Learning context-sensitive synchronous rules*

Anders Søgaard

Dpt. of Linguistics
University of Potsdam
soegaard@ling.uni-potsdam.de

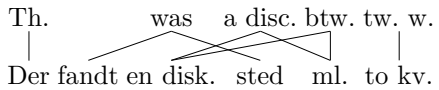
Abstract. Context-sensitive alignments are shown to be frequent in hand-aligned parallel corpora, e.g. in 24%–85% of the sentence pairs in the corpora documented in [GPCC08]. A $\mathcal{O}(|G|n^6)$ time strict extension of inversion transduction grammars (ITGs) [Wu97] called (2,2)-BRCGs is proposed in [Søg08] that induces such alignments. The increase in generative capacity comes from the ability to copy strings in derivations, which means that the (i) intersection of two translations and (ii) the union of two alignment structures are easily defined. The problem for real-life applications is how to induce the grammars from available resources; in particular, how to learn when copying is needed. This paper presents a quadratic time algorithm that reduces the problem of how to induce (2,2)-BRCGs from $m : n$-alignments to the same problem for ITGs by unravelling alignment structures. The algorithm was run on a parallel corpus in the Copenhagen Dependency Treebank [BK07] (Danish–English); the ratio of new alignment structures over the number of sentence pairs was 38.08%. For the ones in [GPCC08], the size of the corpora increased by a factor of 1.74–2.0.

1 Introduction

Consider the simple example of a translation from English into Danish below:

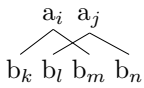
1. There was a discussion between two women.
2. Der fandt en diskussion sted mellem to kvinder.

The discontinuous constituent in Danish, *fandt sted* (lit. 'found place'), is fully idiomatic and therefore necessarily a translation unit. The non-contiguous noun-preposition pairs, *discussion between* and *diskussion mellem*, are perhaps not idiomatic, but conventionalized and idiosyncratic in the sense that the information that the nouns select the prepositions in question must be stored in their lexical entries. It is thus best to treat them as indivisible translation units. The most plausible alignment in this case is thus as follows:



* This work was supported by the German Research Foundation in the Emmy Noether project *Ptolemaios* on grammar learning from parallel corpora.

It is not difficult to see that this alignment cannot be induced by many formalisms for syntax-based machine translation, incl. inversion transduction grammar (ITG) [Wu97], synchronous context-free grammar (SCFG) [Chi07] and synchronous tree substitution grammar (STSG) [Eis03]. The important part of the alignment is this:



The structure is called a *cross-serial discontinuous translation unit* (cross-serial DTU) below. The inability of the above theories to induce cross-serial DTUs follows from the observation that if b_k and b_m in the above are generated or recognized simultaneously in any of the above theories, b_l and b_n cannot be generated or recognized simulaneously. This is a straight-forward consequence of the context-freeness of the component grammars. Context-sensitivity does not, on the other hand, imply the ability to induce cross-serial DTUs. In synchronous tree-adjoining grammar (STAG) [SS90], for instance, the adjunction operation allows us to induce DTUs, but not cross-serial ones.

Cross-serial DTUs are frequent in hand-aligned parallel corpora, e.g. the ratios of cross-serial DTUs over sentences *modulo* translation units in the corpora documented in [GPCC08] are 24%–85%. The lowest ratio was for Spanish–French, the highest for English–Portuguese. The numbers are summarized in Figure 1.

	Snt.	TUs	DTUs	DTUs/Snt.	CDTU-ms	CDTU-ms/Snt.
English–French:	100	937	95	95%	38	38%
English-Portuguese:	100	941	100	100%	85	85%
English–Spanish:	100	950	90	90%	50	50%
Portuguese–French:	100	915	77	77%	27	27%
Portuguese–Spanish:	100	991	80	80%	55	55%
Spanish–French	100	975	74	74%	24	24%

Fig. 1. Frequency of cross-serial DTUs in the hand-aligned parallel corpora documented in [GPCC08].

