Exploiting Named Entity Classes in CCG Surface Realization

Rajakrishnan Rajkumar **Michael White** **Dominic Espinosa**

Department of Linguistics
The Ohio State University
Columbus, OH, USA
{raja,mwhite,espinosa}@ling.osu.edu

Abstract

This paper describes how named entity (NE) classes can be used to improve broad coverage surface realization with the OpenCCG realizer. Our experiments indicate that collapsing certain multi-word NEs and interpolating a language model where NEs are replaced by their class labels yields the largest quality increase, with 4-grams adding a small additional boost. Substantial further benefit is obtained by including class information in the hypertagging (supertagging for realization) component of the system, yielding a state-of-the-art BLEU score of 0.8173 on Section 23 of the CCGbank. A targeted manual evaluation confirms that the BLEU score increase corresponds to a significant rise in fluency.

1 Introduction

Hogan et al. (2007) have recently shown that better handling of named entities (NEs) in broad coverage surface realization with LFG can lead to substantial improvements in BLEU scores. In this paper, we confirm that better NE handling can likewise improve broad coverage surface realization with CCG, even when employing a more restrictive notion of named entities that better matches traditional realization practice. Going beyond Hogan et al. (2007), we additionally show that NE classes can be used to improve realization quality through better language models and better hypertagging (supertagging for realization) models, yielding a state-of-the-art BLEU score of 0.8173 on Section 23 of the CCGbank.

A question addressed neither by Hogan et al. nor anyone else working on broad coverage surface realization recently is whether reported increases in BLEU scores actually correspond to observable improvements in quality. We view this situation as problematic, not only because Callison-Burch et al. (2006) have shown that BLEU does not always rank competing systems in accord with human judgments, but also because surface realization scores are typically much higher than those in MT—where BLEU's performance has been repeatedly assessed—even when using just one reference. Thus, in this paper, we present a targeted manual evaluation confirming that our BLEU score increase corresponds to a significant rise in fluency, a practice we encourage others to adopt.

2 CCG Surface Realization

CCG (Steedman, 2000) is a unification-based categorial grammar formalism defined almost entirely in terms of lexical entries that encode subcategorization as well as syntactic features (e.g. number and agreement). OpenCCG is a parsing/generation library which includes a hybrid symbolic-statistical chart realizer (White, 2006). A vital component of the realizer is the hypertagger (Espinosa et al., 2008), which predicts lexical category assignments using a maxent model trained on contexts within a directed graph structure representing the logical form (LF) input; features and relations in the graph as well as parent child relationships are the main features used to train the model. The realizer takes as input an LF description (see Figure 1 of Espinosa et al., 2008), but here we also

Proceedings of NAACL HLT 2009: Short Papers, pages 161–164,
Boulder, Colorado, June 2009.

use LFs with class information on some elementary predications (e.g. $@_{x:\mathrm{MONEY}}(\mathbf{\$\_10{,}000})$). Chart realization proceeds in iterative beta-best fashion, with a progressively wider hypertagger beam width. If no complete realization is found within the time limit, fragments are greedily assembled. Alternative realizations are ranked using integrated n-gram scoring; n-gram models help in choosing word order and, to a lesser extent, making lexical choices.

3 Collapsing Named Entities

An error analysis of the OpenCCG baseline output reveals that out of 2331 NEs annotated by the BBN corpus, 238 are not realized correctly. For example, multi-word NPs like *Texas Instruments Japan Ltd.* are realized as *Japan Texas Instruments Ltd.*. Inspired by Hogan et al.'s (2007)'s Experiment 1, we decided to use the BBN corpus NE annotation (Weischedel and Brunstein, 2005) to collapse certain classes of NEs. But unlike their experiment where all the NEs annotated by the BBN corpus are collapsed, we chose to collapse into single tokens only NEs whose exact form can be reasonably expected to be specified in the input to the realizer. For example, while some quantificational or comparatives phrases like *more than $ 10,000* are annotated as MONEY in the BBN corpus, in our view only *$_10,000* should be collapsed into an atomic unit, with *more than* handled compositionally according to the semantics assigned to it by the grammar. Thus, after transferring the BBN annotations to the CCGbank corpus, we (partially) collapsed NEs which are CCGbank constituents according to the following rules: (1) completely collapse the PERSON, ORGANIZATION, GPE, WORK OF ART major class type entitites; (2) ignore phrases like *three decades later*, which are annotated as DATE entities; and (3) collapse all phrases with POS tags CD or NNP(S) or lexical items % or *$*, ensuring that all prototypical named entities are collapsed.

4 Exploiting NE Classes

Going beyond Hogan et al. (2007) and collapsing experiments, we also experiment with NE classes in language models and hypertagging models. BBN annotates both major types and subtypes (DATE:AGE, DATE:DATE etc). For all our experiments, we use both of these.

