Chinese Textual Entailment with Wordnet Semantic and Dependency Syntactic Analysis

Chun Tu , Min-Yuh Day

Department of Information Management, Tamkang University, Taiwan

kevincncod2@gmail.com, myday@mail.tku.edu.tw

Abstract

Recognizing Inference in TExt (RITE) is a task for automatically detecting entailment, paraphrase, and contradiction in texts which addressing major text understanding in information access research areas. In this paper, we proposed a Chinese textual entailment system using Wordnet semantic and dependency syntactic approaches in Recognizing Inference in Text (RITE) using the NTCIR-10 RITE-2 subtask datasets. Wordnet is used to recognize entailment at lexical level. Dependency syntactic approach is a tree edit distance algorithm applied on the dependency trees of both the text and the hypothesis. We thoroughly evaluate our approach using NTCIR-10 RITE-2 subtask datasets. As a result, our system achieved 73.28% on Traditional Chinese Binary-Class (BC) subtask and 74.57% on Simplified Chinese Binary-Class subtask with NTCIR-10 RITE-2 development datasets. Thorough experiments with the text fragments provided by the NTCIR-10 RITE-2 subtask showed that the proposed approach can improve system's overall accuracy.

Keywords: Textual Entailment, Semantic Features, Dependency Analysis, WordNet, Syntactic Features, Machine Learning, Support Vector Machine (SVM)

1. INTRODUCTION

Recognizing Textual Entailment (RTE) is a task in which a system is given two text fragments and then determine whether the meaning of hypothesis is entailed from another text [8]. There are two subtasks in RTE: RTE 2-way and RTE 3-way. RTE 2-way output yields two labels: "Entailment" and "No Entailment". The label "Entailment" is given when the Text entails the Hypothesis, while "No Entailment" is given when the Text does not entail the Hypothesis. RTE 3-way output gives three labels: "Entailment" Text entails Hypothesis), "Contradiction" (Text contradicts Hypothesis), and "Unknown". (relationship between Text and Hypothesis is unknown) [22].

RTE is a large European and American project. Its counterpart in East Asia is called, Recognizing Inference in Text (RITE). RITE is a generic benchmark task that addresses major text understanding needs in variety of NLP/Information Access research areas. [8] There are four subtasks in RITE: Binary-Class (BC), Multi-Class (MC), Entrance Exam and RITE4QA. In NTCIR-10 RITE-2, in addition to the four subtasks in NTCIR-9 RITE, the two new subtasks were added: Exam Search subtask and UnitTest subtask. In the Exam Search subtask, instead of a text t1, a set of documents are given to system. Systems are required to search a set of texts in the documents which entails or contradicts t2. In the UnitTest subtasks, the set of examples were developed by providing a breakdown of linguistic phenomena that are necessary for recognizing relations between t1 and t2 in the dataset for the BC subtask.[21] Also, the setting of MC subtasks was slightly changed. The MC Subtask is defined as "A 4-way labeling subtask to detect (forward / bi-directional) entailment or no entailment (contradiction / independence) in a text pair", the expected system output label of RITE MC subtask is "{F,B,C,I}", where F means "forward entailment (t1 entails t2 AND t2 does not entails t1)"; B means "bidirectional entailment (t1 entails t2 AND t2 entails t1)"; C means "contradiction (t1 and t2 contradicts, or cannot be true at the same time)"; I means "independence (otherwise)".[21]

Generally, features used for dealing with TE can be roughly divided into two categories, syntactic features and semantic features. Semantic features include synonyms, antonyms, and negation. In large, semantic features and syntactic features are comprehensively discussed in most studies.

Therefore, WordNet and Dependency Parser are used in this paper. WordNet can be used as a lexical ontology and to find possible forms of the word and synonyms; Dependency Parser can be used to work out the grammatical structure of sentences.

The remainder of this paper is organized as follows. Section 2 describes the literature on RTE, RITE and machine learning. Section 3 details our system framework and the features we adopted. Section 4 shows the experimental setup and the evaluation of our approach. Finally, Section 5 presents our conclusions.

