A Textual Entailment System using Web based Machine Translation System

Partha Pakray[1], Snehasis Neogi[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, snehasis1981@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 (NTCIR-9 RITE) @NTCIR9 for Japanese. NTCIR-9 RITE has four subtasks, Binary-class (BC) subtask, Multi-class (MC) subtask, Entrance Exam and NTCIR-9 RITE4QA. We have submitted a total of three unique runs (Run 1, Run 2 and Run 3) in the BC subtask and one run each in the MC Subtask, Entrance Exam subtask and NTCIR-9 RITE4QA subtask. The first system for BC subtask is based on Machine Translation using the web based Bing translator system. The second system for the BC subtask is based on lexical matching. The third system is based on a voting approach on the outcomes of the first and the second system. The system for MC subtask is based on a learned system that uses different lexical similarity features like Word Net based Unigram Matching, Bigram Matching, Trigram Machine, Skip-gram Matching, LCS Matching and Named Entity (NE) Matching. For Entrance Exam and NTCIR-9 RITE4QA subtask, we develop a single system based on the Ngram matching module similar to the second system of the BC subtask. For the BC subtask, the accuracy for Run 1, Run 2 and Run 3 are 0.490, 0.500 and 0.508 respectively. For the MC subtask, the accuracy is 0.175. The accuracy figures of the Entrance Exam subtask and the NTCIR-9 RITE4QA subtask are 0.5204 and 0.5954 respectively.

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 [1] in 2005, RTE-2 challenge [2] in 2006, RTE-3 challenge [3] in 2007, RTE-4 challenge [4] in 2008, RTE-5 [5] challenge in 2009, RTE-6 challenge [6] in 2010 and RTE-7 challenge in 2011. Another Textual Entailment track was Parser Training and Evaluation using Textual Entailment [7] as part of SemEval-2.

The BC and the MC subtasks, the Entrance Exam and the NTCIR-9 RITE4QA subtask in NTCIR-9 RITE are described in detail [8][1].

Languages in NTCIR-9 RITE task are Japanese, Simplified Chinese and Traditional Chinese. We have participated for the Japanese language only.

Section 2 describes the System Architecture 1 (BC subtask using web based Japanese – English Machine Translation). Section 3 describes the System Architecture 2 for BC subtask on the Japanese test data based on lexical matching. Section 4 details the System Architecture 3 for the BC subtask in which voting is applied on the results of the System 1 and System 2. Section-5 details the System Architecture 4 for the MC subtask. Section 6 details the System Architecture 5 for the Entrance Exam Subtask. Section 7 describes the system architecture 6 for the NTCIR-9 RITE4QA subtask. The experiments carried out on the test data sets are discussed in Section-8 along with the evaluation results. The conclusions are drawn in Section 9.

---

[1] http://artigas.lti.cs.cmu.edu/rite/Task_Guideline

2 System Architecture 1: BC Subtask using web based Machine Translation

The system accepts pairs of text snippets (text and hypothesis) 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 NTCIR-9 RITE BC subtask test set is shown in Figure 1.

```
<dataset type="bc">
 <pair id="1">
 <t1>石垣島は、冬でもハイビスカスが咲
き乱れる楽園だ。</t1>
 <t2>石垣島の冬の気温は高い。</t2>
 </pair>
```

Figure 1: NTCIR-9 RITE BC subtask Test Data

The system first identifies the (t1, t2) Japanese text segments from the NTCIR-9 RITE BC Subtask data. The Japanese (t1, t2) text segment is translated to English (t1, t2) text segment using the Japanese-English Bing translator (http://www.microsofttranslator.com/) machine translation (MT) system. The translated English text segments are then submitted to the textual entailment (TE) recognition system, The TE recognition system is based on Lexical Similarity. The final system is a combination of six different rules working on various lexical knowledge sources. The system computes the entailment decision using the outcome from the each of these rules.
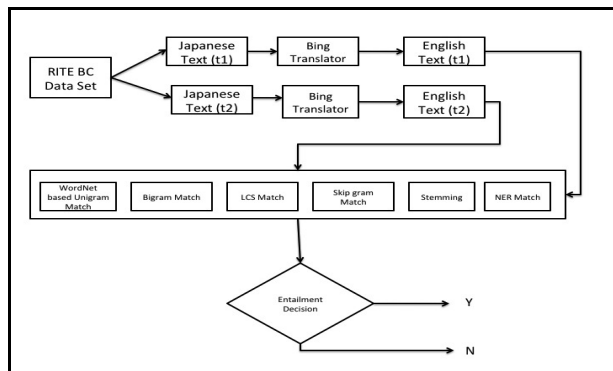

Figure 2. System Architecture BC Subtask (TE Recognition system using Web based MT System)

- *Lexical based Textual Entailment Methods:*

In this section the various lexical based TE methods [9] are described in detail.

