quad (10)$$

Where $w_1, w_2 \in p_{t1}$ and $w_1, w_2 \in p_{t2}$.

2.2.3 Lexical semantic feature

(1) Hownet based similarity

The text semantic similarity based on Hownet can be calculated by the formula below.

$$LS_{Hownet} = \frac{1}{2}(\frac{1}{m}\sum_{i=1}^{m} \max\{sim_w(w_{1i}, w_{2j}) \mid 1 \le j \le n\}) \\ + \frac{1}{n}\sum_{i=1}^{n} \max\{sim_w(w_{1i}, w_{2j}) \mid 1 \le i \le m\} \quad (11)$$

Where the word sets $\{w_{1i} \mid 1 \le i \le m\}$ and $\{w_{2j} \mid 1 \le j \le n\}$ represent the words in text t1 and t2 respectively and we use the same approach to compute the similarity $sim_w(w_1, w_2)$ between two words mentioned in paper (LIU and LI 2002).

(2) Semantic distance similarity

The sentences distance similarity based on HowNet can be calculated by the formula below.

$$sim(t1, t2) = \frac{1}{2}(\frac{1}{m}\sum_{i=1}^{m} \max\{sim(w_{1i}, w_{2j}) \mid 1 \le j \le n\} \\ + \frac{1}{n}\sum_{j=1}^{n} \max\{sim(w_{1i}, w_{2j}) \mid 1 \le i \le m\}) \quad (12)$$

Here, we use tool in paper [2] to calculate the semantic distance similarity based on HowNet.

(3) TongYiCi CiLin based similarity

We hold the assumption that the synonyms in text t1 and text t2 can improve the recognition accuracy of the bidirectional entailment relation, so the text semantic similarity based on the Chinese synonym resource, TongYiCi CiLin, is calculated by the same formula (12) of Hownet based similarity and the word similarity $sim_w(w_1, w_2)$ is computed by the approach mentioned in paper (TIAN and ZHAO 2010)

(4) Antonym

To calculate the pair number of the antonyms in a pair(t1,t2), we must create a antonyms table. The n is the pair number of antonyms in a pair(t1,t2). If n is 0, we think there is no any antonym in the pair(t1,t2). If t1 and t2 are the same, the similarity of t1 and t2 is 1. And if n is not 0, the pair have one or more pairs antonyms may be contradiction.

$$f_{\text{antonym}} = \begin{cases} 0 & (n! = 0) \\ 1 & (n = 0) \end{cases} \quad (13)$$

Where n is the number of the antonym pairs occurring in the text t1 and text t2.

(5) Negative

In order to recognize the contradiction relation more correctly in the text pair (t1, t2), the feature of negative words in the text t1 and text t2 is taken into consideration.

$$f_{neg} = \begin{cases} 0 & (n1 = n2 \,\|\, n1\%2 = n2\%2) \\ 1 & otherwise \end{cases} \tag{14}$$

Where $n1$ and $n2$ are the number of the negative words in the text t1 and text t2 respectively.

2.3 SVM classifier

Here, we choose LIBSVM as the classifier. LIBSVM is a library for support vector classification (SVM) and regression. After preparing and scaling data set in LIBSVM form, our system chooses the RBF kernel function to do the cross-validation.

Figure 4 and Figure 5 are the results of BC and MC respectively.

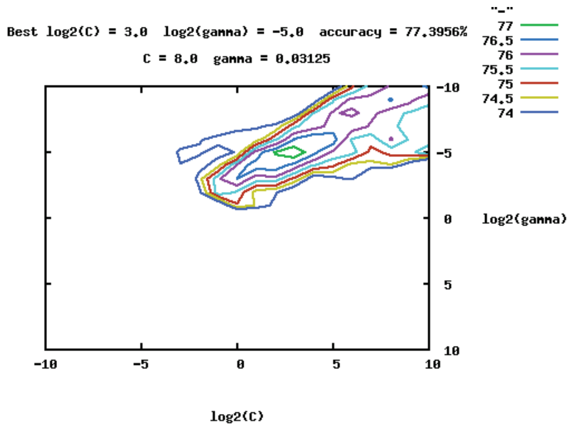


Figure 4. Training data for BC after cross-validation

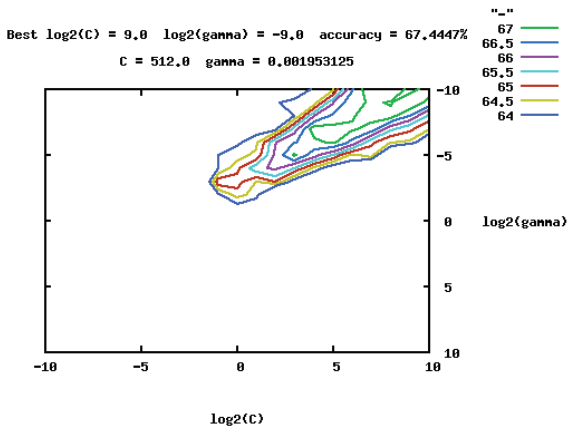


Figure 5. Training data for MC after cross-validation

3. Experiments

We submitted one result of BC and one result of MC to NTCIR-10. The official evaluation results of performance are listed in the Table 1. There is only one type of assessment, automatic assessment by accuracy.