IEEE IRI 2013, August 14-16, 2013, San Francisco, California, USA
978-1-4799-1050-2/13/$31.00 ©2013 IEEE

2. LITERATURE REVIEW

In this section, we provide the research background on RTE, RITE and machine learning with related approaches to this problem. We then lay the foundation for our proposed approach by reviewing the literature on the use of these approaches.

2.1. Recognizing Textual Entailment

RTE mainly uses two sets of features, semantic features and syntactic features. Siblini and Kosseim [15] proposed a Ontology Alignment System (OAS) which adopted syntactic features and semantic features with ontology alignment and acquisition to deal with Text and Hypothesis, respectively. However, the application of OAS is limited by cognitive differences the in text fragments. For example, "bank" might have different meanings in different contexts. In terms of finance, "bank" means "銀行" while in terms of ecology, "bank" means "河岸", which may result in problems when dealing with semantic features.

Burchardt et al. [1] proposed the SALSA system which adopted semantic features for inference analysis of text fragments. They offered a match graph for synonym words. They used 47 features to calculate the similarity in each graph for training. However, this approach encounters the same problem as that of Siblini and Kosseim above: when a word appears in different contexts, it may have different meanings. Bias would thus occur when training with these datasets.

Vanderwende et al. [11], concluded that nearly 48% of text fragments could be inferred merely by syntactic features plus a general-purpose thesaurus. Castillo [9] proposed an approach using Edit Distance and Longest Common Substring (LCS) to recognize the inference of text fragments. Kouylekov and Magnini [12] proposed a Tree Edit Distance approach to analyze the similarity of text fragments.

In sum, compared to Chinese, when we process English text fragments, each word is split explicitly in an English sentence and much information is carried by the use of auxiliaries and by verb inflections. Chinese, on the other hand, is an uninflected language and conveys meaning through word order, adverbials or shared understanding of the context.

2.2. Recognizing Inference in Text

An issue for RITE is that Chinese and Japanese are relatively more complicated than English for text inference. Therefore, understanding the subtle differences in Chinese and Japanese is harder. In Japanese subtasks, Yamana et al. [5] proposed normalized predicate-argument structures for two texts if and only if structure of t1 entails that of t2 based on this structure. Hattori et al. [17] proposed a two-step classification strategy by first assigning a default class to a given text pair by applying a simple rule based on an overlap measure and then examines the necessity of overwriting the default class by applying heuristic rules. Tian et al. [16] used Dependency-based compositional semantics (DCS) which originally used as a natural language interface for database queries. They developed a new framework which eliminated some restrictions of DCS and translated a tree presentation of natural language expression into some algebraic forms to explore the relations among these forms. In Chinese subtasks, some terms may have slightly different meanings when rendered in Simplified Chinese and Traditional Chinese. Shih et al. [2] proposed a two-stage (Entailment recognition and Contradiction recognition) knowledge-based textual inference recognition system for both BC and MC subtasks in Chinese. Wang et al. [20] proposed a labeled-alignment-based RTE method which contains labeled alignment with negative links to explicitly mark the contradictory expressions between the two sentences to justify the non-entailment pairs.

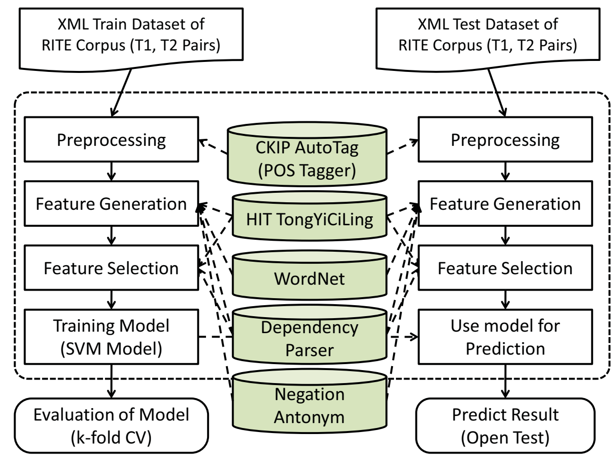

Figure 1. System Architecture of IMTKU Text Entailment System in NTCIR-10 RITE2

Text fragments in Chinese or Japanese are slightly different from English text fragments. For example, "haste" in Chinese has several meanings: "迅速","急躁". The former word has a positive meaning while the latter word has a negative meaning. Therefore, Chinese might encounter one-to-many ambiguity problem.

