

A Hybrid Translation Model using LFG-DOP

Andy Way*,

School of Computer Applications,
Dublin City University.

Email: away@compapp.dcu.ie

Abstract

The Data-Oriented Parsing Model (DOP, [1]) has been presented as a promising paradigm for NLP. It has also been used as a basis for Machine Translation (MT) — Data-Oriented Translation (DOT, [5]). Lexical Functional Grammar (LFG, [3]) has also been used for MT ([4]). LFG has recently been allied to DOP to produce a new LFG-DOP model ([2]) which improves the robustness of LFG.

We summarise the DOT model of translation based on DOP. We demonstrate that DOT is not guaranteed to produce the correct translation, despite provably deriving the most probable translation. Finally, we propose a novel hybrid model for MT based on LFG-DOP which promises to improve upon DOT, as well as the pure LFG-based translation model.

1 Introduction

DOP ([1]) has been presented as a promising paradigm for NLP. It has also been used as a basis for MT — DOT ([5]). LFG ([3]) has also been used for MT ([4]), and has recently been allied to DOP to produce a new LFG-DOP model ([2]) which improves the robustness of LFG.

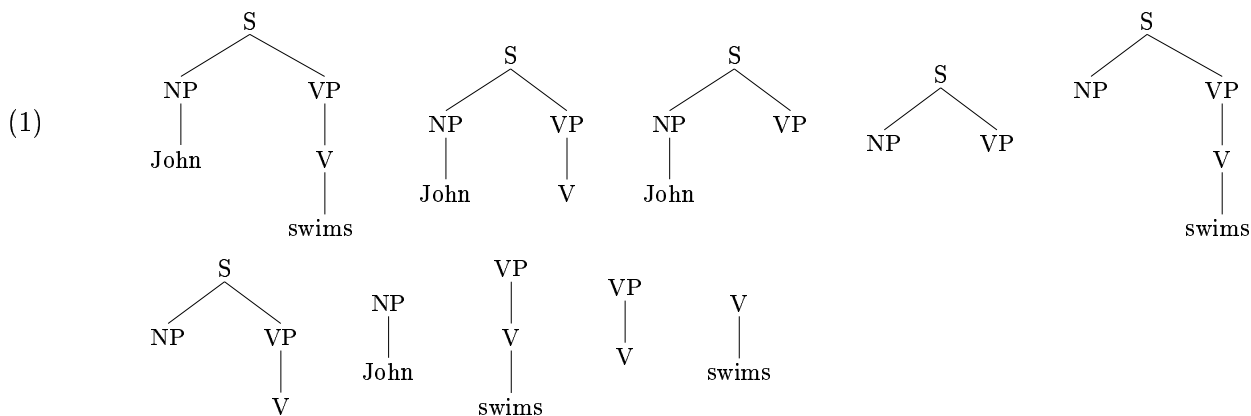
We summarise the DOT model of translation, as well as the DOP model on which it is based. We demonstrate that DOT is not guaranteed to produce the correct translation, despite provably deriving the most probable translation. Finally, we propose a novel hybrid model for MT based on LFG-DOP which promises to improve upon DOT, as well as the pure LFG-based translation model.

*Much of this work was done whilst at the CLMT group at the University of Essex. Thanks to Doug Arnold, Frederik Fouvry and Louisa Sadler for useful discussion there. In addition, thanks are due to Rens Bod, Ron Kaplan, Boris Cormons, Josef Van Genabith and Tony Veale for further helpful comments. Any remaining errors or omissions are of course my fault. Finally, thanks to DCU for funding this research.

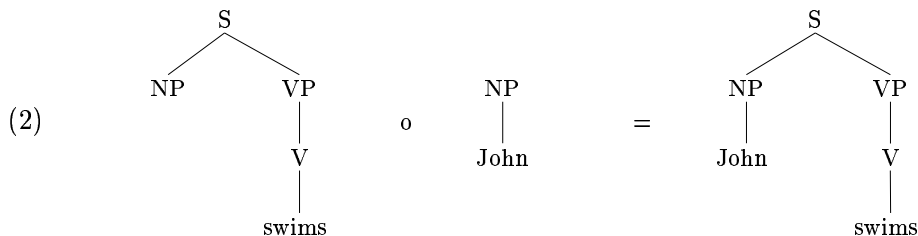
2 The DOP Architecture for NLP

DOP language models ([1]) assume that past experiences of language are significant in both perception and production. DOP prefers performance models over competence grammars, in that abstract grammar rules are eschewed in favour of models based on large collections of previously occurring fragments of language. New language fragments are processed with reference to already existing fragments from the corpus, which are combined using probabilistic techniques to determine the most likely analysis for the new fragment.

