

Exploiting XLE’s Finite State Interface in LFG-based Statistical Machine Translation

Eleftherios Avramidis and Jonas Kuhn

Linguistics Department, University of Potsdam, Germany

In current work on Machine Translation (MT), purely data-driven, statistical approaches, based on very large corpora of sample translations, continue to lead to the best evaluation results, at least when tested on the same text domain as they were trained on (Callison-Burch et al., 2008). At the same time, it is conceptually clear that there are limitations to picking up certain generalizations (which can be easily described in linguistic terms) from unstructured training data – Zipf’s law has it that the multitude of types of linguistic units occur rather infrequently in corpus data. So, it has been identified as one of the major challenges to combine the highly successful statistical techniques with insights from deep linguistic processing. The challenge is greater than one may first think, as it turns out that the straightforward ways of constraining the statistical models to apply only to linguistically warranted units lead to a drop in performance (Koehn et al., 2003; Chiang, 2005). (This is essentially because the unconstrained system will quite often learn to produce a reasonable translation for some combination of words that does not form a linguistic unit at any level.)

LFG is an excellent candidate for an exploration of more sophisticated ways of combining statistics and deep linguistic analysis, thanks to the assumption of parallel correspondence across levels. In this contribution, we present work that addresses the syntax-morphology interface – a very obvious candidate for an exploitation of linguistic generalizations in data-driven MT: when translating into a language that is morphologically more challenging than English (in our case, German), a linguistically motivated morphological analyzer can break down word forms into a lemma and a particular set of morphosyntactic features (e.g., Mann +NN .Masc .Gen .Sg for the form *Mannes* (“man’s”). Lexical generalizations can then be learned for a lemma and generalized to other forms than the ones seen in training. For instance, having learned that *starke Unwetter* (lit. “strong un-weather”) is a good translation for *heavy windstorms*, the system will be able to generalize to translating *a heavy windstorm* as *ein starkes Unwetter* and *after the heavy windstorm* as *nach dem starken Unwetter*. (This is obviously even more critical when translating between *two* morphologically rich languages, where it is much less likely that the system will automatically fall back on a translation that is only incorrect in terms of agreement.)

We build on a statistical tree-to-string translation approach similar to (Hopkins and Kuhn, 2007), working with the XLE system and the grammars developed in the ParGram project (Butt et al., 2002) and a parallel corpus, word-aligned with GIZA++ (Och and Ney, 2003). Information from the source language LFG analysis drives a “tree labelling” approach to translation: a cascade of statistical (discriminative) classifiers is trained that traverses the c-structure analysis, taking into account f-structure information and all previous decisions. The new “labels” assigned to the source c-structure tree will contain target language word forms and re-structuring instructions, so a particular target language string can be read off the final tree.

In the present work, we extend the cascade of statistical classifiers, so it will not insert full target language word forms in the tree labels, but just the lemma in a first step. The morphosyntactic feature specification is then added in separate classification steps, so it can take all available information into account (such as agreement information from previously generated target words). For training, we applied a ParGram LFG grammar not only to the source language (English), but also to the target language (German). Since XLE incorporates finite-state transducers (FSTs) for preprocessing (tokenization) and morphological analysis, the German parses contain a syntactically disambiguated morphological analysis for all words, which is exactly what is needed as training material for the extended tree labelling approach we just described: instead of full form like *starke Unwetter*, we use the following representation of the target language words to train the tree labeller: `stark +ADJ .Pos .MFN .NA .Pl .St Unwetter +NN .Neut .NGA .Pl`.

After training is finished, the tree labelling translator is applied to new input (English sentences) as follows: the English LFG parser is used to produce an annotated c-structure tree. The cascade of statistical classifiers is applied to add the translation labels to this tree, which are then read out to produce a string of lemmata and morphosyntactic features. This again can be fed into the target language morphological analyzer (run in reverse mode, i.e., as a morphological generator). In this final step, a systematic issue arises, which is at the core of our present contribution. To understand the issue, it has to be noted that the feature representations used within the morphological analyzers (Schiller and Steffens, 1990) rely on a compact underspecified feature format to avoid a proliferation of disjunctive analyses for ambiguous word forms. For instance, the form *Mann* (“man”) can be either nominative, dative, or accusative singular (only the genitive singular differs: *Mannes*). The morphological grammar assigns the following analysis to *Mann*: `Mann +NN .Masc .NDA .Sg`. The case tag `.NDA` combines the tags for nominative, dative and accusative in one compact tag. Other singular nouns are case ambiguous for all four cases, e.g., *Frau* (“woman”), which is assigned the case tag `.NGDA`. Similar tag combinations occur for other morphosyntactic features, such as gender, number, and mood.