2.3. Machine Learning

Malakasiotis and Androutsopoulos [13] proposed a Support Vector Machine (SVM) approach with semantic features to tackle text fragments and train a model which contains 128 features in order to increase its accuracy.

In sum, considering SVM as machine learning tool [3] which can solve classification and clustering problem within feasible time limits as well as select the best feature combinations to enhance model accuracy and model efficiency.

3. SYSTEM ARCHITECTURE

We developed a textual entailment system using a hybrid approach that integrates syntactic features, semantic features and machine learning techniques for recognizing inference in text in an NTCIR-10 RITE2 task. Figure 1 shows the proposed system architecture of the IMTKU Textual Entailment System for Recognizing Inference in Text in an NTCIR-10 RITE2 task.

3.1. Preprocessing

We extracted text fragments from NTCIR-9 RITE RITE4QA raw datasets and use CKIP Autotag[24] for producing available datasets.

3.1.1. Data format unification

A word may be expressed in different ways. For example, 1990 may be written "1990 年" or "一九九零年". It is thus necessary to unify the data format. [19]

3.1.2. CKIP Autotag

We adopt the Chinese Knowledge and Information Processing (CKIP) System to process text pairs for analysis.

3.2. Feature Generation

We designated 20 semantic and syntactic features:

Word Similarity, String Length, String Length Difference, String Length Ratio, Longest Common Substring (LCS), Char-Based Edit Distance, Word Length, Word Length Difference, Word Length Ratio, Word-Based Edit Distance, Dependency parser, WordNet, Negation, Antonym.

(1) T1/T2 String Length/Length Difference/Ratio

Basic syntactic approach we adopted as a feature. We use string length difference and ratio as a feature to reduce bias on a length basis. [4]

(2) Longest Common Substring

We use Longest Common Substring [6] to find similarity in text pairs. Find the longest string (or strings) that is a substring (or are substrings) of two or more strings.

(3) Char-based Edit Distance

Edit Distance is a distance in which insertions and deletions have equal cost and replacements have twice the cost of an insertion. It is thus the minimum number of edits needed to transform one string into the other, with the allowable edit operations being insertion, deletion, or substitution of a single character.

(4) T1/T2 Word Length/Difference/Ratio

We use CKIP Autotag to tokenize sentences into every word and calculate the total words numbers, ratio. [4]

Table 1 Syntactic and Semantic Features

Feature ID	Feature
Feature01	T1 String Length
Feature02	T2 String Length
Feature03	String Length Difference
Feature04	String Length Ratio
Feature05	Longest Common Substring
Feature06	Char-Based Edit Distance
Feature07	T1 Word Length
Feature08	T2 Word Length
Feature09	Word Length Difference
Feature10	Word Length Ratio
Feature11	Word-Based Edit Distance
Feature12	Noun Number
Feature13	Verb Number
Feature14	Word Semantic Similarity
Feature15	WordNet Similarity
Feature16	WordNet Similarity Ratio
Feature17	WordNet Similarity Minimum
Feature18	Negation Number
Feature19	Antonym Number
Feature20	Dependency Parser Tree Edit Distance

(5) Word-based Edit Distance

Edit Distance is to measure distance as the number of operations required to transform a string into another where this feature is token-based. [4]

(6) Noun/Verb Number

We incorporated a feature which calculates noun/verb numbers in a sentence, so we could do a simple comparison in advance. [4]

