# Towards an Entity-based recognition of Textual Entailment

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

## Abstract

This paper describes the experiments developed and the results obtained in the participation of UNED in the Fourth Recognising Textual Entailment (RTE) Challenge. This year we decided to change the scope of our work with the aim of beginning to develop a system that performs a deeper analysis than the techniques used in the last editions. This participation has been the first step in the development of our new system.

## 1 Introduction

The task of Recognizing Textual Entailment (RTE) (Dagan et al., 2005) has grown in importance in the last years. There has been several evaluation forums that dealt with this task, as for example the RTE Challenges (Dagan et al., 2005; Bar-Haim et al., 2006; Giampiccolo et al., 2007), and the Answer Validation Exercise (AVE) (Peñas, 2007).

During these years, many approaches have been tested. According to the use of semantics and the results obtained, systems can be divided in three groups:

- The better results (approximately 80% of accuracy) are obtained by the few systems that best deal with the use of large semantic resources and background knowledge (Hickl, 2006; Hickl et al., 2007; Tatu et al., 2007; Iftene, 2007), which are two crucial factors affecting the performance of systems.

- On the other hand, systems that have performed naive methods, mostly based on lexical processing, have achieved better results. However, this kind of systems have an upper bound close to 70% of accuracy. Besides, it is possible to build entailment collections in which these systems would have a low performance (Roth and Sammons, 2007).

- The third group is the one made by systems which employ naive methods of semantics. Despite the fact that deeper semantic analysis is a must for achieving high accuracy, most of the systems that employ this analysis obtain worse results than the naive lexical models approach.

Since we realized that through the use of lexical methods and dependency analysis the performance of our system would be limited, we decided to move to an approach more scalable allowing the use of deeper techniques and more complex resources based on semantics. Then, we have begun to develop a new system even though in the first stages of our work we know the system would be in the group of systems that deal with semantics but do not outperform lexical systems.

The paper is organized as follow: in Section 2 the main characteristics of the new approach are given. Section 3 explains the current stage of development of our system while the details of the implementation are given in Section 4. Section 5 describes the method taken for giving the final decision of entailment for each pair, while the description of the runs submitted is given in Section 6 and the results and their analysis is shown in Section 7. Finally, some conclusions and future work are given.

## 2 An entity/relation approach

We adopted the traditional entity/relation/attribute model with the following definitions:

- **Entity**. An entity is something that has a distinct, separate existence, though it need not be a material existence. This is a definition that includes physical objects such as the entities that are defined in the ACE program (LDCa, 2005) as well as abstractions such as values. These entities may be referenced in a text by their name, indicated by a common noun or noun phrase, or represented by a pronoun.

- **Attribute**. An attribute is a property or abstraction of a characteristic of an entity.

- **Relation**. A relation for us will follow the common definition of a pair of entities with characteristics together (LDCb, 2005).

The first step of our work was to study the RTE collections available (the ones used at RTE-1, RTE-2 and RTE-3) in order to abstract the hypothesis-text pairs to some kind of representation that would make easier the decision of entailment. In the pairs studied, we observed that the hypothesis are formed by a simple sentence in which there are entities and relations among these entities. Besides, there may be additional elements in the sentence such as time expressions, locations, etc, which can be considered as attributes of the relations. Following this approach, we consider two different kinds of hypotheses:

1. Hypotheses formed by two entities and a relation between them. For example, in the sentence *UN peacekeepers abuse children*, the entities are *UN peacekeepers* and *children*, while the relation between them is *abuse*.

2. Hypotheses formed by an entity and attributes about the entity. For example, in the sentence *Becker was a tennis champion*, the entity is Becker, and *tennis* and *champion* are attributes of the entity.

Then, we decided that each hypothesis can be map into an structure made by a set of entities which their own attributes and the relations (which are the core of the structure) among these entities. This structure allow to use different techniques for comparing the hypotheses with the texts in order to find evidences of entailment.

With the structure already generated, the goal is to detect that all the elements (entities, relations and their attributes) or a part of them in the structure are entailed by the given text. For this entailment detection several strategies can be defined at different levels of analysis. These techniques ranked from the most basic ones based on lexical overlapping till more complex ones based on using semantic inferences. Our aim in the near future is to perform strategies based on deep analysis and semantics.

## 3 Current system

After defining the general model for our system, in this edition we developed a first version of it. Since this is the first version, not all the features of the model could be included. The current system includes in the representation the entities of the hypothesis and the main relation between them given by the verb phrase. Attributes of entities or relations have not been included except in the hypotheses formed by a unique entity and its attributes.