[Søg08] introduces a $\mathcal{O}(|G|n^6)$ strict extension of ITGs called binary two-variable bottom-up non-erasing range concatenation grammars ($(2,2)$-BRCGs). In $(2,2)$-BRCGs the above pair of input strings would be copied and parsed twice; the alignment of each DTU is then induced by the parse of a separate copy. Here's an example of a clause that induces an alignment of one of the DTUs, but leaves the other nodes unaligned (see below for a formal definition of $(2,2)$-BRCGs):

$$VP(\mathit{was}\ X_1X_2, \mathit{fandt}\ X_1\mathit{sted}\ X_2) \rightarrow NP(X_1, X_2)PP(X_1, X_2)$$

Intuitively, *was* translates into *fandt sted*, but in Danish an object NP with its PP argument postponed may intervene between the two words.

The problem for real-life application of course is how to induce such grammars from available resources; in particular, how to learn when copying is needed. This paper presents a quadratic time algorithm that reduces the problem of how to induce $(2,2)$-BRCGs from $m:n$-alignments to the same problem for ITGs. In particular, it is linear in the size of the alignment structure and thereby quadratic in the length of the sentence pair.

2 (2,2)-BRCGs

The introduction to (2,2)-BRCGs is very brief, but lengthier introductions and more examples can be found in [Bou98] and [Søg08].

Definition 1 ((2,2)-RCGs). *(2,2)-RCGs are 5-tuples $G = \langle N, T, V, P, S\rangle$. N is a finite set of predicate names with an arity function ρ: $N \rightarrow \{1,2\}$, T and V are finite sets of terminal and non-terminal symbols. P is a finite set of clauses of the form $\psi_0 \rightarrow \phi$, $\phi = \psi_1 \ldots \psi_m$, where $0 \leq m \leq 2$ and each of the $\psi_i, 0 \leq i \leq m$, is a predicate of the form $A(\alpha_1, \ldots, \alpha_{\rho(A)})$. Each $\alpha_j \in (T \cup V)^*$, $1 \leq j \leq \rho(A)$, is an argument. $S \in N$ is the start predicate name with $\rho(S) = 2$.*

A (2,2)-RCG is said to be *bottom-up non-erasing*, i.e. a (2,2)-BRCG, if and only if for all clauses $c \in P$ all variables that occur in the RHS of c also occur in its LHS of c.

The language of a (2,2)-BRCG is based on the notion of *range*. For a string pair $w_1 \ldots w_n, v_{n+2} \ldots v_{n+m}$ a range is a pair of indices $\langle i, j\rangle$ with $0 \leq i \leq j \leq n$ or $n < i \leq j \leq n+m$, i.e. a string span, which denotes a substring $w_{i+1} \ldots w_j$ in the source string or a substring $v_{i+1} \ldots v_j$ in the target string. Only consequtive ranges can be concatenated into new ranges. Terminals, variables and arguments in a clause are bound to ranges by a substitution mechanism. An *instantiated* clause is a clause in which variables and arguments are consistently replaced by ranges; its components are *instantiated predicates*. For example $A(\langle g \ldots h\rangle, \langle i \ldots j\rangle) \rightarrow B(\langle g \ldots h\rangle, \langle i+1 \ldots j-1\rangle)$ is an instantiation of the clause $A(X_1, aY_1b) \rightarrow B(X_1, Y_1)$ if the target string is such that $v_{i+1} = a$ and $v_j = b$. A *derive* relation $\Longrightarrow$ is defined on strings of instantiated predicates. In an instantiated predicate is the LHS of some instantiated clause, it can be replaced by the RHS of that instantiated clause. The language of a (2,2)-BRCG $G = \langle N, T, V, P, S\rangle$ is the set $L(G) = \{\langle w_1 \ldots w_n, v_1 \ldots v_m\rangle \mid S(\langle 0, n\rangle, \langle 0, m\rangle) \stackrel{*}{\Longrightarrow} \epsilon\}$. In other words, an input string pair $\langle w_1 \ldots w_n, v_1 \ldots v_m\rangle$ is recognized if and only if the empty string can be derived from $S(\langle 0, n\rangle, \langle 0, m\rangle)$.