(7) Word Semantic (Synonym) Similarity

We proposed a semantic feature that redesigned HIT TYCCL where each word in the TYCCL is assigned an ID and words with same ID are considered synonyms and with a conversion formula.[4]

(8) WordNet Similarity/Ratio/Minimum

In WordNet, nouns, verbs, adjectives, and adverbs are organized into sets of synonyms, each representing a lexicalized concept. [7] We first searched each CKIP token in the WordNet corpus. Once found, we got its Synset. Synonym words share same Synset ID. If two sentences have more Synset ID in common, the more similar these two sentences are. In other words, these two sentences have a higher similarity.

For instance, we want to calculate the similarity between the word 人(person) and 人類(human being). We first list their Synset ID:

人: 00002086, **07192170**, **05957670**, 01967203, 05961082, 00004123, 07392506, 06126536, **03716629**, 07469674, **05957883**

Total Count: 11

人類: **07192170**, **05957670**, 06079949, **03716629**, **05957883**
Total Count: 5

4 Synset IDs out of 11 are matched. We calculate their similarity: 4/11=0.364.

We also designed different kinds of similarity calculation, such as changing the denominators in order to obverse which calculation can make better performance.

(9) Negation

We proposed a feature which integrated negation words into a total of 52 negation words list. For instance:
沒, 不, 否, 無, 非, 未, 免, 別, 莫..

These are single character negation words. Different combination with words will change their meanings to opposite. We also detect negation words with two characters. For instance:
沒有, 無法, 尚未, 未可, 未得..

We first detected the negation words number of each text pair. By comparing negation words number to determine whether each text pair is opposite or similar.

(10) Antonym

We proposed a feature which integrated antonym words into a total of 568-antonym-pair list. For instance:

開心-傷心, 開心-難過, 快樂-難過...

Each word might correspond to several antonym words, vice versa. Hence, we list down common antonym word pairs and to detect if one appears in text pairs, we could determine if words appeared in the text pair are antonym words or not.

(11) Dependency Parser

Dependency parses give information about grammatical relations between words, instead of constituency information. It can also capture syntactic relations. [14]

We proposed a feature which adopted Stanford Parser to do sentence dependency parsing. In prior research, we found that tree edit distance was common in most dependency parser features.Tree Edit Distance is which the minimum number of edits needed to transform one sentence tree structure into the other , with the allowable edit operations being insertion, deletion, or substitution of a single character. For instance:

T1: 一九九七年香港回歸中國。
(Hong Kong was returned to China in 1997)
T2: 香港的主權和領土是在一九九七年由英國歸還給中國的。
(Hong Kong's sovereignty and territories were returned to China by the United Kingdom in 1997)

We used Stanford Parser to parse this text pair into

T1: (ROOT (IP (NP (NT 一九九七年)) (NP (NR 香港)) (VP (VV 回歸) (NP (NN 中國)))))
T2: (ROOT (IP (NP (DNP (NP (NR 香港)) (DEG 的)) (NP (NN 主權) (CC 和) (NN 領土))) (VP (VC 是) (NP (CP (IP (VP (PP (P 在) (NP (NT 一九九七))) (PP (P 由) (NP (NN 英國))) (VP (VV 歸還給) (NP (NN 中國))))) (DEC 的)))) (PU 。)))

In order to obverse the similarity of these trees, we calculated the tree edit distances between two texts which are 22. Since these trees need at least 22 deletions, insertions or substitutions of a term, basically, we can conclude that the similarity between two trees is low.

This feature can help the system simply calculate the similarity of each text pair on a syntactic basis in advance.

3.3 Machine Learning

We used LibSVM as the machine learning module. LibSVM provides two tools for enhancing model accuracy: grid.py and fselect.py. These two tools select the best parameters and best features for the model.

4. EXPERIMENTAL RESULTS AND ANALYSIS

We use the RITE-2 CT BC/MC Development sets of 1321 training pairs and CT BC/MC Test sets of 881 test pairs, CS BC/MC Development set of 814 training pairs and CS BC/MC Test set of 781 test pairs, provided by NTCIR-10 RITE-2 for model training and prediction. Table 1 shows the syntactic and semantic features list.

