Binary-class and Multi-class Chinese Textural Entailment System Description in NTCIR-9 RITE

Shih-Hung Wu, Wan-Chi Huang,
CSIE, Chaoyang University of Technology
No. 168, Jifong E. Rd., Wufong, Taichung, Taiwan, R.O.C
{shwu, s9727603}@cyut.edu.tw

Liang-Pu Chen, and Tsun Ku
IDEAS, Institute for Information Industry
8F., No. 133, Sec. 4, Minsheng E. Rd.,Taipei, Taiwan, R.O.C
{eit, cujing}@iii.org

ABSTRACT
In this paper, we describe the details of our system for NTCIR-9 RITE. We sent 3 runs for each of the four sub-tasks: CT-BC, CT-MC, CS-BC, and CS-MC. Our approach to the NTCIR-9 RITE task is based on the standard supervised learning classification. We integrate available computational linguistic resources of Chinese language processing to build the system in a statistical natural language processing approach. First, we observed the training corpus and list all possible features. Second, we test the features on training data and find features that can be used to identify textual entailment. The features include surface text, semantic and syntactical information, such as POS tagging, NER tagging, and dependency relation. An automatic annotation subsystem is built to annotate the training corpus. Finally, the annotated data is used in training statistical models and build the classifier for the RITE 1 subtasks.

Categories and Subject Descriptors
H.3.3 [**Information Search and Retrieval**]: Natural language understanding, Textual Entailment

General Terms
Experimentation

Keywords
Chinese Textual Entailment, linguistic feature, classifier

Team Name
III_CYUT_NTHU

Subtasks/Languages
RITE, RITE4QA, Simplified Chinese, Traditional Chinese

External Resources Used
LIBSVM, Stanford Parser

1. INTRODUCTION
Textual Entailment is an important NLP issue; it is hard and has many potential applications [1]. In NTCIR-9 RITE shared task, there are two subtask in both traditional Chinese and simplified Chinese. The binary-class (BC) subtask is "Given a text pair (t1,t2) identify whether t1 entails (infers) a hypothesis t2 or not", and the multi-class (MC) subtask is "5-way labeling task to detect (forward / reverse / bidirectional) entailment or no entailment (contradiction / independence) in a text pair" [2]. The relation between t1 and t2 might be forward entailment, that is, the information in t2 can be deduced from t1 but t2 cannot entail t1. On the other hand, if the information in t1 can be deduced from t2 but t1 cannot entail t2, this relation is called reverse entailment. Bi-direction entailment means both forward and reverse relations exist. Another relation is contradiction, which means the information in t1 is inconsistent to the information in t2, either contradict or cannot be true at the same time. If not in the previous four relation, then t1 and t2 are independent. Examples of the five relations are given in Table 1. Most of the entailment examples listed here can be regarded as paraphrase [3]. Both t1 and t2 have many terms in common. It is easier for computer to check paraphrasing relation than complex entailment relation. In this paper, our analysis focuses on the paraphrasing part of the textual entailment problem.

Table 1. Examples of five entailment relations

Type	example
Forward	t1：日本時間2011 年3 月11 日，日本宮城縣發生芮氏規模9.0強震，造死傷失蹤約3萬多人（Japan time March 11, 2011, Miyagi Prefecture, Japan, a magnitude 9.0 earthquake occurred, causing casualties of about 30,000 people missing or death)
	t2：日本時間2011 年3 月11 日，日本宮城縣發生芮氏規模9.0強震 (Japan time March 11, 2011, Miyagi Prefecture, Japan, a magnitude 9.0 earthquake occurred)
Reverse	t1：美國主權債信評級從最高的ＡＡＡ調降一級到ＡＡ＋ (U.S. sovereign credit rating downgraded from the highest level of AAA to AA +)
	t2：美國主權債信評級從最高的ＡＡＡ調降一級到ＡＡ＋，將造成美國每年的借貸成本增加約一千億美元 (U.S. sovereign credit rating downgraded from the highest level of AAA to AA+, will result in increasing the borrowing costs for the United States each year about one hundred billion U.S. dollars)
Bidirectional	t1：賓拉登在巴基斯坦美軍攻擊中死亡 (Osama bin Laden died in Pakistan under the U.S. military attack)
	t2：巴基斯坦美軍攻擊中殺死賓拉登 (U.S.

