A Perspective-Based Approach for Solving Textual Entailment Recognition

Óscar Ferrández, Daniel Micol, Rafael Muñoz, and Manuel Palomar
Natural Language Processing and Information Systems Group
Department of Computing Languages and Systems
University of Alicante
San Vicente del Raspeig, Alicante 03690, Spain
{ofe, dmicol, rafael, mpalomar}@dlsi.ua.es

Abstract

The textual entailment recognition system that we discuss in this paper represents a perspective-based approach composed of two modules that analyze text-hypothesis pairs from a strictly lexical and syntactic perspectives, respectively. We attempt to prove that the textual entailment recognition task can be overcome by performing individual analysis that acknowledges us of the maximum amount of information that each single perspective can provide. We compare this approach with the system we presented in the previous edition of *PASCAL Recognising Textual Entailment Challenge*, obtaining an accuracy rate 17.98% higher.

1 Introduction

Textual entailment recognition has become a popular Natural Language Processing task within the last few years. It consists in determining whether one text snippet (hypothesis) entails another one (text) (Glickman, 2005). To overcome this problem several approaches have been studied, being the *Recognising Textual Entailment Challenge* (RTE) (Bar-Haim et al., 2006; Dagan et al., 2006) the most referred source for determining which one is the most accurate.

Many of the participating groups in previous editions of RTE, including ourselves (Ferrández et al., 2006), designed systems that combined a variety of lexical, syntactic and semantic techniques. In our contribution to RTE-3 we attempt to solve the textual entailment recognition task by analyzing two different perspectives separately, in order to acknowledge the amount of information that an individual perspective can provide. Later on, we combine both modules to obtain the highest possible accuracy rate. For this purpose, we analyze the provided corpora by using a lexical module, namely *DLSITE-1*, and a syntactic one, namely *DLSITE-2*. Once all results have been obtained we perform a voting process in order to take into account all system's judgments.

The remainder of this paper is structured as follows. Section two describes the system we have built, providing details of the lexical and syntactic perspectives, and explains the difference with the one we presented in RTE-2. Third section presents the experimental results, and the fourth one provides our conclusions and describes possible future work.

2 System Specification

This section describes the system we have developed in order to participate in RTE-3. It is based on surface techniques of lexical and syntactic analysis. As the starting point we have used our previous system presented in the second edition of the RTE Challenge (Ferrández et al., 2006). We have enriched it with two independent modules that are intended to detect some misinterpretations performed by this system. Moreover, these new modules can also recognize entailment relations by themselves. The performance of each separate module and their combination with our previous system will be detailed in section three.

Next, Figure 1 represents a schematic view of the system we have developed.

Proceedings of the Workshop on Textual Entailment and Paraphrasing, pages 66–71,
Prague, June 2007. ©2007 Association for Computational Linguistics

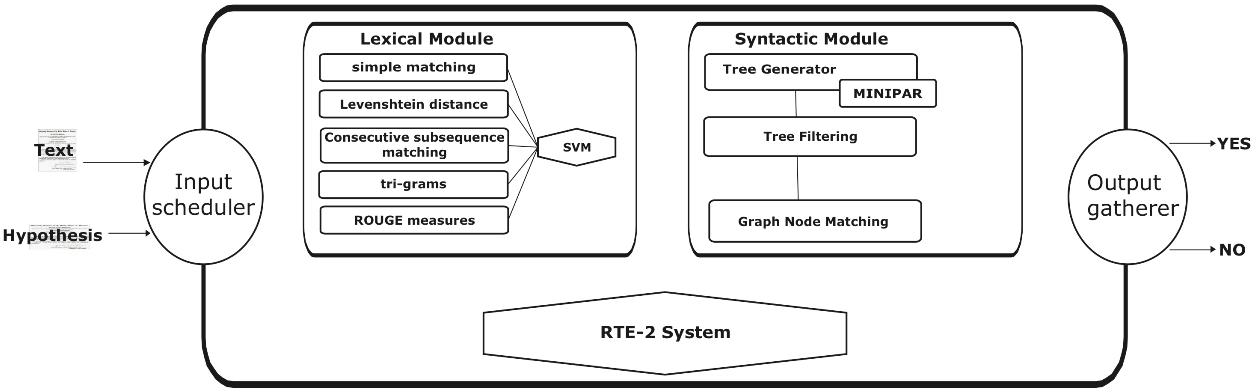

Figure 1: System architecture.

