Semantic Role Labelling with Tree Conditional Random Fields

Trevor Cohn and **Philip Blunsom**
University of Melbourne, Australia
tacohn@csse.unimelb.edu.au and pcbl@csse.unimelb.ed.au

Abstract

In this paper we apply conditional random fields (CRFs) to the semantic role labelling task. We define a random field over the structure of each sentence's syntactic parse tree. For each node of the tree, the model must predict a semantic role label, which is interpreted as the labelling for the corresponding syntactic constituent. We show how modelling the task as a tree labelling problem allows for the use of efficient CRF inference algorithms, while also increasing generalisation performance when compared to the equivalent maximum entropy classifier. We have participated in the CoNLL-2005 shared task closed challenge with full syntactic information.

1 Introduction

The semantic role labelling task (SRL) involves identifying which groups of words act as arguments to a given predicate. These arguments must be labelled with their role with respect to the predicate, indicating how the proposition should be semantically interpreted.

We apply conditional random fields (CRFs) to the task of SRL proposed by the CoNLL shared task 2005 (Carreras and Màrquez, 2005). CRFs are undirected graphical models which define a conditional distribution over labellings given an observation (Lafferty et al., 2001). These models allow for the use of very large sets of arbitrary, overlapping and non-independent features. CRFs have been applied with impressive empirical results to the tasks of named entity recognition (McCallum and Li, 2003; Cohn et al., 2005), part-of-speech (PoS) tagging (Lafferty et al., 2001), noun phrase chunking (Sha and Pereira, 2003) and extraction of table data (Pinto et al., 2003), among other tasks.

While CRFs have not been used to date for SRL, their close cousin, the maximum entropy model has been, with strong generalisation performance (Xue and Palmer, 2004; Lim et al., 2004). Most CRF implementations have been specialised to work with chain structures, where the labels and observations form a linear sequence. Framing SRL as a linear tagging task is awkward, as there is no easy model of adjacency between the candidate constituent phrases.

Our approach simultaneously performs both constituent selection and labelling, by defining an undirected random field over the parse tree. This allows the modelling of interactions between parent and child constituents, and the prediction of an optimal argument labelling for all constituents in one pass. The parse tree forms an acyclic graph, meaning that efficient exact inference in a CRF is possible using belief propagation.

2 Data

The data used for this task was taken from the Propbank corpus, which supplements the Penn Treebank with semantic role annotation. Full details of the data set are provided in Carreras and Màrquez (2005).

2.1 Data Representation

From each training instance we derived a tree, using the parse structure from the Collins parser. The

Proceedings of the 9th Conference on Computational Natural Language Learning (CoNLL),
pages 169–172, Ann Arbor, June 2005.

nodes in the trees were relabelled with a semantic role label indicating how their corresponding syntactic constituent relates to each predicate, as shown in Figure 1. The role labels are shown as subscripts in the figure, and both the syntactic categories and the words at the leaves are shown for clarity only – these were not included in the tree. Additionally, the dashed lines show those edges which were pruned, following Xue and Palmer (2004) – only nodes which are siblings to a node on the path from the verb to the root are included in the tree. Child nodes of included prepositional phrase nodes are also included. This reduces the size of the resultant tree whilst only very occasionally excluding nodes which should be labelled as an argument.

The tree nodes were labelled such that only argument constituents received the argument label while all argument children were labelled as outside, `O`. Where there were parse errors, such that no constituent exactly covered the token span of an argument, the smaller subsumed constituents were all given the argument label.

We experimented with two alternative labelling strategies: labelling a constituent's children with a new 'inside' label, and labelling the children with the parent's argument label. In the figure, the IN and NP children of the PP would be affected by these changes, both receiving either the inside `I` label or `AM-LOC` label under the respective strategies. The inside strategy performed nearly identically to the standard (outside) strategy, indicating that either the model cannot reliably predict the inside argument, or that knowing that the children of a given node are inside an argument is not particularly useful in predicting its label. The second (duplication) strategy performed extremely poorly. While this allowed the internal argument nodes to influence their ancestor towards a particular labelling, it also dramatically increased the number of nodes given an argument label. This lead to spurious over-prediction of arguments.

The model is used for decoding by predicting the maximum probability argument label assignment to each of the unlabelled trees. When these predictions were inconsistent, and one argument subsumed another, the node closest to the root of the tree was deemed to take precedence over its descendants.