Table 2 and 3 shows that config 2 outperformed other configs. Since dependency parser is used to calculate the edit distance in this paper, some sentences' structures which are not properly parsed, or, sentences' structures are different, but the edit distances are nearly identical can also result in biases during calculation.

Table 4 shows that config 9 got better results than others. This config consists of length, edit distance, word length ratio, and tree edit distance. It is likely to get different sentences structures when parser deals with different kinds of languages.

Table 5 shows that all 3 configs does not perform well as expected. The main cause is that due to lack of knowledge-based semantic features and name entity recognition, such MC subtask relations might not be correctly labeled.

Table 2 Cross Validation and Open Test results after features selection (CT BC subtask)

Config	Feature	Cross Validation	Open Test
Config1	Feature1~20	72.14%	65.95%
Config2	Feature 1~19	**73.28%**	**67.65%**
Config3	Feature 2,5,10,20	73.13%	64.13%

Table 3 Cross Validation and Open Test results after features selection (CT MC subtask)

Config	Feature	Cross Validation	Open Test
Config4	Feature1~20	41.48%	55.62%
Config5	Feature 1~19	41.94%	**56.41%**
Config6	Feature2,5,10,20	**49.21%**	40.86%

Table 4 Cross Validation and Open Test results after features selection (CS BC subtask)

Config	Feature	Cross Validation	Open Test
Config7	Feature1~20	73.22%	60.82%
Config8	Feature 1~19	73.96%	60.95%
Config9	Feature2,5,10,20	**74.57%**	**62.7%**

Table 5 Cross Validation and Open Test results after features selection (CS MC subtask)

Config	Feature	Cross Validation	Open Test
Config10	Feature1~20	45.58%	**49.3%**
Config11	Feature1~19	45.82%	45.45%
Config12	Feature2,5,10,20	**48.65%**	40.2%

Table 6 shows 10-fold cross validation model accuracy. We use 4 different kinds of datasets combination using development and test datasets. Figure 2 shows that the best model performance is Model 2, with 73.83% cross validation accuracy.

It is necessary to select appropriate parameters in training. We used grid.py to select the best parameters because different parameters influence model accuracy. Different feature combinations will also result in different models. We adopted fselect.py to select the best feature combinations in order to enhance model accuracy.

4.1 ERROR ANALYSIS

The goal in this experiment is to identify the earliest module that prevents the system to find the right answer, i.e. causes the error. In the following text pairs reveal the system errors:

Original Label: **I** System Prediction Label: **B**

T1:「**代謝症候群**」和國人十大死亡原因中之四項，包括腦血管疾病(中風)、心臟病、糖尿病和高血壓息息相關
(“**Metabolism Syndrome**” is relevant to four causes from top ten leading causes of death including cerebrovascular disease, heart disease and high blood pressure.)

T2: 國人十大死亡原因中的腦血管疾病、心臟病、糖尿病及高血壓都與**膽固醇**有關

(**Cholesterol** is relevant to four causes from top ten leading causes of death including cerebrovascular disease, heart disease and high blood pressure.)

Table 6 IMTKU Experiments for NTCIR-10 RITE-2 Datasets

Model	Datasets	10 Fold CV Accuracy
Model 1	RITE2 BC Dev+Test Dataset: 1321+881=2202 pairs	68.85%
Model 2	Random select 1000 pairs from RITE1 BC Dev+Test Dataset	**73.83%**
Model 3	RITE1 BC Dev+Test Dataset: 421+900=1321 pairs	72.29%
Model 4	RITE1 BC Development Dataset: 421 pairs	72.21%