Regarding the techniques used for checking whether the elements in the structure are or not entailed by the given text, we have taken advantage of WordNet[1] (Herrera et al., 2006a; Herrera et al., 2006b). The system looks for words in the text that entail each of the elements of the structure. Three different values of the degree of entailment found can be returned for each element in the structure:

- If there is some part of the text which strongly entails the element of the hypothesis, the highest value of entailment is returned.

- If the previous condition is not given but a more relaxed entailment relation can be found, then it is returned an intermediate value of entailment.

- If not entailment can be found under the previous conditions, a value indicating that there is not entailment is given.

Once the values of entailment have been calculated for each element in the structure, these values

[1] http://wordnet.princeton.edu/

are used as features of a classifier in order to take the final decision of entailment. Besides this information, in this edition we decided to continue using the information about Named Entities (NE) that was very usefull for us in the last edition of the RTE (Rodrigo et al., 2007) and AVE 2007 (Rodrigo et al., 2008).

## 4 Description of the System

The system accepts pairs of text snippets (text and hypothesis) at the input and gives a boolean value at the output: ENTAILMENT if the system considers that the text entails the hypothesis and NO ENTAILMENT otherwise. This value is obtained by the application of a learned model (or models) by a SVM classifier.

Systems components are the following:

### 4.1 Linguistic processing

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

- NER: the Freeling Named Entity Recogniser (Carreras et al., 2004) is applied in order to recover the information needed by the NE entailment module. Numeric expressions, proper nouns and temporal expressions of each text and hypothesis are tagged.

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

### 4.2 Entailment between named entities

Once the NEs of the hypothesis and the text have been detected, the next step is to determine the entailment relations between the NEs in the text and the named entities in the hypothesis. As it is explained in (Rodrigo et al., 2007), we consider that a named entity NE1 entails a named entity NE2 if the text string of NE1 contains the text string of NE2. However, 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.

### 4.3 Building the structure of the Hypothesis

In this edition we took advantage of the dependency analysis of the hypotheses in order to build the desired hypotheses structures. There are two strategies for building the structures depending on the type of hypothesis:

- If a relation is expressed in the sentence, then the lemma of the verb of the sentence is selected as the relation, while each branch connected to the relation is taken as an entity. Moreover, each entity is annotated with its function in the sentence (subject, object, predicate, etc). For example, in the sentence *Accardo composed 24 Caprices*, the lemma *compose* is taken as the relation and *Accardo* and *24 Caprices* are taken as entities with the function of subject and object respectively.

- If the sentence is formed by only an entity, the structure is built with the entity and the attributes of this entity. For example, in the sentence *Michael Laski was an opponent of China*, the structure would be built with the entity *Michael Laski* and its attribute would be *opponent of China*.

### 4.4 Structure's elements entailment

Once the structures have been obtained, the system checks for each text-hypothesis pair whether there are parts in the text that entails the elements of the hypothesis. This checking of entailment is performed by a module based on WordNet relations and paths (Herrera et al., 2006a; Herrera et al., 2006b).

In this version of the system the three entailment values explained in section 3 are implemented in the following way:

- The highest value of entailment is given when exists a lexical unit LU1 of the text that is a synonym of a lexical unit LU2 of the hypothesis (e.g. obtain entails receive) or there is a path (making use of the hyponym relation) from one synset of LU1 to one synset of LU2

(e.g. glucose entails sugar). Furthermore, part meronym (e.g. Italy entails Europe), and adjective/adverb pertainym (e.g. Italian entails Italy) are used.

- The intermediate value of entailment is given if there is a similarity or derivational relation between LU1 and LU2.

- Otherwise, the value of not entailment is given.

In our study of the RTE collections we observed that it was not necessary to have all the elements of an entity entailed by the text. In fact, the most important element to be entailed in each entity is the head of such entity. Since the module presented above works with lexical units and the entities are usually composed by several lexical units, we used the head of each entity as the lexical units of the hypothesis.

## 5 Entailment decision

A SVM classifier was applied in order to train a model from annotated corpora. 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:

### 5.1 Features

The set of features was selected taking into account that the most important arguments of the hypotheses we studied were (if they exist): the subject, object and predicate. Then, we prepared the following features to feed the SVM model:

1. The value of entailment returned by the WordNet based module for the relation of the hypothesis.

2. The value of entailment returned by the WordNet based module for the entity which function is subject. If there is not any subject in the hypothesis, the highest value of entailment is returned as the default value.

3. The value of entailment returned by the WordNet based module for the entity which function is object or predicate (it is not common to have both in a hypothesis). If there is not any object or predicate in the hypothesis, the highest value of entailment is returned as the default value.

