JU_CSE_NLP: Language Independent Cross-lingual Textual Entailment System

Snehasis Neogi[1], Partha Pakray[2], Sivaji Bandyopadhyay[1], Alexander Gelbukh[3]

[1]Computer Science & Engineering Department
Jadavpur University, Kolkata, India
[2]Computer Science & Engineering Department
Jadavpur University, Kolkata, India
Intern at Xerox Research Centre Europe
Grenoble, France
[3]Center for Computing Research
National Polytechnic Institute
Mexico City, Mexico
{snehasis1981,parthapakray}@gmail.com
sbandyopadhyay@cse.jdvu.ac.in
gelbukh@gelbukh.com

Abstract

This article presents the experiments carried out at Jadavpur University as part of the participation in Cross-lingual Textual Entailment for Content Synchronization (CLTE) of task 8 @ Semantic Evaluation Exercises (SemEval-2012). The work explores cross-lingual textual entailment as a relation between two texts in different languages and proposes different measures for entailment decision in a four way classification tasks (forward, backward, bidirectional and no-entailment). We set up different heuristics and measures for evaluating the entailment between two texts based on lexical relations. Experiments have been carried out with both the text and hypothesis converted to the same language using the Microsoft Bing translation system. The entailment system considers Named Entity, Noun Chunks, Part of speech, N-Gram and some text similarity measures of the text pair to decide the entailment judgments. Rules have been developed to encounter the multi way entailment issue. Our system decides on the entailment judgment after comparing the entailment scores for the text pairs. Four different rules have been developed for the four different classes of entailment. The best run is submitted for Italian – English language with accuracy 0.326.

1 Introduction

Textual Entailment (TE) (Dagan and Glickman, 2004) is one of the recent challenges of Natural Language Processing (NLP). The Task 8 of SemEval-2012[1] [1] defines a textual entailment system that specifies two major aspects: the task is based on cross-lingual corpora and the entailment decision must be four ways. Given a pair of topically related text fragments (T1 and T2) in different languages, the CLTE task consists of automatically annotating it with one of the following entailment judgments:

i. *Bidirectional (T1 ->T2 & T1 <- T2)*: the two fragments entail each other (semantic equivalence)

ii. *Forward (T1 -> T2 & T1!<- T2)*: unidirectional entailment from T1 to T2 .

iii. *Backward (T1! -> T2 & T1 <- T2)*: unidirectional entailment from T2 to T1.

iv. *No Entailment (T1! -> T2 & T1! <- T2)*: there is no entailment between T1 and T2.

CLTE (Cross Lingual Textual Entailment) task consists of 1,000 CLTE dataset pairs (500 for

[1]http://www.cs.york.ac.uk/semeval2012/index.php?id=tasks

*First Joint Conference on Lexical and Computational Semantics (*SEM)*, pages 689–695,
Montréal, Canada, June 7-8, 2012. ©2012 Association for Computational Linguistics

training and 500 for test) available for the following language combinations:

- Spanish/English (spa-eng)
- German/English (deu-eng).
- Italian/English (ita-eng)
- French/English (fra-eng)

Seven Recognizing Textual Entailment (RTE) evaluation tracks have already been held: RTE-1 in 2005 [2], RTE-2 [3] in 2006, RTE-3 [4] in 2007, RTE-4 [5] in 2008, RTE-5 [6] in 2009, RTE-6 [7] in 2010 and RTE-7 [8] in 2011. RTE task produces a generic framework for entailment task across NLP applications. The RTE challenges have moved from 2 – way entailment task (YES, NO) to 3 – way task (YES, NO, UNKNOWN). EVALITA/IRTE [9] task is similar to the RTE challenge for the Italian language. So far, TE has been applied only in a monolingual setting. Cross-lingual Textual Entailment (CLTE) has been proposed ([10], [11], [12]) as an extension of Textual Entailment. In 2010, Parser Training and Evaluation using Textual Entailment [13] was organized by SemEval-2. Recognizing Inference in Text (RITE)[2] organized by NTCIR-9 in 2011 is the first to expand TE as a 5-way entailment task (forward, backward, bi-directional, contradiction and independent) in a monolingual scenario [14].

We have participated in RTE-5 [15], RTE-6 [16], RTE-7 [17], SemEval-2 Parser Training and Evaluation using Textual Entailment Task and RITE [18].

Section 2 describes our Cross-lingual Textual Entailment system. The various experiments carried out on the development and test data sets are described in Section 3 along with the results. The conclusions are drawn in Section 4.

