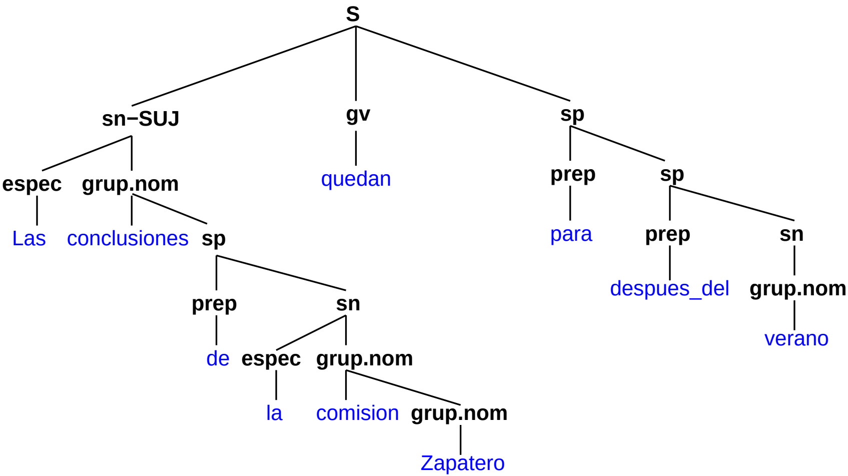
UPC: Experiments with Joint Learning within SemEval Task 9

Lluís Màrquez, Lluís Padró, Mihai Surdeanu, Luis Villarejo

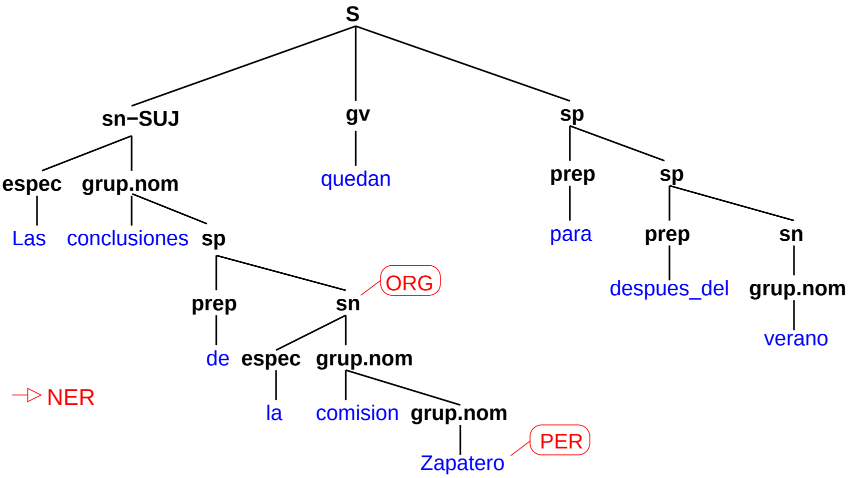
Technical University of Catalonia
{lluism,padro,surdeanu,luisv}@lsi.upc.edu