4.1 Class replaced n-gram models

For both the original CCGbank as well as the collapsed corpus, we created language model training data with semantic classes replacing actual words, in order to address data sparsity issues caused by rare words in the same semantic class. For example, in the collapsed corpus, the Section 00 sentence *Pierre_Vinken , 61 years old , will join the board as a nonexecutive director Nov._29 .* becomes *PERSON , DATE:AGE DATE:AGE old , will join the ORG_DESC:OTHER as a nonexecutive PER_DESC DATE:DATE DATE:DATE .* During realization, word forms are generated, but are then replaced by their semantic classes and scored using the semantic class replaced n-gram model, similar to (Oh and Rudnicky, 2002). As the specific words may still matter, the class replaced model is interpolated at the word level with an ordinary, word-based language model, as well as with a factored language model over POS tags and supertags.

4.2 Class features in hypertagging

We also experimented with a hypertagging model trained over the collapsed corpus, where the semantic classes of the elementary lexical predications, along with the class features of their adjacent nodes, are added as features.

5 Evaluation

5.1 Hypertagger evaluation

As Table 2 indicates, the hypertagging model does worse in terms of per-logical predication accuracy & per-whole-graph accuracy on the collapsed corpus. To some extent this is not surprising, as collapsing eliminates many easy tagging cases; however, a full explanation is still under investigation. Note that class information does improve performance somewhat on the collapsed corpus.

5.2 Realizer evaluation

For a both the original CCGbank and the collapsed corpus, we extracted a section 02–21 lexicogrammars and used it to derive LFs for the development and test sections. We used the language models in Table 1 to score realizations and for the

Condition	Expansion
LM	baseline-LM: word 3g+ pos 3g*stag 3g
HT	baseline Hypertagger
LM4	LM with 4g word
LMC	LM with class-rep model interpolated
LM4C	LM with both
HTC	HT with classes on nodes as extra feats

Table 1: Legend for Experimental Conditions

Corpus	Condition	Tags/pred	Pred	Graph
Uncollapsed	HT	1.0	93.56%	39.14%
	HT	1.5	98.28%	78.06%
Partly Collapsed	HT	1.0	92.22%	35.04%
	HTC	1.0	92.89%	38.31%
	HT	1.5	97.87%	73.14%
	HTC	1.5	98.02%	75.30%

Table 2: Hypertagger testing on Section 00 of the uncollapsed corpus (1896 LFs & 38104 predicates) & partially collapsed corpus (1895 LFs & 35370 predicates)

collapsed corpus, we also tried a class-based hypertagging model. Hypertagger β-values were set for each corpus and for each hypertagging model such that the predicted tags per pred was the same at each level. BLEU scores were calculated after removing the underscores between collapsed NEs.

5.3 Results

Our baseline results are much better than those previously reported with OpenCCG in large part due to improved grammar engineering efforts and bug fixing. Table 3 shows development set results which indicate that collapsing appears to improve realization on the whole, as evidenced by the small increase in BLEU scores. The class-replaced word model provides a big boost on the collapsed corpus, from 0.7917 to 0.7993, much more than 4-grams. Adding semantic classes to the hypertagger improves its accuracy and gives us another half BLEU point increase. Standard test set results, reported in Table 4, confirm the overall increase, from 0.7940 to 0.8173.

In analyzing the Section 00 results, we found that with the collapsed corpus, NE errors were reduced from 238 to 99, which explains why the BLEU score increases despite the drop in exact matches and grammatically complete realizations from the baseline. A semi-automatic analysis reveals that most of the corrections involve proper names that are no longer mangled. Correct adjective ordering is also achieved in some cases; for example, *Dutch publishing group* is enforced by the class-replaced models, while all the other models realize this as *publishing Dutch group*. Additionally, the class-replaced model sometimes helps with animacy marking on relative pronouns, as in *Mr. Otero , who ...* instead of *Mr. Otero , which ....* (Note that our input LFs do not directly specify the choice of function words such as case-marking prepositions, relative pronouns and complementizers, and thus class-based scoring can help to select the correct surface word form.)

Corpus	Condition	%Exact	%Complete	BLEU
Uncollapsed (98.6% coverage)	LM+HT	29.27	84.02	0.7900
	LM4+HT	29.14	83.61	0.7899
	LMC+HT	30.64	83.70	0.7937
	LM4C+HT	30.85	83.65	**0.7946**
Partly collapsed (98.6% coverage)	LM+HT	28.28	82.48	0.7917
	LM4+HT	28.68	82.54	0.7929
	LMC+HT	30.74	82.33	0.7993
	LM4C+HT	31.06	82.33	0.7995
	LM4C+HTC	32.01	83.17	**0.8042**

Table 3: Section 00 blind testing results

Condition	%Exact	%Complete	BLEU
LM+HT	29.38	82.53	0.7940
LM4C+HTC	33.74	85.04	**0.8173**

Table 4: Section 23 results: LM+HT baseline on original corpus (97.8% coverage), LM4C+HTC best case on collapsed corpus (94.8% coverage)

5.4 Targeted manual evaluation

While the language models employing NE classes certainly improve some examples, others are made worse, and some are just changed to different, but equally acceptable paraphrases. For this reason, we carried out a targeted manual evaluation to confirm the BLEU results.

