UPC: Experiments with Joint Learning within SemEval Task 9

Lluís Màrquez, Lluís Padró, Mihai Surdeanu, Luis Villarejo
Technical University of Catalonia
{lluism,padro,surdeanu,luisv}@lsi.upc.edu

1 Introduction

This paper describes UPC's participation in the SemEval-2007 task 9 (Màrquez et al., 2007).[1] We addressed all four subtasks using supervised learning. The paper introduces several novel issues: (a) for the SRL task, we propose a novel re-ranking algorithm based on the re-ranking Perceptron of Collins and Duffy (2002); and (b) for the same task we introduce a new set of global features that extract information not only at proposition level but also from the complete set of frame candidates. We show that in the SemEval setting, i.e., small training corpora, this approach outperforms previous work. Additionally, we added NSD and NER information in the global SRL model but this experiment was unsuccessful.

2 Named Entity Recognition

For the NER subtask we recognize first strong NEs, followed by weak NE identification. Any single token with the `np0000`, `W`, or `Z` PoS tag is considered a strong entity and is classified using the (Atserias et al., 2006) implementation of a multi-label AdaBoost.MH algorithm, with a configuration similar to the NE classification module of Carreras et al. (2003). The classifier yields predictions for four classes (person, location, organization, misc). Entities with NUM and DAT are detected separately solely based on POS tags.

The features used by the strong NE classifier model a [-3,+3] context around the focus word, and include bag-of-words, positional lexical features, PoS tags, orthographic features, as well as features indicating whether the focus word, some of its components, or some word in the context are included in external gazetteers or *trigger words* files.

The second step starts by selecting all noun phrases (`np`) that cover a span of more than one token and include a strong NE as weak entity candidates. This strategy covers more than 95% of the weak NEs. A second AdaBoost.MH classifier is then applied to decide the right class for the noun phrase among the possible six (person, location, organization, misc, number, date) plus a *NONE* class indicating that the noun phrase is not a weak NE.

The features used for weak NE classification are: (1) *simple features* – length in tokens, head word, lemma, and POS of the `np`, syntactic function of the `np` (if any), minimum and maximum number of `np` nodes in the path from the candidate noun phrase to any of the strong NEs included in it, and number and type of the strong NEs predicted by the first–level classifier that fall inside the candidate; (2) *bag of content words* inside the candidate; and (3) *pattern-based features*, consisting in codifying the sequence of lexical tokens spanned by the candidate according to some generalizations. When matching, tokens are generalized to: the POS tag (in case of `np0000`, `W`, `Z`, and punctuation marks), *trigger-word* of class X, *word-in-gazetteer* of class X, and *strong-NE* of type X, predicted by the first level classifier. The rest of words are abstracted to a common form ("w" standing for a single word and "w+" standing for a sequence of $n > 1$ words). Beginning and end of the span are also codified explicitly in the pattern–based features. Finally, to avoid sparsity, only paths of up

[1] Two of the authors of this paper, Lluís Màrquez and Luis Villarejo, are organizers of the SemEval-2007 task 9.

Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007), pages 426–429,
Prague, June 2007. ©2007 Association for Computational Linguistics

to length 6 are codified as features. Also, for each path, n–grams of length 2, 3 and 4 are considered. We filter out features that occur less than 10 times.

3 Noun Sense Disambiguation

We have approached the NSD subtask using supervised learning. In particular, we used SVM$^{\text{light}}$ (Joachims, 1999), which is a freely available implementation of Support Vector Machines (SVM).

We trained binary SVM classifiers for every sense of words with more than 15 examples in the training set and a probability distribution over its senses in which no sense is above 90%. The words not covered by the SVM classifiers are disambiguated using the most frequent sense (MFS) heuristic. The MFS was calculated from the relative frequencies in the training corpus. To the words that do not appear in the training corpus we assigned the first WordNet sense.

