Binary-class and Multi-class based Textual Entailment System

Partha Pakray[1], Sivaji Bandyopadhyay[1], Alexander Gelbukh[2]
[1] Computer Science and Engineering Department,
Jadavpur University, Kolkata, India
[2] Center for Computing Research, National Polytechnic Institute,
Mexico City, Mexico
parthapakray@gmail.com, sbandyopadhyay@cse.jdvu.ac.in
gelbukh@gelbukh.com

Abstract

The article presents the experiments carried out as part of the participation in Recognizing Inference in TExt (RITE-2)[1] @NTCIR-10 for Japanese. RITE-2 has four subtasks Binary-class (BC) subtask for Japanese and Chinese, Multi-class (MC) subtask for Japanese and Chinese, Entrance Exam for Japanese and RITE4QA for Chinese. We have submitted three runs in BC subtask for Japanese (JA) (one run), Chinese Simplified (CS) (one run) and Chinese Traditional (CT) (one run). Three runs have been submitted in MC Subtask, one run for each language. We have developed Textual Entailment system which is based on Machine Translation using the web based Google translator system[2]. The system is based on the Support Vector Machine that uses features from lexical similarity, lexical distance, and syntactic similarity.

1 Introduction

Recognizing Textual Entailment (RTE) is one of the recent challenges of Natural Language Processing (NLP). Textual Entailment has many applications in Natural Language Processing (NLP) tasks. For example, in Summarization (SUM), a summary should be entailed by the text; Paraphrases (PP) can be seen as mutual entailment between a text T and a hypothesis H; in Information Extraction (IE), the extracted information should also be entailed by the text; in Question Answering (QA) the answer obtained for one question after the Information Retrieval (IR) process must be entailed by the supporting snippet of text.

There were seven Recognizing Textual Entailment competitions RTE-1 challenge (Dagan et al., 2005) in 2005, RTE-2 challenge (Bar-Haim et al., 2006) in 2006, RTE-3 challenge (Giampiccolo et al., 2007) in 2007, RTE-4 challenge (Giampiccol et al., 2008) in 2008, RTE-5 (Bentivogli et al., 2009) challenge in 2009, RTE-6 challenge (Bentivogli et al., 2010) in 2010 and RTE-7 challenge in 2011. Textual Entailment track was Parser Training and Evaluation using Textual Entailment (Yuret et al., 2010) as part of SemEval-2.

We have participated in TAC RTE-5 (Pakray et al., 2009), TAC RTE-6 Challenge (Pakray et al., 2010a), TAC RTE-7 Challenge, SemEval-2 Parser Training and Evaluation using Textual Entailment Task, RITE (Pakray et al., 2011) in NTCIR-9 and RITE-2 in NTCIR-10 (Watanabe et al., 2013).

Section 2 describes the System Architecture using web based Machine Translation. Section 3 describes Binary Class Identification. Section 4 details Multiclass Class Identification. The experiments carried out on test data sets are discussed in Section 6 along with the results. The conclusions are drawn in Section 7.

2 System Architecture: Using web based Machine Translation

The various components of the textual entailment recognition system are pre-processing module, Lexical Textual Entailment module, Syntactic Textual Entailment, Support Vector Machine and Entailment Decision module. The system architecture has shown in Figure 1. The system is a combination of different rules working on various lexical knowledge sources, lexical distance, and syntactic similarity. The system computes the entailment decision using the outcome from the each of these rules.

[1] http://www.cl.ecei.tohoku.ac.jp/rite2/doku.php
[2] http://translate.google.com/

