Parser lexicalisation through self-learning

Marek Rei & Ted Briscoe

Computer Laboratory, University of Cambridge, UK

UNIVERSITY OF CAMBRIDGE

BACKGROUND

- The use of lexically-conditioned features, such as relations between lemmas or word forms, is important for creating accurate parsers.
- However, utilising such features leads the parser to learn information that is often specific to the domain and/or genre of the training data. Furthermore, manual creation of in-domain treebanks is expensive and time-consuming.
- Unlexicalised parsers avoid using lexical information and select a syntactic analysis using only more general features, such as POS tags. While they cannot be expected to achieve optimal performance when trained and tested in a single domain, unlexicalised parsers can be surprisingly competitive with their lexicalised counterparts.
- Instead of trying to adapt a lexicalised parser to new domains, perhaps we can directly integrate lexical features with any unlexicalised parser.

HYPOTHESIS

- We hypothesise that a large corpus will often contain examples of dependency relations in non-ambiguous contexts, and these will mostly be correctly parsed by an unlexicalised parser.
- Lexical statistics derived from the corpus can then be used to select the correct parse in a more difficult context.
- This would allow the unlexicalised parser to learn lexical features directly from its own output, without any manual annotation.

EXAMPLES

Interest can be both noun or a verb, which can lead to ambiguous sentences:

- *Government raises interest rates*
- *Government projects interest researchers*

After learning from non-ambiguous cases, we could infer that *interest* is likely to modify *rates*:

- *Interest rates are increasing*
- *Government projects receive funding*

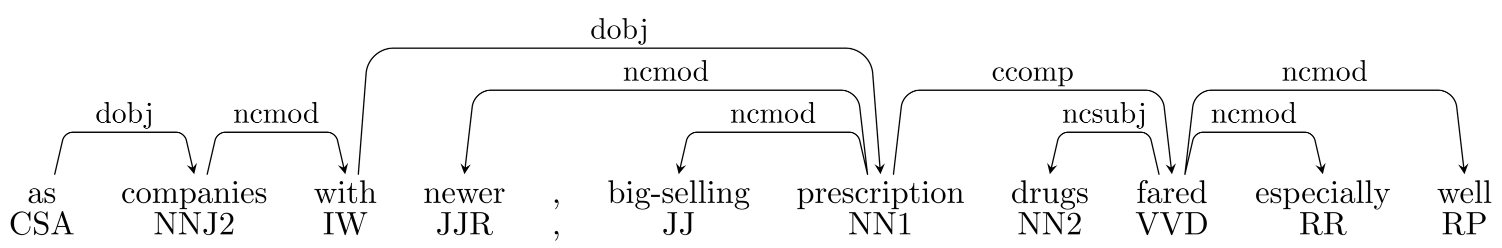


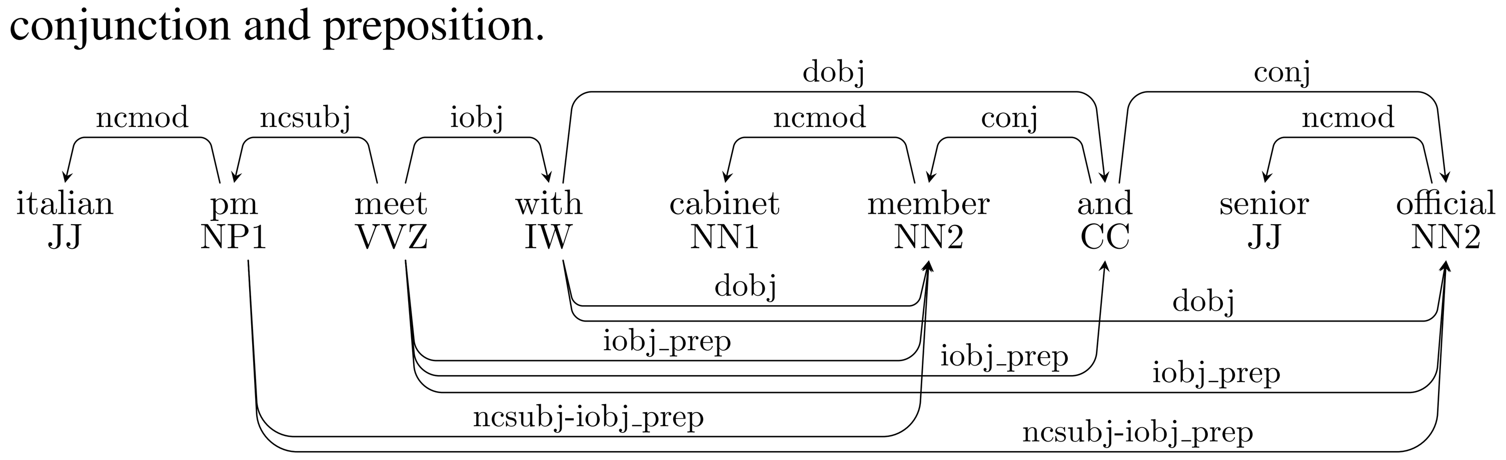
Figure1: Incorrect dependency graph found by the unlexicalised parser.