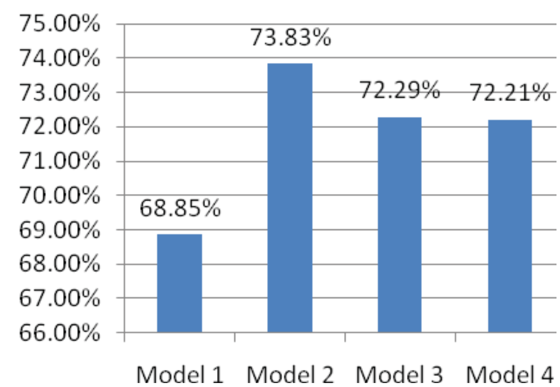

Figure 2. 10-fold cross validation for NTCIR-10 RITE-2 datasets

In dependency parser, the similarity between two sentences and structures are high that misleads the system. System lacks knowledge-based feature to identify the grammatical meaning such as "A is relevant to B".

Original Label: **I** System Prediction Label: **F**

T1: **如果**卡翠娜颶風過境導致石油生產與供應出問題，對美國經濟的衝擊會比過去兩年來得嚴重
(The economic impact will be worse than the past two years **if** hurricane Katrina strikes and leads to oil production and supplement problem.)
T2: 卡翠娜颶風對美國整體經濟的衝擊相當嚴重
(The overall economic impact of hurricane Katrina was seriously affected.)

This system error is due to grammatical structure where the system could not identify If-clause in the sentence. This problem also lack of knowledge-based feature.

Original Label: **I** System Prediction Label: **B**

T1: 流感病毒可在人體外存活三到六小時
(**Flu virus** can live outside human body from 3 to 6 hours.)
T2: 冠狀病毒通常可在人體外存活二到三小時
(**Coronavirus** can live outside human body from 2 to 3 hours.)

These two sentence structures are nearly identical, but due to lack of Name Entity Recognition, treating the first words the same.

We found that in these cases, knowledge-base and NER features influence the most. Hence, we expect to add these features into the system to enhance the overall model accuracy.

5. CONCLUSIONS

In this paper, we propose a novel system using both semantic and syntactic features for performing a RITE-2 BC and MC subtasks. The results showed that 67.65% on Traditional Chinese Binary-Class (BC) subtask, 56.41% on Multi-Class (MC), 62.7% on Simplified Chinese Binary-Class subtask, and 49.3% on Multi-Class with NTCIR-10 RITE-2.

The contributions of this paper include:

(1) Dependency parser tree edit distances have been influenced by different language contexts (Traditional Chinese and Simplified Chinese) to some extent.

(2) We thoroughly evaluate our approach in the context of the subtasks of the NTCIR-10 RITE-2. The results of our system attest the effectiveness of the approaches we propose for the NTICR-10 RITE-2 and its subtasks.

6. ACKNOWLEDGEMENT

This research was supported in part by the National Science Council of Taiwan under Grants NSC 101-3113-P-032-001 and TKU research grant. We would like to thank the support of IASL, IIS, Academia Sinica, Taiwan.

7. REFERENCES

[1] A. Burchardt, N. Reiter, S. Thater and A. Frank, "A semantic approach to textual entailment: System evaluation and task analysis.," presented at the Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing, Prague, Czech Republic, 2007.

[2] C.-W. Shih, C. Liu and C.-W. Lee, "IASL RITE System at NTCIR-10", Proceeding of the 10th NTCIR Conference, Tokyo, Japan., June 18-21, 2013.

[3] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol* vol. 2, pp. 1-27, 2011.

[4] C. Tu, M.-Y. Day, "A statistical approach with syntactic and semantic features for Chinese Textual Entailment", Proceeding of Information Reuse and Integration (IRI), 2012

[5] D. Ito, M. Tanaka and H. Yamana, *et al.*, "WSD Team's Approaches for Textual Entailment Recognition at the NTCIR10 (RITE2)", Proceeding of the 10th NTCIR Conference, Tokyo, Japan., June 18-21, 2013,

[6] D. S. Hirschberg, "Algorithms for the Longest Common Subsequence Problem," *Journal of the Assocrauon for Computing Machinery,* vol. 24:4, **pp.** 664-675, 1997.