2 System Architecture

Our system for CLTE task is based on a set of heuristics that assigns entailment scores to a text pair based on lexical relations. The text and the hypothesis in a text pair are translated to the same language using the Microsoft Bing machine translation system. The system separates the text pairs (T1 and T2) available in different languages and preprocesses them. After preprocessing we have used several techniques such as Word Overlaps, Named Entity matching, Chunk matching, POS matching to evaluate the separated text pairs. These modules return a set of score statistics, which helps the system to go for multi-class entailment decision based on the predefined rules. We have submitted 3 runs for each language pair for the CLTE task and there are some minor differences in the architectures that constitute the 3 runs. The three system architectures are described in section 2.1, section 2.2 and section 2.3.

2.1 System Architecture 1: CLTE Task with Translated English Text

The system architecture of Cross-lingual textual entailment consists of various components such as Preprocessing Module, Lexical Similarity Module, Text Similarity Module. Lexical Similarity module again is divided into subsequent modules like POS matching, Chunk matching and Named Entity matching. Our system calculates these measures twice once considering T1 as text and T2 as hypothesis and once T2 as text and T1 as hypothesis. The mapping is done in both directions T1-to-T2 and T2-to-T1 to arrive at the appropriate four way entailment decision using a set of rules. Each of these modules is now being described in subsequent subsections. Figure 1 shows our system architecture where the text sentence is translated to English.

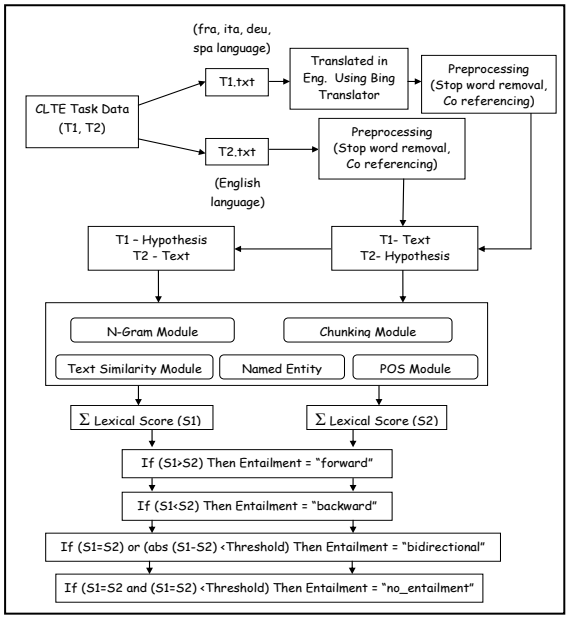

Figure 1: System Architecture

[2] http://artigas.lti.cs.cmu.edu/rite/Main_Page

2.1.1 Preprocessing Module

The system separates the T1 and T2 pair from the CLTE task data. T1 sentences are in different languages (In French, Italian, German and Spanish) where as T2 sentences are in English. Microsoft Bing translator[3] API for Bing translator (microsoft-translator-java-api-0.4-jar-with-dependencies.jar) is being used to translate the T1 text sentences into English. The translated T1 and T2 sentences are passed through the two sub modules.

i. Stop word Removal: Stop words are removed from the T1 and T2 sentences.

ii. Co-reference: Co–reference chains are evaluated for the datasets before passing them to the TE module. The objective is to increase the entailment score after substituting the anaphors with their antecedents. A word or phrase in the sentence is used to refer to an entity introduced earlier or later in the discourse and both having same things then they have the same referent or co-reference. When the reader must look back to the previous context, co-reference is called "*Anaphoric Reference*". When the reader must look forward, it is termed "*Cataphoric Reference*". To address this problem we used a tool called JavaRAP[4] (A java based implementation of Anaphora Procedure (RAP) - an algorithm by Lappin and Leass (1994)). It has been observed that the presence of co – referential expressions are very small in sentence based paradigm.

2.1.2 Lexical Based Textual Entailment (TE) Module

T1 - T2 pairs are the inputs to the system. The TE module is executed once by considering T1 as text and T2 as hypothesis and again by considering T2 as text and T1 as hypothesis. The overall TE module is a collection of several lexical based sub modules.

i. N-Gram Match module: The N-Gram match basically measures the percentage match of the unigram, bigram and trigram of hypothesis present in the corresponding text. These scores are simply combined to get an overall N – Gram matching score for a particular pair. By running the module we get two scores, one for T1-T2 pair and another for T2-T1 pair.

ii. Chunk Similarity module: In this sub module our system evaluates the key NP-chunks of both text and hypothesis identified using NP Chunker v1.1[5]. Then our system checks the presence of NP-Chunks of hypothesis in the corresponding text. System calculates the overall value for the chunk matching, i.e., number of text NP-chunks that match with hypothesis NP-chunks. If the chunks are not similar in their surface form then our system goes for WordNet matching for the words and if they match in WordNet synsets information, the chunks are considered as similar.