We used a fairly regular set of features from the WSD literature. We included: (1) a bag of content words appearing in a $\pm$10-word window; (2) a bag of content words appearing in the clause of the target word; (3) $\{1, \ldots, n\}$–grams of POS tags and lemmas in a $\pm n$-word window (n is 3 for POS and 2 for lemmas); (4) unigrams and bigrams of (POS-tag,lemma) pairs in a $\pm$2-word window; and (5) syntactic features, i.e., label of the syntactic constituent from which the target noun is the head, syntactic function of that constituent (if any), and the verb.

Regarding the empirical setting, we filtered out features occurring less than 3 times, we used linear SVMs with a 0.5 value for the C regularization parameter (trade-off between training error and margin), and we applied one-vs-all binarization.

4 Semantic Role Labeling

The SRL approach deployed here implements a re-ranking strategy that selects the best argument frame for each predicate from the top N frames generated by a base model. We describe the two models next.

4.1 The Local Model

The local (i.e., base) model is an adaption of Model 3 of Màrquez et al. (2005). This SRL approach maps each frame argument to one syntactic constituent and trains one-vs-all AdaBoost (Schapire and Singer, 1999) classifiers to jointly identify and classify constituents in the full syntactic tree of the sentence as arguments. The model was adapted to the languages and corpora used in the SemEval evaluations by removing the features that were specific either to English or PropBank (governing category, content word, and temporal cue words) and adding several new features: (a) *syntactic function* features – the syntactic functions available in the data often point to specific argument labels (e.g., `SUJ` usually indicates an `Arg0`); and (b) *back-off* features for syntactic labels and POS tags – for the features that include POS tags or syntactic labels we add a back-off version of the feature where the POS tags and syntactic labels are reduced to a small set.

In addition to feature changes we modified the candidate filtering heuristic: we select as candidates only syntactic constituents that are immediate descendents of `S` phrases that include the corresponding predicate (for both languages, over 99.6% of the candidates match this constraint).

4.2 The Global Model

We base our re-ranking approach on a variant of the re-ranking Perceptron of Collins and Duffy (2002). We modify the original algorithm in two ways to make it more robust to the small training set available: (a) instead of comparing the score of the correct frame only with that of the best candidate for each frame, we sequentially compare it with the score of *each* candidate in order to acquire more information, and (b) we learn not only when the prediction is incorrect but also when the prediction is not confident enough.

The algorithm is listed in Algorithm 1: $\mathbf{w}$ is the vector of model parameters, $\mathbf{h}$ generates the feature vector for one example, and $\mathbf{x_{ij}}$ denotes the jth candidate for the ith frame in the training data. $\mathbf{x_{i1}}$, which denotes the "correct" candidate for frame i, is selected to maximize the F_1 score for each frame. The algorithm sequentially inspects all candidates for each frame and learns when the difference between the scores of the correct and the current candidate is less than a threshold τ. During testing we use the average of all acquired model vectors, weighted by the number of iterations they survived in training. We tuned all system parameters through cross-validation on the training data. For both languages we set $\tau = 10$ (we do not normalize feature vectors)

Algorithm 1: Re-ranking Perceptron

$\mathbf{w} = \vec{0}$
for $i = 1$ **to** n **do**
  for $j = 2$ **to** n_i **do**
    if $\mathbf{w} \cdot \mathbf{h}(\mathbf{x_{ij}}) > \mathbf{w} \cdot \mathbf{h}(\mathbf{x_{i1}}) - \tau$ **then**
      $\mathbf{w} \leftarrow \mathbf{w} + \mathbf{h}(\mathbf{x_{i1}}) - \mathbf{h}(\mathbf{x_{ij}})$

and the number of training epochs to 2.

