TFLEX: Speeding up Deep Parsing with Strategic Pruning

Myroslava O. Dzikovska
Human Communication Research Centre
University of Edinburgh
Edinburgh, EH8 9LW, UK
mdzikovs@inf.ed.ac.uk

Carolyn P. Rose
Carnegie Mellon University
Language Technologies Institute
Pittsburgh PA 15213, USA
cprose@cs.cmu.edu

1 Introduction

This paper presents a method for speeding up a deep parser through backbone extraction and pruning based on CFG ambiguity packing.[1] The TRIPS grammar is a wide-coverage grammar for deep natural language understanding in dialogue, utilized in 6 different application domains, and with high coverage and sentence-level accuracy on human-human task-oriented dialogue corpora (Dzikovska, 2004). The TRIPS parser uses a best-first beam search algorithm and a chart size limit, both of which are a form of pruning focused on finding an n-best list of interpretations. However, for longer sentences limiting the chart size results in failed parses, while increasing the chart size limits significantly impacts the parsing speed.

It is possible to speed up parsing by implementing faster unification algorithms, but this requires considerable implementation effort. Instead, we developed a new parser, TFLEX, which uses a simpler technique to address efficiency issues. TFLEX combines the TRIPS grammar with the fast parsing technologies implemented in the LCFLEX parser (Rosé and Lavie, 2001). LCFLEX is an all-paths parser which uses left-corner prediction and ambiguity packing, and which was shown to be efficient on other unification augmented context-free grammars. We describe a way to transfer the TRIPS grammar to LCFLEX, and a pruning method which achieves significant improvements in both speed and coverage compared to the original TRIPS parser.

[1]This material is based on work supported by grants from the Office of Naval Research under numbers N000140510048 and N000140510043.

2 TFLEX

To use the TRIPS grammar in LCFLEX we first extracted a CFG backbone from the TRIPS grammar, with CFG non-terminals corresponding directly to TRIPS constituent categories. To each CFG rule we attach a corresponding TRIPS rule. Whenever a CFG rule completes, a TRIPS unification function is called to do all the unification operations associated with the TRIPS rule. If the unification fails, the constituent built by the CFG is cancelled.

The TFLEX pruning algorithm uses ambiguity packing to provide good pruning points. For example, in the sentence "we have a heart attack victim at marketplace mall" the phrase "a heart attack victim" has two interpretations depending on whether "heart" modifies "attack" or "attack victim". These interpretations will be ambiguity packed in the CFG structure, which offers an opportunity to make pruning more strategic by focusing specifically on competing interpretations for the same utterance span. For any constituent where ambiguity-packed non-head daughters differ only in local features, we prune the interpretations coming from them to a specified prune beam width based on their TRIPS scores. In the example above, pruning will happen at the point of making a VP "have a heart attack victim". The NP will be ambiguity packed, and we will prune alternative VP interpretations resulting from combining the same sense of the verb "have" and different interpretations of the NP.

This approach works better than the original TRIPS best-first algorithm, because for long sentence the TRIPS chart contains a large number

Proceedings of the Ninth International Workshop on Parsing Technologies (IWPT), pages 194–195,
Vancouver, October 2005.

of similar constituents, and the parser frequently reaches the chart size limit before finding the correct constituent to use. Ambiguity packing in TFLEX helps chose the best constituents to prune by pruning competing interpretations which cover the same span and have the same non-local features, thus making it less likely that a constituent essential for building a parse will be pruned.

3 Evaluation

Our evaluation data is an excerpt from the Monroe corpus that has been used in previous TRIPS research on parsing speed and accuracy (Swift et al., 2004). The test contained 1042 utterances, from 1 to 45 words in length (mean 5.38 words/utt, st. dev. 5.7 words/utt). Using a hold-out set, we determined that a beam width of 3 was an optimal setting for TFLEX. We then compared TFLEX at beam width 3 to the TRIPS parser with chart size limits of 1500, 5000, and 10000. As our evaluation metrics we report are average parse time per sentence and probability of finding at least one parse, the latter being a measure approximating parsing accuracy.

The results are presented in Figure 1. We grouped sentences into equivalence classes based on length with a 5-word increment. On sentences greater than 10 words long, TFLEX is significantly more likely to produce a parse than any of the TRIPS parsers (evaluated using a binary logistic regression, $p < .001$). Moreover, for sentences greater than 20 words long, no form of TRIPS parser returned a complete parse. TFLEX is significantly faster than TRIPS-10000, statistically indistinguishable in terms of parse time from TRIPS-5000, and significantly slower than TRIPS-1500 ($p < .001$).

Thus, TFLEX presents a superior balance of coverage and efficiency especially for long sentences (10 words or more) since for these sentences it is significantly more likely to find a parse than any version of TRIPS, even a version where the chart size is expanded to an extent that it becomes significantly slower (i.e., TRIPS-10000).

4 Conclusions

In this paper, we described a combination of efficient parsing techniques to improve parsing speed and coverage with the TRIPS deep parsing grammar.

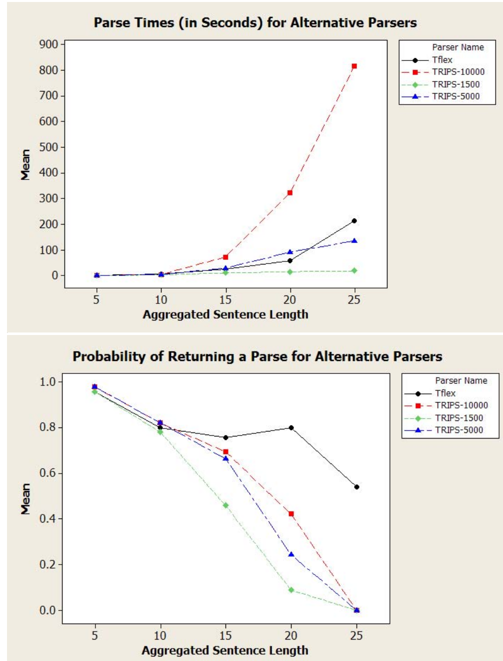

Figure 1: Parse times and probability of getting a parse depending on (aggregated) sentence lengths. 5 denotes sentences with 5 or fewer words, 25 sentences with more than 20 words.

The TFLEX system uses an all-paths left-corner parsing from the LCFLEX parser, made tractable by a pruning algorithm based on ambiguity packing and local features, generalizable to other unification grammars. Our pruning algorithm provides a better efficiency-coverage balance than best-first parsing with chart limits as utilised by the TRIPS parser.

References

M. O. Dzikovska. 2004. *A Practical Semantic Representation For Natural Language Parsing*. Ph.D. thesis, University of Rochester.

C. P. Rosé and A. Lavie. 2001. Balancing robustness and efficiency in unification-augmented context-free parsers for large practical applications. In J.C. Junqua and G Van Noord, editors, *Robustness in Language and Speech Technology*. Kluwer Academic Press.

M. Swift, J. Allen, and D. Gildea. 2004. Skeletons in the parser: Using a shallow parser to improve deep parsing. In *Proceedings of COLING-04*.

J. Tetreault, M. Swift, P. Prithviraj, M. Dzikovska, and J. Allen. 2004. Discourse annotation in the monroe corpus. In *ACL-04 workshop on Discourse Annotation*.