23rd June 2007

Task overview

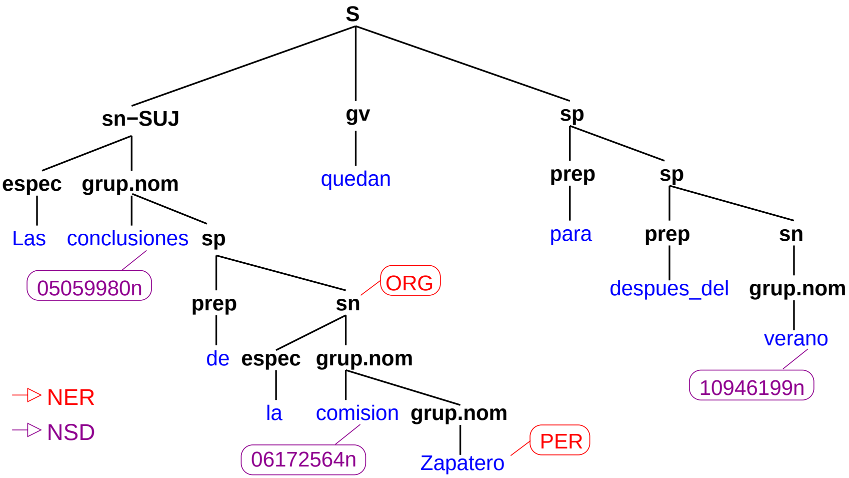


Task overview

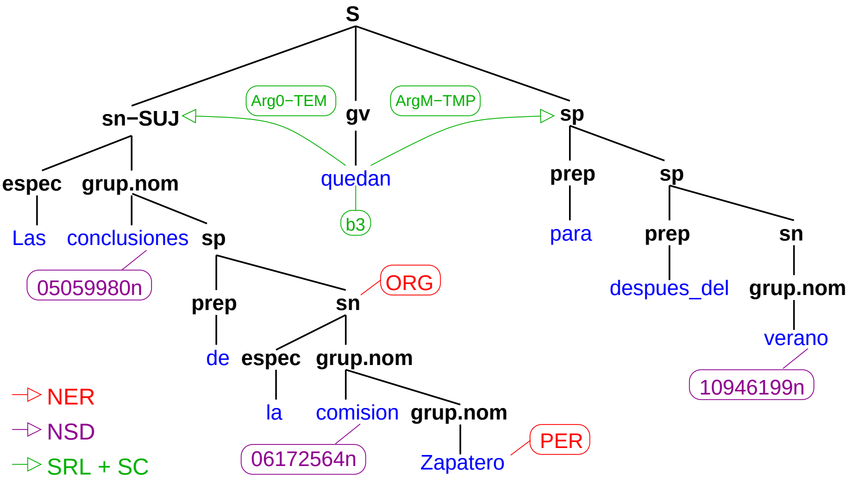


NER

Task overview



Task overview



Why is this different?

- Differences from PropBank:
  - More input information: more syntactic labels, syntactic functions (`SUJ`, `OD`, `OI`).
  - But, the corpus is significantly smaller: es 10.5 times smaller, ca 9.8 times smaller.
  - Plus, we have to label more argument types (SRL): es - 46 and ca - 36 versus en - 21.
- What it means: sparsity and overfitting significant problems in this setup.
- Solutions (for SRL):
  - Elegant: novel re-ranking approach that extracts global information from each proposition and from *all* candidates for one proposition $\rightarrow$ less sparse than the local model.
  - Ugly: learned post-processing rules to correct `ArgM` arguments $\rightarrow$ increase coverage for these arguments (post eval).

Outline

Classification of strong and weak entities

1. Strong entities:
   - Candidates: all tokens with the POS tags `np0000`, `W`, `Z`.
   - Classifier: multi-label AdaBoost.
   - Features: lexical, POS, trigger words, gazetteers; in a $[-3, +3]$ context window.
2. Weak entities:
   - Candidates: noun phrases (`np`) that have span > 1 and include one strong NE. Coverage: $> 95\%$ of weak NEs.
   - Classifier: multi-label AdaBoost.
   - Features:
     - Simple features: length, lexical and POS head information, strong NE information (number and type, `np` count in path to strong NE), syntactic function.
     - Bag of content words inside the candidate.
     - Pattern-based features $\rightarrow$ codify the sequence of tokens inside the candidate. Tokens are generalized to: (a) POS (`np0000`, `W`, `Z`), trigger word of class X, gazetteer of class X, strong NE of class X, `w`, or `w+`.