With respect to the features used, we focus only on global features that can be extracted independently of the local models. We show in Section 6 that this approach performs better on the small SemEval corpora than approaches that include features from the local models. We group the features into two sets: (a) features that extract information from the whole candidate set, and (b) features that model the structure of each candidate frame:

Features from the whole candidate set:

(1) Position of the current candidate in the whole set. Frame candidates are generated using the dynamic programming algorithm of Toutanova et al. (2005), and then sorted in descending order of the log probability of the whole frame (i.e., the sum of all argument log probabilities as reported by the local model). Hence, smaller positions indicate candidates that the local model considers better.

(2) For each argument in the current frame, we store its number of repetitions in the whole candidate set. The intuition is that an argument that appears in many candidate frames is most likely correct.

Features from each candidate frame:

(3) The complete sequence of argument labels, extended with the predicate lemma and voice, similar to Toutanova et al. (2005).

(4) Maximal overlap with a frame from the verb lexicon. Both the Spanish and Catalan TreeBanks contain a static lexicon that lists the accepted sequences of arguments for the most common verbs. For each candidate frame, we measure the maximal overlap with the lexicon frames for the given verb and use the precision, recall, and F_1 scores as features.

(5) Average probability (from the local model) of all arguments in the current frame.

(6) For each argument label that repeats in the current frame, we add combinations of the predicate lemma, voice, argument label, and the number of label repetitions as features. The intuition is that argument repetitions typically indicate an error (even if allowed by the domain constraints).

5 Semantic Class Detection

The semantic class detection subtask has been performed using a naive cascade of heuristics: (1) the predicted frame for each verb is compared with the frames present in the provided verbal lexicon, and the class of the lexicon frame with the largest number of matching arguments is chosen; (2) if there is more than one verb with the maximum score, the first one in the lexicon (i.e., the most frequent) is used; (3) if the focus verb is not found in the lexicon, its most frequent class in the training corpus is used; (4) if the verb does not appear in the training data, the most frequent class overall (D2) is assigned. The results obtained on the training corpus are 81.1% F_1 for Spanish and 86.6% for Catalan. As a baseline, assigning the most frequent class for each verb (or D2 if not seen in training), yields F_1 values of 48.1% for Spanish and 64.0% for Catalan.

6 Results and Discussion

Table 1 lists the results of our system on the SemEval test data. Our results are encouraging considering the size of the training corpus (e.g., the English PropBank is 10 times larger than the corpus used here) and the complexity of the problem (e.g., the NER task includes both weak and strong entities; the SRL task contains 33 core arguments for Spanish vs. 6 for English). We analyze the behavior of our system next.

The first issue that deserves further analysis is the contribution of our global SRL model. We list the results of this analysis in Table 2 as improvements over the local SRL model. We report results for 6 corpora: the 4 test corpora and the 2 training corpora, where the results are generated through 5-fold cross validation. The first block in the table shows the contribution of our best re-ranking model. The second block shows the results of a re-ranking model using our best feature set but the original re-ranking Perceptron of Collins and Duffy (2002). The third block shows the performance of our re-ranking algorithm configured with the features proposed by Toutanova et al. (2005). We draw several conclusions from this experiment: (a) our re-ranking model

	NER			NSD			SRL			SC
	P	R	F_1	P	R	F_1	P	R	F_1	F_1
ca.CESS-ECE	79.92%	76.63%	78.24	87.47%	87.47%	87.47	82.16%	70.05%	75.62	85.71
es.CESS-ECE	72.53%	68.48%	70.45	83.30%	83.30%	83.30	86.24%	75.58%	80.56	87.74
ca.3LB	82.04%	79.42%	80.71	85.69%	85.53%	85.61	86.36%	85.30%	85.83	87.35
es.3LB	62.03%	53.85%	57.65	88.14%	88.14%	88.14	82.23%	80.78%	81.50	76.01

Table 1: Official results on the test data. Due to space constraints, we show only the F_1 score for SC.