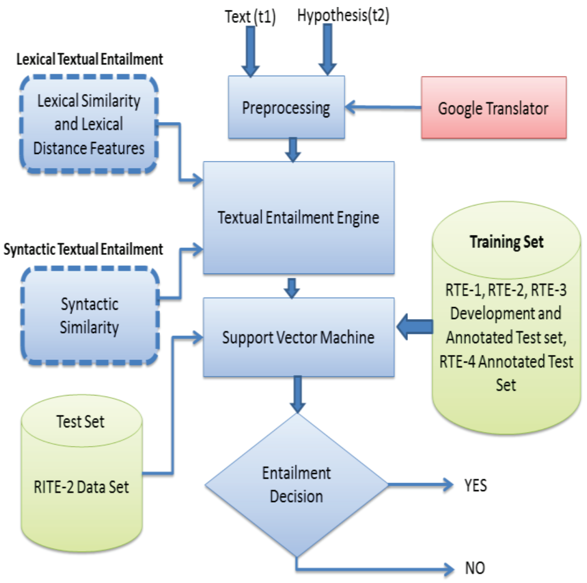

Text (t1)
Hypothesis(t2)
Lexical Textual Entailment
Lexical Similarity and Lexical Distance Features
Preprocessing
Google Translator
Textual Entailment Engine
Syntactic Textual Entailment
Syntactic Similarity
Training Set
RTE-1, RTE-2, RTE-3 Development and Annotated Test set, RTE-4 Annotated Test Set
Support Vector Machine
Test Set
RITE-2 Data Set
Entailment Decision
YES
NO

Figure 1: System Architecture BC Task

2.1 Pre-processing Module

The system accepts pairs of text snippets (t1 and t2) at the input and gives a Boolean value at the output: "Y" if the t1 entails the t2 and "N" otherwise. An example t1-t2 pair from the RITE BC development set is shown in Figure 2.

```
<dataset type="bc">
 <pair id="2" label="Y">
   <t1>伊坂幸太郎は直木賞候補になった2003 年の『重力ピエロ』で一般読者に広く認知されるようになった。</t1>
   <t2>『重力ピエロ』は伊坂幸太郎による小説で直木賞候補作品だった。</t2>
  </pair>
```

Figure 2: RITE BC Task Test Data

At first we have identified the t1 and t2 text segments in Japanese. Then the Japanese (t1, t2) gets converted to English (t1, t2) using the Japanese – English Google Translator.

2.2 Lexical Textual Entailment Methods

In this section the various lexical based TE methods (Pakray et al., 2009) are described in detail.

i. **WordNet based Unigram Match:** In this method, the various unigrams in the hypothesis (t2) for each text (t1)-hypothesis (t2) pair are checked for their presence in the text. WordNet synsets are identified for each of the unmatched unigrams in the hypothesis. If any synset for the hypothesis unigram matches with any synset of a word in the text then the hypothesis unigram is considered as a WordNet based unigram match.

If n1= common unigram or WordNet Synonyms be tween text and hypothesis and n2= number of unigram in Hypothesis, i.e. *Wordnet_Unigram_Match=n1/n2*. If the value of Wordnet_Unigram_Match is 0.75 or more, i.e., 75% or more unigrams in the hypothesis match either directly or through WordNet synonyms, then the text-hypothesis pair is considered as an entailment. The text-hypothesis pair is then assigned the value of 1 meaning entailment (Y); otherwise, the pair is assigned a value of 0 meaning entailment (N). The cut-off value for the Wordnet_Unigram_Match is based on experiments carried out on the RITE BC task development set.

ii. **Bigram Match:** Each bigram in the hypothesis is searched for a match in the corresponding text part. The measure Bigram_Match is calculated as the fraction of the hypothesis bigrams that match in the corresponding text, i.e., *Bigram_Match= (Total number of matched bigrams in a text-hypothesis pair / Number of hypothesis bigrams)*. If the value of Bigram_Match is 0.5 or more, i.e., 50% or more bigrams in the hypothesis match in the corresponding text, then the text-hypothesis pair is considered as an entailment. The text-hypothesis pair is then assigned the value of 1 meaning entailment (Y); otherwise, the pair is assigned a value of 0 meaning entailment (N). The cut-off value for the Bigram_Match is based on experiments carried out on the RITE BC task development set.