[7] G. A. Miller, "WordNet: A lexical database for English", Communications *of the ACM*, Volume 38 Issue 11, Nov. 1995 Pages 39-41.

[8] H. Shima, H. Kanayama, C.-W. Lee, C.-J. Lin, T. Mitamura, Y. Miyao, S. Shi and K. Takeda., "Overview of NTCIR-9 RITE: Recognizing Inference in TExt," presented at the Proceedings of NTCIR-8 Workshop Meeting, Tokyo, Japan, 2011.

[9] J. Castillo, "A Machine Learning Approach for Recognizing Textual Entailment in Spainish," 2010.

[10] J.-J. Mei, Y.-M. Zhu, Y.-Q. Gao and H.-X Yin, *TongYiCi CiLin (Chinese Synonym Forest): Shanghai Press of Lexicon and Books*, 1983.

[11] L. Vanderwende. and W. B. Dolan , "What Syntax can Contribute in Entailment Task," *Microsoft Research,* 2006.

[12] M. Kouylekov and B. Magnini, "Recognizing textual entailment with tree edit distance algorithms.," presented at the Proceedings of the PASCAL Recognizing Textual Entailment Challenge, 2005.

[13] P. Malakasiotis and I. Androutsopoulos, "Learning textual entailment using SVMs and string similarity measures.," presented at the Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing, Prague. ,2007.

[14] P. C. Chang, H. Tseng, D. Jurafsky, and C. D. Manning, "Discriminative Reordering with Chinese Grammatical Relations Features", *In Proceedings of the Third Workshop on Syntax and Structure in Statistical Translation*, 2009.

[15] R. Siblini and L. Kosseim, "Using Ontology Alignment for TAC RTE Challenge," presented at the Proceedings of the Text Analysis Conference, Gaithersburg, MD, 2008.

[16] R. Tian and Y. Miyao, "BnO at NTCIR-10 RITE: A Strong Shallow Approach and an Inference-based Textual Entailment Recognition System", Proceeding of the 10th NTCIR Conference, Tokyo, Japan., June 18-21, 2013.

[17] S. Hattori and S. Satoshi, "Team SKL's Strategy and Experience in RITE2", Proceeding of the 10th NTCIR Conference, Tokyo, Japan., June 18-21, 2013.

[18] W.-J. Huang and C.-L. Liu, "NCCU-MIG at NTCIR-10: Using Lexical, Syntactic, and Semantic Features for the RITE Tasks", Proceeding of the 10th NTCIR Conference, Tokyo, Japan., June 18-21, 2013.

[19] W.-C. Huang and S.-H. Wu, "Feature Analysis of Chinese Textural Entailment Systems," presented at the Proceedings of the 23rd Conference on Computational Linguistics and Speech Processing 2011.

[20] X.-L. Wang*., H. Zhao and B.-L. Liu,* "BCMI-NLP Labeled-Alignment-Based Entailment System for NTCIR-10 RITE-2 Task", Proceeding of the 10th NTCIR Conference, Tokyo, Japan., June 18-21, 2013.

[21] Y. Watanabe and Y. Miyao and J. Mizuno and T. Shibata, H. Kanayama, C. -W. Lee, C. -J. Lin and K. Takeda, "Overview of Recognizing Inference in TExt(RITE-2) at the NTCIR-10 Workshop," in Proceedings of NTCIR-10 Workshop Meeting, Tokyo, Japan, 2013.

[22] (2011.10.27). *Text Analysis Conference*. Available: http://www.nist.gov/tac/2010/RTE/RTE6_Main_NoveltyDetection_Task_Guidelines.pdf

[23] (2013). *NTCIR-10 RITE-2*. Available: http://www.cl.ecei.tohoku.ac.jp/rite2/doku.php?id=start

[24] (2011.10.27). *CKIP AutoTag*. Available: http://ckipsvr.iis.sinica.edu.tw/