Effects of Using Simple Semantic Similarity on Textual Entailment Recognition

TEAM ID : u_tokyo

Ken-ichi Yokote, Shohei Tanaka and Mitsuru Ishizuka
Department of Information and Communication Eng.
School of Information Science and Technology
The University of Tokyo
{yokote, tanaka, ishizuka}@mi.ci.i.u-tokyo.ac.jp

Abstract

We applied various WordNet based similarity measures to the RTE (Recognizing Textual Entailment) task in order to compare the effects of them on Textual Entailment Recognition. Although the improvements over a baseline system are not big, many of them show positive effects.

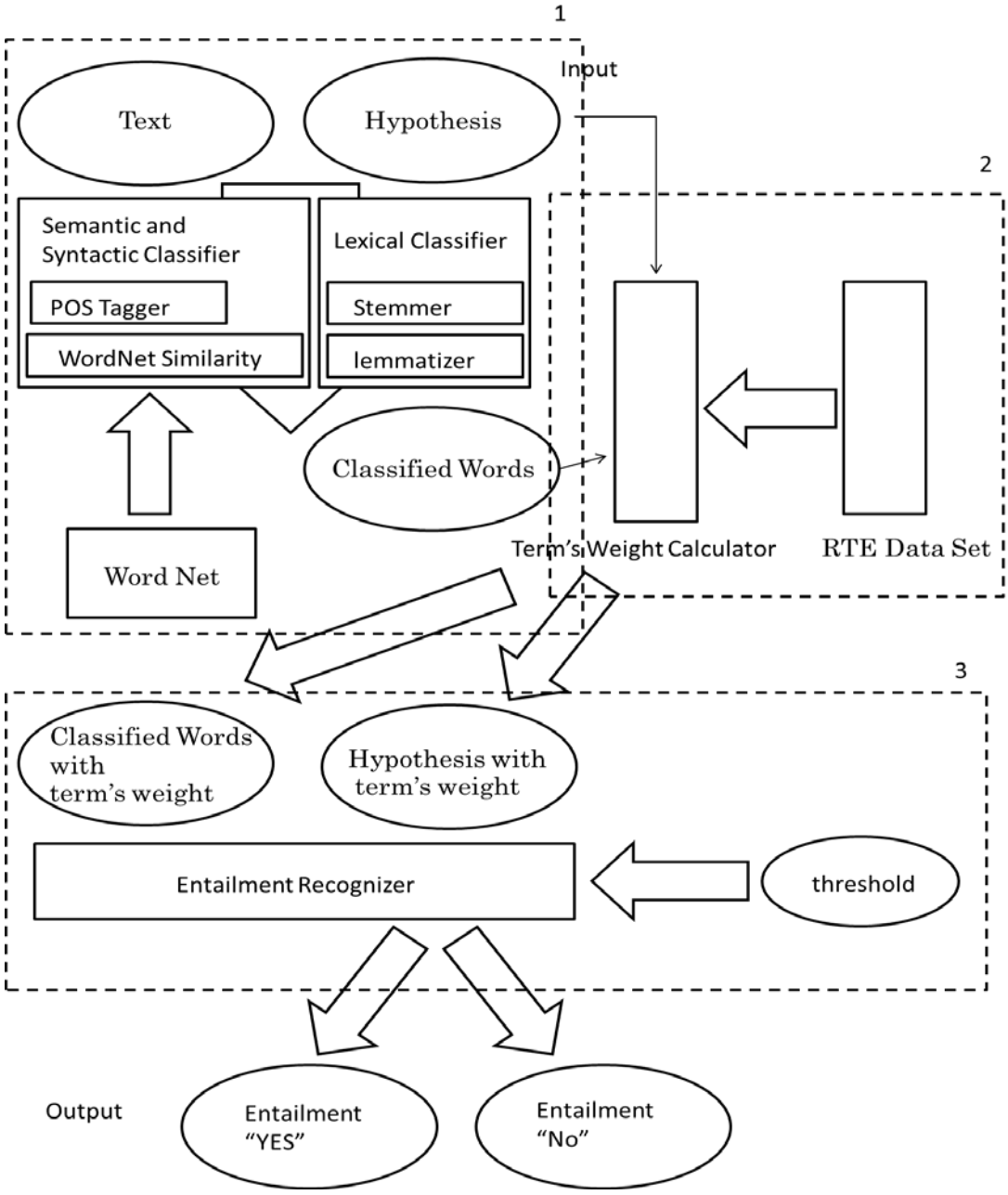
1. Introduction

In RTE (Recognizing Textual Entailment) tasks, it becomes effective to consider semantic similarities between given sentences -- T(precedent text) and H(hypothesis)--, while word-level matching is mainly employed in many present systems. However, the definition of "semantic similarity" is ambiguous and it is unclear what is the best way to measure the similarity for textual entailment. Thus, in our research, we tried to apply various WordNet based similarity measures to the RTE task in order to compare the effects of them on Textual Entailment Recognition. We used WordNet::Similarity [WordNet similarity] which is a freely available software package that makes it possible to measure the semantic similarity and relatedness between a pair of concepts (or synsets).

2. Our RTE System

The following figure shows the overview of our RTE system. This is roughly divided into three stages.

Overview of the Recognizing Textual Entailment System



2.1 Stage 1 -- Classifying terms in H

The system classifies terms in H into two groups, ones which are closely related to T, and the other. (The former are called "classified words" in the figure.) We employed two criteria in this classification. One is "Lexical Classifier", which is based on lexical coincidence. Another one is "Semantic and Syntactic Classifier", which is based on POS (part-of-speech) coincidence and Semantic Score.

Here, the Semantic Score of h (h ∈ H) is defined as:

$$\text{score}(h) = \text{Max}\{\text{WordNet-Similarity}(h, t) \text{ s.t. } t \in T\}$$

2.2 Stage 2 -- Calculating the term's weights

After the term classification in Stage 1, the system calculates the term's weight for all terms in H (including the classified words) as follows:

$$w(t) = \log_2 \frac{|T|}{\text{textfreq}(t)+1}$$ (|T| is amount of sentences in the Topic)

This is almost equivalent to IDF (Inverse Document Frequency).