4. The average of the values of entailment returned by the WordNet based module for the entities which function is different to subject, object and predicate. If there is not any entity of this kind in the hypothesis, the highest value of entailment is returned as the default value.

5. 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 4.2.

### 5.2 Training

About the decision of how to perform the training in our SVM models and based in the experience of our last year's participation (Rodrigo et al., 2007), we perform two kinds of training strategies in this edition:

1. To train a unique model for all pairs.

2. To train one model for each task. Each model is trained with only pairs from the same task that the model will predict (one model for IE, another for IR, etc). With this strategy it is supposed that it is easier for the model to capture the special characteristics of each task because the noise produced by the pairs of other tasks is removed.

Despite the fact that strategy (2) gave us better results in the last edition and the experiments performed before the RTE-4, we decided to use also the strategy (1) because we wanted to compare again in this edition the results of using one or another training strategy.

## 6 Runs Submitted

We sent three different runs to the RTE-4 with the aim of comparing different approaches about the way of training and the corpus used for training. The three runs used the set of features described in section 5.1. The characteristics of each run are the following:

- Run 1 was obtained training a unique model for all the tasks (the method (1) of section 5.2). For the training collection, we join the collections of development and test of RTE-3. The first purpose was to check the performance of using a unique model for all the tasks, while the second purpose was to check whether a bigger training collection produces better results.

- Run 2 was obtained using the same training collection that in Run 1, but the method of training was different. We trained a different model for each task, and each of these models was used for predicting the value of entailment of pairs of this task in the RTE-4 test collection. The purpose of this run was to compare the results with Run 1 in order to check the possible gain in performance obtained by training different SVM models for each task.

- Run 3 was obtained using the same method of training used in Run 2 (one model per task). However, we used for training purposes only the RTE-3 test set. The objective of using this collection was to compare with Run 2 whether more training examples means a better performance.

## 7 Analysis of the results

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

The results obtained over the test corpus for the three runs submitted are shown in table 1.

Table 1: Results for Run 1, Run 2 and Run 3.

| | Accuracy | | |
|---|---|---|---|
| | Run 1 | Run 2 | Run 3 |
| IE | 47% | 44.33% | 50.33% |
| IR | 56.67% | 53.5% | 49% |
| QA | 53.5% | 65.5% | 53.5% |
| SUM | 65.5% | 56.33% | 54% |
| Overall | 54.9% | 54% | 51.3% |

As we can see in the runs, different accuracy values are obtained depending on the task. The worst results are obtained in the IE task. This is typically the task in which all the systems perform worst because the inferences that are necessary to use are more complex than in the other tasks. So, it is expected that as we add deeper analysis to our system, this will be one of the task with a higher increase in performance.

On the other hand, one of the best results have been obtained in the SUM pairs. Our system has behave in such way because in this pairs the lexical overlap is usually higher than in other tasks. Therefore, it seams that the naive version of our system has taken advantage of this characteristic.

Regarding the results among different runs, comparing the results obtained in Run 1 with the ones of Run 2, it seams that not significance variations are obtained changing the way of training. We do not know whether this is a consequence on the way the test collection has been generated this year (maybe there is less differences between pairs of different pairs). However, we think that more experiments on different collections should be performed taking into account statistical relevance measures in order to obtain some conclusions.

The only difference between Run 2 and Run 3 is the amount of data used for training. Since the results in Run 2, which is the one which uses more training examples (the double that Run 3), are better, it seams that the performance of the system we have presented can differ depending on the amount of training examples used. This follows the intuition obtained at RTE-2 that the size of the training corpora is one of the most important factors in this task (Bar-Haim et al., 2006). In fact, the best result of the RTE-2 was obtained by a system that utilized a very large entailment corpus for training, which contributed 10% to the overall accuracy obtained by the system (Hickl, 2006).

Regarding the analysis of the errors, since an entity in the hypothesis is entailed if its head is entailed, there are entities which head is entailed, but the complete full entity is not entailed because of modifiers affecting the head. This has produced that the system returned NO ENTAILMENT when the real value of the pair was ENTAILMENT. An example of this behaviour is shown in Figure 1. In the Figure, the entity of the hypothesis *the same facilities* is not entailed by the entity of the text *separate hotels* while the head *facilities* is entailed by *hotels*. Therefore, the systems fails giving the value of this pair.

```
<pair id=``302'' entailment=CONTRADICTION'' task=``QA''>
   <t>There have been no face-to-face talks yet - Israeli and
   Syrian delegations sit in separate hotels while Turkish
   mediators shuttle back and forth with messages </t>
   <h>Israeli and Syrian delegations are staying
   at the same facilities.</h>
</pair>
```

Figure 1: Example of a pair where the use of relation is incorrect.

```
<pair id=``729'' entailment=``UNKNOWN'' task=``IE''>
   <t>The UN in Congo have said that four people may have been
   killed in Kinshasa, but that is not yet confirmed.  A journalist
   in the area saw bruised and bandaged protestors who said they
   had been beaten by police, and a local TV station broadcast
   images of a protester who they said had been shot dead.
   The station was raided shortly afterwards by police.</t>
   <h>Police killed a protester in Congo.</h>
</pair>
```

Figure 2: Example of a pair where the use of relation is incorrect.

