ICL Participation at RTE-7*

XU Xing, WANG Houfeng
Key Laboratory of Computational linguistics (Peking University), Ministry of Education, Beijing 100871
E-mail: xuxing@pku.edu.cn

Abstract

This paper describes ICL's participation at RTE-7. We chose the Main task. Textual entailment is a problem to predict whether an entailment holds for a given test-hypothesis pair. We built an inference model to solve this problem by means of using dependency syntax analysis (by Stanford Parser), lexical knowledge base (e.g. Wordnet), web information (e.g. Wikipedia) and probability method.

Keywords: textual entailment; lexical entailment probability; dependency syntax analysis; Wikipedia; Wordnet

1. Introduction

Textual Entailment is that a task to detect the entail relationship between a text pair, it can be applied to Q&A system, Intelligent search, etc. In RTE Main task, we need to know both the entail relationship for the given text pair.

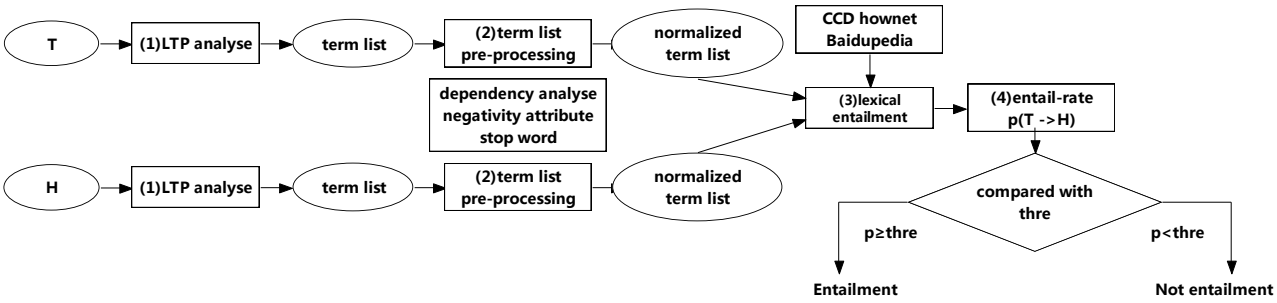

Figure 1: System architecture

2. Pre-processing

Stanford Parser: The functions of Stanford Parser include tokenization, POS tagging,

*This project is supported by NSFC(Natural Science Foundation of China), Fund No: 90920011, 91024009

Syntactic Analysis, etc. We get a word list after parsing a English sentence. Every word is converted to its lemma style by TreeTagger tool. We merge the words in the same named entity to one word. For each word in the list, we have its lexical information (lemma , POS), syntactic information (father and children in the dependency syntax tree) and semantic information (named entity, semantic role).

More Lexical information:

(1) **Negativity Attribute**: If a word is linked by a negative words (like no, not) in the syntax tree, then its negativity attribute is true, else is false.

(2) **Stop Word**: Stop Word stands for the word doesn't have real meaning, include punctuation, negative words and function word. All stop words will not participate in the calculation of the sentence entail-rate. We use the Stop attribute to identify if a word is a stop word.

3. Main Module

When given a pair of two sentences, T stands for the text, and H stands for the hypothesis, we then calculate the entail-rate of $p(T \to H)$.

(1)word pair entail-rate $p(T_w \to H_w)$***:***

We choose a word T_w from the text and a word H_w from the hypothesis, we have the following rules to get the entail-rate $p(T_w \to H_w)$:

First, Direct Rule: $p_{identify}(T_w \to H_w)$=1 if H_w is the same as T_w or a part of T_w

Second, Wordnet based Rule: $p_{\mathrm{CCD}}(T_w \to H_w)$=1 if H_w is in the synset, hypernym set, hyponym set, holonym set, meronym set or attribute set of T_w in Wordnet. And $p_{wordnet}(T_w \to H_w)$ = -1 if H_w is in the antonym set of T_w.

Third, Wikipedia based Rule: $p_{web}(T_w \to H_w)$=1 if H_w can be extracted from T_w's Wikipedia page. Three are many ways to extracted related word pairs from the web, we can choose the words that are linked with the source-word by be-verb, the hyper-link text and the text it links to, or the word in the parenthesis(as supplement explain for the former word) and the former word.

(2)the entail-rate of one word $p(H_w)$***:***

First, get the word from text that entails H_w***:***

H_w is a word in the hypothesis, all the words in the text is the candidate to entail this word, we use the word that have the maximal absolute value of entail-rate, named $T_{w-entail}$,

$$T_{w-entail} = \arg \operatorname*{maxabs}_{T_w \in T} p(T_w \to H_w)$$

Second, take into account the Negativity Attribute:

$$p(H_w) = P(T_{w-entail} \to H_w) * neg(T_{w-entail}) * neg(H_w)$$

Where $neg(w)$ is -1 is the Negativity Attribute is false and 1, otherwise.

(3)the entail-rate of the whole hypothesis sentence $p(H)$***:***

For each non-stop word in hypothesis, we get:

$$\begin{cases} polarity(H_w) = \text{sgn}(p(H_w)) \\ simlarity(H_w) = \mid p(H_w) \mid \end{cases}$$

The we calculate the final entail-rate of $P(T \to H)$,

$$polarity(H) = \prod_{H_w \in H} polarity(H_w)$$

$$simlarity(H) = \frac{1}{n} \sum_{H_w \in H} simlarity(H_w)$$, n is the count amount of non-stop word in H

$$p(T \to H) = polarity(H) * similarity(H)$$

(4)from $P(T \to H)$ ***to the entail-type:***

Binary-class:

$$result = \begin{cases} p(T \to H) \geq thre & \text{entailment} \\ otherwise & \text{not entailment} \end{cases}$$

In the training data, precision get maximal value when thre=0.62(BC), we use this value to label the test data.

4. Results

The result of RTE-7 Main task is presented below:

	Precision	**Recall**	**F-value**
ICL1	47.88%	21.56%	29.73%

Table 1: Results in RTE-7 Main task on test data

The result is not ideal. One reason is that the texts are extracted from a whole article, and there are some pronouns is the texts, it is necessary to use Coreference Resolution to find the true entities of the pronouns.

5. Conclusion and Future work

The paper presents the architecture of the system used in RTE-7. We used Wordnet to get lexical entailment, most words can be found in Wordnet but some Named Entities and OOV can't be found, so we introduced Wikipedia to improve the entailment of these words. But three are too many noises in the web, we will find more exact rules to extract

the helpful information from the web in the future. Our system didn't work well in recognize contradict cases, we will improve the algorithm from lexical entailment to sentence entailment.

Reference

Glickman, Oren, Ido Dagan and Moshe Koppel. *A Probabilistic Lexical Approach to Textual Entailment[C]*. In Proceedings of the Nineteenth Inter national Joint Conference on Artificial Intelligence. 2005:1682-1683.

Adrian Iftene. *TEXTUAL ENTAILMENT[D]*. Iasi, Romania: "Al. I. Cuza" University, 2009.

A. Iftene and M.A. Moruz. 2009. *UAIC participation at rte5*.In TAC 2009 Notebook Papers.

Eyal Shnarch. *Lexical entailment and its extraction from wikipedia. Master thesis*, Computer Science Department, Bar-Ilan University, Israel, 2008.