

PeMoZa submission to TAC 2008

Fabio Massimo Zanzotto
DISP, University of Rome “Tor Vergata”, Roma, Italy
zanzotto@info.uniroma2.it

Marco Pennacchiotti
Computerlinguistik, Universität des Saarlandes, Saarbrücken, Germany
pennacchiotti@coli.uni-sb.de

Alessandro Moschitti
DISI, University of Trento, Povo di Trento, Italy
moschitti@disi.unitn.it

February 2, 2009

Abstract

In this paper we describe the PeMoZa system participating to the fourth Recognizing of Textual Entailment (RTE) challenge. The major novelties with respect to our systems of the RTE3 challenge is the exploration on combining different data sets, coming from different challenges and from automatically acquired corpora.

1 Introduction

The design of our RTE4 system capitalizes our previous experiences in Recognizing of Textual Entailment challenges (RTE2 [1] and RTE3 [6]). It is based on a machine learning model that automatically derives first-order (logic) rules from annotated examples. In contrast with the previous challenges [4, 1, 6], the participants were not provided with a development set. This motivated our study on the use of data coming from the different challenges to improve the accuracy of our system. In such work we considered (a) previous work showing the failure on the use of data merged from different challenges and (b) the exploitation of automatically acquired datasets.

In this paper, we introduce the model in Sec. 2 and present the experiments and the results in Sec. 3.

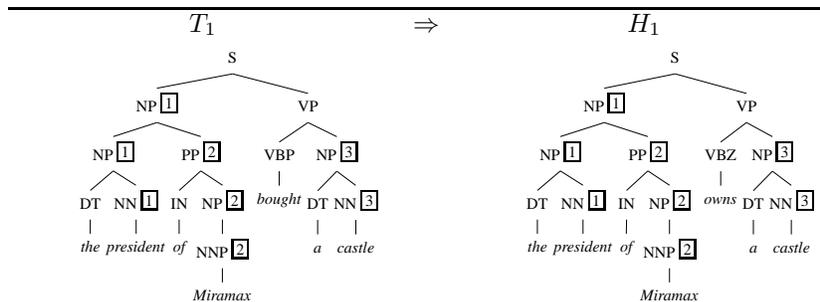


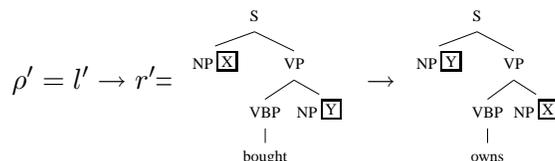
Figure 1: A syntactically analyzed textual entailment pair

2 Learning first-order rewrite rules from examples

Our model is based on a feature space, which can represent first-order syntactic rewrite rules as described in [12]. This section describes such space (Sec. 2.1) as well as the kernel function which implicitly defines such space (Sec. 2.2.1).

2.1 First-order syntactic rewrite rules feature space

In the feature space of first-order syntactic rewrite rule (FOSR), each feature f_ρ represents a syntactic first-order or grounded rewrite rule ρ . For example, the rule:



is represented with the feature $\langle l', r' \rangle$. A (T, H) pair p activates a feature f_ρ if it unifies with the rule ρ . For example, the above feature $f_{\rho'}$ is activated for the example in Fig. 1.

2.2 Kernels for the FOSR feature space

Since the full FOSR feature space has an exponential number of features, we use the kernel trick to optimize computations. It consists in defining the scalar product $K(i_1, i_2)$ between two instances i_1 and i_2 in such space, instead of first defining the function \mathcal{F} mapping instances in the feature space, i.e., $\mathcal{F}(i_1)$ and $\mathcal{F}(i_2)$ and then computing the distance. This is possible because kernel-machines, e.g. SVMs, only use $K(i_1, i_2)$ and not directly the feature values.

substructures shared by two trees τ_1 and τ_2 , the whole fragment space is not used. In the following, we report the formal definition presented in [3].

