BUPTTeam Participation at TAC 2011 Recognizing Textual Entailment

Yongmei Tan, Junyu Zeng, Xiaojie Wang
Center for Intelligence Science and
Technology and Technology
Beijing University of Posts and
Telecommunications
Beijing, China
{ymtan, jyzeng,
xjwang}@bupt.edu.cn

Eduard Hovy
Information Sciences Institute

University of Southern California

Marina del Rey, California 90292
hovy@isi.edu

Abstract

This paper overviews BUPTTeam's participation in the main task organized within the RTE7 Evaluation. In this paper we propose a method to calculate the similarity between text and hypothesis based on the TF/IDF values. Our system designed to recognize textual entailment typically employ lexical information. The evaluation results show that our method is effective for RTE task.

1 Introduction

The TAC 2011 recognizing textual entailment (RTE7) main task is similar to the RTE6 main task, in which textual entailment is performed on a real corpus.

The challenge of the task is threefold. First, the texts and hypotheses are not modified as compared to the original source, so they may contain incomplete sentences, spelling errors, grammar errors and abbreviations, etc. Second, texts and hypotheses are interpreted within the context of the topic, as they rely on explicit and implicit references to entities, dates, places, events, etc. pertaining to the corpus (Jia, Huang, Ma, Wan, Xiao, 2009). Third, there are much more negative pairs than positive pairs, as for RTE7 development set there are totally 21420 candidate pairs, while 1136 positive pairs (entailing) and 20284 negative pairs (not entailing).

We focus on the similarity estimated as the degree of word overlap between text and hypothesis based on the intuition that entailment is related to the similarity between text and hypothesis.

This paper describes BUPTTeam's participation in the main task and a preliminary analysis the challenge of RTE. In section 2 the related work is presented. Section 3 is dedicated to a detailed presentation of our system architecture, and the processing procedures are described in section 4. In section 5 the experimental results, together with the discussions, are presented. Conclusions and perspectives on future work are outlined in section 6.

2 Related work

The textual entailment is defined as a directional relationship between two text fragments - T, the entailing text and H, the entailed text - so that T entails H if, typically, a human reading T would infer that H is most likely true (Dagan et al., 2006).

This definition of entailment is based on common human understanding of language as well as background knowledge; in fact, for textual entailment to hold it is required that text and knowledge entail hypothesis, but knowledge alone cannot entail hypothesis (Bentivogli, Clark, Dagan, Dang and Giampiccolo, 2010). In other words, hypothesis is not entailed if hypothesis is true regardless of text (Bentivogli, Clark, Dagan, Dang and Giampiccolo, 2010). TF-IDF is a weight often used in information retrieval and text mining, and can be found in previous RTE paper.

In RTE6 18 teams participated in the Search Task, submitting a total of 48 runs (Bentivogli, Clark, Dagan, Dang and Giampiccolo, 2010).

3 System architecture

The proposed system is based on the similarity estimated as the degree of word overlap between text and hypothesis because we have the intuition that entailment is related to the similarity between text and hypothesis. For the most positive pairs, the similarity values are high and for the most negative pairs, the similarity values are low.

If a word appears frequently in the given text and hypothesis, the word might have less important than other infrequent words. So that we use TF-IDF algorithm to reduce the weights of frequent words while increase the weights of infrequent words. The system architecture is illustrated in Figure. 1.

4 Processing

4.1 Preprocessing

This step is to improve the quality of the text and hypothesis pairs. There are a lot of noises in the data set. We delete the tag ‘Q:’ and ‘A’ within text. Uppercase is converted to lowercase in order to improve the performance of word overlapping. We replace “hasn’t” with “has not”, “isn’t” with “is not” (Iftene and Moruz, 2009) within text. Sometimes the following signs maybe occur more than one time in the same place, so we just keep one and remove the rest: '.', '...', '"', '\'', '\'s', ',', '?', '!', ';', '--', '(', ')', ':', '_', etc.

4.2 POS Tagging and Stemming

We use the TreeTagger tool[1] to do Part-Of-Speech (POS) tagging and stemming, with a higher degree of precision. This step is very important, because our algorithm calculates words overlapping and builds the comparison on the basis of words.

