Adjunction in hierarchical phrase-based translation

Arnoult, S.I.

Publication date
2021

Document Version
Final published version

Link to publication

Citation for published version (APA):
Arnoult, S. I. (2021). *Adjunction in hierarchical phrase-based translation*. [Thesis, fully internal, Universiteit van Amsterdam]. Institute for Logic, Language and Computation.

Adjunction in Hierarchical Phrase-Based Translation

Sophie Arnoult

Adjunction in Hierarchical Phrase-Based Translation

ILLC Dissertation Series DS-2021-04

INSTITUTE FOR LOGIC, LANGUAGE AND COMPUTATION

For further information about ILLC-publications, please contact

Institute for Logic, Language and Computation
Universiteit van Amsterdam
Science Park 107
1098 XG Amsterdam
phone: +31-20-525 6051
e-mail: illc@uva.nl
homepage: http://www.illc.uva.nl/

The investigations were supported by the Netherlands Organization for Scientific Research (NWO), VC EW grant 612.001.122.

Front cover: Scribe A42 (Musée du Louvre); photograph by Rama, CC BY-SA 3.0 FR, https://commons.wikimedia.org/wiki/File:Scribe-A_42-IMG_4488-gradient.jpg

Printed and bound by Ipskamp Printing.

ISBN: 978-94-6421-283-9

Adjunction in Hierarchical Phrase-Based Translation

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Universiteit van Amsterdam
op gezag van de Rector Magnificus
prof. dr. ir. K.I.J. Maex
ten overstaan van een door het College voor Promoties ingestelde commissie,
in het openbaar te verdedigen
op dinsdag 13 april 2021, te 15.00 uur

door Sophie Isabelle Arnoult
geboren te Suresnes

Promotiecommissie

Promotor:	prof. dr. K. Sima'an	Universiteit van Amsterdam
Copromotor:	dr. W.H. Zuidema	Universiteit van Amsterdam
Overige leden:	prof. dr. J. van Genabith	Universität des Saarlandes
	prof. dr. G.J.M. van Noord	Rijksuniversiteit Groningen
	prof. dr. L.W.M. Bod	Universiteit van Amsterdam
	dr. C. Monz	Universiteit van Amsterdam
	dr. W. Ferreira Aziz	Universiteit van Amsterdam

Faculteit der Natuurwetenschappen, Wiskunde en Informatica *to Ari, Jean and Caroline*

Contents

Acknowledgments

This dissertation would not have been possible without the generosity and support of my promotor, Khalil Sima'an. Khalil shaped a place for me I would never have envisioned, mentored me and helped me rise over many weaknesses in the course of my PhD. I am grateful to say the least, and proud to be able at last to follow some of his example. I am also thankful to Jelle Zuidema for his supervision. Jelle never failed to impress me with the originality and sharpness of his insights and observations, and his support in the last part of my PhD was critical.

I am grateful to Gertjan van Noord and my anonymous reviewers at SSST14 for comments that made me reflect and develop this dissertation further. I thank Wilker Ferreira Aziz, Gideon Maillette de Buy Wenniger and Miloš Stanojević for their comments and generally advancing this dissertation through useful discussions.

Chapters of this thesis have benefitted from the scrutiny and patient reading of Jasmijn Bastings, Gideon Maillette de Buy Wenniger and Miguel Rios Gaona. The cover of this book is the work of my partner, Ari de Jong, who diligently turned my idea of using a scribe into a beautiful piece of craftmanship.

The ILLC has been a wonderful environment for me, and I have been lucky to be surrounded with so many bright people. I want to thank Ivan Titov for advancing my understanding of Machine Learning; Desmond Elliot and Stella Frank for their kind guidance; Jenny Batson, Reut Tsarfaty, Lisa Beinborn, Tejaswini Deoskar and Raquel Fernández Rovira for their encouragements, and Joaquim Daiber and Markos Mylonakis for wise words on the hardships of the PhD; Miguel Rios Gaona for telling me to write.