iii. **Longest Common Subsequence (LCS):** The Longest Common Subsequence of a text-hypothesis pair is the longest sequence of words which is common to both the text and hypothesis. LCS (T, H) estimates the similarity between text T and hypothesis H, as *LCS_Match= LCS (T, H)/ length of H*. If the value of LCS_Match is 0.8 or more, i.e., the length of the longest common subsequence between text T and hypothesis H is 80% or more of the length of the hypothesis, then the text-hypothesis pair is considered as an entailment. The text-hypothesis pair is then assigned the value of 1 meaning entailment (Y); otherwise, the pair is assigned a value of 0 meaning entailment (N). The cut-off value for the LCS_Match is based on experiments carried out on the RITE BC task development set.

iv. **Skip-grams:** A skip-gram is any combination of n words in the order as they appear in a sentence, allowing arbitrary gaps. In the present work, only 1-skip-bigrams are considered where 1-skip-bigrams are bigrams with one word gap between two subsequent words in a sentence. The measure 1-skip_bigram_Match is defined as *1_skip_bigram_Match* = *skip_gram*(*T,H*) / n, where *skip_gram*(*T,H*) refers to the number of common 1-skip-bigrams (pair of words in sentence order with one word gap) found in T and H and *n* is the number of 1-skip-bigrams in the hypothesis H. If the value of 1_skip_bigram_Match is 0.5 or more, then the text-hypothesis pair is considered as an entailment. The text-hypothesis pair is then assigned the value of 1 meaning entailment (Y); otherwise, the pair is assigned a value of 0 meaning entailment (N). The cut-off value for the skip_gram is based on experiments carried out on the RITE BC task development set.

v. **Stemming**: Stemming is the process of reducing terms to their root form. For example, the plural forms of a noun such as ‘boxes’ are transformed into ‘box’. Derivational endings with ‘ing’, ‘es’, ‘s’ and ‘ed’ are removed from verbs. Each word in the text and hypothesis pair is stemmed using the stemming function provided along with the WordNet 2.0. If s1= number of common stemmed unigrams between text and hypothesis and s2= number of stemmed unigrams in Hypothesis, then the measure Stemming_match is defined as *Stemming_Match=s1/s2*. If the value of Stemming_Match is 0.7 or more, i.e., 70% or more stemmed unigrams in the hypothesis match in the stemmed text, then the text-hypothesis pair is considered as an entailment. The text-hypothesis pair is assigned the value of 1 meaning entailment; otherwise, the pair is assigned a value of 0. The cut-off value for the Stemming_Match is based on experiments carried out on the RITE BC task development set.

vi. **Named Entity Match**: It is based on the detection and matching of Named Entities (NEs) in the text-hypothesis pair. Once the NEs of the hypothesis and the text have been detected, the next step is to determine the number of NEs in the hypothesis that match in the corresponding text. The measure NE_Match is defined as NE_Match=number of common NEs between text and hypothesis/Number of NE in Hypothesis. If the value of NE_Match is 0.5 or more, i.e., 50% or more NEs in the hypothesis match in the text, then the text-hypothesis pair is considered as an entailment. The text-hypothesis pair is assigned the value of 1 meaning entailment; otherwise, the pair is assigned a value of 0. The cut-off value for the NE_Match is based on experiments carried out on the RITE BC task development set.

WordNet (Fellbaum, 1998) is one of most important resource for lexical analysis. The WordNet 2.0 has been used for WordNet based unigram match and stemming step. API for WordNet Searching (JAWS) [3] provides Java applications with the ability to retrieve data from the WordNet database.

2.3 Lexical Distance

The textual entailment system for Lexical distance measurement uses the SimMetrics[4] Tool. The important lexical distance features[5] are as follows:

i. Vector Based Measured: Vector based measures are described for vector based model i.e. Block distance, Consine similarity, Dice similarity.

a. Block distance: This is a vector based approach so where ‘x’ and ‘y’ are defined in n-dimensional vector space The L or block distance[6] is calculated from summing the edge distances.

$$L(x,y)=\sum_{p}|x(p)-y(p)|$$

b. Cosine similarity: Cosine similarity[7] is a common vector based similarity measure. The cosine similarity is often paired with other approaches to limit the dimensionality of the problem. For instance with simple strings at list of stop words are used to exclude from the dimensionality of the comparison. In theory this problem has as many dimensions as terms exist.