1. **WordNet based Unigram Match:** In this method, the various unigrams in the hypothesis (t2) for each text (t1)-hypothesis (t2) pair are checked for there in the text. WordNet synsets are identified for each of the unmatched unigrams in the hypothesis. If any synset member for the hypothesis unigram matches with any synset member 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 between text and hypothesis and n2= number of unigram in Hypothesis,

$$Wordnet\_Unigram\_Match=n1/n2.$$

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, the text-hypothesis pair is considered as an entailment. The pair is assigned the value of 1 meaning entailment (Y); otherwise, the pair is assigned a value of 0 meaning no entailment (N). The cut-off value for the Wordnet_Unigram_Match is based on experiments carried out on the NTCIR-9 RITE BC subtask development set.

2. **Bigram Match:** Each bigram in the hypothesis is searched for a match in the corresponding text part. The measure Bigram_Match is calculated as
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 pair is then assigned the value of 1 meaning entailment (Y); otherwise, the pair is assigned a value of 0 meaning no entailment (N). The cut-off value for the Bigram_Match is based on experiments carried out on the NTCIR-9 RITE BC subtask development set.

3. **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 pair is then assigned the value of 1 meaning entailment (Y); otherwise, the pair is assigned a value of 0 meaning no entailment (N). The cut-off value for the LCS_Match is based on experiments carried out on the NTCIR-9 RITE BC subtask development set.

4. **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 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 1_skip_bigram_Match is based on experiments carried out on the NTCIR-9 RITE subtask development set.

5. **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 pair is assigned the value of 1 meaning entailment (Y); otherwise, the pair is assigned a value of 0 meaning no entailment (N). The cut-off value for the Stemming_Match is based on experiments carried out on the NTCIR-9 RITE BC subtask development set.

6. **Named Entity Match**: It is based on the detection and matching of Named Entities (NEs) in the text-hypothesis pair. 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 pair is assigned the value of 1 meaning entailment (Y); otherwise, the pair is assigned a value of 0 meaning no entailment (N). The cut-off value for the NE_Match is

based on experiments carried out on the NTCIR-9 RITE BC subtask development set.

For named entity recognition, the RASP Parser [10] nertag component has been used. The nertag component is a rule-based named entity recognizer which recognizes and marks up the following kinds of named entities: numex (sums of money and percentages), timex (dates and times) and enamex (persons, organizations and locations).

WordNet [11] is one of most important resource for lexical analysis. The WordNet 2.0 has been used for WordNet based unigram match and stemming . API for WordNet Searching (JAWS) [12] provides Java applications with the ability to retrieve data from the WordNet database.

3 System Architecture 2: BC Task in Japanese Language

The system architecture 2 is based on N-Gram matching on the input Japanese text pair {t1, t2} to identify whether t1 entails (infers) a hypothesis t2 or not. The system first learns the N-Gram (unigram and bigram) word overlap on the development data. The learned system is then applied on the test data to classify whether the text entails the hypothesis or not.
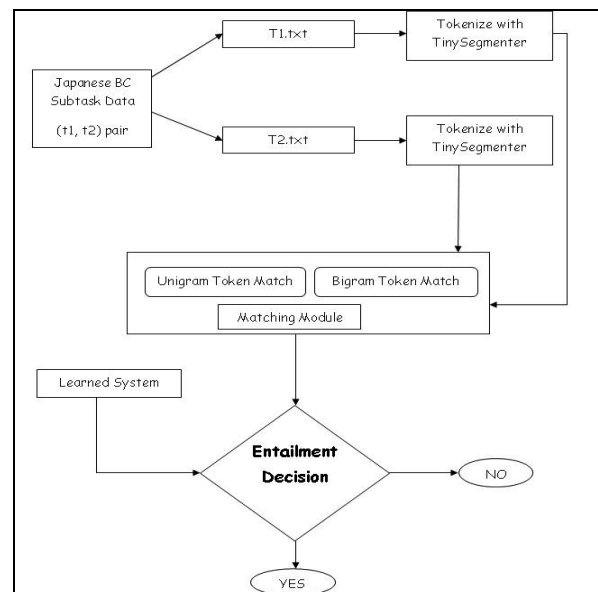

Figure 3. System Architecture Japanese BC subtask

- *Pre-processing:*
The system first separates Japanese text and hypothesis pairs t1 and t2.The text and the hypothesis are tokenized using the perl module of TinySegmenter[1] (Super compact Japanese Tokenizer). TinySegmenter separates the Japanese tokens by the symbol "#".The output of the

---

[1] http://chasen.org/~taku/software/TinySegmenter/

tokenizer on the text pairs t1 and t2 (as shown in Figure 1) is shown in the Figure 4. The parsed and tokenized forms of all the text and hypothesis sentences are stored in the system.