DOP models typically use surface PS-trees as the chosen **representation** for strings (hence “Tree-DOP”), but nothing hangs on this choice. However, given that LFG c-structures are little more than annotated PS-trees allows us to proceed very much on the same lines as in Tree-DOP, which has two **decomposition** operations to produce subtrees from sentence representations: (i) the *Root* operation, which takes any node in a tree as the root of a new subtree, deleting all other nodes except this new root and all nodes dominated by it; and (ii) the *Frontier* operation, which selects a (possibly empty) set of nodes in the newly created subtree, excluding the root, and deletes all subtrees dominated by these selected nodes. For instance, the full set of DOP trees derived from the sentence *John swims* would be as in (1):



Tree-DOP **recombines** fragments starting from the leftmost non-terminal frontier node, and replaces this with a fragment having the same root symbol. For instance, assuming the treebank in (1), *John swims* has (2) as a possible derivation (among many others):



Finally, the chosen **probability model** for Tree-DOP is based quite simply on the relative fre-

quencies of fragments in the corpus.

These elements enable representations of new strings to be constructed from previously occurring fragments in a number of ways. If each derivation t has a probability $P(t)$ (i.e. its relative frequency), then the probability of deriving a Tree-DOP representation is the sum of the probabilities of the individual derivations, as in (3):

$$(3) \quad \sum_i \prod_j \frac{\#(t_{ij})}{\#(t \mid \text{root}(t) = \text{root}(t_{ij}))}$$

The probability of each individual derivation t is calculated as the product of the probabilities of all the constituent elements ($\langle t_1, t_2 \dots t_n \rangle$, here) involved in choosing tree t from the corpus, as in:

$$(4) \quad P(\langle t_1, t_2 \dots t_n \rangle) = \prod_{i=1}^n \frac{P(t)}{\sum_{t' \in \text{corpus}} P(t')}$$

2.1 Opportunities for Hybridity—LFG DOP

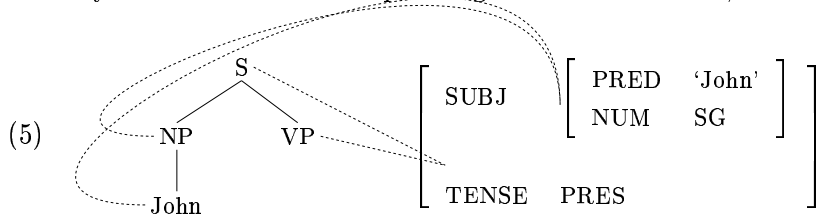
DOP-based approaches are necessarily limited to those contextual dependencies actually occurring in the corpus, which is a reflection of surface phenomena only. Given its facility to capture and provide representations of linguistic phenomena other than those occurring at surface structure, the functional structures of LFG have been allied to the techniques of DOP to create a new model, LFG-DOP ([2]), which adds a measure of robustness not available to models based solely on LFG. We suggest that this framework has the potential to be utilised for MT.

As with DOP, LFG-DOP needs to be defined using four parameters. Its **representations** are simply lifted *en bloc* from LFG theory, so that each string is annotated with a c-structure, an f-structure, and a mapping ϕ between them, with well-formedness conditions operating solely on f-structure, as usual.

Since we are now dealing with $\langle c, f \rangle$ pairs of structure, the *Root* and *Frontier decomposition* operations of DOP need to be adapted to stipulate exactly which c-structure nodes are linked to which f-structure components, thereby maintaining the fundamentals of c- and f-structure correspondence. As in DOP, *Root* erases all nodes outside of the selected node, and in addition deletes all ϕ -links leaving the erased nodes, as well as all f-structure units that are not ϕ -accessible from the remaining nodes, reflecting the intuitive notion that nodes in a tree carry information only about the f-structure elements to which the root node of the tree permits access.

