Experiments of UNED at the Third Recognising Textual Entailment Challenge

Álvaro Rodrigo, Anselmo Peñas, Jesús Herrera, Felisa Verdejo
Departmento de Lenguajes y Sistemas Informáticos
Universidad Nacional de Educación a Distancia
Madrid, Spain
{alvarory, anselmo, jesus.herrera, felisa}@lsi.uned.es

Abstract

This paper describes the experiments developed and the results obtained in the participation of UNED in the Third Recognising Textual Entailment (RTE) Challenge. The experiments are focused on the study of the effect of named entities in the recognition of textual entailment. While Named Entity Recognition (NER) provides remarkable results (accuracy over 70%) for RTE on QA task, IE task requires more sophisticated treatment of named entities such as the identification of relations between them.

1 Introduction

The systems presented to the Third Recognizing Textual Entailment Challenge are based on the one presented to the Second RTE Challenge (Herrera et al., 2006b) and the ones presented to the Answer Validation Exercise (AVE) 2006 (Rodrigo et al., 2007).

Since a high quantity of pairs of RTE-3 collections contain named entities (82.6% of the hypotheses in the test collection contain at least one named entity), the objective of this work is to study the effect of named entity recognition on textual entailment in the framework of the Third RTE Challenge.

In short, the techniques involved in the experiments in order to reach these objectives are:

- Lexical overlapping between ngrams of text and hypothesis.
- Entailment between named entities.
- Branch overlapping between dependency trees of text and hypothesis.

In section 2, the main components of the systems are described in detail. Section 3 describes the information our systems use for the entailment decision. The description of the two runs submitted are given in Section 4. The results obtained and its analysis are described in Section 5. Section 6 shows a discussion of the results. Finally, some conclusions and future work are given.

2 Systems Description

The proposed systems are based on surface techniques of lexical and syntactic analysis considering each task (Information Extraction, Information Retrieval, Question Answering and Text Summarization) of the RTE Challenge independently.

The systems accept pairs of text snippets (text and hypothesis) at the input and give a boolean value at the output: YES if the text entails the hypothesis and NO otherwise. This value is obtained by the application of the learned model by a SVM classifier.

The main components of the systems are the following:

2.1 Linguistic processing

Firstly, each text-hypothesis pair is preprocessed in order to obtain the following information for the entailment decision:

- POS: a Part of Speech Tagging is performed in order to obtain lemmas for both text and hypothesis using the Freeling POS tagger (Carreras et al., 2004).

Proceedings of the Workshop on Textual Entailment and Paraphrasing, pages 89–94,
Prague, June 2007. ©2007 Association for Computational Linguistics

```
<t>...Iraq invaded Kuwait on <TIMEX>August_2_1990</TIMEX>...</t>
<h>Iraq invaded Kuwait in <NUMEX>1990</NUMEX></h>
```

Figure 1: Example of an error when disambiguating the named entity type.

```
<t>...Chernobyl accident began on
   <ENTITY>Saturday_April_26_1986</ENTITY>...</t>
<h>The Chernobyl disaster was in <ENTITY>1986</ENTITY></h>
```

Figure 2: Example of a pair that justifies the process of entailment.

```
<pair id=``5'' entailment=``NO'' task=``IE'' length=``short''>
   <t>The Communist Party USA was a small Maoist political party
   which was founded in 1965 by members of the Communist Party around
   Michael Laski who took the side of China in the Sino-Soviet split.
   </t>
   <h>Michael Laski was an opponent of China.</h>
</pair>

<pair id=``7'' entailment=``NO'' task=``IE'' length=``short''>
   <t>Sandra Goudie was first elected to Parliament in the 2002
   elections, narrowly winning the seat of Coromandel by defeating
   Labour candidate Max Purnell and pushing incumbent Green MP
   Jeanette Fitzsimons into third place.</t>
   <h>Sandra Goudie was defeated by Max Purnell.</h>
</pair>

<pair id=``8'' entailment=``NO'' task=``IE'' length=``short''>
   <t>Ms.  Minton left Australia in 1961 to pursue her studies in
   London.</t>
   <h>Ms.  Minton was born in Australia.</h>
</pair>
```

Figure 3: IE pairs with entailment between named entities but not between named entities relations.