Praise go to the ILLC office and Jenny Batson, Karine Gigengack, Tanja Kassenaar, Debbie Klaassen and Peter van Ormondt for dealing so kindly with my lack of organisation.

Thanks to Miloš, Raquel, Gideon, Phong, Jasmijn, Samira, Bas and Bryan for sharing office with me and filling it with a perfect balance of sense and nonsense, application and chatter; to Iacer, Miguel and Wilker for their hospitality and spirit; to Carlos, Amir, Bushra, Hoang, Jo, Olivier, Benno, Ehsan, Diego, Des, Stella, Marion, Tom, Lisa and Jelke for talks in all kind.

Finally, I want to thank Piek Vossen and my colleagues at the Vrije Universiteit, Antske Fokkens, Filip Ilievski, Minh Lê, Isa Maks, Roser Morante Vallejo, Marten Postma, Pia Sommerauer, Chantal van Son, Hennie van der Vliet and later members of the CLTL for welcoming me in their midst and supporting me in the last stage of my PhD.

Amsterdam Sophie Arnoult
March, 2021.

Chapter 1

Introduction

1.1 Motivation

Machine Translation is one of the most complex tasks in Natural Language Processing; intuitively, one has to deal with the complexity of not a single, but of two languages and their relationship. Reordering, which results from word-order differences between languages, constitutes a central problem in modelling this relationship. For word-based models (and phrase-based models by extension), that assume one-to-one equivalence between source and target words, a sentence of n words has $n!$ possible target reorderings. Exploring this space is infeasible for natural-language sentences that average 20 to 40 tokens in Machine Translation datasets, occasionally exceeding 100 tokens.

How do sentences grow that long? Consider this sentence, taken from the European Parliament proceedings[1]:

(1) Therefore, let me conclude by expressing my special thanks to the European parliament for its support for the Commission's efforts towards better financial management of the European union's budget.

This sentence is built around a main verb frame *let me conclude*, recursively modified to form the full sentence:

(2) Let me conclude.
Therefore, let me conclude *by expressing my thanks.*
(. . .) expressing my *special* thanks *to the parliament for its support.*
(. . .) thanks to the *European* parliament for its support *for the Commission's efforts.*
(. . .)

[1] https://www.statmt.org/europarl/

At the clause level, the verb frame is modified by the discourse marker *therefore* and the adjunct of manner *by expressing my thanks*. This phrase is modified in turn by the qualifying adjective *special* and the prepositional adjunct phrases *to the parliament* and *for its support*. Both phrases are modified in turn by more adjectives and prepositional phrases, and so on until we obtain the full sentence.

At all stages of modification, we find syntactically complete sentences. At the same time, modifiers may fill different semantic roles: while *special* only qualifies *thanks*, *European* restricts the meaning of *parliament*; and both *to the parliament* and *for its support* are semantic arguments of *thanks* [2] [3]. Most modifiers in this example contribute significantly to the meaning of the full sentence, and *let me conclude* would in fact poorly summarize it—modifiers are optional syntactically, not per se semantically. That modifiers may correspond to core arguments and or be restrictive is a fact of linguistic economy; without it, one could never thank anybody without providing a reason, and *Thank you!* would be an ill-formed expression.

1.1.1 Adjunction and recursion

Syntactic modification partakes in a general and simple mechanism to build complex sentences from simple ones: *adjunction*, as modifiers, or *adjuncts*, simply *adjoin* to the phrases they modify. Adjunction plays a large part in linguistic recursion, even if it does not account for it entirely. Recursion can in fact also apply to syntactic complements:

(3) I see that you believe that she thinks that we consider ...

In Tree-Adjoining Grammar (TAG, Joshi et al., 1975; Joshi and Schabes, 1997), recursion is accounted for by substitution on one hand, for syntactic complements, and adjunction on the other hand. Adjunction then does not only apply to syntactic modification, but also to other phenomena (Kroch and Joshi, 1985), like *wh*-fronting:

(4) *This is what* I see that you believe that she thinks that we consider []

and raising:

(5) I see that you *are inclined to* believe that she thinks that we consider ...