3 Model

We define a CRF over the labelling $\mathbf{y}$ given the observation tree $\mathbf{x}$ as:

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \exp \sum_{c \in \mathcal{C}} \sum_{k} \lambda_k f_k(c, \mathbf{y}_c, \mathbf{x})$$

where $\mathcal{C}$ is the set of cliques in the observation tree, λ_k are the model's parameters and $f_k(\cdot)$ is the feature function which maps a clique labelling to a vector of scalar values. The function $Z(\cdot)$ is the normalising function, which ensures that p is a valid probability distribution. This can be restated as:

$$\begin{aligned} p(\mathbf{y}|\mathbf{x}) &= \frac{1}{Z(\mathbf{x})} \exp \left\{ \sum_{v \in \mathcal{C}_1} \sum_{k} \lambda_k g_k(v, \mathbf{y}_v, \mathbf{x}) \right. \\ &\quad \left. + \sum_{u,v \in \mathcal{C}_2} \sum_{j} \lambda_j h_j(u, v, \mathbf{y}_u, \mathbf{y}_v, \mathbf{x}) \right\} \end{aligned}$$

where $\mathcal{C}_1$ are the vertices in the graph and $\mathcal{C}_2$ are the maximal cliques in the graph, consisting of all *(parent, child)* pairs. The feature function has been split into g and h, each dealing with one and two node cliques respectively.

Preliminary experimentation without any pair-wise features (h), was used to mimic a simple maximum entropy classifier. This model performed considerably worse than the model with the pair-wise features, indicating that the added complexity of modelling the parent-child interactions provides for more accurate modelling of the data.

The log-likelihood of the training sample was optimised using limited memory variable metric (LMVM), a gradient based technique. This required the repeated calculation of the log-likelihood and its derivative, which in turn required the use of dynamic programming to calculate the marginal probability of each possible labelling of every clique using the sum-product algorithm (Pearl, 1988).

4 Features

As the conditional random field is conditioned on the observation, it allows feature functions to be defined over any part of the observation. The tree structure requires that features incorporate either a node labelling or the labelling of a parent and its

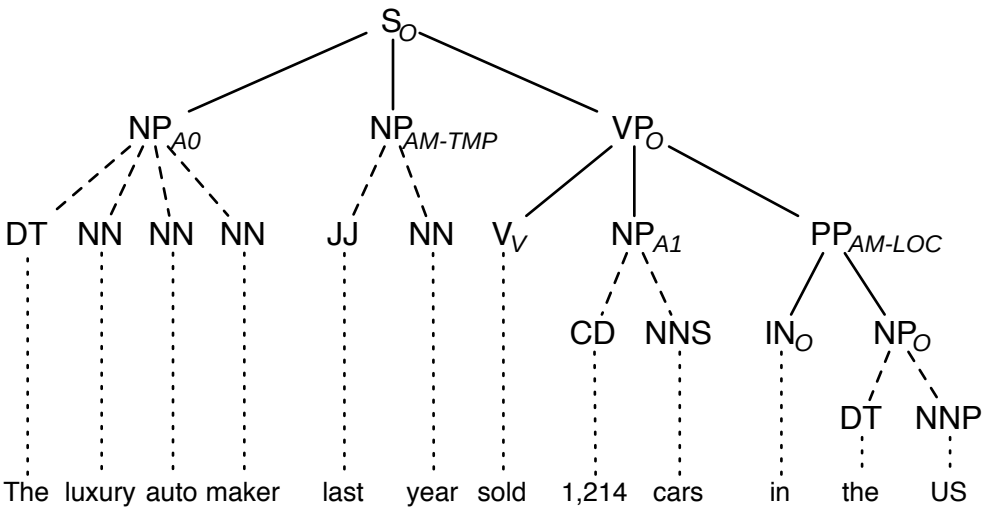

Figure 1: Syntax tree labelled for semantic roles with respect to the predicate *sell*. The subscripts show the role labels, and the dotted and dashed edges are those which are pruned from the tree.

child. We have defined node and pairwise clique features using data local to the corresponding syntactic node(s), as well as some features on the predicate itself.

Each feature type has been made into binary feature functions g and h by combining *(feature type, value)* pairs with a label, or label pair, where this combination was seen at least once in the training data. The following feature types were employed, most of which were inspired by previous works:

Basic features: {*Head word, head PoS, phrase syntactic category, phrase path, position relative to the predicate, surface distance to the predicate, predicate lemma, predicate token, predicate voice, predicate sub-categorisation, syntactic frame*}. These features are common to many SRL systems and are described in Xue and Palmer (2004).

Context features {*Head word of first NP in preposition phrase, left and right sibling head words and syntactic categories, first and last word in phrase yield and their PoS, parent syntactic category and head word*}. These features are described in Pradhan et al. (2005).

Common ancestor of the verb The syntactic category of the deepest shared ancestor of both the verb and node.

Feature conjunctions The following features were conjoined: { *predicate lemma + syntactic category, predicate lemma + relative position, syntactic category + first word of the phrase*}.

Default feature This feature is always on, which allows the classifier to model the prior probability distribution over the possible argument labels.

Joint features These features were only defined over pair-wise cliques: {*whether the parent and child head words do not match, parent syntactic category + and child syntactic category, parent relative position + child relative position, parent relative position + child relative position + predicate PoS + predicate lemma*}.