<t1>自主運#用移#行#を#契機#に#存在#感#が#高まっ#て#いる#公的#年金#資金#の#０#２#年度#の#資産#配分#計画#が#決まる#。 </t2>

<t2>公的#年金#資金#は#、#自主#運用#さ#れ#て#いる#。 </t2>

Figure 4. Tokenizer Output

- *N-Gram match:*

The N-Gram matching module is applied on the tokenized data and the percentage of N-Gram matching on the text and hypothesis are identified. These percentage scores are stored along with the pair id and Gold Standard entailment labels. The system applies unigram-bigram matching on the tokenized data.

1) **Unigram Match:** In this method, the various unigrams in the hypothesis for each text-hypothesis pair are checked for their presence in the text. For example in the above pair the common unigrams are 公的, 年金, 資金, て, いる.

$$Percentage\ Unigram\ Match = \frac{Common\ Unigrams\ between\ text\ and\ hypothesis}{Number\ of\ Unigrams\ in\ hypothesis} \times 100$$

2) **Bigram Match:** This method is similar to the unigram module; each bigram in the hypothesis is searched for a match in the text and the number of matches is considered in our system.

After the module runs on the development data we get the unigram and bigram matching percentage scores for each pair. Using these percentage values the system is trained so that it can classify the text – hypothesis pair in the RITE BC subtask test data.

The test data is tokenized and the unigram and bigram-matching modules are run on the tokenized test data. The unigram and bigram matching percentages are evaluated. Based on the statistics learned on the development data, if the percentage of unigram Match is 0.70 or more, i.e., 70% or more unigrams in the hypothesis match, the text-hypothesis pair is considered as an entailment. The text-hypothesis pair is assigned the label of "Y" meaning entailment; otherwise, the pair is assigned the label "N".

Similarly if the percentage of bigram Match is 0.40 or more, i.e., 40% or more bigrams in the hypothesis, match, the text-hypothesis pair is considered as an entailment. The text-hypothesis pair is assigned the label of "Y" meaning entailment; otherwise, the pair is assigned the label "N".

4 System Architecture 3: BC Subtask using Voting

The system considers the output of the previous two systems (Run 1 and Run 2) as input. The entailment decision is taken based on voting on the decisions in Run 1 and Run 2.

- *ANDing Module to match labels*

The system reads the labels of the system generated output files of Run 1 and Run 2 for a particular pair id. If the entailment decision of both the runs agree then this is outputted as the final entailment label. Otherwise, if they do not agree, the final entailment label will be "N" (NO).

5 System Architecture 4: NTCIR-9 RITE MC Subtask

The RITE MC subtask development data is similar to that of the BC Subtask except the entailment label. Here multi – way entailment label is used to detect entailment or no entailment in the text – hypothesis pair. The 5-way labeling subtask detects (forward / reverse / bidirectional) entailment or no entailment (contradiction / independence) in a text pair. The Labels for this task are {F, R, B, C, I} and are defined as follows.

*F: forward entailment** (t1 entails t2 AND t2 does not entails t1).
*R: reverse entailment** (t_2 entails t_1 AND t_1 does not entails t2).
*B: bidirectional entailment** (t1 entails t2 AND t2 entails t1).
*C: contradiction** (t1 and t2 contradicts, or cannot be true at the same time).
*I: independence** (otherwise).

Table 1: NTCIR-9 RITE MC subtask

In this subtask, the percentage of N-Gram (unigram, bigram, and trigram) word overlap on the text and hypothesis pair is evaluated and stored along with Gold Standard entailment labels. The architecture combines different modules such as preprocessing module, lexical based N-Gram similarity module and entailment decision module.