WordNet [19] is one of most important resource for lexical analysis. The WordNet 2.0 has been used for WordNet based chunk matching. The API for WordNet Searching (JAWS)[6] is an API that provides Java applications with the ability to retrieve data from the WordNet database. Let us consider the following example taken from training data:

T1: *Due/JJ to/TO [an/DT error/NN of/IN communication/NN] between/IN [the/DT police/NN] …*

T2: *On/IN [Tuesday/NNP] [a/DT failed/VBN communication/NN] between/IN…*

The chunk in T1 [error communication] matches with T2 [failed communication] via WordNet based synsets information. A weight is assigned to the score depending upon the nature of chunk matching.

$$\text{score (S)} = \sum_{i=1}^{N} M[i] / N$$

$$M[i] = W_m[i] * \rho / Wc[i]$$

Where N= Total number of chunk containing hypothesis.

M[i] = Match Score of the i^{th} Chunk.

$W_m[i]$ = Number of words matched in the i^{th} chunk.

$W_c[i]$ = Total words in the i^{th} chunk.

and ρ = { 1 if surface word matches. ; ½ if matche via WordNet

[3] http://code.google.com/p/microsoft-translator-java-api/

[4] http://aye.comp.nus.edu.sg/~qiu/NLPTools/JavaRAP.html

[5] http://www.dcs.shef.ac.uk/~mark/phd/software/

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

System takes into consideration several text similarity measures calculated over the T1-T2 pair. These text similarity measures are summed up to produce a total score for a particular text pair. Similar to the Lexical module, text similarity module is also executed for both T1-T2 and T2-T1 pairs.

iii. Text Distance Module: The following major text similarity measures have been considered by our system. The text similarity measure scores are added to generate the final text distance score.

- *Cosine Similarity*
- *Levenshtein Distance*
- *Euclidean Distance*
- *MongeElkan Distance*
- *NeedlemanWunch Distance*
- *SmithWaterman Distance*
- *Block Distance*
- *Jaro Similarity*
- *MatchingCoefficient Similarity*
- *Dice Similarity*
- *OverlapCoefficient*
- *QGrams Distance*

iv. Named Entity Matching: It is based on the detection and matching of Named Entities in the T1-T2 pair. Stanford Named Entity Recognizer[7] (NER) is used to tag the Named Entities in both T1 and T2. System simply matches the number of hypothesis NEs present in the text. A score is allocated for the matching.

NE_match = (Number of common NEs in Text and Hypothesis)/(Number of NEs in Hypothesis).

v. Part-of-Speech (POS) Matching: This module basically deals with matching the common POS tags between T1 and T2 pair. Stanford POS tagger[8] is used to tag the part of speech in both T1 and T2. System matches the verb and noun POS words in the hypothesis that match in the text. A score is allocated based on the number of POS matching.

POS_match = (Number of verb and noun POS in Text and Hypothesis)/(Total number of verb and noun POS in hypothesis).

[7] http://nlp.stanford.edu/software/CRF-NER.shtml
[8] http://nlp.stanford.edu/software/tagger.shtml

System adds all the lexical matching scores to evaluate the total score for a particular T1- T2 pair, i.e.,

Pair1: (T1 – Text and T2 – Hypothesis)
Pair2: (T1 – Hypothesis and T2 - Text).

Total lexical score for each pair can be mathematically represented by:

$$Score(S2) = \sum (Lexical\ Scores\ of\ pair2)$$

$$Score(S1) = \sum (Lexical\ Scores\ of\ pair1)$$

where S1 represents the score for the pair with T1 as text and T2 as hypothesis while S2 represents the score from T1 to T2. The figure 2 shows the sample output values of the TE module.

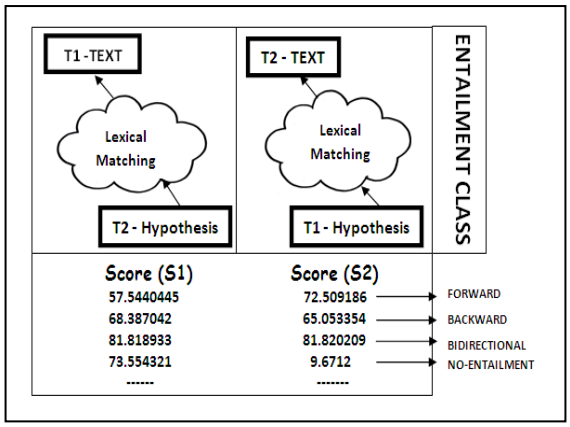

Figure 2: output values of this module

The system finally compares the above two values S1 and S2 as obtained from the lexical module to go for four-class entailment decision. If 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 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 "backward". 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 lexical scores. As our system produces a final score (S1 and S2) that is basically the sum over different similarity measures,

the tendency of identical S1 – S2 will be quite small. As a result we establish another heuristic for "bidirectional" class. If the absolute value difference between S1 and S2 is below the threshold value, our 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 are below a certain threshold, again set based on the observation in the training file, then system concludes the entailment class as "no_entailment". This threshold has been set as 20 based on observation from the training file.