	forces kill bin Laden in Pakistan attack)
Contradiction	t1：張學友在1961 年7 月10 日，生於香港，祖籍天津(Jacky Cheung in the July 10, 1961, was born at Hong Kong, native of Tianjin)
	t2：張學友生於 1960 年(Jacky Cheung was born in 1960)
Independence	t1 ： 黎 姿 與 ” 残 障 富 豪 ” 馬 廷 強 結 婚 (Gigi married with the “disability rich” Mating Jiang married)
	t2：馬廷強為香港“東方報業集團”創辦人之一馬惜如之子(Mating Jiang is the son of Ma Xi Ru, one of the founders of Hong Kong, "the Oriental Press Group")

2. Research Methodology

There are many different approaches are used on textual entailment recognition in English, such as theorem proofing [4][5][6][7], using different semantic resources like WordNet [8] [9]. Our research methodology is based on the standard supervised learning classification [10]. Given a training corpus, we integrate available computational linguistic resources of Chinese language processing to build the system in a statistical natural language processing approach. The developing stages can be described as follows. At the first stage, we observed the training corpus and list all possible features. The features include surface text, semantic and syntactical information, such as POS tagging, NER tagging, and dependency relation. At the second stage, we implement how to extract the features form the training set. An automatic annotation subsystem is built to annotate the training corpus. At the third stage, we build a classification system and test the features on training data in a 10 fold cross validation way and find which features are more useful on textual entailment recognition. Finally, the classification system is used in the RITE tasks.

3. System architecture

The flowchart of our system is shown in Figure 1. The basic components are “preprocessing”, “word segmentation”, “Chinese character conversion”, “feature extraction” and “SVM” classifier.

3.1 Preprocessing

In this implementation, all the terms that have the same meaning in both t1 and t2 will be replaced to the same term, thus safe the time of matching in later steps. This is a short cut to integrate semantic information into the system.

3.1.1 Normalizations

One of the goals of the preprocessing module is to normalize the expression in sentences. In our system, the preprocessing module normalizes the terms in parentheses as a synonym of the term in front of the parentheses. Since parentheses in document usually represents the transliteration or translation. For example, in “車諾比核事故(切爾諾貝利核事故)”, the term in parentheses is another translation of the same term, “Chernobyl nuclear accident”.

There are many ways to represent time in Chinese text. For example, table 2 shows several cases. Our preprocessing module also normalizes the expressions of time in Chinese. We normalize the expressions in a formal way for further computation, such as exact matching, partial matching, duration comparison, and etc.

Table 2. Examples of time expressions

Type	Time expressions in text
Chinese only	一九九七年二月廿三日
Full type digit with Chinese	１９９７年２月２３日
Half type digit with Chinese	1997年2月23日
Digit only	1999-05-07
Duration	1999年延長至2001年

3.1.2 Background knowledge matching and substitution

The second goal of the preprocessing module is to substitute terms with their synonyms to simplify the need of matching in the following steps. The synonyms are collected from Wikipedia. The time express and location express are also treated as a synonym substitution problem.

There is another issue about the time expression is the year era of different dynasties in Chinese or Japanese history. For example, the year of “乾隆(Qianlong)” is 1735, and the year of “昭和(Showa)” is 1925. The time expressions need to be normalized with background knowledge.

Another similar requirement is the acronyms of locations need to be expanded into the full expression before text matching. For example, “台、印、美” refers to “Taiwan, India, and the United States”, therefore, it is normalized into “台灣(Taiwan)、印度(India)、美國(U.S.A)”.

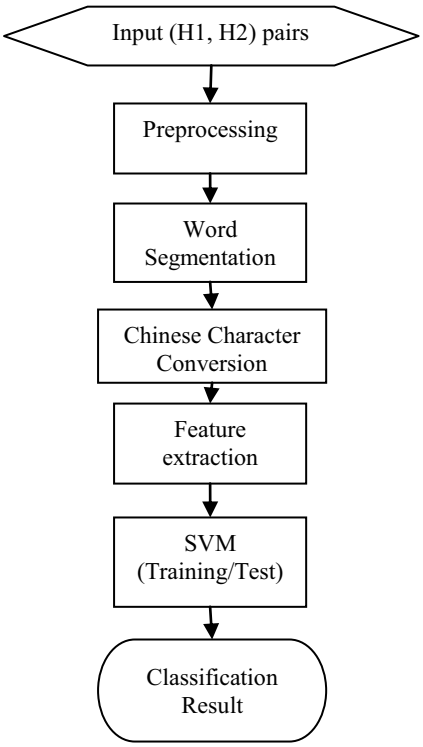

Figure 1. System flowchart

3.2 Word Segmentation and Chinese character conversion

The word segmentation toolkit in used is the ICTCLAS word segmentation system [24], which is provided by the Institute of Computing Technology Chinese Academy of Sciences. The toolkit functions include word segmentation, POS tagging, NE recognition, new word identification, and customized dictionary.