Each module in turn divided into different sub – modules. The descriptions of these modules are given below. Figure 6 shows the development data of RITE MC Subtask.
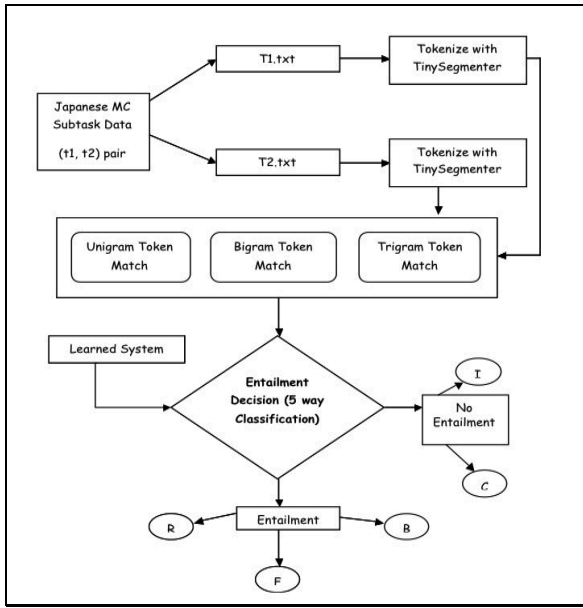
Figure 5: NTCIR-9 RITE MC subtask System Architecture

Figure 6: NTCIR-9 RITE MC Subtask Development Data

- *Pre – Processing Module*

1) Separate t1 – t2 pair: The system first accepts the development data and separates (t1, t2) pair id and the entailment label.

2) Tokenization with TinySegmenter: This module tokenizes the text and hypothesis using TinySegmenter (Super compact Japanese Tokenizer), a perl module that

reads every line and separates the tokens using the special symbols "#".

- *Lexical Based N-Gram Similarity Module:*

- *N-gram Match:*
This module is similar to that used in the BC subtask System architecture 2.

$$P(w_n \mid w_1^{n-1}) \approx P(w_n \mid w_{n-N+1}^{n-1})$$

$$P(w_n \mid w_1^{n-1}) \approx P(w_n \mid w_{n-1})$$

1) Unigram Match: In this method, the various unigrams in the hypothesis for each text-hypothesis pair are checked for their presence in the text.

$$Percentage\ Unigram\ Match = \frac{Common\ Unigrams\ between\ text\ and\ hypothesis}{Number\ of\ Unigrams\ in\ hypothesis} \times 100$$

2) Bigram Match: This method is similar to the unigram module; each Bigram in the hypothesis is searched for a match in the text.

$$Percentage\ Bigram\ Match = \frac{Common\ Bigrams\ between\ text\ and\ hypothesis}{Number\ of\ Bigrams\ in\ hypothesis} \times 100$$

3) Trigram Match: As per trigram every token in the text is depending on the previous two tokens of the text. This sub module is similar to the unigram and bigram module; each trigram in the hypothesis is searched for a match in the text.

$$Percentage\ Trigram\ Match = \frac{Common\ Trigrams\ between\ text\ and\ hypothesis}{Number\ of\ Trigrams\ in\ hypothesis} \times 100$$

After the percentage values of the unigram, bigram and trigram matches are obtained, these are used to train the system so that it can classify the test data text – hypothesis pair.
The MC subtask test data is also preprocessed and the different N-Gram percentages are evaluated. These percentage values are then compared with the percentage values calculated over the development data. If a match is found for the corresponding (*unigram,bigram*) *or* (*unigram,trigram*) *or* (*trigram, bigram*) **pairwise** percentage values for each text – hypothesis pair of the test data, the same entailment label {F,R,B,C,I} of the development data is used to classify the text – hypothesis

of the test set. Otherwise it classifies the pair as I (Independence).

6   System Architecture 5: NTCIR-9 RITE Entrance Exam Subtask

The Entrance Exam subtask is similar to that of BC subtask in terms of input and output. All the data is created from actual college-level entrance exams. In this subtask the entailment label is same as the BC subtask {Y, N}. The system for the entrance exam Subtask is similar to the system architecture 2 used for the BC subtask with some additions in the matching module. The complexity of the matching module is increased compared to the BC subtask architecture.