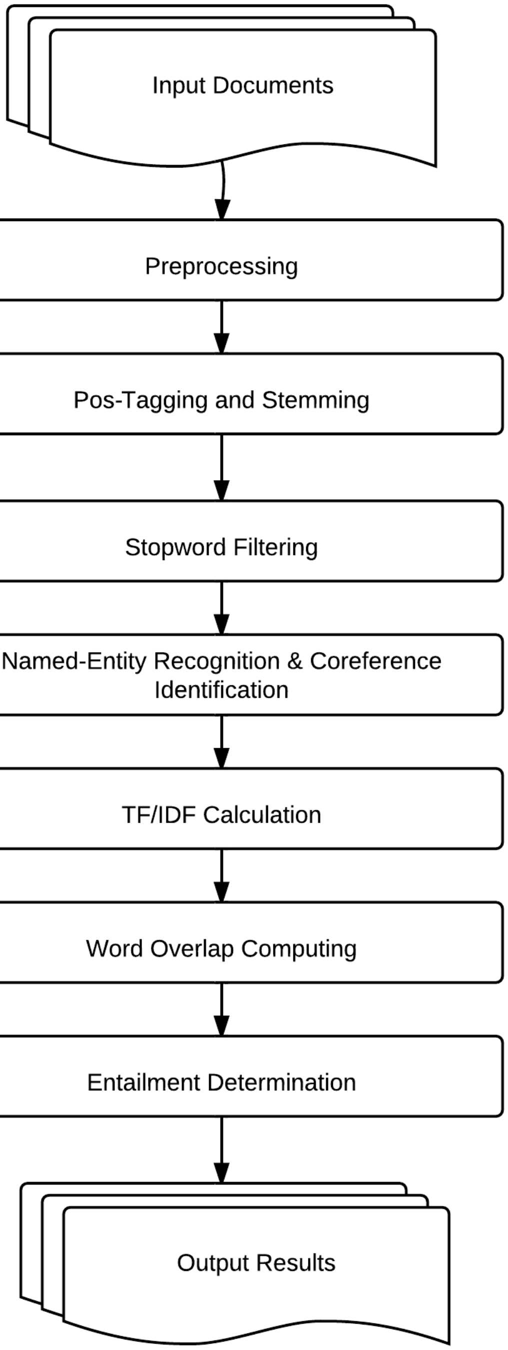

Figure 1. System Architecture

After sending the text to TreeTagger, we replace ‘<unkonwn>’ with the initial word and replace ‘@card@’ with the initial number. Our system also fixes the bug that TreeTagger produces many ‘\t’ in the output file. The meaning of the text is the same, but the quality of the LingPipe output is better after this transformation.

[1] http://www.cele.nottingham.ac.up/~ccztk/treetagger.php

4.3 Stopwords Removal

Stopwords usually are high-frequency words like the, to, and etc. They have little lexical content and their presence in a text fails to distinguish it from other texts. We try to use the following 3 methods to remove the stopwords, but we get the best experimental results using the method 1 on the data set of RTE6.

(1) All words of length 1.
(2) Stop list with 118 terms[2].
(3) Stop list with 571 terms[3].

4.4 Named Entity Recognition and Coreference Identification

Several words in different sentences are not exactly matched but they are referred to the same identity, such as 'Bush' and 'George. W. Bush'. Within a single sentence, pronoun can be referred to a noun phrase. This problem can be resolved by coreference identification in order to ignore the differences in form and indicate the same identity.

We use LingPipe coreference tool to identify coreference, which can give ids to certain noun phrases and pronouns, the same id indicates the same identity. Coreference identification is based on named entity recognition, so that LingPipe extracts mentions of people, companies, locations, organizations, and etc. firstly. But there are still many errors in the result file, for example, the tool labels the word 'he' in different sentences as the same id, which is an obvious mistake. We do some post-processing and fix these problems. After coreference identification, we replace the different names in texts and hypotheses with the same identity to calculate word overlap.

4.5 TF-IDF Calculation

In the standard TF-IDF algorithm, TF measures the term frequency in a document, we simply define our TF is equal to 1.0 because we use the intersection of the text and hypothesis pair to calculate word overlap, and every word of a set must be unique. Our IDF formula is as the same as the standard one.

We use all the unique sentences (including text and hypothesis) in the given text corpus to train the TF-IDF model, especially, if a test pair includes a word that never appeared in the given corpus, we consider it as a rare word and give it the maximum value of all the appeared words. Moreover, we think TF-IDF weight can cover the effect by using stop list, which merely consider several common words as zero-weighted.