Since the Stanford parser [23] that we used can only take simplified Chinese [22], we must convert the character set for traditional Chinese subtasks. The character conversion tool in used is Google translate [25].

3.3 Feature extraction

The features used in our system are listed here, most of them were used in previous textual entailment recognition works [11]:

1.unigram_recall
2.unigram_precision
3.unigram_F_measure
4.log_bleu_recall
5.log_bleu_precision
6.log_bleu_F_measure
7.difference in sentence length (character)
8.absolute difference in sentence length (character)
9.difference in sentence length (term)
10.absolute difference in sentence length (term)
11.Sub-tree mapping
12.Time mapping

The first three features measure how similarity between t1 and t2 according to the common Chinese character. Where unigram recall, precision, and F measure can be interpreted as the percentage of t1's characters in t2, the percentage of t2's characters in t1, and the geometric average of the two percentages respectively.

Then our system use bleu as another three features [12][13]. Bleu was designed as a way to measure the quality of machine translation by comparing the n-grams in the text and its translation. [20][21] Our system treats t1 as the original text and t2 as its translation and gets the log Bleu recall, log Bleu precision, and log Bleu F measure values.

The following four features are the comparison of the sentence lengths of t1 and t2. Our system counts the difference of sentence lengths of t1 and t2 according to characters or terms, and also provides the absolute values as another two features.

These features can be treated as surface features after preprocessing. We also tested another two advanced features in our system.

3.3.1 Sub-tree mapping

The syntax of a sentence is also important issue. The dependency in a sentence [15] has been used to identify the paraphrasing relation [14]. There are several previous works suggest different way to measure the similarity between two parsed trees, such as tree editing distance [17][18][19]. Sub-tree mapping is a way to calculate the similarity between two sentences by comparing the parsing trees of two sentences. [16] We believe that tree mapping is promising; however, in our experimental results, the performance drops slightly with it.

3.3.2 Time mapping

When we observe the training data, we found that many pairs have time expressions as an important part of its meaning. Therefore, we analyzed the matching types as shown in Table 3.

Table 3. Examples of time expression matching types

Matching type	Examples
Exact match	t1：據他所知，這是查爾斯首度參加雪梨-荷芭特帆船賽，而查爾斯一向是注重安全、非常謹慎的人，他更想參加**2000年**雪梨奧運帆船賽。(Year 2000)
	t2：**2000年**奧運在雪梨舉辦(Year 2000)
Partial match (1)	t1：若望保祿二世**一九七八年十月十六日**被選為教宗(1978.10.10)
	t2：若望保祿二世於**1978年**當上教宗(1978)
Partial match (2)	t1：蘇哈托 **1921年6月8日**出生(1921.06.08)
	t2：蘇哈托（Suharto，民間常用「Soeharto」，**1921年6月8日－2008年1月27日**）(1921.06.08 to 2008.01.27)
No match	t1：張藝謀**1987年**以「紅高粱」拿下柏林影展金熊獎(1987)
	t2：柏林電影節應該是張藝謀的福地。**1988年**，他執導的《紅高粱》贏得了最佳影片金熊獎，成為中國電影的首個金熊獎(1988)

Table 4. Number of the three matching types in TC training set

	F	R	I	B	C
Exact match	9	17	3	30	17
Partial match	12	8	7	0	0
No match	1	1	2	0	7

According to Table 4, this feature is very useful to identify the Bidirectional entailment in training set. Unfortunately, it is not that useful in the test.

3.4 Support vector machine

The SVM tool used in our system is the LIBSVM [26], which can be used to train both binary class classifier and multiple class classifiers.

4. Experiment Result

In this section, we report the experiment results on training set and test set.

4.1 Result on training set

In order to find which features might give best result, several different settings of experiments were conducted. Our system is

trained and tested with 10 fold cross validation on the given 421 training pairs. The best result is shown in Table 5. In that setting, the first 10 features in our feature list are used.