These phenomena, syntactic modification included, can separate syntactic dependents over long distances:

[2] Filling the roles of 'Recipient' and 'Cause' for PropBank (Palmer et al., 2005), see: `http://verbs.colorado.edu/propbank/framesets-english-aliases/thank.html`

[3] In FrameNet and Frame Semantics Fillmore (1976, 1982, 1985); Fillmore and Baker (2001), both are 'core Frame Elements' ('Addressee' and 'Reason') of the 'Judgment-direct-address' frame for 'thank': `https://framenet2.icsi.berkeley.edu/fnReports/data/frameIndex.xml?frame=Judgment_direct_address`

(6) I see *now, but it had been on my mind for a while, and the look you gave me the other day has made it completely clear to me*, that you believe (. . .)

This makes them interesting for translation and for reordering, as long-distance dependencies extend the range of reorderings that need accounting for.

1.1.2 Adjuncts in translation

How do adjuncts behave in translation? Consider the French translation of Example 1:

(7) Permettez-moi donc de conclure en remerciant tout particulièrement le parlement européen pour le soutien qu'il apporte aux efforts de la commission en vue d'une meilleure gestion financière du budget de l'union européenne.

Like its English counterpart, this sentence is built by adjunction around a main verb frame, *permettez-moi de conclure*:

(8) Permettez-moi de conclure
Allow me to conclude
"Let me conclude"

(9) Permettez-moi *donc* de conclure *en remerciant le parlement.*
Allow me therefore to conclude by thanking the Parliament.
"*Therefore,* let me conclude *by expressing my thanks to the Parliament*"

(10) (. . .) remerciant *tout particulièrement* le parlement *européen*
(. . .) thanking all particularly the Parliament European
"(. . .) expressing my *special* thanks to the *European* Parliament"

(11) (. . .) pour le soutien *qu'il apporte aux efforts de la commission*
(. . .) for the support that it brings to the efforts of the commission
"(. . .) for its support *for the Commission's efforts*"

Adjunction appears to proceed largely in parallel in the French sentence and its English counterpart. This parallelism can be explained in part by syntactic similarity in both languages. For instance, the adjuncts of manner *en remerciant le parlement* and *by expressing my thanks* are both expressed with gerunds. But the parallelism goes further than that, aligning adjuncts of different syntactic types, like the adverbial/adjectival adjunct pair *tout particulièrement*/*special*. The parallelism is not total however: *le parlement* in Example 9 is a syntactic argument of *remerciant*, whereas it is expressed as an adjunct in English; *qu'il apporte* and *de la commission* in Example 11 are adjuncts with a restrictive interpretation to *le soutien* and *aux efforts*, whereas they are expressed as determiners in the English sentence.

These cases can be attributed to different choices of expressions and structures in English and French. A classical example of this is the distinction between verb-framed and satellite-framed constructions and languages (Talmy, 1991), where motion paths are predominantly expressed through verbs or adjuncts:

(12) elle a *traversé* la Manche à la nage
she crossed the Channel swimming
"she swam *across* the Channel"

French and other Romance languages, but also Hebrew and Turkish are verb-framed as they express the path of motion through verbs, while English and other Germanic languages but also Russian or Mandarin to some extent are satellite-framed as they express the path through adjuncts (Slobin, 2004). Similar alternations let heads, adjuncts and arguments switch between languages, or conflate lexically (Dorr, 1994; Nikolaev et al., 2020).

