Pipeline and Reranker-based Multilingual Semantic Role Labeling

Contribution to the CoNLL 2009 Shared Task SRL-only track

Anders Björkelund, Love Hafdell, and Pierre Nugues
Dept. of Computer Science, Lund university, Sweden

Method Overview

- Pipeline of linear classifiers divided into three steps
- Beam search used to generate multiple parses
- Reranker and pipeline combined for final choice
- Classifiers are linear logistic classifiers trained using LibLinear (Fan et al 2008)
- Logistic classifiers output probabilities, used for beam search and combination of models

Pipeline

- Pipeline divided in three steps, inspired by Johansson and Nugues (2008)
- Classifiers are trained on predicted dependencies
- Specialized feature sets for each language

Predicate Disambiguation

- One classifier for each lemma
- Greedy search (no beam search)
- Default labels for unknown lemmas

Argument Identification

- Binary classifier outputs probability of being an argument
- No pruning – all tokens are considered
- Probability of an unlabeled proposition defined as
  P_{AI} = the product of the probabilities of each choice
- Beam search is used to select the top propositions with respect to the P_{AI} score. We used a beam width of *4*.

Features used

- Pool of 32 feature templates
- Greedy forward feature selection performed in each step for each language
- Incrementally adds best feature until no further gain is possible
- First adds unigram features, then bigram features

Argument Classification

- Multiclass classifier outputs probabilities for each label
- Composite labels treated as unique labels
- Probability of a labeling defined as
  P_{AC} = the product of the probabilities of every label
- Beam search is used to select the top propositions with respect to the P_{AC} score. We used a beam width of *4*.

Pipeline probability

- Probability of a labeled proposition defined as
  $P_{Local} = P_{AI} \cdot (P_{AC})^{1/a}$, where *a* is the number of arguments
- The geometric mean is applied to P_{AC} to avoid penalizing propositions with more arguments, hence the exponent.
- The probabilities of the complete candidate pool is normalized by dividing the probability of each proposition with the total sum. This normalized probability is denoted P'_{Local}

Reranker

- Similar to previous rerankers (Johansson and Nugues 2008; Toutanova et al 2008)
- Binary classifier that outputs probabilities on complete propositions
- Training data generated in a cross-validation manner using local subclassifers

Features used

- All features from local classifiers
- Argument Identification features prefixed with AI-
- Argument Classification features prefixed with *lab*-, where *lab* denotes the label
- Core Argument Label Sequence, ie the concatenation of core argument labels and predicate sense with respect to the word ordering, e.g.
  A0 + pred.02 + A2 + A1

Reranker Probability

- Single classifier used to score complete propositions
- Outputs probabilities on each proposition independently
- The probability of a proposition is denoted $P_{Reranker}$

Combination

- To select the best proposition from the pool of $4 \cdot 4 = 16$ candidates the reranker and pipeline probabilities are combined
- The final score of a proposition is defined as
  $P_{Final} = P'_{Local} \cdot (P_{Reranker})^{\alpha}$
- The proposition that maximizes P_{Final} is selected
- We used $\alpha = 1$ since it performed best on the development set

Conclusion

- Our system achieved the second best semantic score, both tracks.
- The method is rather simple and streamlined, and produces decent results even with greedy search.

Further Work

- Argument pruning should be considered.
- The potential in the beam search is much greater than what we achieve. We believe more could be gained from this. The reranker feature space, as well as the combination of pipeline and reranker probabilities, should probably be reconsidered.
- Incorporate the semantic lexicons, for predicate disambiguation as well as constrainting argument labels.

Results

- The table displays the performance of our system given using the labeled semantic F1 measure
- Greedy refers to beam widths set to 1, which is equivalent to pipeline only.
- For the submission to the Shared Task we used a wrongly trained reranker classifier for Spanish, yielding the poor results. Post-evaluation figures obtained after retraining this classifier are denoted by *.

	Greedy	Reranker	Gain
Catalan	79.54	80.01	0.47
Chinese	77.84	78.60	0.76
Czech	84.99	85.41	0.42
English	84.44	85.63	1.19
German	79.01	79.71	0.70
Japanese	75.61	76.30	0.69
Spanish	79.28	76.52	-2.76
Spanish*	79.28	79.91	0.63
Average	80.10	80.31	0.21
Average*	80.10	80.80	0.70

* denotes post-evaluation figures after bugfix