5 Experimental Results

The model was trained on the full training set after removing unparsable sentences, yielding 90,388 predicates and 1,971,985 binary features. A Gaussian prior was used to regularise the model, with variance $\sigma^2 = 1$. Training was performed on a 20 node PowerPC cluster, consuming a total of 62Gb of RAM and taking approximately 15 hours. Decoding required only 3Gb of RAM and about 5 minutes for the 3,228 predicates in the development set. Results are shown in Table 1.

	Precision	Recall	$F_{\beta=1}$
Development	73.51%	68.98%	71.17
Test WSJ	75.81%	70.58%	73.10
Test Brown	67.63%	60.08%	63.63
Test WSJ+Brown	74.76%	69.17%	71.86

Test WSJ	Precision	Recall	$F_{\beta=1}$
Overall	75.81%	70.58%	73.10
A0	82.21%	79.48%	80.82
A1	74.56%	71.26%	72.87
A2	63.93%	56.85%	60.18
A3	63.95%	54.34%	58.75
A4	68.69%	66.67%	67.66
A5	0.00%	0.00%	0.00
AM-ADV	54.73%	48.02%	51.16
AM-CAU	75.61%	42.47%	54.39
AM-DIR	54.17%	30.59%	39.10
AM-DIS	77.74%	73.12%	75.36
AM-EXT	65.00%	40.62%	50.00
AM-LOC	60.67%	54.82%	57.60
AM-MNR	54.66%	49.42%	51.91
AM-MOD	98.34%	96.55%	97.44
AM-NEG	99.10%	96.09%	97.57
AM-PNC	49.47%	40.87%	44.76
AM-PRD	0.00%	0.00%	0.00
AM-REC	0.00%	0.00%	0.00
AM-TMP	77.20%	68.54%	72.61
R-A0	87.78%	86.61%	87.19
R-A1	82.39%	75.00%	78.52
R-A2	0.00%	0.00%	0.00
R-A3	0.00%	0.00%	0.00
R-A4	0.00%	0.00%	0.00
R-AM-ADV	0.00%	0.00%	0.00
R-AM-CAU	0.00%	0.00%	0.00
R-AM-EXT	0.00%	0.00%	0.00
R-AM-LOC	0.00%	0.00%	0.00
R-AM-MNR	0.00%	0.00%	0.00
R-AM-TMP	71.05%	51.92%	60.00
V	98.73%	98.63%	98.68

Table 1: Overall results (top) and detailed results on the WSJ test (bottom).

6 Conclusion

Conditional random fields proved useful in modelling the semantic structure of text when provided with a parse tree. Our novel use of a tree structure derived from the syntactic parse, allowed for parent-child interactions to be accurately modelled, which provided an improvement over a standard maximum entropy classifier. In addition, the parse constituent structure proved quite appropriate to the task, more so than modelling the data as a sequence of words or chunks, as has been done in previous approaches.

Acknowledgements

We would both like to thank our research supervisor Steven Bird for his comments and feedback on this work. The research undertaken for this paper was supported by an Australian Postgraduate Award scholarship, a Melbourne Research Scholarship and a Melbourne University Postgraduate Overseas Research Experience Scholarship.

References

Xavier Carreras and Lluís Màrquez. 2005. Introduction to the CoNLL-2005 Shared Task: Semantic Role Labeling. In *Proceedings of the CoNLL-2005*.

Trevor Cohn, Andrew Smith, and Miles Osborne. 2005. Scaling conditional random fields using error correcting codes. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*. To appear.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labelling sequence data. In *Proceedings of the 18th International Conference on Machine Learning*, pages 282–289.

Joon-Ho Lim, Young-Sook Hwang, So-Young Park, and Hae-Chang Rim. 2004. Semantic role labeling using maximum entropy model. In *Proceedings of the CoNLL-2004 Shared Task*.

Andrew McCallum and Wei Li. 2003. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the 7th Conference on Natural Language Learning*, pages 188–191.

Judea Pearl. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.

David Pinto, Andrew McCallum, Xing Wei, and Bruce Croft. 2003. Table extraction using conditional random fields. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 235–242.

Sameer Pradhan, Kadri Hacioglu, Valerie Krugler, Wayne Ward, James Martin, and Daniel Jurafsky. 2005. Support vector learning for semantic argument classification. In *To appear in Machine Learning journal, Special issue on Speech and Natural Language Processing*.

Fei Sha and Fernando Pereira. 2003. Shallow parsing with conditional random fields. In *Proceedings of the Human Language Technology Conference and North American Chapter of the Association for Computational Linguistics*, pages 213–220.

Nianwen Xue and Martha Palmer. 2004. Calibrating features for semantic role labeling. In *Proceedings of EMNLP*.