Figure 7 shows the system architecture of the NTCIR-9 RITE Entrance exam subtask and RITE4QA Subtask. The matching module is now described.
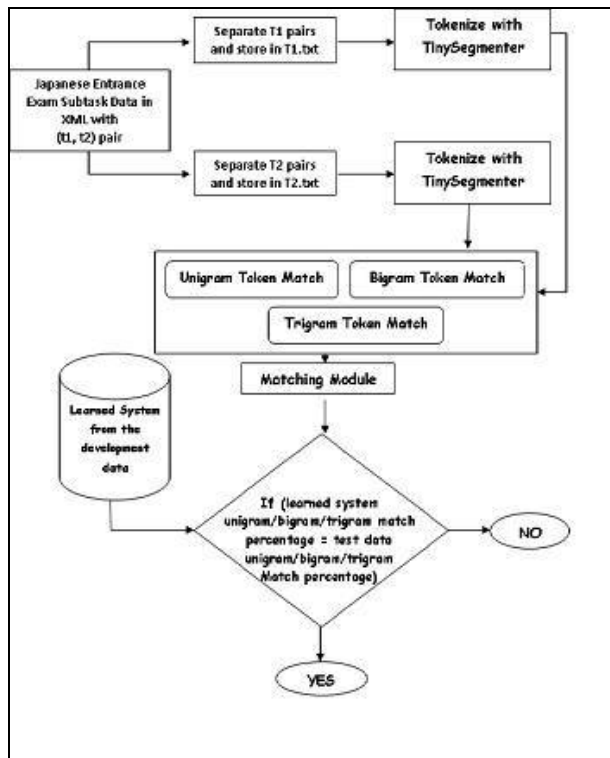

Figure 7: System Architecture for Entrance Exam and NTCIR-9 RITE4QA Subtask

- ***Matching Module***

This module is designed to take the entailment decision from the learned system based on the development data. The system stores the values of unigram, bigram, trigram and total percentage match between every pair (T1, T2) of development data with the corresponding pair id and gold standard label.

Now test data is also preprocessed and the values of unigram, bigram, trigram and total percentage match between every pair (T1, T2) of test data with the pair id are identified.

This module reads both the files of development and test data and checks weather the combined matching percentage of unigram – bigram, bigram – trigram, trigram – unigram, unigram – total, bigram – total or trigram – total of test data matches with any of the development data, then the corresponding gold standard label of the development data is used to classify the test data. If no entailment decision could be taken, the system checks the unigram match percentage of the test data pairs. If the unigram entailment score is 0.40 or greater, the entailment label "YES" is associated with the pair. Otherwise, the entailment label "NO" is associated.

7   System Architecture 6: NTCIR-9 RITE4QA Subtask

The RITE4QA Subtask is also similar to the BC subtask in terms of input and output, but it includes an embedded answer validation component in the Question Answering system. This way, the impact of NTCIR-9 RITE to an overall end-to-end application can be measured.

In this subtask, the same architecture and modules as in the NTCIR-9 RITE Entrance Exam Subtask are followed.

8   Experiments and Results

- ➢ **For NTCIR-9 RITE BC Subtask**: **(RITE1-JUCS-JA-BC)**

The NTCIR-9 RITE BC sub task development set was used to train the various RITE methods to identify the cut-off values for the various measures towards entailment decision. The RITE development set consisted of 500 text-hypothesis pairs. The RITE BC subtask test set includes 500 text-hypothesis pairs. In the lexical based textual entailment system, each method was run separately on the NTCIR-9 RITE BC task development set and two-way entailment (Y or N) decisions were obtained for each text-hypothesis pair.

- • **Run 1(RITE1-JUCS-JA-BC-01)**:

Based on Machine Translation using Bing Translator and Lexical Textual Entailment (Results are shown in Table 2 and Table 3)

	Y	N	
Y	64	69	133
N	186	181	367
	250	250	

Table 2. Confusion Matrix NTCIR-9 RITE BC Subtask on Test Set for Run 1

Accuracy	0.490

Table 3. Accuracy for NTCIR-9 RITE BC Subtask on Test Set for Run 1

- **Run 2(RITE1-JUCS-JA-BC-02)**

Based on Lexical Textual Entailment in Japanese Language. (Results are shown in Table 4 and Table 5)

	Y	N	
Y	85	85	170
N	165	165	330
	250	250	

Table 4. Confusion Matrix NTCIR-9 RITE BC Subtask on Test Set for Run 2

Accuracy	0.500

Table 5. Accuracy for NTCIR-9 RITE BC Subtask on Test Set for Run 2

- **Run 3(RITE1-JUCS-JA-BC-03)**:

Based on Run 1 and Run 2 Voting. (Results are shown in Table 6 and Table 7)