As we can see in the previous Figure, our system is composed of three modules that are coordinated by an input scheduler. Its commitment is to provide the text-hypothesis pairs to each module in order to extract their corresponding similarity rates. Once all rates for a given text-hypothesis pair have been calculated, they will be processed by an output gatherer that will provide the final judgment. The method used to calculate the final entailment decision consists in combining the outputs of both lexical and syntactic modules, and these outputs with our RTE-2 system's judgment. The output gatherer will be detailed later in this paper when we describe the experimental results.

2.1 RTE-2 System

The approach we presented in the previous edition of RTE attempts to recognize textual entailment by determining whether the text and the hypothesis are related using their respective derived logic forms, and by finding relations between their predicates using WordNet (Miller et al., 1990). These relations have a specific weight that provide us a score representing the similarity of the derived logic forms and determining whether they are related or not.

For our participation in RTE-3 we decided to apply our previous system because it allows us to handle some kinds of information that are not correctly managed by the new approaches developed for the current RTE edition.

2.2 Lexical Module

This method relies on the computation of a wide variety of lexical measures, which basically consists of overlap metrics. Although in other related work this kind of metrics have already been used (Nicholson et al., 2006), the main contribution of this module is the fact that it only deals with lexical features without taking into account any syntactic nor semantic information. The following paragraphs list the considered lexical measures.

Simple matching: initialized to zero. A boolean value is set to one if the hypothesis word appears in the text. The final weight is calculated as the sum of all boolean values and normalized dividing it by the length of the hypothesis.

Levenshtein distance: it is similar to simple matching. However, in this case we use the mentioned distance as the similarity measure between words. When the distance is zero, the increment value is one. On the other hand, if such value is equal to one, the increment is 0.9. Otherwise, it will be the inverse of the obtained distance.

Consecutive subsequence matching: this measure assigns the highest relevance to the appearance of consecutive subsequences. In order to perform this, we have generated all possible sets of consecutive subsequences, from length two until the length in words, from the text and the hypothesis. If we proceed as mentioned, the sets of length two extracted from the hypothesis will be compared to the sets of the same length from the text. If the same element is present in both the text and the hypothesis set, then a unit is added to the accumulated weight. This procedure is applied for all sets of different length extracted from the hypothesis. Finally, the sum of the weight obtained from each set of a specific length is normalized by the number of sets corresponding to this length, and the final accumulated weight is also normalized by the length of the hypothesis in words minus one. This measure is defined as follows:

$$CSmatch = \frac{\sum_{i=2}^{|H|} f(SH_i)}{|H| - 1} \qquad (1)$$

where SH_i contains the hypothesis' subsequences of length i, and $f(SH_i)$ is defined as follows:

$$f(SH_i) = \frac{\sum_{j \in SH_i} match(j)}{|H| - i + 1} \qquad (2)$$

being $match(j)$ equal to one if there exists an element k that belongs to the set that contains the text's subsequences of length i, such that $k = j$.

One should note that this measure does not consider non-consecutive subsequences. In addition, it assigns the same relevance to all consecutive subsequences with the same length. Furthermore, the longer the subsequence is, the more relevant it will be considered.

Tri-grams: two sets containing tri-grams of letters belonging to the text and the hypothesis were created. All the occurrences in the hypothesis' tri-grams set that also appear in the text's will increase the accumulated weight in a factor of one unit. The weight is normalized by the size of the hypothesis' tri-grams set.

ROUGE measures: considering the impact of n-gram overlap metrics in textual entailment, we believe that the idea of integrating these measures[1] into our system is very appealing. We have implemented them as defined in (Lin, 2004).

Each measure is applied to the words, lemmas and stems belonging to the text-hypothesis pair. Within the entire set of measures, each one of them is considered as a feature for the training and test stages of a machine learning algorithm. The selected one was a Support Vector Machine due to the fact that its properties are suitable for recognizing entailment.

2.3 Syntactic Module

The syntactic module we have built is composed of few submodules that operate collaboratively in order to obtain the highest possible accuracy by using only syntactic information.