2.2 System Architecture 2: CLTE Task with translated hypothesis

System Architecture 2 is based on lexical matching between the text pairs (T1, T2) and basically measures the same attributes as in the architecture 1. In this architecture, the English hypothesis sentences are translated to the language of the text sentence (French, Italian, Spanish and German) using the Microsoft Bing Translator. The CLTE dataset is preprocessed after separating the (T1, T2) pairs. Preprocessing module includes stop word removal and co-referencing. After preprocessing, the system executes the TE module for lexical matching between the text pairs. This module comprises N-Gram matching, Text Similarity, Named Entity Matching, POS matching and Chunking. The TE module is executed once with T1 as text and T2 as hypothesis and again with T1 as hypothesis and T2 as text. But in this architecture N-Gram matching and text similarity modules differ from the previous architecture. In system architecture 1, the N-Gram matching and text similarity values are calculated on the English text translated from T1 (i.e., Text in Spanish, German, French and Italian languages). In system architecture 2, the Microsoft Bing translator is used to translate T2 texts (in English) to different languages (i.e. in Spanish, German, French and Italian) and calculate N – Gram matching and Text Similarity values on these (T1 – newly translated T2) pairs. Other lexical sub modules are executed as before. These lexical matching scores are stored and compared according to the heuristic defined in section 2.1.

2.3 System Architecture 3: CLTE task using Voting

The system considers the output of the previous two systems (Run 1 from System architecture 1 and Run 2 from System architecture 2) as input. If the entailment decision of both the runs agrees then this is output as the final entailment label. Otherwise, if they do not agree, the final entailment label will be "no_entailment". The voting rule can be defined as the ANDing rule where logical AND operation of the two inputs are considered to arrive at the final evaluation class.

3 Experiments on Datasets and Results

Three runs (Run 1, Run 2 and Run 3) for each language were submitted for the SemEval-3 Task 8. The descriptions of submissions for the CLTE task are as follows:

- *Run1:* Lexical matching between text pairs (Based on system Architecture – 1).
- Run2: Lexical matching between text pairs (Based on System Architecture – 2).
- *Run3:* ANDing Module between *Run1* and *Run2*. (Based on System Architecture –3).

The CLTE dataset consists of 500 training CLTE pairs and 500 test CLTE pairs. The results for Run 1, Run 2 and Run 3 for each language on CLTE Development set are shown in Table 1.

Run Name	Accuracy
JU-CSE-NLP_deu-eng_run1	0.284
JU-CSE-NLP_deu-eng_run2	0.268
JU-CSE-NLP_deu-eng_run3	0.270
JU-CSE-NLP_fra-eng_run1	0.290
JU-CSE-NLP_fra-eng_run2	0.320
JU-CSE-NLP_fra-eng_run3	0.278
JU-CSE-NLP_ita-eng_run1	0.302
JU-CSE-NLP_ita-eng_run2	0.298
JU-CSE-NLP_ita-eng_run3	0.298
JU-CSE-NLP_spa-eng_run1	0.270
JU-CSE-NLP_spa-eng_run2	0.262
JU-CSE-NLP_spa-eng_run3	0.262

Table 1: Results on Development set

The comparison of the runs for different languages shows that in case of deu-eng language pair system architecture – 1 is useful for development data whereas system architecture – 2 is more accurate for test data. For fra-eng language pair, system architecture - 2 is more accurate for development data whereas voting helps to get more accurate results for test data. Similar to the deu-eng language pair, ita-eng language pair shows same trends, i.e., system architecture – 1 is more helpful for development data and system architecture – 2 is more accurate for test data. In case of spa-eng language pair system architecture – 1 is helpful for both development and test data.

The results for Run 1, Run 2 and Run 3 for each language on CLTE Test set are shown in Table 2.