Given the set of fragments $\{f_1, f_2, \dots\} = \mathcal{F}$, the indicator function $I_i(n)$ is equal to 1 if the target f_i is rooted at node n and 0 otherwise. A tree kernel is then defined as:

$$TK(\tau_1, \tau_2) = \sum_{n_1 \in N_{\tau_1}} \sum_{n_2 \in N_{\tau_2}} \Delta(n_1, n_2) \quad (1)$$

where N_{τ_1} and N_{τ_2} are the sets of the τ_1 's and τ_2 's nodes, respectively and $\Delta(n_1, n_2) = \sum_{i=1}^{|\mathcal{F}|} I_i(n_1)I_i(n_2)$. This latter is equal to the number of common fragments rooted in the n_1 and n_2 nodes and Δ can be evaluated with the following algorithm:

1. if the productions at n_1 and n_2 are different then $\Delta(n_1, n_2) = 0$;
2. if the productions at n_1 and n_2 are the same, and n_1 and n_2 have only leaf children (i.e. they are pre-terminal symbols) then $\Delta(n_1, n_2) = 1$;
3. if the productions at n_1 and n_2 are the same, and n_1 and n_2 are not pre-terminals then

$$\Delta(n_1, n_2) = \prod_{j=1}^{nc(n_1)} (1 + \Delta(c_{n_1}^j, c_{n_2}^j)) \quad (2)$$

where $nc(n_1)$ is the number of the children of n_1 and c_n^j is the j -th child of the node n . Note that, since the productions are the same, $nc(n_1) = nc(n_2)$.

Additionally, we add the decay factor λ by modifying steps (2) and (3) as follows¹:

2. $\Delta(n_1, n_2) = \lambda$,
3. $\Delta(n_1, n_2) = \lambda \prod_{j=1}^{nc(n_1)} (1 + \Delta(c_{n_1}^j, c_{n_2}^j))$.

The computational complexity of Eq. 1 is $O(|N_{\tau_1}| \times |N_{\tau_2}|)$ although the average running time tends to be linear [10].

The next section shows a technique to assign the same placeholders to similar text and hypothesis pair.

¹To have a similarity score between 0 and 1, we also apply the normalization in the kernel space, i.e. $K'(\tau_1, \tau_2) = \frac{TK(\tau_1, \tau_2)}{\sqrt{TK(\tau_1, \tau_1) \times TK(\tau_2, \tau_2)}}$.

where (1) $t(\cdot, c)$ returns the syntactic tree enriched with placeholders replaced by means of the substitution c , (2) i is the identity substitution and (3) $TK(\tau_1, \tau_2)$ is a tree kernel function (e.g. the one specified by Eq. 1) applied to the two trees τ_1 and τ_2 .

At the same time, the desired similarity value to be used in the learning algorithm is given by $TK(t(T, c_{max}), t(T', i)) + TK(t(H, c_{max}), t(H', i))$, i.e. by solving the following optimization problem:

$$K_p(\langle T, H \rangle, \langle T', H' \rangle) = \max_{c \in C} (TK(t(T, c), t(T', i)) + TK(t(H, c), t(H', i))), \quad (3)$$

As a final remark, it should be noted that, (a) $K_s(\langle T, H \rangle, \langle T', H' \rangle)$ is a symmetric function since the set of derivation C are always computed with respect to the pair that has the largest anchor set and (b) it is not a valid kernel as the *max* function does not in general produce valid kernels. However, in [7], it is shown that when kernel functions are not positive semidefinite like in this case, SVMs still solve a data separation problem in pseudo Euclidean spaces. The drawback is that the solution may be only a local optimum. Nevertheless, such a solution can still be valuable as the problem is modeled with a very rich feature space.

3 Preliminary analysis

We study how effectively using data from different challenges and automatically acquired corpora. Our preliminary analysis demonstrates that (a) naively merging data is not effective and (b) the use of automatically acquired corpora decreases accuracy.

3.1 Experimental Setup

For our experiments, we used the following sets:

news: a corpus of 1600 examples obtained using the methods described for building the LCC corpus, both for the negative and positive examples [8].² We randomly divided the corpus in two parts: 800 training and 800 testing examples. Each set contains an equal number of 400 positive and negative pairs.

RTE1, RTE2, and RTE3: the corpora from the first three RTE challenges [4, 1, 6]. We used the standard split between development and testing, where the former is used for training.

²For negative examples, we adopt the headline - first paragraph extraction methodology.

<i>Training Corpus</i>	<i>Accuracy</i>
RTE2	60.62
RTE1	51.25
RTE3	57.25
news	53.25
RTE2+RTE1	58.5
RTE2+RTE3	59.62
RTE2+news	56.75

Table 1: Accuracy of different training corpora over RTE2 test

We use the Charniak Parser [2] for parsing sentences, and SVM-light [9] extended with the syntactic first-order rule kernels described in [12, 11]. Additionally, we used the lexical overlap similarity (*lex* model) score described in (Corley and Mihalcea, 2005).