Example 1 *The grammar $G = \langle N, T, V, P, S\rangle$ with the clauses P below induces the alignment structure discussed in the introduction. The initial substrings, there and der, and the final substrings, two women and to kvinder, are ignored for brevity, since they translate directly into each other.*

$$
\begin{array}{lr}
(1) & A_0(X_1, Y_1) \rightarrow A_1(X_1, Y_1)A_2(X_1, Y_1) \\
(2) & A_1(was\ X_1, fandt\ Y_1 sted\ Y_2) \rightarrow NPP(X_1)NP(Y_1)P(Y_2) \\
(3) & A_2(X_1 a\ disc.\ btw., Y_1 en\ disk.\ Y_2 ml.\ Y_2) \rightarrow V(X_1)V(Y_1)Prt(Y_2) \\
(4) & NPP(a\ disc.\ btw.) \rightarrow \epsilon \\
(5) & NP(en\ disk.) \rightarrow \epsilon \\
(6) & P(ml.) \rightarrow \epsilon \\
(7) & V(was) \rightarrow \epsilon \\
(8) & V(fandt) \rightarrow \epsilon \\
(9) & Prt(sted) \rightarrow \epsilon
\end{array}
$$

A possible derivation of ⟨*was a discussion between, fandt en diskussion sted mellem*⟩ *is:*

$$
\begin{array}{ll}
 & A_0(\langle 0,4\rangle, \langle 0,5\rangle) \\
\Longrightarrow & A_1(\langle 0,4\rangle, \langle 0,5\rangle)A_2(\langle 0,4\rangle, \langle 0,5\rangle) \quad \textit{by (1)} \\
\Longrightarrow & NPP(\langle 1,4\rangle)NP(\langle 1,3\rangle)P(\langle 4,5\rangle)A_2(\langle 0,4\rangle, \langle 0,5\rangle) \quad \textit{by (2)} \\
\Longrightarrow & A_2(\langle 0,4\rangle, \langle 0,5\rangle) \quad \textit{by (4–6)} \\
\Longrightarrow & V(\langle 0,1\rangle)V(\langle 0,1\rangle)Prt(\langle 3,4\rangle) \quad \textit{by (3)} \\
\Longrightarrow & \epsilon \quad \textit{by (7–9)}
\end{array}
$$

Note, however, that what buys us the extra expressivity is clauses of the form:

$$A_0(X_1, Y_1) \rightarrow A_1(X_1, Y_1)A_2(X_1, Y_1)$$

Clauses of this form allows us to take the intersection of two arbitrary translations recognized by (2,2)-BRCGs. Since there is a simple translation from ITGs into (2,2)-BRCGs, this means that (2,2)-BRCG recognizes the intersective closure of translations recognized by ITGs, incl., for instance, $\{\langle a^n b^m c^n d^m, a^n c^n b^m d^m\rangle \mid m, n \geq 0\}$.

3 Unraveling alignments with DTUs

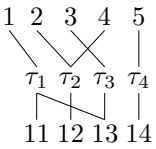
The following algorithm reduces the induction problem of (2,2)-BRCG to the same problem for ITGs by unraveling the relevant subgraphs. Say A is an alignment structure, and $CoAligned(A)$ is the set of tuples of the ordered sequences of integers $\langle i \ldots j, k \ldots l\rangle$ such that the words in the source string at positions $i \ldots j$ and the words in the target string at positions $k \ldots l$ form a translation unit. Inside-out alignments [Wu97] are ignored, since the task is only to reduce the induction problem to that for ITGs, but they are easily handled too. Simply add a subprocedure *insideout* that removes a translation unit to a new alignment structure if it is the left-most source string element in an inside-out alignment. Costly search is avoided if, as e.g. in the Copenhagen Dependency Treebank, A is read as an ordered sequence of the elements of $CoAligned(A)$, ordered by the first elements in the sequences in the first arguments of the tuples. The overall runtime will turn quadratic in the size of the alignment structure, i.e. cubic in the length of the sentence pair.

function *unravel* (A):
for $\alpha \in A$
  if *contiguous* (α) **returns false**
    move α **to** A'
    print A'
print A
end function

