Joint Parsing and Semantic Role Labeling

Charles Sutton and Andrew McCallum
Department of Computer Science
University of Massachusetts
Amherst, MA 01003 USA
{casutton,mccallum}@cs.umass.edu

Abstract

A striking feature of human syntactic processing is that it is *context-dependent*, that is, it seems to take into account semantic information from the discourse context and world knowledge. In this paper, we attempt to use this insight to bridge the gap between SRL results from gold parses and from automatically-generated parses. To do this, we jointly perform parsing and semantic role labeling, using a probabilistic SRL system to rerank the results of a probabilistic parser. Our current results are negative, because a locally-trained SRL model can return inaccurate probability estimates.

1 Introduction

Although much effort has gone into developing statistical parsing models and they have improved steadily over the years, in many applications that use parse trees errors made by the parser are a major source of errors in the final output. A promising approach to this problem is to perform both parsing and the higher-level task in a single, *joint* probabilistic model. This not only allows uncertainty about the parser output to be carried upward, such as through an k-best list, but also allows information from higher-level processing to improve parsing. For example, Miller et al. (2000) showed that performing parsing and information extraction in a joint model improves performance on both tasks. In particular, one suspects that attachment decisions, which are both notoriously hard and extremely important for semantic analysis, could benefit greatly from input from higher-level semantic analysis.

The recent interest in semantic role labeling provides an opportunity to explore how higher-level semantic information can inform syntactic parsing. In previous work, it has been shown that SRL systems that use full parse information perform better than those that use shallow parse information, but that machine-generated parses still perform much worse than human-corrected gold parses.

The goal of this investigation is to narrow the gap between SRL results from gold parses and from automatic parses. We aim to do this by jointly performing parsing and semantic role labeling in a single probabilistic model. In both parsing and SRL, state-of-the-art systems are probabilistic; therefore, their predictions can be combined in a principled way by multiplying probabilities. In this paper, we rerank the k-best parse trees from a probabilistic parser using an SRL system. We compare two reranking approaches, one that linearly weights the log probabilities, and the other that learns a reranker over parse trees and SRL frames in the manner of Collins (2000).

Currently, neither method performs better than simply selecting the top predicted parse tree. We discuss some of the reasons for this; one reason being that the ranking over parse trees induced by the semantic role labeling score is unreliable, because the model is trained locally.

2 Base SRL System

Our approach to joint parsing and SRL begins with a base SRL system, which uses a standard architecture from the literature. Our base SRL system is a cascade of maximum-entropy classifiers which select the semantic argument label for each constituent of a full parse tree. As in other systems, we use three stages: pruning, identification, and classification. First, in *pruning*, we use a deterministic preprocessing procedure introduced by Xue and Palmer (2004) to prune many constituents which are almost certainly not arguments. Second, in *identification*, a binary MaxEnt classifier is used to prune remaining constituents which are predicted to be null with

Proceedings of the 9th Conference on Computational Natural Language Learning (CoNLL),
pages 225–228, Ann Arbor, June 2005.

Base features [GJ02]
Path to predicate
Constituent type
Head word
Position
Predicate
Head POS [SHWA03]
All conjunctions of above

Table 1: Features used in base identification classifier.

high probability. Finally, in *classification*, a multiclass MaxEnt classifier is used to predict the argument type of the remaining constituents. This classifer also has the option to output NULL.

It can happen that the returned semantic arguments overlap, because the local classifiers take no global constraints into account. This is undesirable, because no overlaps occur in the gold semantic annotations. We resolve overlaps using a simple recursive algorithm. For each parent node that overlaps with one of its descendents, we check which predicted probability is greater: that the parent has its locally-predicted argument label and all its descendants are null, or that the descendants have their optimal labeling, and the parent is null. This algorithm returns the non-overlapping assignment with globally highest confidence. Overlaps are uncommon, however; they occurred only 68 times on the 1346 sentences in the development set.

We train the classifiers on PropBank sections 02–21. If a true semantic argument fails to match any bracketing in the parse tree, then it is ignored. Both the identification and classification models are trained using gold parse trees. All of our features are standard features for this task that have been used in previous work, and are listed in Tables 1 and 2. We use the maximum-entropy implementation in the Mallet toolkit (McCallum, 2002) with a Gaussian prior on parameters.

3 Reranking Parse Trees Using SRL Information

Here we give the general framework for the reranking methods that we present in the next section. We write a joint probability model over semantic frames F and parse trees t given a sentence $\mathbf{x}$ as

$$p(F, t|\mathbf{x}) = p(F|t, \mathbf{x})p(t|\mathbf{x}), \quad (1)$$

where $p(t|\mathbf{x})$ is given by a standard probabilistic parsing model, and $p(F|t, \mathbf{x})$ is given by the baseline SRL model described previously.

Base features [GJ02]
Head word
Constituent type
Position
Predicate
Voice
Head POS [SHWA03]
From [PWHMJ04]
Parent Head POS
First word / POS
Last word / POS
Sibling constituent type / head word / head POS
Conjunctions [XP03]
Voice & Position
Predicate & Head word
Predicate & Constituent type

Table 2: Features used in baseline labeling classifier.