5.4.1 Procedure

Along the lines of (Callison-Burch et al., 2006), two native speakers (two of the authors) provided ratings for a random sample of 49 realizations that differed between the baseline and best conditions on the collapsed corpus. Note that the selection procedure excludes exact matches and thus focuses on sentences whose realization quality may be lower on average than in an arbitrary sample. Sentences were rated in the context of the preceding sentence (if any) for both fluency and adequacy in comparison to the original sentence. The judges were not

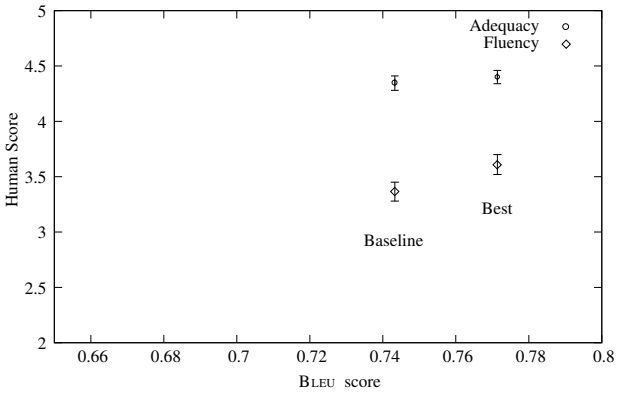

Figure 1: BLEU scores plotted against human judgements of fluency and adequacy

aware of the condition (best/baseline) while doing the rating. Ratings of the two judges were averaged for each item.

5.4.2 Results

In the human evaluation, the best system's mean scores were 4.4 for adequacy and 3.61 for fluency, compared with the baseline's scores of 4.35 and 3.36 respectively. Figure 1 shows these results including the standard error for each measurement, with the BLEU scores for this specific test set. The sample size was sufficient to show that the increase in fluency from 3.36 to 3.61 represented a significant difference (paired t-test, 1-tailed, p = 0.015), while the adequacy scores did not differ significantly.

5.4.3 Brief comparison to related systems

While direct comparisons cannot really be made when inputs vary in their semantic depth and specificity, we observe that our all-sentences BLEU score of 0.8173 exceeds that of Hogan et al. (2007), who report a top score of 0.6882 (though with coverage near 100%). Nakanishi et al. (2005) and Langkilde-Geary (2002) report scores of 0.7733 and 0.7570, respectively, though the former is limited to sentences of length 20 or less, and the latter's coverage is much lower.

6 Conclusion and Future Work

In this paper, we have shown how named entity classes can be used to improve the OpenCCG realizer's language models and hypertagging models, helping to achieve a state-of-the-art BLEU score of 0.8173 on CCGbank Section 23. We have also confirmed the increase in quality through a targeted manual evaluation, a practice we encourage others working on surface realization to adopt. In future work, we plan to investigate the unexpected drop in hypertagger performance on our NE-collapsed corpus, which we conjecture may be resolved by taking advantage of Vadas and Curran's (2008) corrections to the CCGbank's NP structures.

7 Acknowledgements

This work was supported in part by NSF IIS-0812297 and by an allocation of computing time from the Ohio Supercomputer Center. Our thanks also to Josef Van Genabith, the OSU Clippers group and the anonymous reviewers for helpful comments and discussion.

References

Chris Callison-Burch, Miles Osborne, and Philipp Koehn. 2006. Re-evaluating the role of BLEU in machine translation research. In *Proc. EACL*.

Dominic Espinosa, Michael White, and Dennis Mehay. 2008. Hypertagging: Supertagging for surface realization with CCG. In *Proc. ACL-08:HLT*.

Deirdre Hogan, Conor Cafferkey, Aoife Cahill, and Josef van Genabith. 2007. Exploiting multi-word units in history-based probabilistic generation. In *Proc. EMNLP-CoNLL*.

Irene Langkilde-Geary. 2002. An empirical verification of coverage and correctness for a general-purpose sentence generator. In *Proc. INLG-02*.

Hiroko Nakanishi, Yusuke Miyao, and Jun'ichi Tsujii. 2005. Probabilistic methods for disambiguation of an HPSG-based chart generator. In *Proc. IWPT-05*.

Alice H. Oh and Alexander I. Rudnicky. 2002. Stochastic natural language generation for spoken dialog systems. *Computer, Speech & Language*, 16(3/4):387–407.

Mark Steedman. 2000. *The syntactic process*. MIT Press, Cambridge, MA, USA.

David Vadas and James R. Curran. 2008. Parsing noun phrase structure with CCG. In *Proc. ACL-08:HLT*.

Ralph Weischedel and Ada Brunstein. 2005. BBN pronoun coreference and entity type corpus. Technical report, BBN.

Michael White. 2006. Efficient Realization of Coordinate Structures in Combinatory Categorial Grammar. *Research on Language and Computation*, 4(1):39–75.