- NER: the Freeling Named Entity Recogniser is also applied to recover the information needed by the named entity entailment module that is described in the following section. Numeric expressions, proper nouns and temporal expressions of each text and hypothesis are tagged.

- Dependency analysis: a dependency tree of each text and hypothesis is obtained using Lin's Minipar (Lin, 1998).

2.2 Entailment between named entities

Once the named entities of the hypothesis and the text are detected, the next step is to determine the entailment relations between the named entities in the text and the named entities in the hypothesis. In (Rodrigo et al., 2007) the following entailment relations between named entities were defined:

1. A Proper Noun E1 entails a Proper Noun E2 if the text string of E1 contains the text string of E2.

2. A Time Expression T1 entails a Time Expression T2 if the time range of T1 is included in the time range of T2.

3. A numeric expression N1 entails a numeric expression N2 if the range associated to N2 encloses the range of N1.

Some characters change in different expressions of the same named entity as, for example, in a proper noun with different wordings (e.g. *Yasser, Yaser, Yasir*). To detect the entailment in these situations, when the previous process fails, we implemented a modified entailment decision process taking into account the edit distance of Levenshtein (Levensthein, 1966). Thus, if two named entities differ in less than 20%, then we assume that exists an entailment relation between these named entities.

However, this definition of named entities entailment does not support errors due to wrong named entities classification as we can see in Figure 1. The expression 1990 represents a year but it is recognised as a numeric expression in the hypothesis. However the same expression is recognised as a temporal expression in the text and, therefore, the expression in the hypothesis cannot be entailed by it according to the named entities entailment definition above.

We quantified the effect of these errors in recognising textual entailment. For this purpose, we developed the following two settings:

1. A system based in dependency analysis and WordNet (Herrera et al., 2006b) that uses the categorization given by the NER tool, where the entailment relations between named entities are the previously ones defined.

2. The same system based on dependency analysis and WordNet but not using the categorization given by the NER tool. All named entities detected receive the same tag and a named entity E1 entails a named entity E2 if the text string of E1 contains the text string of E2 (see Figure 2).

We checked the performance of these two settings over the test corpus set of the Second RTE Challenge. The results obtained, using the accuracy measure that is the fraction of correct responses according to (Dagan et al., 2006), are shown in table 1. The table shows that with an easier and a more robust processing (NER without classification) the performance is not only maintained, but it is even slightly higher.

This fact led us to ignore the named entity categorization given by the tool and assume that text and hypothesis are related and close texts where same expressions must receive same categories, without the need of classification. Thus, all detected named entities receive the same tag and we consider that a named entity E1 entails a named entity E2 if the text string of E1 contains the text string of E2.

Table 1: Entailment between numeric expressions.

	Accuracy
Setting 1	0.610
Setting 2	0.614

2.3 Sentence level matching

A tree matching module, which searches for matching branches into the hypotheses' dependency trees, is used. There is a potential matching branch per leaf. A branch from the hypothesis is considered a "matching branch" only if all its nodes from the root to the leaf are involved in a lexical entailment (Herrera et al., 2006a). In this way, the subtree conformed by all the matching branches from a hypothesis' dependency tree is included in the respective text's dependency tree, giving an idea of tree inclusion.

We assumed that the larger is the included subtree of the hypothesis' dependency tree, the more semantically similar are the text and the hypothesis. Thus, the existence or absence of an entailment relation from a text to its respective hypothesis considers the portion of the hypothesis' tree that is included in the text's tree.