In our experiments, TF-IDF can greatly improve the F-measure of our system based on word overlap.

4.6 Determination of Entailment

A higher degree of matching between text and hypothesis has been taken as indication of a semantic relation.

In our word overlap algorithm, we first get the word set of hypothesis and the intersection of text and hypothesis, then use the TF-IDF weighted set of the intersection divide the weighted set of hypothesis, to get the overlap score of candidate pair. Finally we use the threshold trained with previous corpus as a criterion to compare with the overlap score. Candidate pair with score higher than the threshold will be marked as true; and candidate pair with score lower than the threshold will be marked as false. So that the threshold is important to determine whether there is an entailment relation between a text and a hypothesis.

5 Results and Discussion

The RTE7 data set is composed of 20 topics, 10 used for the development set and 10 for the test set. The development set is composed of 100 documents and contains globally 284 hypotheses. The test set is also composed of 100 documents and contains globally 272 hypotheses. There are much more negative pairs than positive pairs.

System results are compared to a human-annotated gold standard and the metrics used to evaluate system performances are Precision, Recall, and F-measure.

Our system uses a threshold to judge whether the hypothesis can be entailed from the relative text or not. We use the development set and test set of RTE6, and develop set of RTE7, to train an appropriate threshold for the unseen test set of RTE7. We get 0.46, 0.50 and 0.53 relatively, so we use 0.46 (the minimum value), 0.49 (the average value) and 0.53 (the maximum value) as three

[2] http://www.lextek.com/manuals/onix/stopwords2.html
[3] http://nltk.googlecode.com/svn/trunk/doc/book/ch02.html

running threshold to get our result. The micro averaged scores and macro averaged scores are shown in Table 1.

Run	Micro-Average			Macro-Average		
	Precision	Recall	F-measure	Precision	Recall	F-measure
1	45.02	44.95	44.99	47.53	46.41	46.96
2	48.93	40.37	44.24	52.22	41.88	46.48
3	51.99	36.93	43.18	56.21	38.63	45.79

Table 1: Main task results for RTE7 test set

Analyzing text and hypothesis pairs from the development set, we find that the negative pairs have very low word overlap, and there are fewer negative pairs with high overlap than positive pairs with high overlap, and there are more positive pairs with low overlap than positive pairs with high overlap.

Based on several experiments, we decide to simply use word overlap as the main algorithm in our system. Our first experiment is SVM-based classification, trying to classify text and hypothesis pairs to positive (entailment) and negative (not entailment) by extracting several features, but most of the results are positive, due to the imbalance of the number of positive pairs to the number of negative pairs. This method results in a low F-measure. Another experiment based on linear regression by using SPSS meets the similar problem.

6 Conclusions

In this paper we propose a method, to calculate the similarity between text and hypothesis based on the IF-IDF values. In future we will focus on semantic and syntactic information to improve system performance. The experimental results suggest that lexical information alone is inadequate.

Acknowledgments

We gratefully acknowledge the support of the Fundamental Research Funds for the Central Universities and the National Natural Science Foundation of China (NSFC) (90920006). Any opinions, findings, and conclusion or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the view of the Ministry of Education of the People’s Republic of China. Part of this work was performed while the author was visiting USC/ISI, where the project was supported by the Defense Advanced Research Projects Agency (DARPA) Machine Reading Program under Air Force Research Laboratory (AFRL) prime contract no. FA8750-09-C-0172. Any opinions, findings, and conclusion or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the view of the DARPA, AFRL, or the US government.

References

Luisa Bentivogli, Peter Clark, Ido Dagan, Hoa Trang Dang and Ganilo Giampiccolo. 2010. The Sixth PASCAL Recognizing Textual Entailment Challenge

Adrian Iftene and Mihai Alex Moruz. UAIC Participation at RTE5. In Proceedings of TAC 2009.

Houping Jia, Xiaojiang Huang, Tengfei Ma, Xiaojun Wan, Jianguo Xiao. PKUTM Participation at TAC 2010 RTE and Summarization Track. In Proceedings of TAC 2009.

Gerard Salton and Christopher Buckley. 1988 Term-weighting approaches in automatic text retrieval. Information Processing & Management 24(5): 513–523.