SYSTEM OVERVIEW

1. A large corpus of in-domain text is parsed with the unlexicalised parser.
2. New edges are added to dependency graphs, to model selected higher-order dependencies.
3. Maximum-likelihood probabilities are found for all lexical relations.
4. New confidence scores are calculated for alternative parses of each sentence.
5. Parses are reranked, improving the accuracy of the top parse.

GRAPH MODIFICATIONS

For every dependency graph g_r the graph expansion procedure creates a modified representation g'_r which contains a wider range of bilexical relations. We normalise the lemmas and create additional second-order edges for each verb, conjunction and preposition.



RELATION SCORING

Every edge in the modified graph is assigned a confidence score, using various methods:

- The probability of edge e belonging to the best possible parse, based on ranking from the unlexicalised parser:

$$\mathrm{RES}(e) = \frac{\sum_{r=1}^{R}[\frac{1}{r} \times \mathrm{contains}(g'_r, e)]}{\sum_{r=1}^{R}\frac{1}{r}}$$

- The probability of a specific relation type occurring between two words, given that the words are seen in a sentence together, calculated from the background corpus:

$$\mathrm{CES}_2(e) = \frac{\mathrm{P}(rel, w_1, w_2)}{\mathrm{P}(*, w_1, w_2)}$$

- Smoothing these probabilities with distributional similarity:

$$\mathrm{ECES}{*}_2(rel, w_1, w_2) = \frac{\sum_{c_1 \in C_1} \mathrm{sim}(c_1, w_1) \times \frac{\mathrm{P}(rel, c_1, w_2)}{\mathrm{P}(*, c_1, w_2)}}{\sum_{c_1 \in C_1} \mathrm{sim}(c_1, w_1)}$$

- Combining together all alternative scoring methods.

GRAPH SCORING

Edge scores are combined into graph scores by first averaging over all edges for each node, and then over all nodes.

$$\mathrm{NodeScore}(n) = \frac{\sum_{e \in E_g} \mathrm{EdgeScore}(e) \times \mathrm{isDep}(e, n)}{\sum_{e \in E_g} \mathrm{isDep}(e, n)}$$

$$\mathrm{GraphScore}(g) = \frac{\sum_{n \in N_g} \mathrm{NodeScore}(n)}{|N_g|}$$

EXPERIMENTS

We make use of the unlexicalised RASP parser (Briscoe, 2006) as the baseline system. Experiments were performed on Wall Street Journal (DepBank/GR) and biomedical (Genia-GR) datasets.