Outline

The disambiguation approach

- Three step approach:
  1. Words that appear > 15 times in training and have a probability distribution in which no sense is over 90%: trained linear SVM classifiers.
  2. Words that appear in training: select most frequent sense in training.
  3. Words unseen in training: most frequent WordNet sense.
- Features of the SVM classifiers:
  - Bag of words in a $[-10, +10]$ context window.
  - Bag of words in the clause of the target word.
  - $\{1, \ldots, n\}$–grams of POS tags and lemmas in a $[-n, +n]$ window (n is 3 for POS and 2 for lemmas).
  - Unigrams and bigrams of (POS, lemma) tuples in a $[-2, +2]$ window.
  - Syntactic label and syntactic function of the constituent that has the target noun as head.

Outline

The local model

- Overall SRL approach: re-ranking strategy that selects the best argument frame for each predicate from the top *N* frames generated by a base model.
- The local model:
  - Adapted from the English SRL system SwiRL: `http://www.lsi.upc.edu/~surdeanu/swirl.html`.
  - Removed features:
    - The governing category (does not apply to this corpus: objects not attached to the verb).
    - Content word (rules were English Treebank specific) $\longrightarrow$ bad decision...
    - Temporal cue words (English specific).
  - New features:
    - Syntactic functions. E.g., `SUJ` usually indicates an `Arg0`.
    - Back-off features for syntactic labels and POS tags.
  - Candidate selection: constituents that are *immediate* descendents of any S phrase that includes the target predicate. Coverage: over 99.6% of arguments.

Re-ranking Perceptron

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**
      $\mathbf{w} \leftarrow \mathbf{w} + \mathbf{h}(\mathbf{x_{i1}}) - \mathbf{h}(\mathbf{x_{ij}})$

- **w** is the vector of model parameters.
- **h** generates the feature vector for one example.
- $\mathbf{x_{i1}}$ is the "best" candidate for frame *i* $\rightarrow$ maximizes F_1.

- Two changes:
  1. We compare the score of the best candidate ($\mathbf{x_{i1}}$) with *each* candidate not just the current prediction $\rightarrow$ acquire more information.
  2. We learn not only when the prediction is incorrect but also when it is not confident enough (delta $<$ threshold τ) $\rightarrow$ large margin re-ranking.

Features of the global model (1/2)

Features from the whole candidate set:

1. Position of the current candidate in the list ordered by log probability of the whole frame (as reported by the local model) $\longrightarrow$ smaller positions indicate candidates that the local model considers better.
   - Frame candidates generated using the dynamic programming algorithm of Toutanova et al. (2005).
   - This feature is sufficient to replicate the behavior of the local model!
2. For each argument in the current frame, we store its number of repetitions in the whole candidate set $\longrightarrow$ intuition: an argument that appears in many candidate frames is most likely correct.

Features of the global model (2/2)

Features from each candidate frame:

3. The complete sequence of argument labels, extended with the predicate lemma and voice, same as Toutanova et al. (2005).
4. Maximal overlap with a frame from the lexicon of the target predicate.
   - Corpus lexicon lists the accepted frames for each verb.
   - Use precision, recall and F_1 of maximal overlap as features.
5. Average probability (from the local model) of all arguments in the current frame.
6. For each argument that repeats in the frame $\rightarrow$ tuples of (predicate lemma, predicate voice, argument label, number of repetitions). Intuition: argument repetitions typically indicate an error.

Outline

Class detection through cascade of heuristics

1. For each verb the best frame predicted compared with the lexicon frames $\longrightarrow$ select class with the largest number of matching arguments.
2. Disambiguate ties by picking the most frequent class.
3. If the target verb not found in lexicon $\longrightarrow$ use the most frequent class of the verb in training.
4. If the target verb does not appear in training $\longrightarrow$ assign the most frequent class overall (`D2`).