2.3 Stage 3 -- Judging textual entailment

First in this stage, the system constructs feature vectors of H and the set of the classified words, where each feature component corresponds to each word. Then, Entailment Recognizer judges whether entailment is YES or NO by comparing a threshold with the cosine similarity between H and the classified words. (The result of this similarity can be approximated by the degree of the overlaps of H and the classified words.)

3. Experimental Results

3.1 Baseline system

As a baseline system, we used only lexical classifier in the stage1. For the development data set, it brought the best result shown below when the threshold was 0.7 in the experiments.

DEVELOPMENT-SET	
Recall	43.58
Precision	61.92

F-measure (macro)	50.6	(threshold = 0.7)

Using this threshold value, experimental results for the test data set were as follows:.

TEST-SET		
Recall	41.36	
Precision	50.00	
F-measure (macro)	45.27	(threshold = 0.7)

3.2 Applying WordNet Similarity Functions

We applied various WordNet similarity functions [WordNet similarity] to the classifier, and obtained their performance for the development data set as:.

DEVELOPMENT-SET	
F-measure (macro)	
Path Similarity	51.0
Res (Resnik) Similarity	50.1
Wup (Wu-Palmer) Similarity	50.8
Lin Similarity	51.2
Lch (Leacock-Chodorow) Similarity	51.2
Jcn (Jiang-Conrath) Similarity	51.7

where the threshold in each case was chosen to attain the best result. Applying the same threshold in each case, we obtained the experimental results for the test data set. Below shows only top two cases.

TEST-SET	
F-measure (macro)	
46.78	using Jcn (Jiang-Conrath) Similarity
46.04	using Lch (Leacock-Chodorow) Similarity

If we multiply these two similarity measures to generate a new measure, a bit better result has been obtained as:

46.87	using Jcn and Lch Similarities

where the threshold was also determined by multiplying two thresholds of Jcn and Lcn cases.

4. Discussion and Conclusion

The experimental results to date show that Jcn (Jiang-Conrath) Similarity in the WordNet similarity functions is the most effective to RTE-7 task. There are rooms for further improvements by applying several WordNet similarity functions simultaneously. Also, we are interested in applying more comprehensive measures as the semantic similarity.

Acknowledgments

We are grateful to Kai Ishikawa, Masaaki Tsuchida and Toshi-ichi Fukushima (NEC Corp.) for their advice and help.

References

[WordNet similarity] http://nltk.googlecode.com/svn/trunk/doc/howto/wordnet.html