The commitment of the first two submodules is to generate an internal representation of the syntactic dependency trees generated by *MINIPAR* (Lin, 1998). For this purpose we obtain the output of such parser for the text-hypothesis pairs, and then process it to generate an on-memory internal representation of the mentioned trees. In order to reduce our system's noise and increase its accuracy rate, we only keep the relevant words and discard the ones that we believe do not provide useful information, such as determinants and auxiliary verbs. After this step has been performed we can proceed to compare the generated syntactic dependency trees of the text and the hypothesis.

The graph node matching, termed alignment, between both the text and the hypothesis consists in finding pairs of words in both trees whose lemmas are identical, no matter whether they are in the same position within the tree. Some authors have already designed similar matching techniques, such as the one described in (Snow et al., 2006). However, these include semantic constraints that we have decided not to consider. The reason of this decision is that we desired to overcome the textual entailment recognition from an exclusively syntactic perspective. The formula that provides the similarity rate between the dependency trees of the text and the hypothesis in our system, denoted by the symbol ψ, is shown in Equation 3:

$$\psi(\tau, \lambda) = \sum_{\nu \in \xi} \phi(\nu) \qquad (3)$$

where τ and λ represent the text's and hypothesis' syntactic dependency trees, respectively, and ξ is the set that contains all synsets present in both trees, being $\xi = \tau \cap \lambda \;\; \forall \alpha \in \tau, \beta \in \lambda$. As we can observe in Equation 3, ψ depends on another function, denoted by the symbol ϕ, which provides the relevance of a synset. Such a weight factor will depend on the grammatical category and relation of the synset. In addition, we believe that the most relevant words of a phrase occupy the highest positions in the dependency tree, so we desired to assign different weights depending on the depth of the synset. With all these factors we define the relevance of a word as shown

[1] The considered measures were ROUGE-N with n=2 and n=3, ROUGE-L, ROUGE-W and ROUGE-S with s=2 and s=3.

in Equation 4:

$$\phi(\beta) = \gamma \cdot \sigma \cdot \mu^{-\delta_\beta} \tag{4}$$

where β is a synset present in both τ and λ, γ represents the weight assigned to β's grammatical category (Table 1), σ the weight of β's grammatical relationship (Table 2), μ an empirically calculated value that represents the weight difference between tree levels, and δ_β the depth of the node that contains the synset β in λ. The performed experiments reveal that the optimal value for μ is 1.1.

Grammatical category	**Weight**
Verbs, verbs with one argument, verbs with two arguments, verbs taking clause as complement	1.0
Nouns, numbers	0.75
Be used as a linking verb	0.7
Adjectives, adverbs, noun-noun modifiers	0.5
Verbs *Have* and *Be*	0.3

Table 1: Weights assigned to the relevant grammatical categories.

Grammatical relationship	**Weight**
Subject of verbs, surface subject, object of verbs, second object of ditransitive verbs	1.0
The rest	0.5

Table 2: Weights assigned to the grammatical relationships.

We would like to point out that a requirement of our system's similarity measure is to be independent of the hypothesis length. Therefore, we must define the normalized similarity rate, as represented in Equation 5:

$$\overline{\psi(\tau, \lambda)} = \frac{\sum\limits_{\nu \in \xi} \phi(\nu)}{\sum\limits_{\beta \in \lambda} \phi(\beta)} \tag{5}$$

Once the similarity value has been calculated, it will be provided to the user together with the corresponding text-hypothesis pair identifier. It will be his responsibility to choose an appropriate threshold that will represent the minimum similarity rate to be considered as entailment between text and hypothesis. All values that are under such a threshold will be marked as not entailed.

3 System Evaluation

In order to evaluate our system we have generated several results using different combinations of all three mentioned modules. Since the lexical one uses a machine learning algorithm, it has to be run within a training environment. For this purpose we have trained our system with the corpora provided in the previous editions of RTE, and also with the development corpus from the current RTE-3 challenge. On the other hand, for the remainder modules the development corpora was used to set the thresholds that determine if the entailment holds.

The performed tests have been obtained by performing different combinations of the described modules. First, we have calculated the accuracy rates using only each single module separately. Later on we have combined those developed by our research group for this year's RTE challenge, which are *DLSITE-1* (the lexical one) and *DLSITE-2* (the syntactic one). Finally we have performed a voting process between these two systems and the one we presented in RTE-2.