Outline

Official results

	NER			NSD	SRL			SC
	P	R	F_1	A	P	R	F_1	A
ca.out	79.92%	76.63%	78.24	87.47%	82.16%	70.05%	75.62	85.71%
es.out	72.53%	68.48%	70.45	83.30%	86.24%	75.58%	80.56	87.74%
ca.in	82.04%	79.42%	80.71	85.61%	86.36%	85.30%	85.83	87.35%
es.in	62.03%	53.85%	57.65	88.14%	82.23%	80.78%	81.50	76.01%
overall	76.93%	73.08%	74.96	85.87%	84.18%	78.24%	81.10	83.86%

- Encouraging results considering the complexity of the problem and the size of the corpus. NER 50+ F_1 points over task baseline. NSD 1 point over. SRL score over 80.
- Recall remains the largest problem, e.g., 70.05% for SRL in ca.out.

SRL analysis: contribution of global model (1/2)

	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.out	+0.77	+0.66	+0.71	**+0.99**	**+0.84**	**+0.91**	-8.11	-6.29	-7.10
es.out	**+1.85**	**+1.94**	**+1.91**	+1.45	+1.85	+1.68	-10.84	-8.46	-9.54
ca.in	**+1.58**	**+1.47**	**+1.53**	+1.48	+1.39	+1.44	-7.71	-7.57	-7.64
es.in	+2.57	+2.83	+2.71	**+2.71**	**+2.91**	**+2.82**	-10.53	-11.95	-11.26

- Three models compared:
    1. *Re-ranking*: our best model.
    2. *Collins*: our best feature set + re-ranking Perceptron of Collins and Duffy (2002).
    3. *Toutanova*: our best re-ranking Perceptron + feature set of Toutanova et al. (2005).

SRL analysis: contribution of global model (2/2)

- Our re-ranking model using only global information always outperforms the local model, with F_1 score improvements ranging from 0.71 to 3.14 points.
- The re-ranking Perceptron proposed here performs better than the original algorithm, but the improvement is small (0.51 F_1 points overall).
- The feature set proposed here achieve significant better performance on the SemEval corpora than the feature set of the current state of the art.
  - The Toutanova et al. feature set too sparse for SemEval.
  - We replicate the behavior of the local model just with feature (1). All the other 5 global features proposed have a positive contribution.

Post eval: new features + bug fixes in SRL

	SRL		
	P	R	F_1
ca.out	82.17%	69.67%	75.41
es.out	86.63%	76.26%	81.12
ca.in	87.83%	86.61%	87.22
es.in	83.06%	81.47%	82.26
overall	84.92%	78.83%	81.76

	Contribution of re-ranking		
	P	R	F_1
ca.out	+1.44	+1.22	+1.32
es.out	+1.90	+2.33	+2.16
ca.in	+2.35	+2.18	+2.27
es.in	+4.33	+4.36	+4.35
overall	+2.67	+2.62	+2.64

- Bug fix: correct handling of Unicode strings.
- New feature: “content” word for prepositional phrases.

Post eval: post processing for TMP/LOC arguments

	SRL on ca.out		
	P	R	F_1
...			
ArgM-LOC	39.02%	14.29%	20.92
ArgM-TMP	33.33%	5.38%	9.26
...			
overall	82.17%	69.67%	75.41

1. Acquired all TMP and LOC rules with precision $>$ 60% in training. Rule LHS types: head word; head word + content word; head word + content POS tag.
2. Post process: unassigned candidates that match a rule assigned the corresponding argument label (TMP or LOC).

	SRL overall		
	P	R	F_1
local	82.25%	76.21%	79.12
+global	84.92%	78.83%	81.76 (+2.64)
+post process	85.26%	82.92%	84.07 (+2.31)

Conclusions

- A first approach for complex semantic analysis of Spanish and Catalan (NER, NSD, SRL, SC). Encouraging results considering the complexity of the problem and the size of the corpus.
- Proposed a novel re-ranking algorithm based on the re-ranking Perceptron of Collins and Duffy (2002): large margin support, explores all candidates not just the best prediction.
- Proposed a new model for re-ranking that extract information from each proposition and from the whole set of candidates.
- The proposed re-ranking SRL performs significantly better than the local model alone and the previous state of art in re-ranking SRL.
- Future work: LOC/TMP patterns should be included in the ML model; joint learning.

Thank you! Questions?