c. Dice similarity: Dice coefficient[8] is a term based similarity measure (value between 0-1) whereby the similarity measure is defined as twice the number of terms common to the compared entities divided by the total number of terms in both

[3] http://lyle.smu.edu/~tspell/jaws/index.html

[4] http://sourceforge.net/projects/simmetrics/

[5] http://en.wikipedia.org/wiki/String_metric

[6] http://en.wikipedia.org/wiki/Block_distance

[7] http://en.wikipedia.org/wiki/Cosine_similarity

[8] http://en.wikipedia.org/wiki/Dice%27s_coefficient

tested entities. The Dice coefficient result of 1 indicates identical vectors whereas a 0 value signifies orthogonal vectors.

ii. Set-based similarities: Those text similarity functions are based on a set representation of the texts where set elements are words. Different set-based resemblance coefficients are used to obtain a similarity score between 0 and 1, some of them are:

i. $$Dice(A,B) = \frac{2|A \cap B|}{|A| + |B|}$$

ii. $$Jaccard(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

iii. $$overlap(A,B) = \frac{|A \cap B|}{\min(|A|,|B|)}$$

iv. $$\cos ine(A,B) = \frac{|A \cap B|}{\sqrt{|A| \cdot |B|}}$$

v. $$harmonic(A,B) = \frac{|A \cap B| \cdot (|A| + |B|)}{2 \cdot |A| \cdot |B|}$$

2.4 Syntactic Textual Entailment

In this section the syntactic based TE methods (Pakray et al., 2010b) are described in detail. The dependency relations are identified by the Stanford Parser[9] for each text and the hypothesis pair. The hypothesis relations are then compared with the text relations. The different features that are compared are noted below. In all the comparisons, a matching score of 1 is considered when the complete dependency relation along with all of its arguments matches in both the text and the hypothesis. In case of a partial match for a dependency relation, the matching process continues to the next relation in order.

a. Subject-Verb Comparison: The system compares hypothesis subject and verb with text subject and verb that are identified through the nsubj and nsubjpass dependency relations for Stanford parser. A matching score of 0.5 is assigned in case of a complete match. If match not fount then the system considers the following matching process i.e. WordNet Based Subject-Verb Comparison.

b. WordNet Based Subject-Verb Comparison: If the corresponding hypothesis and text subjects do match in the subject-verb comparison, but the verbs do not match, then the WordNet distance between the hypothesis and the text is compared. If the value of the WordNet distance is less than 0.5, indicating a closeness of the corresponding verbs, then a match is considered and a matching score of 0.5 is assigned. Otherwise, the subject-subject comparison process is applied.

c. Subject-Subject Comparison: The system compares hypothesis subject with text subject. If a match is found, a score of 0.5 is assigned to the match.

d. Object-Verb Comparison: The system compares hypothesis object and verb with text object and verb that are identified through dobj dependency relation. In case of a match, a matching score of 0.5 is assigned. If match not found then system considers the following matching process i.e. WordNet Based Object-Verb Comparison.

e. WordNet Based Object-Verb Comparison: The system compares hypothesis object with text object. If a match is found then the verb corresponding to the hypothesis object is compared with the verb corresponding to the text object. If the two verbs do not match then the WordNet distance between the two verbs is calculated. If the value of WordNet distance is below 0.5 then a matching score of 0.5 is assigned.

f. Cross Subject-Object Comparison: The system compares hypothesis subject and verb with text object and verb or hypothesis object and verb with text subject and verb. In case of a match, a matching score of 0.5 is assigned.

g. Number Comparison: The system compares numbers along with units in the hypothesis with similar numbers along with units in the text. Units are first compared and if they match then the corresponding numbers are compared. In case of a match, a matching score of 1 is assigned.

h. Noun Comparison: The system compares hypothesis noun words with text noun words that are identified through nn dependency relation. In case of a match, a matching score of 1 is assigned.

i. Prepositional Phrase Comparison: The system compares the prepositional dependency relations in the hypothesis with the corresponding relations in the text and then checks for the noun words that are arguments of the relation. In case of a match, a matching score of 1 is assigned.