The combination of *DLSITE-1* and *DLSITE-2* is described as follows. If both modules agree, then the judgement is straightforward, but if they do not, we then decide the judgment depending on the accuracy of each one for true and false entailment situations. In our case, *DLSITE-1* performs better while dealing with negative examples, so its decision will prevail over the rest. Regarding the combination of the three approaches, we have developed a voting strategy. The results obtained by our system are represented in Table 3. As it is reflected in such table, the highest accuracy rate obtained using the RTE-3 test corpus was achieved applying only the lexical module, namely *DLSITE-1*. On the other hand, the syntactic one had a significantly lower rate, and the same happened with the system we presented in RTE-2. Therefore, a combination of them will most likely produce less accurate results than the lexical module, as it is shown in Table 3. However, we would like to point out that these results depend heavily on the corpus idiosyncrasy. This can be proven with the results obtained for the RTE-2 test corpus, where the grouping of the three modules provided the highest accuracy rates of all possible combinations.

	RTE-2 test	RTE-3 dev	RTE-3 test				
	Overall	Overall	Overall	IE	IR	QA	SUM
RTE-2 system	0.5563	0.5523	0.5400	0.4900	0.6050	0.5100	0.5550
DLSITE-1	0.6188	0.7012	**0.6563**	**0.5150**	**0.7350**	**0.7950**	**0.5800**
DLSITE-2	0.6075	0.6450	0.5925	0.5050	0.6350	0.6300	0.6000
DLSITE-1&2	0.6212	0.6900	0.6375	0.5150	0.7150	0.7400	0.5800
Voting	0.6300	0.6900	0.6375	0.5250	0.7050	0.7200	0.6000

Table 3: Results obtained with the corpora from RTE-2 and RTE-3.

3.1 Results Analysis

We will now perform an analysis of the results shown in the previous section. First, we would like to mention the fact that our system does not behave correctly when it has to deal with long texts. Roughly 11% and 13% of the false positives of *DLSITE-1* and *DLSITE-2*, respectively, are caused by misinterpretations of long texts. The underlying reason of these failures is the fact that it is easier to find a lexical and syntactic match when a long text is present in the pair, even if there is not entailment.

In addition, we consider very appealing to show the accuracy rates corresponding to true and false entailment pairs individually. Figure 2 represents the mentioned rates for all system combinations that we displayed in Table 3.

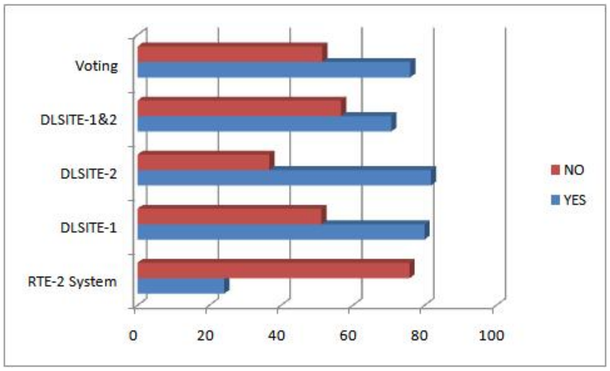

Figure 2: Accuracy rates obtained for true and false entailments using the RTE-3 test corpus.

As we can see in Figure 2, the accuracy rates for true and false entailment pairs vary significantly. The modules we built for our participation in RTE-3 obtained high accuracy rates for true entailment text-hypothesis pairs, but in contrast they behaved worse in detecting false entailment pairs. This is the opposite to the system we presented in RTE-2, since it has a much higher accuracy rate for false cases than true ones. When we combined *DLSITE-1* and *DLSITE-2*, their accuracy rate for true entailments diminished, although, on the other hand, the rate for false ones raised. The voting between all three modules provided a higher accuracy rate for false entailments because the system we presented at RTE-2 performed well in these cases.

Finally, we would like to discuss some examples that lead to failures and correct forecasts by our two new approaches.

Pair 246 entailment=YES task=IR

T: Overall the accident rate worldwide for commercial aviation has been falling fairly dramatically especially during the period between 1950 and 1970, largely due to the introduction of new technology during this period.