	Y	N	
Y	41	37	78
N	209	213	422
	250	250	

Table 6. Confusion Matrix NTCIR-9 RITE BC Subtask on Test Set for Run 3

Accuracy	0.508

Table 7. Accuracy for NTCIR-9 RITE BC Subtask on Test Set for Run 3

➢ **For NTCIR-9 RITE MC Subtask (RITE1-JUCS-JA-MC-01)**

The NTCIR-9 RITE MC subtask development set was used to train the various NTCIR-9 RITE methods to identify the cut-off values for the various measures towards entailment decision. The NTCIR-9 RITE development set consisted of 440 text-hypothesis pairs. The NTCIR-9 RITE MC subtask test set consisted of 440 text-hypothesis pair. In the lexical based textual entailment system, each method was run separately on the NTCIR-9 RITE MC subtask development set and five-way entailment (F, R, B, C, I) decisions were obtained for each text-hypothesis pair. Results are shown in Table 8 and Table 9.

	F	R	B	C	I	
F	32	35	22	28	30	147
R	29	18	14	9	17	87
B	7	9	3	5	4	28
C	14	13	8	4	9	48
I	28	35	28	19	20	130
	110	110	75	65	80	

Table 8. Confusion Matrix NTCIR-9 RITE MC Subtask on Test Set

Accuracy	0.175

Table 9. Accuracy for NTCIR-9 RITE BC Subtask on Test Set for Run 3

➢ **For NTCIR-9 RITE Entrance Exam Subtask(RITE1_JA_test_exam)**

The Evaluation Results in the form of Confusion Matrix and Accuracy are shown in Table 10 and Table 11 respectively.

	Y	N	
Y	89	120	209
N	92	141	233
	181	261	

Table 10. Confusion Matrix NTCIR-9 RITE Entrance Exam Subtask on Test Set

Accuracy	0.5204

Table 11. Accuracy for NTCIR-9 RITE Entrance Exam Subtask on Test Set

➢ **For NTCIR-9 RITE RITE4QA Subtask (RITE1_JA_test_rite4qa)**

The Evaluation Results are for Confusion Matrix shown in Table 12 and Accuracy shown in Table 13)

	Y	N	
Y	59	343	402
N	47	515	562
	106	858	

Table 12. Confusion Matrix NTCIR-9 RITE4QA Subtask on Test Set

Accuracy	0.5954

Table 13. Accuracy for NTCIR-9 RITE Entrance Exam Subtask on Test Set

9 Conclusions and Future Work

In this paper, textual entailment systems mainly based on the lexical similarity modules have been developed. MC subtask requires multi way classifications for the {T1,T2} pair. In this subtask, the on the lexical similarity based system has been used. The future works are oriented towards improving the system accuracy in two ways. First, in system architecture – 1, after translating the Japanese {t1,t2} pair into English {t1,t2} pair using the Web based Bing Japanese – English translator, anaphoric expressions in the English pair may be resolved to increase the accuracy of the textual entailment system. Second, in the system architecture for the MC subtask, the system can learn using some supervised learning methods (e.g., Support Vector Machine, Naive Bayes Classifier) that uses six different features based on lexical similarity.

Acknowledgments

References

[1] Dagan, I., Glickman, O., Magnini, B.: The PASCAL Recognising Textual Entailment Challenge Proceedings of the First PASCAL Recognizing Textual Entailment Workshop. (2005).

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

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

[4] Giampiccolo, D., Dang, H. T., Magnini, B., Dagan, I., Cabrio, E.:The Fourth PASCAL Recognizing Textual Entailment Challenge. In TAC 2008 Proceedings.
http://www.nist.gov/tac/publications/2008/papers.html (2008)

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

[6] Luisa Bentivogli, Peter Clark, Ido Dagan, Hoa Trang Dang,Danilo Giampiccolo: The Sixth PASCAL Recognizing Textual Entailment Challenge. In TAC 2010 Notebook Proceedings. (2010)

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

[8] Overview of NTCIR-9 RITE: Recognizing Inference in Text-H. Shima,H. Kanayama,C.-W. Lee,C.-J.Lin,T. Mitamura,Y. Miyao,S. Shi and K. Takeda.

[9] Partha Pakray, Sivaji Bandyopadhyay, Alexander Gelbukh, "Lexical based two-way RTE System at RTE-5", System Report, TAC RTE Notebook, 2009.

[10] E. Briscoe, J. Carroll, and R. Watson: The Second Release of the RASP System. In Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions.

[11] Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT Press (1998).

[12] Java API for WordNet Searching, Available: http://lyle.smu.edu/~tspell/jaws/index.html