3 Entailment decision

A SVM classifier was applied in order to train a model from the development corpus. The model was trained with a set of features obtained from the processing described above. The features we have used and the training strategies were the following:

3.1 Features

We prepared the following features to feed the SVM model:

1. Percentage of nodes of the hypothesis' dependency tree pertaining to matching branches according to section 2.3 considering, respectively:
   - Lexical entailment between the words of the snippets involved.
   - Lexical entailment between the lemmas of the snippets involved.
2. Percentage of words of the hypothesis in the text (treated as bags of words).
3. Percentage of unigrams (lemmas) of the hypothesis in the text (treated as bags of lemmas).
4. Percentage of bigrams (lemmas) of the hypothesis in the text (treated as bags of lemmas).
5. Percentage of trigrams (lemmas) of the hypothesis in the text (treated as bags of lemmas).
6. A boolean value indicating if there is or not any named entity in the hypothesis that is not entailed by one or more named entities in the text according to the named entity entailment decision described in section 2.2.

Table 2: Experiments with separate training over the development corpus using cross validation.

	Accuracy with the same model for all tasks	Accuracy with a different model for each task
Setting 1	0.64	0.67
Setting 2	0.62	0.66

Table 3: Experiments with separate training over the test corpus.

	Accuracy with the same model for all tasks	Accuracy with a different model for each task
Setting 1	0.59	0.62
Setting 2	0.60	0.64

Table 4: Results for run 1 and run 2.

	Accuracy	
	run 1	run 2
IE	52.50%	53.50%
IR	67%	67%
QA	72%	72%
SUM	58%	60%
Overall	62.38%	63.12%

3.2 Training

About the decision of how to perform the training in our SVM models, we wanted to study the effect of training a unique model compared to training one different model per task.

For this purpose we used the following two settings:

1. A SVM model that uses features 2, 3, 4 and 5 from section 3.1.
2. A SVM model that uses features 2, 3, 4, 5 and 6 from section 3.1.

Each setting was training using cross validation over the development set of the Third RTE Challenge in two different ways:

1. Training a unique model for all pairs.

2. Training one model for each task. Each model is trained with only pairs from the same task that the model will predict.

The results obtained in the experiments are shown in table 2. As we can see in the table, with the training of one model for each task results are slightly better, increasing performance of both settings. Taking into account these results, we took the decision of using a different training for each task in the runs submitted.

Our decision was confirmed after the runs submission to RTE-3 Challenge with new experiments over the RTE-3 test corpus, using the RTE-3 development corpus as training (see table 3 for results).

4 Runs Submitted

Two different runs were submitted to the Third RTE Challenge. Each run was trained using the method described in section 3.2 with the following subset of the features described in section 3.1:

- Run 1 was obtained using the features 2, 3, 4 and 5 from section 3.1. These features obtained good results for pairs from the QA task, as we can see in (Rodrigo et al., 2007), and we wanted to check their performance in other tasks.

- Run 2 was obtained using the following features for each task:
  - IE: features 2, 3, 4, 5 and 6 from section 3.1. These ones were the features that obtained the best results for IE pairs in our experiments over the development set.
  - IR: features 2, 3, 4 and 5 from section 3.1. These ones were the features that obtained best results for IR pairs in our experiments over the development set.
  - QA: feature 6 from section 3.1. We chose this feature, which had obtained an accuracy over 70% in previous experiments over the development set in QA pairs, to study the effect of named entities in QA pairs.
  - SUM: features 1, 2 and 3 from section 3.1. We selected these features to show the importance of dependency analysis in SUM pairs as it is shown in section 6.

5 Results

Accuracy was applied as the main measure to the participating systems.

The results obtained over the test corpus for the two runs submitted are shown in table 4.

As we can see in both runs, different accuracy values are obtained depending on the task. The best result is obtained in pairs from QA with a 72% accuracy in the two runs, although two different systems are applied. This result pushes us to use this system for Answer Validation (Peñas et al., 2007). Results in run 2, which uses a different setting for each task, are slightly better than results in run 1, but only in IE and SUM. However, results are too close to accept a confirmation of our initial intuition that pairs from different tasks could need not only a different training, but also the use of different approaches for the entailment decision.