H: Airplane accidents are decreasing.

Pair 246 is incorrectly classified by *DLSITE-1* due to the fact that some words of the hypothesis do not appear in the same manner in the text, although they have similar meaning (e.g. airplane and aviation). However, *DLSITE-2* is able to establish a true entailment for this pair, since the hypothesis' syntactic dependency tree can be matched within the text's, and the similarity measure applied between lemmas obtains a high score. This fact produces that, in this case, the voting also achieves a correct prediction for pair 246.

Pair 736 entailment=YES task=SUM

T: In a security fraud case, Michael Milken was sentenced to 10 years in prison.

H: Milken was imprisoned for security fraud.

Pair 736 is correctly classified by *DLSITE-1* since there are matches for all hypothesis' words (except *imprisoned*) and some subsequences. In contrast, *DLSITE-2* does not behave correctly with this example because the main verbs do not match, being this fact a considerable handicap for the overall score.

4 Conclusions and Future Work

This research provides independent approaches considering mainly lexical and syntactic information. In order to achieve this, we expose and analyze a wide variety of lexical measures as well as syntactic structure comparisons that attempt to solve the textual entailment recognition task. In addition, we propose several combinations between these two approaches and integrate them with our previous RTE-2 system by using a voting strategy.

The results obtained reveal that, although the combined approach provided the highest accuracy rates for the RTE-2 corpora, it has not accomplished the expected reliability in the RTE-3 challenge. Nevertheless, in both cases the lexical-based module achieved better results than the rest of the individual approaches, being the optimal for our participation in RTE-3, and obtaining an accuracy rate of about 70% and 65% for the development and test corpus, respectively. One should note that these results depend on the idiosyncrasies of the RTE corpora. However, these corpora are the most reliable ones for evaluating textual entailment recognizers.

Future work can be related to the development of a semantic module. Our system achieves good lexical and syntactic comparisons between texts, but we believe that we should take advantage of the semantic resources in order to achieve higher accuracy rates. For this purpose we plan to build a module that constructs characterized representations based on the text using named entities and role labeling in order to extract semantic information from a text-hypothesis pair. Another future research line could consist in applying different recognition techniques depending on the type of entailment task. We have noticed that the accuracy of our approach differs when the entailment is produced mainly by lexical or syntactic implications. We intend to establish an entailment typology and tackle each type by means of different points of view or approaches.

Acknowledgments

This research has been partially funded by the QALL-ME consortium, which is a 6^{th} Framework Research Programme of the European Union (EU), contract number FP6-IST-033860 and by the Spanish Government under the project CICyT number TIN2006-1526-C06-01. It has also been supported by the undergraduate research fellowships financed by the Spanish Ministry of Education and Science, and the project ACOM06/90 financed by the Spanish Generalitat Valenciana.

References

Roy Bar-Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. 2006. The Second PASCAL Recognising Textual Entailment Challenge. *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, pages 1–9.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL Recognising Textual Entailment Challenge. In *Quiñonero-Candela et al., editors, MLCW 2005, LNAI Volume 3944*, pages 177–190. Springer-Verlag.

Oscar Ferrández, Rafael M. Terol, Rafael Muñoz, Patricio Martínez-Barco, and Manuel Palomar. 2006. An approach based on Logic forms and wordnet relationships to textual entailment performance. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, pages 22–26, Venice, Italy.

Oren Glickman. 2005. *Applied Textual Entailment Challenge*. Ph.D. thesis, Bar Ilan University.

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

Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In Stan Szpakowicz Marie-Francine Moens, editor, *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, Barcelona, Spain, July. Association for Computational Linguistics.

George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J. Miller. 1990. Introduction to WordNet: An On-line Lexical Database. *International Journal of Lexicography*, 3(4):235–244.

Jeremy Nicholson, Nicola Stokes, and Timothy Baldwin. 2006. Detecting Entailment Using an Extended Implementation of the Basic Elements Overlap Metrics. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, pages 122–127, Venice, Italy.

Rion Snow, Lucy Vanderwende, and Arul Menezes. 2006. Effectively using syntax for recognizing false entailment. In *Proceedings of the North American Association of Computational Linguistics*, New York City, New York, United States of America.