Frontier operates as in DOP, and furthermore deletes all ϕ -links of these erased nodes together

with any semantic form corresponding to the same nodes, as in (5):



which illustrates the ability of *Root* nodes to access certain features (TENSE, here) even after subnodes have been deleted. (5) can be pruned still further by applying a third, and new operation, *Discard*, to the TENSE feature. *Discard* adds considerably to LFG’s robustness by providing generalised fragments from those derived via *Root* and *Frontier*.

Composition is also a two-step operation. C-structures are combined by leftmost substitution, as in DOP, subject to the matching of their nodes. F-structures corresponding to these nodes are then recursively unified, and the resulting f-structures are subjected to the grammaticality checks of LFG.

Finally, $P(f \mid CS)$ denotes the probability of choosing a fragment f from a competition set CS of competing fragments. [2] describe four possible competition sets linked to the **probability models** for LFG-DOP: (i) a straightforward extension of the DOP probability model where the choice of a fragment depends only on its *Root* node and not on the Grammaticality conditions of LFG; (ii) c-structure nodes must match, and f-structures must be unifiable if two LFG fragments are to be combined, i.e. taking into account the LFG Uniqueness condition as well as the *Root* category; (iii) furthermore, the LFG Coherence check is enforced at each step; and (iv) finally, all LFG grammaticality checks, as well as the DOP category-matching stipulation, are left to the end. Note that in models (i)–(iii) the category matching condition is enforced on-line whilst all LFG checks are either performed on-line or *post hoc*, whereas given the non-monotonic nature of the Completeness check, this can only ever be enforced *post hoc*.

3 Data-Oriented Translation (DOT)

[5] has developed a DOP-based model of translation — Data-Oriented Translation — which relates POS-fragments between two languages (English and Dutch), with an accompanying probability. Once a derivation for the source language sentence has been arrived at, the target structure is assembled, and a string produced. Since there are typically many different derivations for the source sentence, there may be as many different translations available. As is the case when DOP is used monolingually, the probability of a translation is calculated by summing the probabilities of all possible derivations forming the translation. Poutsma shows that the most probable translation

can be computed using Monte-Carlo disambiguation, and exemplifies this using sentence idioms, where corresponding source-target translations are linked at all possible nodes.

3.1 Some Limitations of DOT

DOT is an interesting model, yet it fails to capture the correct translation when this is non-compositional and considerably less probable than the default, compositional alternative. When LFG-DOP MT is used instead this problem can be overcome. Furthermore, DOP’s statistical model also gives a “level of correctness” figure to alternative translations. This is useful in cases where the default translation in LFG-MT (and in many other systems) cannot be suppressed when the specific translation is required. For example, assuming the basic default rules in (6):

- (6) a. commettre \Leftrightarrow commit
- b. suicide \Leftrightarrow suicide

in order to deal with the sentences in (7):

- (7) a. Jean commet un crime \Leftrightarrow Jean commits a crime
- b. Le suicide est tragique \Leftrightarrow Suicide is tragic

we would get the wrong translation where *John commits suicide* \Leftrightarrow **John commet le suicide* (cf. *John se suicide*). Since we require specific rules to override the default translation where applicable, in LFG-MT we would get both translations here, i.e. a correct one and a wrong one. Assuming a DOP treebank built from the French sentences in (7) as well as *Marie se suicide*, the ill-formed string *Jean commet le suicide* is preferred (in the French language model) about half as much again as the correct alternative *Jean se suicide*. There are several reasons for this: the preference for *Jean* as subject of *commettre*, the co-occurrence of *le* and *suicide*, plus the fact that *commettre* is followed by an NP consisting of a Det + N sequence. Note also that these results are obtained with the same number of instances of each verb — in a larger corpus *commettre* would surely greatly outnumber instances of *se suicider*.

This is by no means an unexpected result. As an example, in the LOB Corpus, there are 66 instances of *commit* as a verb (including its morphological variants), only 4 of which have *suicide* as its object, out of the 15 occurrences of *suicide* as an NP. Consequently, even for this small sample, we can see that 94% of these examples need to be translated compositionally (by *commettre* + NP), while only the *commit suicide* examples require a specific rule to apply (i.e. *se suicider*). In the on-line Canadian Hansards covering 1986-1993, there are just 106 instances of *se suicider* (including its morphological variants). There will, of course, be many thousands of instances of *commettre*. Given occurrences of *suicide* as an NP in French corpora, it is not an unreasonable hypothesis to

expect that the wrong, compositional translations will be much more probable than those derived via the specific rule.