Another kind of errors has been detected when the relation of the hypothesis was entailed but the entities connected to this relation in the hypothesis were different to the entities connected to the word in the text that entails the relation of the hypothesis. An example can be seen in the Figure 2. In the Figure the relation *killed* of the hypothesis is entailed by *killed* in the text. However, the entities affected by the relation in the hypothesis are *Police* and *a protester* while in the text there are only one entity affected, which is *four people*. So, the systems returns a wrong value of entailment.

## 8 Conclusions and future work

Lexical methods for detecting textual entailment has an upper bound that can only be overcome by systems that performed a deeper analysis and take into account semantic resources. However, systems that perform these techniques not always achieve the expected results at the beginning.

With the aim of moving our system from the methods used in the past editions of the RTE Challenges to the use of more complex techniques, we have presented a new approach in this paper. Here we present the first stage of such work in which we have begun to develop our approach, not achieving the best possible results. However, we are encouraged to continue the development of our system including new resources and deeper analysis techniques since the model defined allow it to us.

Since this paper shows the first stages of our new RTE system, there is a lot of work to be made. Firstly, this future work is focused on improving the model presented here for the hypothesis including maybe information extracted from external resources such as FrameNet (Baker, 1998). Then, future work will continue in adding new and more complex ways of checking entailment among elements in the text and the hypothesis.

## Acknowledgments

This work has been partially supported by the Spanish Ministry of Science and Innovation within the project QEAVis-Catiex (TIN2007-67581-C02-01),

the Education Council of the Regional Government of Madrid and the European Social Fund.

## References

C. Baker, C. Fillmore, J. Lowe 1998. *The Berkeley FrameNet project*. In Proceedings of the COLING-ACL, Montreal, Canada, 1998.

R. Bar-Haim, I. Dagan, B. Dolan, L. Ferro, D. Giampiccolo, B. Magnini, I. Szpektor 2006. *The Second PASCAL Recognising Textual Entailment Challenge*. In Proceedings of the Challenges Workshop, April 2006, pages 1-9, Venice.

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. 2005. *The PASCAL Recognising Textual Entailment Challenge*. In Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment, Southampton, UK, pages 18, April 2005.

D. Giampiccolo, B. Magnini, I. Dagan , B. Dolan 2007. *The Third PASCAL Recognizing Textual Entailment Challenge*. In Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing. 2007.

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.

A. Hickl, J. Bensley 2007. *A Discourse Commitment-Based Framework for Recognizing Textual Entailment*. In Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing. 2007.

A. Hickl, J. Williams, J. Bensley, K. Roberts, B. Rink, Y. Shi 2006. *Recognizing Textual Entailment with LCCs GROUNDHOG System*. In Second PASCAL RTE Wokrshop. 2006

A. Iftene, A. Balahur-Dobrescu 2007. *Hypothesis Transformation and Semantic Variability Rules Used in Recognizing Textual Entailment*. In Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing. 2007.

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.

Linguistic Data Consortium a 2005. *ACE (Automatic Content Extraction) English Annotation Guidelines for Entities*.

Linguistic Data Consortium b 2005. *ACE (Automatic Content Extraction) English Annotation Guidelines for Relations*.

A. Peñas, Á. Rodrigo, V. Sama, F. Verdejo 2007. *Overview of the Answer Validation Exercise 2006*. In CLEF 2006, Lecture Notes in Computer Science LNCS 4730. Springer, Berlin. 2007

Á. Rodrigo, A. Peñas, J. Herrera, F. Verdejo 2007. *Experiments of UNED at the Third Recognising Textual Entailment Challenge*. In Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing. 2007.

Á. Rodrigo, A. Peñas, F. Verdejo 2008. *UNED at Answer Validation Exercise 2007*. In CLEF 2007, Lecture Notes in Computer Science LNCS 5152. Springer, Berlin. 2008.

D. Roth, M. Sammons 2007. *Semantic and Logical Inference Model for Textual Entailment*. In Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing. 2007.

M. Tatu, D. Moldovan 2007. *COGEX at RTE 3*. In Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing. 2007.