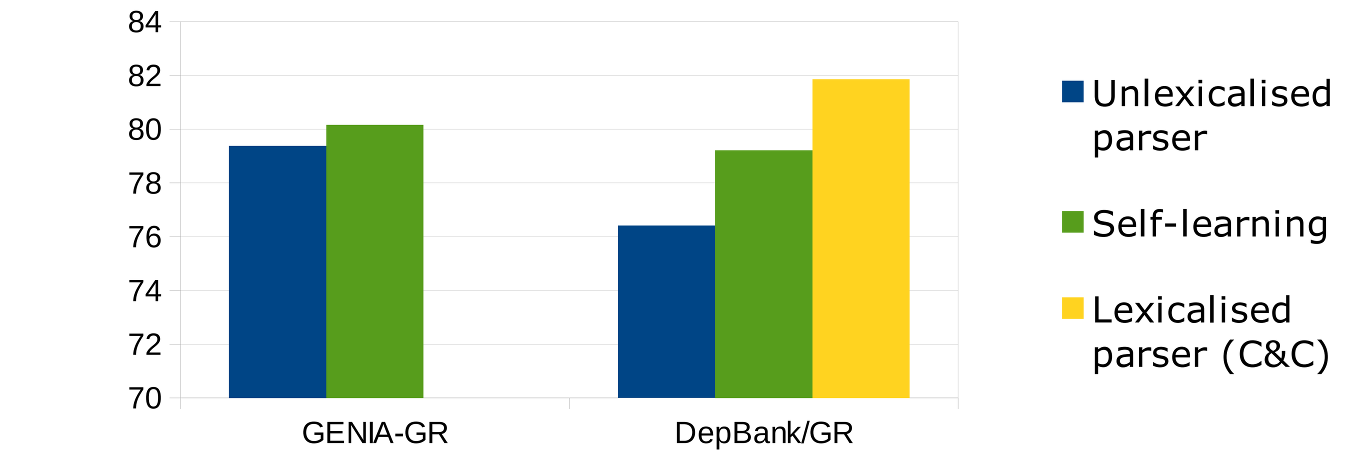


Figure 2: F-scores on the Genia and Wall Street Journal datasets.

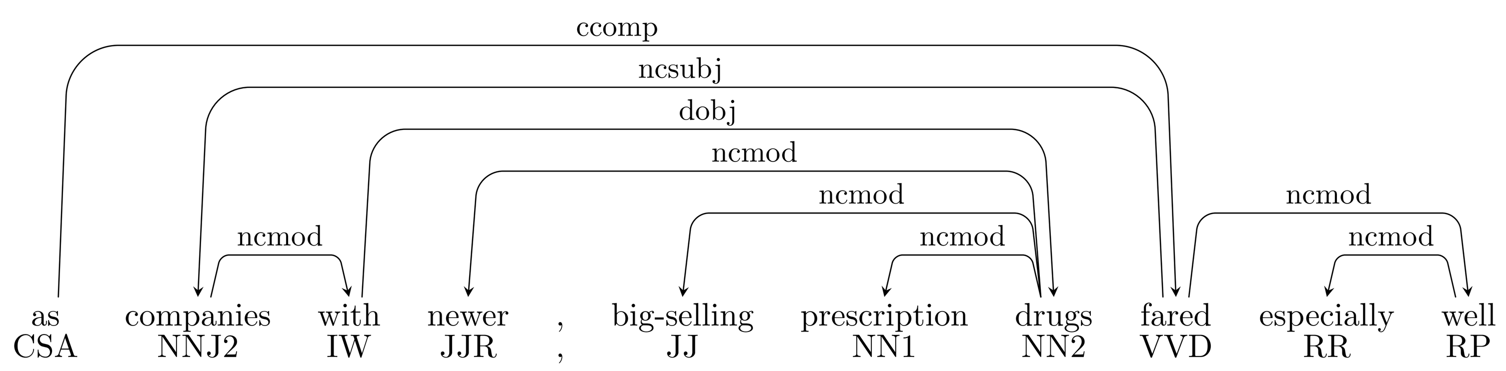


Figure 3: An improved parse of the example from Figure 1. The self-learning framework has learned that *prescription* is likely to modify *drugs*.

SUMMARY

- The unlexicalised parser is able to learn lexical features from its own output.
- Significant improvement in F-score achieved on both WSJ and Genia data.
- The method managed to close more than half of the gap between the performance of a fully-supervised in-domain lexicalised parser and a weakly-supervised unlexicalised one.
- The framework requires only a large corpus of in-domain text. No manual annotation or supervised training is needed.