Table 5. Best 10 fold cross validation result of training set

Subtask	Accuracy
CT-BC	0.7381
CT-MC	0.5368

4.2 Formal run results

The best result of our system in the formal run is shown in Table 6. In that runs, the first 10 features in our feature list is used. The performance of runs with 12 features is lower than the run with 10 features. Note that we use only the traditional Chinese training set in our training phrase, and use the same model to test both traditional Chinese test set and simplified Chinese test set.

Table 6. Best result of each run in RITE 1

Subtask	Accuracy
III_CYUT_NTHU CS-BC-02	0.683
III_CYUT_NTHU CS-MC-02	0.590
III_CYUT_NTHU CT-BC-02	0.650
III_CYUT_NTHU CT-MC-02	0.491

The following tables are the confusion matrices of our best runs for each subtask:

Table 7. The confusion matrix of CS-BC-02 run

Predicted	Actual		Total
	Y	N	
Y	172	38	210
N	91	106	197
Total	263	144	407

Table 8. The confusion matrix of CS-MC-02 run

Predicted	Actual					Total
	F	R	B	C	I	
F	61	0	5	5	10	81
R	8	76	4	11	32	131
B	8	8	50	24	6	96
C	11	0	12	34	3	60
I	13	7	0	0	19	39
Total	101	91	71	74	70	407

Table 9. The confusion matrix of CT-BC-02 run

Predicted	Actual		Total
	Y	N	
Y	294	159	453
N	156	291	447
Total	450	450	900

Table 10. The confusion matrix of CT-MC-02 run

Predicted	Actual					Total
	F	R	B	C	I	
F	98	2	29	18	34	181
R	31	147	24	43	76	321
B	17	16	95	40	23	191
C	8	1	25	60	5	99
I	26	14	7	19	42	108
Total	180	180	180	180	180	900

4.3 RITE4QA results

The best result of our system in the RITE4QA is shown in Table 11. In that runs, the first 10 features in our feature list is used. The performance of runs with 12 features is higher than the run with 10 features. Note that we use both the traditional Chinese and simplified Chinese training set in our training phrase, and use the same model to test both traditional Chinese test set and simplified Chinese test set. Therefore, the test results are exactly the same.

Table 11. Best result of RITE4QA in RITE 1

Subtask	Accuracy
RITE1-III_CYUT_NTHU-CS-RITE4QA	0.7525
RITE1-III_CYUT_NTHU-CT-RITE4QA	0.7525

Table 12. The confusion matrix of both CT and CS-RITE4QA

Predicted	Actual		Total
	Y	N	
Y	31	66	97
N	99	486	585
Total	130	552	682

5. CONCLUSIONS

This paper reports our system in the RITE1 CT-BC, CT-MC, CS-BC, and CS-MC. Although the process is almost the same, but the results in each subtasks are quite different. In the CT-BC subtask, the accuracy of our system is 0.650, which is quite high among all participant systems, while the accuracy values in other subtasks are not that well. Our system is build for simplified Chinese; therefore, it has better performance in CS runs than in CT runs. However, the differences between CS runs and CT runs are quit different. In BC subtask, the difference is 0.03, and in the MC

subtask, the difference is 0.10. Since we used only the traditional Chinese training set in our training phrase and test our system on both traditional Chinese and simplified Chinese, we can conclude that the test set of simplified Chinese is different but not very different from the test set of traditional Chinese.

According to the experience on the system developing, we find that it takes a lot of time to build background knowledge. For example, the era name of Chinese or Japanese must be converted into the same representation before time matching. Geographical knowledge is also necessary to determine the relation between two place names. The knowledge is beyond the content of any normal size training set and linguistic knowledge.

6. ACKNOWLEDGMENTS

This study was conducted under the 'Intelligent Web-Enabled Service Research and Development Project' of the Institute for Information Industry which is subsidized by the Ministry of Economic Affairs of the Republic of China. This research was partly supported by the National Science Council under NSC 100-2221-E-324 -025-MY2.

7. REFERENCES

[1] Ido Dagan and Oren Glickman, Probabilistic textual entailment: Generic applied modeling of language variability, In Proceedings of the Workshop on Learning Methods for Text Understanding and Mining, Grenoble, France,2004.

[2] NTCIR 9, Recognizing Inference in TExt task, http://artigas.lti.cs.cmu.edu/rite/Main_Page.