Run Name	Accuracy
JU-CSE-NLP_deu-eng_run1	0.262
JU-CSE-NLP_deu-eng_run2	0.296
JU-CSE-NLP_deu-eng_run3	0.264
JU-CSE-NLP_fra-eng_run1	0.288
JU-CSE-NLP_fra-eng_run2	0.294
JU-CSE-NLP_fra-eng_run3	0.296
JU-CSE-NLP_ita-eng_run1	0.316
JU-CSE-NLP_ita-eng_run2	0.326
JU-CSE-NLP_ita-eng_run3	0.314
JU-CSE-NLP_spa-eng_run1	0.274
JU-CSE-NLP_spa-eng_run2	0.266
JU-CSE-NLP_spa-eng_run3	0.272

Table 2: Results on Test Set

4 Conclusions and Future Works

We have participated in Task 8 of Semeval-2012 named Cross Lingual Textual Entailment mainly based on lexical matching and translation of text and hypothesis sentences in the cross lingual corpora. Both lexical matching and translation have their limitations. Lexical matching is useful for simple sentences but fails to retain high accuracy for complex sentences with number of clauses. Semantic graph matching or conceptual graph is a good substitution to overcome these limitations. Machine learning technique is another important tool for multi-class entailment task. Features can be trained by some machine learning tools (such as SVM, Naïve Bayes or Decision tree etc.) with multi-way entailment (forward, backward, bi-directional, no-entailment) as its class. Works have been started in these directions.

Acknowledgments

The work was carried out under partial support of the DST India-CONACYT Mexico project "*Answer Validation through Textual Entailment*" funded by DST, Government of India and partial support of the project *CLIA Phase II (Cross Lingual Information Access)* funded by DIT, Government of India.

References

[1] Negri, M., Marchetti, A., Mehdad, Y., Bentivogli, L., and Giampiccolo, D.: *Semeval-2012 Task 8: Cross-lingual Textual Entailment for Content Synchronization*. In Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012).

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

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

[4] 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).

[5] Giampiccolo, D., Dang, H. T., Magnini, B., Dagan, I., Cabrio, E.: *The Fourth PASCAL Recognizing Textual Entailment Challenge*. In TAC 2008 Proceedings. (2008)

[6] 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).

[7] Luisa Bentivogli, Peter Clark, Ido Dagan, Hoa Trang Dang,Danilo Giampiccolo: *The Sixth PASCAL Recognizing Textual Entailment Chal-*

lenge. In TAC 2010 Notebook Proceedings. (2010)

[8] Bentivogli, L., Clark, P., Dagan, I., Dang, H., Giampiccolo, D.: *The Seventh PASCAL Recognizing Textual Entailment Challenge*. In TAC 2011 Notebook Proceedings. (2011)

[9] Bos, Johan, Fabio Massimo Zanzotto, and Marco Pennacchiotti. 2009. *Textual Entailment at EVALITA 2009*: In Proceedings of EVALITA 2009.

[10] Mehdad, Yashar, Matteo Negri, and Marcello Federico.2010. *Towards Cross-Lingual Textual entailment.* In Proceedings of the 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics, NAACL-HLT 2010. LA, USA.

[11] Negri, Matteo, and Yashar Mehdad. 2010. *Creating a Bilingual Entailment Corpus through Translations with Mechanical Turk: $100 for a 10-day Rush.* In Proceedings of the NAACL-HLT 2010, Creating Speech and Text Language Data With Amazon's Mechanical Turk Workshop. LA, USA.

[12] Mehdad, Yashar, Matteo Negri, Marcello Federico. 2011. *Using Bilingual Parallel Corpora for Cross-Lingual Textual Entailment.* In Proceedings of ACL 2011.

[13] 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).

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

[15] Pakray, P., Bandyopadhyay, S., Gelbukh, A.: *Lexical based two-way RTE System at RTE-5*. System Report, TAC RTE Notebook. (2009)

[16] Pakray, P., Pal, S., Poria, S., Bandyopadhyay, S., , Gelbukh, A.: *JU_CSE_TAC: Textual Entailment Recognition System at TAC RTE-6*. System Report, Text Analysis Conference Recognizing Textual Entailment Track (TAC RTE) Notebook. (2010)

[17] Pakray, P., Neogi, S., Bhaskar, P., Poria, S., Bandyopadhyay, S., Gelbukh, A.: *A Textual Entailment System using Anaphora Resolution*. System Report. Text Analysis Conference Recognizing Textual Entailment Track Notebook, November 14-15. (2011)

[18] Pakray, P., Neogi, S., Bandyopadhyay, S., Gelbukh, A.: *A Textual Entailment System using Web based Machine Translation System*. NTCIR-9, National Center of Sciences, Tokyo, Japan. December 6-9, 2011. (2011)

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