This compact feature representation leads to the following issue in translation: as the assignment of labels is trained from output of the morphology, the system will of course pick up generalizations that involve combined tags like `.NDA`.

It may turn out however that the translator ends up using such a tag with a lemma that has a slightly different inflection paradigm (e.g., producing `Frau +NN .Fem .NDA .Sg` instead of `Frau +NN .Fem .NGDA .Sg`). Running the incorrect sequence through the morphological generator will result in a failure.

One may argue that one should try to improve the training so it will only produce “legal” sequences. However, this would reduce the effectiveness of the training with a given amount of data. In addition, it should be noted that we are seeing the effect of a representational short-hand that was intended for a different applicational context.

Fortunately, it is relatively straightforward to augment the pre-processing FSTs used in XLE by a “correction” module: using the FST composition operation, we can map combined tags like `.NDA` to other, overlapping combined tags like `.NGDA`, operating in two stages (which are however compiled out in the resulting FST). A new “recombination” FST is defined that adds a set of replace rules in order to (a) explicate the combined tags towards their component features (e.g., replacing `.NDA` with `.Nom`, `.Dat` or `.Acc`, disjunctively) and then (b) to generalize these to get a disjunction of all the (other) possible tag combinations that may contain them. This way, the desired tags would be taken into consideration, even if they are more or less explicit than the expected.

In other cases, the step-by-step generation of morphological tags performed in translation may even lead to an incorrect assignment of unambiguous features tags, e.g., assigning `.Fem` to a noun that is actually masculine, e.g., leading to `Mann +NN .Fem .NDA .Sg`. It is clearly desirable to rely on the morphological grammar for overriding such incorrect feature markings. Of course, we only want to change a feature like `.Fem` into `.Masc` in situations where the former analysis is indeed incompatible with the morphological grammar. For this, a default cascade is needed: we only want to apply certain correction modules in case the original FST cascade fails. The finite-state operation of *priority union* (Guingne et al., 2003) can be used to this effect (as a unification operation, it was proposed in 1985 by Kaplan (1995)). Combining two FSTs with priority union has the effect that the second one is only applied to a given input in case the input is not included in the upper side of the first FST. For instance, we may generally apply the mentioned “recombination” FST, and if this combination does not lead to a result, we prefix an additional feature correction FST:

$$(T_{recomb} \circ T_{morph}) \bigcup^P (T_{correct} \circ T_{recomb} \circ T_{morph}) \quad (1)$$

$$\equiv (T_{recomb} \circ T_{morph}) \cup (\neg upper(T_{recomb} \circ T_{morph}) \circ (T_{correct} \circ T_{recomb} \circ T_{morph})) \quad (2)$$

where T_{morph} is the existing morphology generator, T_{recomb} is the tag recombination transducer and $T_{correct}$ a FST cascade for substituting tags that may fail during the first generation.

In experimental results on a part of the Europarl corpus (Koehn, 2005), we managed to reduce the word generation failure rate from 19.4 % to 12.3 % by using the recombination transducer and to 7.0 % by using recombination, plus correction when needed.

References

- Butt, M., Dyvik, H., King, T., Masuichi, H., and Rohrer, C. (2002). The Parallel Grammar project. *International Conference On Computational Linguistics*, pages 1–7.
- Callison-Burch, C., Koehn, P., Monz, C., Schroeder, J., and Fordyce, C. S., editors (2008). *Proceedings of the Third Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Columbus, Ohio.
- Chiang, D. (2005). A hierarchical phrase-based model for statistical machine translation. In *Proceedings of ACL*, pages 263–270.
- Guingne, F., Nicart, F., Champarnaud, J., Karttunen, L., Gaal, T., and Kempe, A. (2003). Virtual operations on virtual networks: The priority union. *International Journal of Foundations of Computer Science*, 14(6):1055–1070.
- Hopkins, M. and Kuhn, J. (2007). Deep Grammars in a Tree Labeling Approach to Syntax-based Statistical Machine Translation. In *Proceedings of ACL Workshop on Deep Linguistic Processing 2007*.
- Kaplan, R. M. (1995). Three seductions of computational psycholinguistics. In Dalrymple, M., Kaplan, R. M., Maxwell, J. T., and Zaenen, A., editors, *Formal Issues in Lexical-Functional Grammar*. CSLI Publications.
- Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. *MT Summit*, 5.
- Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. In *Proceedings of the Human Language Technology Conference 2003 (HLT-NAACL 2003)*, Edmonton, Canada.
- Och, F. J. and Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.
- Schiller, A. and Steffens, P. (1990). A two-level environment for morphological descriptions and its application to problems of german inflectional morphology. *Terminology and Knowledge Engineering*, 1:318–329.