[3] Ion Androutsopoulos and Prodromos Malakasiotis, "A survey of paraphrasing and textual entailment methods", Journal of Artificial Intelligence Research, Volume 38, pages 135-187, 2010.

[4] Michael Collins and Nigel Duffy, " New ranking algorithms for parsing and agging: Kernels over discrete structures, and the voted perceptron", Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, 2002.

[5] Johan Bos, Katja Markert, "Recognising textual entailment with logical inference", Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, Vancouver, B.C., Canada, 2005.

[6] Marta Tatu and Dan Moldovan, "COGEX at RTE 3", In Proceedings of ACL-PASCAL Workshop on Textual Entailment and Paraphrasing, pages 22–27, Prague, Czech Republic, 2007.

[7] Marta Tatu, Dan Moldovan, "A semantic approach to recognizing textual entailment", In Proceedings of HLT/EMNLP 2005, pages 371–378,Vancouver, Canada, 2005

[8] Christiane Fellbaum, "WordNet: An Electronic Lexical Database", The MIT Press, 1998.

[9] Dan I. Moldovan and Vasile Rus, "Logic form transformation of WordNet and its applicability to question answering", In Proceedings of the 39th Annual Meeting of ACL, pages 402–409, Toulouse, France, 2001.

[10] Prodromos Malakasiotis, Ion Androutsopoulos, "Learning textual entailment using SVMs and string similarity measures", In Proceedings of ACL-PASCAL Workshop on Textual Entailment and Paraphrasing, pages 42–47, Prague, Czech Republic, 2007.

[11] Wan, S., Dras, M., Dale, R., & Paris, C., "Using dependency-based features to take the "parafarce" out of paraphrase", In Proceedings of the Australasian Language Technology Workshop, pages 131–138, Sydney,Australia, 2006.

[12] Kishore Papineni, Salim Roukos, Todd Ward and Wei-Jing Zhu, "BLEU: a method for automatic evaluation of machine translation", In Proceedings of the 40th Annual Meeting on ACL, pages 311–318, Philadelphia, PA, 2002.

[13] Liang Zhou, Chin-Yew Lin and Eduard Hovy, "Re-evaluating machine translation results with paraphrase support", In Proceedings of the Conference on EMNLP, pages 77–84, Sydney, Australia, 2006.

[14] Prodromos Malakasiotis, "Paraphrase recognition using machine learning to combine similarity measures", In Proceedings of the 47th Annual Meeting of ACL and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, Suntec, Singapore, 2009.

[15] Igor Mel'cuk, "Dependency Syntax: Theory and Practice", State Universityof New York Press, 1987.

[16] Sandra Kübler, Ryan McDonald, and Joakim Nivre, "Dependency Parsing". Synthesis Lectures on Human Language Technologies. Morgan and Claypool Publishers, 2009.

[17] Selkow, S., "The tree-to-tree editing problem", The Journal of Information Processing Letters, Volume 6, No. 6, 184–186, 1977.

[18] Kuo-Chung Tai, "The tree-to-tree correction problem", The Journal of ACM, Volume 26, No. 3, 422–433, 1979.

[19] Kaizhong Zhang and Dennis Shasha, "Simple fast algorithms for the editing distance between trees and related problems", SIAM Journal of Computing, Volume 18, No. 6, pages 1245–1262, 1989.

[20] Eduard Hovy, "Toward finely differentiated evaluation metrics for machine translation", In Proceedings of the Eagles Workshop on Standards and Evaluation, Pisa, Italy, 1999.

[21] J.S. White and T. O'Connell, "The ARPA MT evaluation methodologies: evolution, lessons, and future approaches", In Proceedings of the First Conference of the Association for Machine Translation in the Americas, pages 193–205, Columbia, Maryland, 1994.

[22] Roger Levy and Christopher Manning, "Is it harder to parse Chinese, or the Chinese Treebank?", In Proceeding of the 41st Annual Meeting on Association for Computational Linguistics, Volume 1, pages 439-446,Sapporo Convention Center, Japan, 2003.

[23] Stanford parser, http://nlp.stanford.edu/software/lex-parser.shtml

[24] ICTCLAS, http://ictclas.org/

[25] Google Translate, http://translate.google.com.tw/

[26] LIBSVM, http://www.csie.ntu.edu.tw/~cjlin/lib