3.2 Experimental Results

For the exploratory experiments, we used the FORS feature space described in Section 2. The first goal of the experiment is to check the quality of the automatically acquired corpus. We then independently experiment with the *news* corpora with the standard training-test splits as reported above. The accuracy of the system is 94.875% on the *news* corpus. The *news* corpus is very easy to separate: pilot experiments show that when increasing the size of the *news* corpus the accuracy reaches nearly 100%. This suggests that positive examples are very differently from negative ones. Indeed, we note that the lexical overlap in the negative examples is extremely low. This makes such dataset not really representative for the entailment phenomenon.

As a second step, we tested the use of RTE corpora from different challenges. Some experiments (e.g. [5]) show that RTE corpora are usually not homogeneous making difficult their joint exploitation. Following such work, we used RTE2 test set and different training sets obtained as combination of the remaining RTE sets. Results are reported in Table 1. The best result is achieved by training and testing on RTE2 (second row). As expected, the models learnt on RTE1 and RTE3 perform worse than on RTE2. RTE1 is extremely different from RTE2 according to our FOSR feature space. It is interesting to notice that all the experimented extensions of the RTE2 training lead to a drop in accuracy, suggesting that none of the corpora is homogeneous to RTE2. Yet, the performance drop of the *news* corpus (RTE2 + news) is much larger than when using the other two RTE corpora (i.e. RTE2 + RTE1 and RTE2 + RTE3). This suggests that *news* is very different from RTE and

Run	Training	Accuracy	Average Precision
1	RTE1+RTE2+RTE3	0.563	0.5619
2	RTE2+RTE3	0.59	0.6287
2	RTE3	0.586	0.603

Table 2: Result submission

cannot be used to improve our systems.

3.3 Submission results

The results of the submission are presented in Tab. 2. Here the model used is the FORS feature space combined with the lexical feature space. As expected the results are lower than those obtained in the other years since we expected RTE4 test data different from previous the one developed previous challenges. The best result has been obtained excluding RTE1 from the learning set.

Acknowledgments

This work has been partially supported by the European Commission - LIVING-KNOWLEDGE project, Grant agreement no.: 231126.

References

- [1] Roy Bar-Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, and Idan Magnini, Bernardo Szpektor. The second pascal recognising textual entailment challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*. Venice, Italy, 2006.
- [2] Eugene Charniak. A maximum-entropy-inspired parser. In *Proc. of the 1st NAACL*, pages 132–139. Seattle, Washington, 2000.
- [3] Michael Collins and Nigel Duffy. New ranking algorithms for parsing and tagging: Kernels over discrete structures, and the voted perceptron. In *Proceedings of ACL02*. 2002.
- [4] Ido Dagan, Oren Glickman, and Bernardo Magnini. The pascal recognising textual entailment challenge. In Quionero-Candela et al., editor, *LNAI 3944: MLCW 2005*, pages 177–190. Springer-Verlag, Milan, Italy, 2006.

- [5] Marie-Catherine de Marneffe, Bill MacCartney, Trond Grenager, Daniel Cer, Anna Rafferty, and Christopher D. Manning. Learning to distinguish valid textual entailments. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, Venice, Italy, 2006.
- [6] Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. The third pascal recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 1–9. Association for Computational Linguistics, Prague, June 2007.
- [7] Bernard Haasdonk. Feature space interpretation of SVMs with indefinite kernels. *IEEE Trans Pattern Anal Mach Intell*, 27(4):482–92, Apr 2005.
- [8] Andrew Hickl, John Williams, Jeremy Bensley, Kirk Roberts, Bryan Rink, and Ying Shi. Recognizing textual entailment with LCCs GROUNDHOG system. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*. Venice, Italy, 2006.
- [9] Thorsten Joachims. Making large-scale svm learning practical. In B. Schlkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods-Support Vector Learning*. MIT Press, 1999.
- [10] Alessandro Moschitti. Efficient Convolution Kernels for Dependency and Constituent Syntactic Trees. In *Proceedings of The 17th European Conference on Machine Learning*. Berlin, Germany, 2006.
- [11] Alessandro Moschitti and Fabio Massimo Zanzotto. Fast and effective kernels for relational learning from texts. In *Proceedings of the International Conference of Machine Learning (ICML)*. Corvallis, Oregon, 2007.
- [12] Fabio Massimo Zanzotto and Alessandro Moschitti. Automatic learning of textual entailments with cross-pair similarities. In *Proceedings of the 21st Coling and 44th ACL*, pages 401–408. Sydney, Australia, July 2006.