	Re-ranking			Collins			Toutanova		
	P	R	F_1	P	R	F_1	P	R	F_1
ca.train	**+1.87**	**+1.79**	**+1.83**	+1.56	+1.48	+1.52	-6.81	-6.67	-6.73
es.train	**+3.16**	**+3.12**	**+3.14**	+2.96	+2.93	+2.95	-6.51	-6.96	-6.75
ca.CESS-ECE	+0.77	+0.66	+0.71	**+0.99**	**+0.84**	**+0.91**	-8.11	-6.29	-7.10
es.CESS-ECE	**+1.85**	**+1.94**	**+1.91**	+1.45	+1.85	+1.68	-10.84	-8.46	-9.54
ca.3LB	**+1.58**	**+1.47**	**+1.53**	+1.48	+1.39	+1.44	-7.71	-7.57	-7.64
es.3LB	+2.57	+2.83	+2.71	**+2.71**	**+2.91**	**+2.82**	-10.53	-11.95	-11.26

Table 2: Analysis of the re-ranking model for SRL.

using only global information always outperforms the local model, with F_1 score improvements ranging from 0.71 to 3.14 points; (b) the re-ranking Perceptron proposed here performs better than the original algorithm, but the improvement is minimal; and (c) the feature set proposed here achieve significant better performance on the SemEval corpora than the set proposed by Toutanova et al., which never improves over the local model. The model configured with the Toutanova et al. feature set performs modestly because the features are too sparse for the small SemEval corpora (e.g., all features from the local model are included, concatenated with the label of the corresponding argument). On the other hand, we replicate the behavior of the local model just with feature (1), and furthermore, all the other 5 global features proposed have a positive contribution.

In a second experiment we investigated simple strategies for model combination. We incorporated NER and NSD information in the re-ranking model for SRL as follows: for each frame argument, we add features that concatenate the predicate lemma, the argument label, and the NER or NSD labels for the argument head word (we add features both with and without the predicate lemma). We used only the best NER/NSD labels from the local models. To reduce sparsity, we converted word senses to coarser classes based on the corresponding WordNet semantic files. This new model boosts the F_1 score of our best re-ranking SRL model with an average of 0.13 points on two corpora (es.3LB and ca.CESS-ECE), but it reduces the F_1 of our best SRL model with an average of 0.17 points on the other 4 corpora. We can conclude that, in the current setting, NSD and NER do not bring useful information to the SRL problem. However, it is soon to state that problem combination is not useful. To have a conclusive answer one will have to investigate true joint learning of the three subtasks.

References

J. Atserias, B. Casas, E. Comelles, M. Gonzàlez, L. Padró, and M. Padró. 2006. Freeling 1.3: Syntactic and semantic services in an open-source NLP library. In *Proc. of LREC*.

X. Carreras, L. Màrquez, and L. Padró. 2003. A simple named entity extractor using AdaBoost. In *CoNLL 2003 Shared Task Contribution*.

M. Collins and N. Duffy. 2002. New ranking algorithms for parsing and tagging: Kernels over discrete structures, and the voted perceptron. In *Proc. of ACL*.

T. Joachims. 1999. *Making large-scale SVM learning practical, Advances in Kernel Methods - Support Vector Learning*. MIT Press, Cambridge, MA.

L. Màrquez, M. Surdeanu, P. Comas, and J. Turmo. 2005. A robust combination strategy for semantic role labeling. In *Proc. of EMNLP*.

L. Màrquez, M.A. Martí, M. Taulé, and L. Villarejo. 2007. SemEval-2007 task 09: Multilevel semantic annotation of Catalan and Spanish. In *Proc. of SemEval-2007, the 4th Workshop on Semantic Evaluations. Association for Computational Linguistics*.

R.E. Schapire and Y. Singer. 1999. Improved boosting algorithms using confidence-rated predictions. *Machine Learning*, 37(3).

K. Toutanova, A. Haghighi, and C. Manning. 2005. Joint learning improves semantic role labeling. In *Proc. of ACL*.