[9] http://nlp.stanford.edu/software/lex-parser.shtml

j. Determiner Comparison: The system compares the determiner in the hypothesis and in the text that are identified through det relation. In case of a match, a matching score of 1 is assigned.

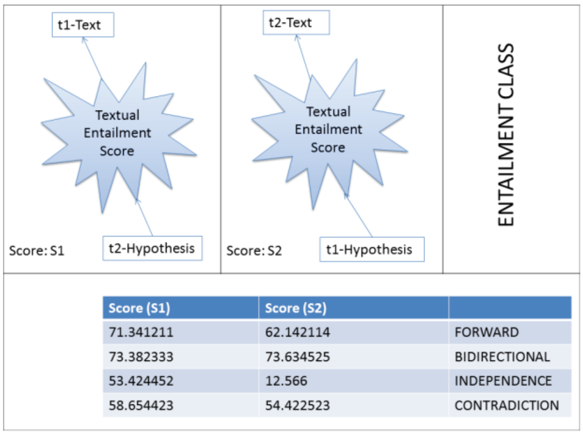
k. Other relation Comparison: Besides the above relations that are compared, all other remaining relations are compared verbatim in the hypothesis and in the text. In case of a match, a matching score of 1 is assigned.

3 Binary Class Identification

The LibSVM[10] has used to find the textual entailment relation. The system has used LIBSVM for building the model file. The TE system has used the following data sets: RTE-1 development and test set, RTE-2 development and annotated test set, RTE-3 development and annotated test set, RTE-4 annotated test set to deal with the two-way classification task for training purpose to build the model file. The LIBSVM tool is used by the SVM classifier to learn from this data set. For training purpose, 3967 text-hypothesis pairs have been used. After training the system, it has tested on the RITE-2. Finally, system gives the entailment score with entailment decisions (i.e., "*YES" / "NO*").

4 Multiclass Identification

The system finally compares the above two score S1 and S2 values as obtained from the BC Class Identification to take the four-class entailment decision. If the score S1, i.e., the mapping score with t1 as text and t2 as hypothesis is greater than the score S2, i.e., mapping score with t2 as text and t1 as hypothesis, then the entailment class will be "***forward***". Similarly if both the scores S1 and S2 are equal the entailment class will be "***bidirectional***" (entails in both directions). Measuring "bidirectional" entailment is much more difficult than any other entailment decision due to combinations of different scores. As the system produces a final score (S1 and S2) that is basically the sum over different similarity measures, the tendency of identical S1 – S2 scores will be quite small. As a result, system establishes another heuristic for "**bidirectional**" class. If the absolute value difference between S1 and S2 is below the threshold value, the system recognizes the pair as "bidirectional" *(abs (S1 – S2) < threshold)*. This threshold has been set as 5 based on observation from the training file. If the individual scores S1 and S2 fall below a certain threshold, again set based on the observation in the training file, the system concludes the entailment class as "***independence***". If S1 is less than S2, i.e., T2 now acts as the text and T1 acts as the hypothesis then the entailment class will be "***contradiction***". This threshold has been set as 20 based on observation from the training file. An example has shown in Figure 3.



Score (S1)	Score (S2)	
71.341211	62.142114	FORWARD
73.382333	73.634525	BIDIRECTIONAL
53.424452	12.566	INDEPENDENCE
58.654423	54.422523	CONTRADICTION

Figure 3: Output of Entailment Class

5 Experiments and Results

For BC Task: The RITE-2 BC task result has shown in Table 1.

Language	MacroF1	Accuracy
JA	48.83	49.02
CS	48.49	48.66
CT	48.72	50.82

Table 1: RITE-2 BC Subtask on Test Set

For MC Task: The RITE-2 BC task result has shown in Table 2.