Given Poutsma’s model, it would appear that the adherence to left-most substitution in the target given *a priori* left-most substitution in the source is too strictly linked to linear order of words, so that, as soon as this deviates to any significant extent even between similar languages, DOT has a huge bias in favour of the incorrect translation. Even if the correct, non-compositional translation is achievable in such circumstances via DOT, it is likely to be so outranked by other wrong alternatives that it will be dismissed, unless all possible translations are maintained for later scrutiny by the user.

One line of investigation which we now develop that can overcome this linear restriction is to use LFG’s τ -equations to relate translation fragments between languages.

4 LFG-DOP MT

The DOT model cannot explicitly relate parts of the source language structure to the corresponding, correct parts in the target structure. We therefore propose the use of LFG-DOP ([2]) as the basis for an innovative MT system, given that LFG’s τ -equations are able to link exactly those source-target elements which are translations of each other.

Using separate language corpora, one simple, linear model would build a target f-structure f' from a source c-structure c and f-structure f , the mapping between them ϕ , and the tau-equations τ . From this target f-structure f' , a target string will be generated via the standard LFG generation algorithms ([6]). The different components needed are (i) a source language LFG-DOP model; (ii) the τ mapping; and (iii) a target language LFG-DOP model. The advantage of this model over DOT is the availability of the explicit τ -equations to link source-target correspondences, as in (8):

$$\begin{aligned}
 (8) \quad & \textit{commit} \\
 & (\tau \uparrow \text{PRED}) = \textit{se suicider} \\
 & \tau(\uparrow \text{SUBJ}) = (\tau \uparrow \text{SUBJ}) \\
 & (\uparrow \text{OBJ PRED}) =_c \textit{suicide}
 \end{aligned}$$

Using LFG τ -equations ensures the derivation of the correct target f-structure, along with some wrong alternatives via the default rules. We cannot be sure that the generation of a target string via the correct target f-structure will be a more probable translation than any wrong alternative, but it will exist as one of a small number of high-ranking candidate solutions from which the final translation can be selected. Of course, the better the target language LFG-DOP model, the more likely the correct translation is to be the most preferred translation.

5 Conclusions

The DOT translation system, despite provably deriving the most probable translation, is not guaranteed to produce the best, or even a correct translation, since it is unable to explicitly link exactly those fragments which are playing the decisive role in translation.

[2] have shown how DOP and LFG can be integrated to provide a powerful mechanism for the treatment of parsing. We described how such a model may be extended to provide a robust solution for the problems of MT in the spirit of the current trend for hybrid approaches. LFG-DOP MT promises to improve on previous attempts at LFG-MT, particular where robustness is concerned, being able to handle both unseen and ill-formed input with relative ease. It also ensures that the correct target f-structure is input into the generation process. It is reasonable to expect it to outperform pure statistics-based systems, in having the additional facility of grammatical information at hand to use where necessary.

Much of this work is ongoing, and a number of issues remain for the future. Finally, of course, we seek to complete the development of the systems described, leading to greater experimentation on a larger scale.

References

- [1] Bod, R. (1995): *Enriching Linguistics with Statistics: Performance Models of Natural Language*, ILLC Dissertation Series 1995-14, University of Amsterdam, The Netherlands.
- [2] Bod, R. & R. Kaplan (1998): "A Probabilistic Corpus-Driven Model for Lexical-Functional Analysis", in *COLING: Proceedings of the 17th International Conference on Computational Linguistics & 36th Conference of the Association for Computational Linguistics*, Montreal, Canada, 1:145–151.
- [3] Kaplan, R. & J. Bresnan, (1982): "Lexical Functional Grammar: A Formal System for Grammatical for Grammatical Representation", in J. Bresnan (ed.) *The Mental Representation of Grammatical Relations*, MIT Press, Cambridge, Mass., pp.173–281.
- [4] Kaplan, R., K. Netter, J. Wedekind & A. Zaenen (1989): "Translation by Structural Correspondences", in *Fourth Conference of the European Chapter of the Association for Computational Linguistics*, Manchester, pp.272–281.
- [5] Poutsma, A. (1998): "Data-Oriented Translation", in *Ninth Conference of Computational Linguistics In the Netherlands*, Leuven, Belgium.
- [6] Wedekind, J. (1988): "Generation as Structure Driven Derivation", in *COLING: 12th International Conference on Computational Linguistics*, Budapest, Hungary, pp.732–737.