Parse Trees Used	SRL F1
Gold	77.1
1-best	63.9
Reranked by gold parse F1	68.1
Reranked by gold frame F1	74.2
Simple SRL combination ($\alpha = 0.5$)	56.9
Chosen using trained reranker	63.6

Table 3: Comparison of Overall SRL F1 on development set by the type of parse trees used.

In this paper, we choose (F^*, t^*) to approximately maximize the probability $p(F, t|\mathbf{x})$ using a reranking approach. To do the reranking, we generate a list of k-best parse trees for a sentence, and for each predicted tree, we predict the best frame using the base SRL model. This results in a list $\{(F^i, t^i)\}$ of parse tree / SRL frame pairs, from which the reranker chooses. Thus, our different reranking methods vary only in which parse tree is selected; given a parse tree, the frame is always chosen using the best prediction from the base model.

The k-best list of parses is generated using Dan Bikel's (2004) implementation of Michael Collins' parsing model. The parser is trained on sections 2–21 of the WSJ Treebank, which does not overlap with the development or test sets. The k-best list is generated in Bikel's implementation by essentially turning off dynamic programming and doing very aggressive beam search. We gather a maximum of 500 best parses, but the limit is not usually reached using feasible beam widths. The mean number of parses per sentence is 176.

4 Results and Discussion

In this section we present results on several reranking methods for joint parsing and semantic role la-

beling. Table 3 compares F1 on the development set of our different reranking methods. The first four rows in Table 3 are baseline systems. We present baselines using gold trees (row 1 in Table 3) and predicted trees (row 2). As shown in previous work, gold trees perform much better than predicted trees.

We also report two cheating baselines to explore the maximum possible performance of a reranking system. First, we report SRL performance of ceiling parse trees (row 3), i.e., if the parse tree from the k-best list is chosen to be closest to the gold tree. This is the best expected performance of a parse reranking approach that maximizes parse F1. Second, we report SRL performance where the parse tree is selected to maximize SRL F1, computing using the gold frame (row 4). There is a significant gap both between parse-F1-reranked trees and SRL-F1-reranked trees, which shows promise for joint reranking. However, the gap between SRL-F1-reranked trees and gold parse trees indicates that reranking of parse lists cannot by itself completely close the gap in SRL performance between gold and predicted parse trees.

4.1 Reranking based on score combination

Equation 1 suggests a straightforward method for reranking: simply pick the parse tree from the k-best list that maximizes $p(F, t|\mathbf{x})$, in other words, add the log probabilities from the parser and the base SRL system. More generally, we consider weighting the individual probabilities as

$$s(F, t) = p(F|t, \mathbf{x})^{1-\alpha} p(t|\mathbf{x})^{\alpha}. \quad (2)$$

Such a weighted combination is often used in the speech community to combine acoustic and language models.

This reranking method performs poorly, however. No choice of α performs better than $\alpha = 1$, i.e., choosing the 1-best predicted parse tree. Indeed, the more weight given to the SRL score, the worse the combined system performs. The problem is that often a bad parse tree has many nodes which are obviously not constituents: thus $p(F|t, \mathbf{x})$ for such a bad tree is very high, and therefore not reliable. As more weight is given to the SRL score, the unlabeled recall drops, from 55% when $\alpha = 0$ to 71% when $\alpha = 1$. Most of the decrease in F1 is due to the drop in unlabeled recall.

4.2 Training a reranker using global features

One potential solution to this problem is to add features of the entire frame, for example, to vote against predicted frames that are missing key arguments. But such features depend globally on the entire frame, and cannot be represented by local classifiers. One way to train these global features is to learn a linear classifier that selects a parse / frame pair from the ranked list, in the manner of Collins (2000). Reranking has previously been applied to semantic role labeling by Toutanova et al. (2005), from which we use several features. The difference between this paper and Toutanova et al. is that instead of reranking k-best SRL frames of a single parse tree, we are reranking 1-best SRL frames from the k-best parse trees.

Because of the the computational expense of training on k-best parse tree lists for each of 30,000 sentences, we train the reranker only on sections 15–18 of the Treebank (the same subset used in previous CoNLL competitions). We train the reranker using LogLoss, rather than the boosting loss used by Collins. We also restrict the reranker to consider only the top 25 parse trees.

This globally-trained reranker uses all of the features from the local model, and the following global features: (a) *sequence features*, i.e., the linear sequence of argument labels in the sentence (e.g. `A0_V_A1`), (b) the log probability of the parse tree, (c) *has-arg* features, that is, for each argument type a binary feature indicating whether it appears in the frame, (d) the conjunction of the predicate and has-arg feature, and (e) the number of nodes in the tree classified as each argument type.