6 Discussion

In run 2 we used NER for IE and QA, the two tasks with the higher percentage of pairs with at least one named entity in the hypothesis (98.5% in IE and 97% in QA).

Our previous work about the use of named entities in textual entailment (Rodrigo et al., 2007) suggested that NER permitted to obtain good results. However, after the RTE-3 experience, we found that the use of NER does not improve results in all tasks, but only in QA in a solid way with the previous work.

We performed a qualitative study over the IE pairs showing that, as it can be expected, in pairs from IE the relations between named entities are more important that named entities themselves.

Figure 3 shows some examples where all named entities are entailed but not the relation between them. In pair 5 both *Michael Laski* and *China* are entailed but the relation between them is *took the side of* in the text, and *was an opponent of* in the hypothesis. The same problem appears in the other pairs with the relation *left* instead *was born in* (pair 8) or passive voice instead active voice (pair 7).

Comparing run 1 and run 2, dependency analysis shows its usefulness in SUM pairs, where texts and hypotheses have a higher syntactic parallelism than in pairs from other tasks. This statement is shown

Table 5: Percentage of hypothesis nodes in matching branches.

	Percentage
SUM	75,505%
IE	7,353%
IR	6,422%
QA	8,496%

in table 5 where the percentage of hypothesis nodes pertaining to matching branches in the dependency tree is much higher in SUM pairs than in the rest of tasks.

This syntactic parallelism seems to be the responsible for the 2% increasing between the first and the second run in SUM pairs.

7 Conclusions and future work

The experiments have been focused on the study of the importance of considering entailment between named entities in the recognition of textual entailment, and the use of a separate training for each task. As we have seen, both approaches increase slightly the accuracy of the proposed systems. As we have also shown, different approaches for each task could also increase the system performance.

Future work is focused on improving the performance in IE pairs taking into account relations between named entities.

Acknowledgments

This work has been partially supported by the Spanish Ministry of Science and Technology within the project R2D2-SyEMBRA (TIC-2003-07158-C04-02), the Regional Government of Madrid under the Research Network MAVIR (S-0505/TIC-0267), the Education Council of the Regional Government of Madrid and the European Social Fund.

References

X. Carreras, I. Chao, L. Padró, and M. Padró. 2004. *FreeLing: An Open-Source Suite of Language Analyzers.*. In Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC04). Lisbon, Portugal, 2004.

I. Dagan, O. Glickman and B. Magnini. 2006. *The PASCAL Recognising Textual Entailment Challenge*. In Quiñonero-Candela et al., editors, MLCW 2005, LNAI Volume 3944, Jan 2006, Pages 177 - 190.

J. Herrera, A. Peñas and F. Verdejo. 2006a. *Textual Entailment Recognition Based on Dependency Analysis and WordNet*. In Quiñonero-Candela et al., editors, MLCW 2005, LNAI Volume 3944, Jan 2006, Pages 231-239.

J. Herrera, A. Peñas, Á. Rodrigo and F. Verdejo. 2006b. *UNED at PASCAL RTE-2 Challenge*. In Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment, Venice, Italy.

V. I. Levensthein. 1966. *Binary Codes Capable of Correcting Deletions, Insertions and Reversals*. In Soviet Physics - Doklady, volume 10, pages 707710, 1966.

D. Lin. 1998. *Dependency-based Evaluation of MINIPAR*. Workshop on the Evaluation of Parsing Systems, Granada, Spain, May, 1998.

A. Peñas, Á. Rodrigo, V. Sama and F. Verdejo. 2007. *Overview of the Answer Validation Exercise 2006*. In Lecture Notes in Computer Science. In press.

Á. Rodrigo, A. Peñas, J. Herrera and F. Verdejo. 2007. *The Effect of Entity Recognition on Answer Validation*. In Lecture Notes in Computer Science. In press.