Language	MacroF1	Accuracy
JA	21.42	22.63
CS	24.38	24.71
CT	24.21	21.22

Table 2: RITE-2 MC Subtask on Test Set

[10] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

For ExamBC Task: This subtask is same as BC subtask. The RITE-2 ExamBC subtask result has shown in Table 3.

Language	MacroF1	Accuracy
JA	50.46	50.89

Table 3: RITE-2 ExamBC subtask on Test Set

Acknowledgments

We acknowledge the support of the INDO-MEXICO funded Indo-Mexico project “Answer Validation through Textual Entailment”.

References

Dagan, I., Glickman, O., Magnini, B. 2005. The PASCAL Recognising Textual Entailment Challenge. Proceedings of the First PASCAL Recognizing Textual Entailment Workshop.

Bar-Haim, R., Dagan, I., Dolan, B., Ferro, L., Giampiccolo, D., Magnini, B., Szpektor, I. 2006. The Second PASCAL Recognising Textual Entailment Challenge. Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment, Venice, Italy.

Giampiccolo, D., Magnini, B., Dagan, I., Dolan, B. 2007. The Third PASCAL Recognizing Textual Entailment Challenge, In Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing, Prague, Czech Republic.

Giampiccolo, D., Dang, H. T., Magnini, B., Dagan, I., Cabrio, E. 2008. The Fourth PASCAL Recognizing Textual Entailment Challenge. In TAC 2008 Proceedings.

Bentivogli, L., Dagan, I., Dang. H.T., Giampiccolo, D., Magnini, B. 2009. The Fifth PASCAL Recognizing Textual Entailment Challenge, In TAC 2009 Workshop, National Institute of Standards and Technology Gaithersburg, Maryland USA.

Bentivogli, L., Clark, P., Dagan, I., Dang, H.T. , Giampiccolo, D. 2010. The Sixth PASCAL Recognizing Textual Entailment Challenge. In TAC 2010 Notebook Proceedings.

Yuret, D., Han, A., Turgut, Z. 2010. SemEval-2010 Task 12: Parser Evaluation using Textual Entailments, Proceedings of the SemEval-2010 Evaluation Exercises on Semantic Evaluation.

Pakray, P., Bandyopadhyay, S., and Gelbukh, A. 2009. Lexical based two-way RTE System at RTE-5. System Report, TAC RTE Notebook.

Pakray, P., Pal, S., Poria, S., Bandyopadhyay, S. and Gelbukh, A. 2010a. JU_CSE_TAC: Textual Entailment Recognition System at TAC RTE-6. System Report, TAC RTE Notebook, 2010.

Pakray, P., Neogi, S., Bandyopadhyay, S., Gelbukh, A. 2011. A Textual Entailment System using Web based Machine Translation System. NTCIR-9: The 9th NTCIR Workshop Meeting "Evaluation of Information Access Technologies: Information Retrieval, Question Answering, and Cross-Lingual Information Access". RITE competition: Recognizing Inference in TExt@NTCIR9. National Institute of Informatics (NII), National Center of Sciences, Tokyo, Japan. December 6-9.

Pakray, P., Gelbukh, A. and Bandyopadhyay, S. 2010b. A Syntactic Textual Entailment System Using Dependency Parser. Springer Berlin / Heidelberg, Volume Volume 6008/2010, Book Computational Linguistics and Intelligent Text Processing, ISBN 978-3-642-12115-9, Pages 269-278.

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

Watanabe, Y., Miyao, Y., Mizuno, J., Shibata, T., Kanayama, H., Lee, C.-W., Lin, C.-J, Shi, S., Mitamura, T., Kando, N., Shima, H., Takeda, K. 2013. Overview of the Recognizing Inference in Text (RITE-2) at the NTCIR-10 Workshop. NTCIR-10: The 10th NTCIR Workshop Meeting "Evaluation of Information Access Technologies: Information Retrieval, Question Answering, and Cross-Lingual Information Access". RITE-2 competition: Recognizing Inference in TExt@NTCIR-10. National Institute of Informatics (NII), National Center of Sciences, June 18-21, 2013, NII, Tokyo, Japan