The results of this system on the development set are given in Table 3 (row 6). Although this performs better than the score combination method, it is still no better than simply taking the 1-best parse tree. This may be due to the limited training set we used in the reranking model. A base SRL model trained only on sections 15–18 has 61.26 F1, so in comparison, reranking provides a modest improvement. This system is the one that we submitted as our official submission. The results on the test sets are given in Table 4.

5 Summing over parse trees

In this section, we sketch a different approach to joint SRL and parsing that does not use reranking at all. Maximizing over parse trees can mean that poor parse trees can be selected if their semantic labeling has an erroneously high score. But we are not actually interested in selecting a good parse tree; all we want is a good semantic frame. This means that we should select the semantic frame

	Precision	Recall	$F_{\beta=1}$
Development	64.43%	63.11%	63.76
Test WSJ	68.57%	64.99%	66.73
Test Brown	62.91%	54.85%	58.60
Test WSJ+Brown	67.86%	63.63%	65.68

Test WSJ	Precision	Recall	$F_{\beta=1}$
Overall	68.57%	64.99%	66.73
A0	69.47%	74.35%	71.83
A1	66.90%	64.91%	65.89
A2	64.42%	61.17%	62.75
A3	62.14%	50.29%	55.59
A4	72.73%	70.59%	71.64
A5	50.00%	20.00%	28.57
AM-ADV	55.90%	49.60%	52.57
AM-CAU	76.60%	49.32%	60.00
AM-DIR	57.89%	38.82%	46.48
AM-DIS	79.73%	73.75%	76.62
AM-EXT	66.67%	43.75%	52.83
AM-LOC	50.26%	53.17%	51.67
AM-MNR	54.32%	51.16%	52.69
AM-MOD	98.50%	95.46%	96.96
AM-NEG	98.20%	94.78%	96.46
AM-PNC	46.08%	40.87%	43.32
AM-PRD	0.00%	0.00%	0.00
AM-REC	0.00%	0.00%	0.00
AM-TMP	72.15%	67.43%	69.71
R-A0	0.00%	0.00%	0.00
R-A1	0.00%	0.00%	0.00
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	0.00%	0.00%	0.00
V	99.21%	86.24%	92.27

Table 4: Overall results (top) and detailed results on the WSJ test (bottom).

that maximizes the posterior probability: $p(F|\mathbf{x}) = \sum_t p(F|t,\mathbf{x})p(t|\mathbf{x})$. That is, we should be *summing* over the parse trees instead of maximizing over them. The practical advantage of this approach is that even if one seemingly-good parse tree does not have a constituent for a semantic argument, many other parse trees in the k-best list might, and all are considered when computing F^*. Also, no single parse tree need have constituents for all of F^*; because it sums over all parse trees, it can mix and match constituents between different trees. The optimal frame F^* can be computed by an $O(N^3)$ parsing algorithm if appropriate independence assumptions are made on $p(F|\mathbf{x})$. This requires designing an SRL model that is independent of the bracketing derived from any particular parse tree. Initial experiments performed poorly because the marginal model $p(F|\mathbf{x})$ was inadequate. Detailed exploration is left for future work.

6 Conclusion and Related Work

In this paper, we have considered several methods for reranking parse trees using information from semantic role labeling. So far, we have not been able to show improvement over selecting the 1-best parse tree. Gildea and Jurafsky (Gildea and Jurafsky, 2002) also report results on reranking parses using an SRL system, with negative results. In this paper, we confirm these results with a MaxEnt-trained SRL model, and we extend them to show that weighting the probabilities does not help either.

Our results with Collins-style reranking are too preliminary to draw definite conclusions, but the potential improvement does not appear to be great. In future work, we will explore the max-sum approach, which has promise to avoid the pitfalls of max-max reranking approaches.

Acknowledgements

This work was supported in part by the Center for Intelligent Information Retrieval, in part by National Science Foundation under NSF grants #IIS-0326249 ond #IIS-0427594, and in part by the Defense Advanced Research Projec ts Agency (DARPA), through the Department of the Interior, NBC, Acquisition Services Division, under contract number NBCHD030010. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsor.

References

Daniel M. Bikel. 2004. Intricacies of Collins' parsing model. *Computational Linguistics*.

Michael Collins. 2000. Discriminative reranking for natural language parsing. In *Proc. 17th International Conf. on Machine Learning*, pages 175–182. Morgan Kaufmann, San Francisco, CA.

Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3):245–288.

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. `http://mallet.cs.umass.edu`.

Scott Miller, Heidi Fox, Lance A. Ramshaw, and Ralph M. Weischedel. 2000. A novel use of statistical parsing to extract information from text. In *ANLP 2000*, pages 226–233.

Mihai Surdeanu, Sanda Harabagiu, John Williams, and Paul Aarseth. 2003. Using predicate-argument structures for information extraction. In *ACL-2003*.

Kristina Toutanova, Aria Haghighi, and Christopher D. Manning. 2005. Joint learning improves semantic role labeling. In *ACL 2005*.

Nianwen Xue and Martha Palmer. 2004. Calibrating features for semantic role labeling. In *Proceedings of 2004 Conference on Empirical Methods in Natural Language Processing*.