Table 1. Official results of WUST formal run experiment

Run	Subtask	MacroF1	Accuracy
WUST-CS-BC-01	BC	0.501	0.588
WUST-CS-MC-02	MC	0.409	0.524

3.1 BC subtask

The label result is evaluated from three parameters, i.e. F-measure, precision, recall. The respective formulas are listed as follows:

$$F-measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{15}$$

$$Precision = \frac{TP}{TP + FP} \tag{16}$$

$$Recall = \frac{TP}{TP + FN} \tag{17}$$

Where TP（True Positives）and FP（False Positives）are the true number and false number of all the positives of this class in the test dataset.

The experiment results of the BC subtask are shown in the following Table 2.

Table 2. Experiment results of the BC

Label	F-measure	Precision	Recall
Y	0.709	0.573	0.929
N	0.294	0.691	0.187

According to Table 2, we consider the accuracy of “Y” is better than that of “N”. And we think that the most influence factors of accuracy are the judgment of the N mistakes.

In BC subtask, we use the same features with MC subtask, but the characteristics of the BC and MC subtasks should be different, which may cause the dissatisfaction of BC result. We think if we choose some different features to BC, the accuracy of BC would be higher.

3.2 MC subtask

The experiment results of the MC subtask are shown in the following Table 3.

Table 3. Experiment results of the MC

Label	F-measure	Precision	Recall
B	0.597	0.564	0.635
F	0.628	0.479	0.906
C	0.036	0.333	0.019
I	0.374	0.719	0.253

In Table3, B stands for bi-direction entailment, F stands for forward entailment, C stands for contradiction and I stands for independence.

According to Table 3, we can find that contradiction is almost wrong and independence is not ideal. And we think that the most influence factors of accuracy are the judgment of the C and I mistake.

In the experiment, we have found that contradiction is difficult to judge. We added two features, antonyms and negative words, to solve this problem. We need to try more features to solve the contradiction problem.

4. Conclusions

In this paper, we construct the classification model based on support vector machine to recognize semantic inference in Chinese text pair using multiple features, including statistical, syntactic and lexical semantic ones. From the result, we find that using multiple features to recognize textual entailment in Chinese text pairs is workable and effective. However inadequacies still exist. Through further analysis, we find that in our system, we use the same features in BC and MC subtasks, but the characteristics of the BC and MC subtasks should be different, which may cause the dissatisfaction of BC result. We should extract and select different features for the BC and MC subtasks. In the MC subtask, we should optimize features to improve the accuracy. We have compared the use of word segmenter and found the accuracy of Stanford with PKU standard is much better than ICTCLAS. Moreover, we mostly consider statistical features in our system, if we add some rule features, the accuracy may be significantly improved.

ACKNOWLEDGMENTS

The work presented in this paper is supported by the National Nat ural Science Foundation of China (No. 61100133 and 61173062), the Major Projects of National Social Science Foundation of Chi na (No. 11&ZD189) and the Major Projects of Social Science Fo undation of Department of Education of Hubei Province of China (No. 2011jyte126).

REFERENCES

[1] Tian Jiule, Zhao Wei. Words Similarity Algorithm Based on Tongyici Cilin in Semantic. Journal of Jilin University (Information Science Edition),2010,28(6):602-608

[2] Qun Liu , Sujian Li. Word Similarity Computing Based on How-net.Computational Linguistics and Chinese Language Processing,2002,7(2):59-76

[3] CONDORAVDI, C., DICK C., VALERIA de P., RERINHARD S., and DANIEL G.B.. 2003. Entailment, intensionality and text understanding. Proceedings of the HLT-NAACL 2003 workshop on Text meaning, Vol.9, pp. 38-45.

[4] Jonathan Berant, Ido Dagan, Jacob Goldberger. Global Leaning of Focused Entailment Graphs. Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, 2010:1220-1229.

[5] Roni Ben Aharon, Idan Szpektor, Ido Dagan. Generating Entailment Rules from FrameNet. Proceedings of the ACL 2010 Conference Short Papers, 2010:241-246.

[6] Ion Androutsopoulos, Prodromos Malakasiotis. A Survey of Paraphrasing and Textual Entailment Methods. Journal of Artificial Intelligence Research, 2010: 135-187.

[7] P.Schlenker. Meaning II: Entailments. Linguistics1: Introduction to the Study of Language, 2006.

[8] Shima H., Kanayama H., Lee C.-W., Lin C.-J., Mitamura T., Miyao S. S. Y., and Takeda K. 2011. Overview of ntcir-9 rite: Recognizing inference in text. In Proceedings of NTCIR-9 workshop meeting, 291-301.