T	LA	AS		EM	
		U	L	U	L
None	MBL	79.08	72.83	28.99	21.15
τ_p	MBL	80.79	74.39	31.54	22.53
$\tau_p \circ \tau_v \circ \tau_{c+^*}$	MBL	82.93	76.31	34.17	23.01
None	SVM	81.09	75.68	32.24	25.02
τ_p	SVM	82.93	77.28	35.99	27.05
$\tau_p \circ \tau_v \circ \tau_{c+^*}$	SVM	84.55	78.82	37.63	27.69

Table 5: Optimized parsing results (SVM, Δ_e); T = transformation; LA = learning algorithm; AS = attachment score, EM = exact match; U = unlabeled, L = labeled

T	P:S	R:S	P:C	R:C	P:A	R:A	P:M	R:M
None	52.63	72.35	55.15	67.03	82.17	82.21	69.95	69.07
$\tau_p \circ \tau_v \circ \tau_{c+^*}$	63.73	82.10	63.20	75.14	90.89	92.79	80.02	81.40

Table 6: Detailed results for SVM; T = transformation; P = unlabeled precision, R = unlabeled recall

costly to train (Sagae and Lavie, 2005).

Table 5 shows the results, for both MBL and SVM, of the baseline, the pure pseudo-projective parsing, and the combination of pseudo-projective parsing with PS-to-MS transformations. We see that pseudo-projective parsing brings a very consistent increase in accuracy of at least 1.5 percentage points, which is more than that reported by Nivre and Nilsson (2005), and that the addition of the PS-to-MS transformations increases accuracy with about the same margin. We also see that SVM outperforms MBL by about two percentage points across the board, and that the positive effect of the graph transformations is most pronounced for the unlabeled exact match score, where the improvement is more than five percentage points overall for both MBL and SVM.

Table 6 gives a more detailed analysis of the parsing results for SVM, comparing the optimal parser to the baseline, and considering specifically the (unlabeled) precision and recall of the categories involved in coordination (separators S and conjuncts C) and verb groups (auxiliary verbs A and main verbs M). All figures indicate, without exception, that the transformations result in higher precision and recall for all directly involved words. (All differences are significant beyond the 0.01 level.) It is worth noting that the error reduction is actually higher for A and M than for S and C, although the former are less frequent.

With respect to unlabeled attachment score, the results of the optimized parser are slightly below the best published results for a single parser. Hall and Novák (2005) report a score of 85.1%, applying a corrective model to the output of Charniak's parser; McDonald and Pereira (2006) achieve a score of 85.2% using a second-order spanning tree algorithm. Using ensemble methods and a pool of different parsers, Zeman and Žabokrtský (2005) attain a top score of 87.0%. For unlabeled exact match, our results are better than any previously reported results, including those of McDonald and Pereira (2006). (For the labeled scores, we are not aware of any comparable results in the literature.)

5 Conclusion

The results presented in this paper confirm that choosing the right representation is important in parsing. By systematically transforming the representation of coordinate structures and verb groups in PDT, we achieve a 10% error reduction for a data-driven dependency parser. Adding graph transformations for non-projective dependency parsing gives a total error reduction of about 20% (even more for unlabeled exact match). In this way, we achieve state-of-the-art accuracy with a deterministic, classifier-based dependency parser.

Acknowledgements

The research presented in this paper was partially supported by the Swedish Research Council. We are grateful to Jan Hajič and Daniel Zeman for help with the Czech data and to three anonymous reviewers for helpful comments and suggestions.