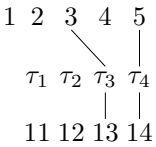
function *contiguous* (α):
if $\alpha = \langle \ldots i\ (i+j) \ldots, \ldots \rangle$, $j > 1$
  return false
elsif $\alpha = \langle \ldots, \ldots i\ (i+j) \ldots \rangle$, $j > 1$
  return false
else return true
end function

The procedure only visits each translation unit once. Consequently, the overall runtime remains quadratic in the length of the sentence pair and linear in the size of the alignment structure. Say an induction algorithm for ITGs runs in time $\mathcal{O}(n^k)$. It now follows that there is an extension of this algorithm for (2,2)-BRCG that runs in time $\mathcal{O}(n^2 + n^{k+1})$, which for all $k \geq 1$ equals $\mathcal{O}(n^{k+1})$.

Say, for instance, we have the following alignment structure:



Our algorithm reads the alignment as an ordered sequence of translation units: $\langle\langle 1, 11\ 13\rangle, \langle 2\ 4, 12\rangle, \langle 3, 13\rangle, \langle 5, 14\rangle$. It then unravels the two first translation units, $\langle 1, 11\ 13\rangle$ and $\langle 2\ 4, 12\rangle$. The translation units $\langle 3, 13\rangle$ and $\langle 5, 14\rangle$ stay in the original structure, which is now reduced to:



The three new alignment structures can all be induced by ITGs.

The unravelling algorithm was applied to the Danish–English parallel corpus in the Copenhagen Dependency Treebank [BK07]. The texts are from the Parole corpora. The aligned corpus is hand-aligned and contains 4,729 sentence pairs with a total of 110,511 translation units. Our unravelling algorithm produced 1801 new alignment structures. This number reflects that 1.63% of the translation units were DTUs.

Our unravelling algorithm was also run on the hand-aligned parallel corpora documented in [GPCC08], i.e. the first 100 sentences of the Europarl corpus

for six different language pairs. The size of the corpora increased by a factor of 1.74–2.0. See Figure 1 for details.

4 Conclusion and future work

This paper provides empirical motivation for context-sensitive synchronous rules. The main obstacle for real-life applications to machine translation is how to induce context-sensitive grammars from available resources. This paper describes a linear time algorithm that reduces the induction problem for (2,2)-BRCGs to the induction problem for ITGs.

An alignment and translation system based on (2,2)-BRCGs is currently being implemented at the University of Potsdam. It assigns (2,2)-BRCG derivations to all aligned sentence pairs in a parallel corpus and estimates a probabilistic grammar from the derivations. It introduces copying clauses for all alignment structures that are unravelled and uses them to induce complex alignment structures.

References

[BK07] Matthias Buch-Kromann. Computing translation units and quantifying parallelism in parallel dependency treebanks. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, the Linguistic Annotation Workshop*, pages 69–76, 2007.

[Bou98] Pierre Boullier. Proposal for a natural language processing syntactic backbone. Technical report, INRIA, Le Chesnay, France, 1998.

[Chi07] David Chiang. Hierarchical phrase-based translation. *Computational Linguistics*, 33(2):201–228, 2007.

[Eis03] Jason Eisner. Learning non-isomorphic tree mappings for machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 205–208, Sapporo, Japan, 2003.

[GPCC08] Joao Graca, Joana Pardal, Luísa Coheur, and Diamantino Caseiro. Building a golden collection of parallel multi-language word alignments. In *Proceedings of the 6th International Conference on Language Resources and Evaluation*, Marrakech, Morocco, 2008.

[Søg08] Anders Søgaard. Range concatenation grammars for translation. In *Proceedings of the 22nd International Conference on Computational Linguistics*, Manchester, England, 2008. To appear.

[SS90] Stuart Shieber and Yves Schabes. Synchronous tree-adjoining grammars. In *Proceedings of the 13th Conference on Computational Linguistics*, pages 253–258, Helsinki, Finland, 1990.

[Wu97] Dekai Wu. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3):377–403, 1997.