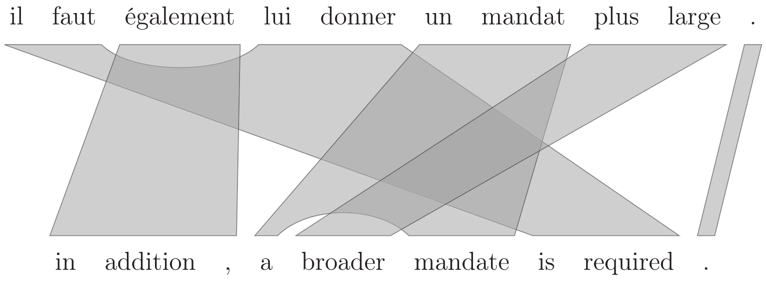
In chapter 3, I report on an analysis of adjunct alignment in French and English. This analysis shows a high degree of parallelism in adjunction between both languages in translation data. While the parallelism may be less far-reaching in different language pairs, it generally rests on the fact that adjunction provides a simple, generic syntactic operation to express secundary semantic arguments and meaning specification.

1.1.3 Compositional translation

The question of how to represent each side of a translation system, and how both adapted representations should be put in relation for translation, is a central problem of translation modelling (Yngve, 1957). In particular, models can either emphasize the *transfer* aspect of translation, or the *translation equivalence* between a sentence and its translation. This is what compositional translation models do, as they explain meaning equivalence between the different parts of a sentence and its translation through a compositional process.

In (M.T. Rosetta, 1994), Appelo and colleagues describe a compositional, rule-based Machine Translation system based on Montague Grammar. In that system, both sides of the data are described with a variant of Montague Grammar, but the translation grammar itself is also designed to be compositional, as the authors propose to transfer Frege's principle of compositionality to translation:

> *Two expressions are each other's translation if they are built up from parts which are each other's translation, by means of translation-equivalent rules.*
>
> (M.T. Rosetta, 1994, p.17)


il faut également lui donner un mandat plus large .
in addition , a broader mandate is required .

r$_0$: $S \rightarrow \langle\ X \mid X\ \rangle$

r$_1$: $X \rightarrow \langle\ X\ .\ \mid X\ .\ \rangle$

r$_2$: $X \rightarrow \langle$ il faut X^1 lui donner X^2 $\mid$ X^1 X^2 is required $\rangle$

r$_3$: $X \rightarrow \langle$ également $\mid$ in addition , $\rangle$

r$_4$: $X \rightarrow \langle$ un mandat X $\mid$ a X mandate $\rangle$

r$_5$: $X \rightarrow \langle$ plus large $\mid$ broader $\rangle$

Figure 1.1: Example rule decomposition with Hiero.

Inventorizing translation-equivalent rules to adequately cover translation phenomena and linguistic situations forms a major challenge for such a system.

In Statistical Machine Translation (SMT), syntax-based models like GHKM (Galley et al., 2004, 2006) and hierarchical phrase-based models like Hiero (Chiang, 2005) are compositional from the onset, as they are built on syntax-directed-transduction grammars and Synchronous Context-Free Grammar (SCFG, Lewis and Stearns, 1968; Aho and Ullman, 1972). Such grammars explain sentence pairs by decomposition in translation-equivalent fragments through transduction rules like those shown in Figure 1.1. These grammars are compositional in the sense of the Rosetta principle, as each translation rule puts translation-equivalent parts in relation, and as rule application preserves translation equivalence.

These models are robust compared to rule-based models, as they are data-based and thus capture translation equivalences that would be hard to predict. However, the relation to monolingual compositionality is generally lost: syntax-based models generally apply syntax to only one side of the data, whereas Hiero grammars are asyntactic on both sides. In Hiero, SCFG translation rules are acquired on word-aligned training data, and are subsequently applied on the source side, producing candidate translations that may or may not form grammatical, meaningful sentences. Linguistically motivated adaptations of the model like

Syntax-Augmented Machine Translation (Zollmann and Venugopal, 2006), presented in Chapter 2, allow to guide rule selection, but at a high computational cost. Besides, syntactic refinement of SCFG nonterminals introduces rule sparsity, and constraints that diminish the generality of the model. It is not a coincidence either that syntax would in general be used on only one side in Syntax-Based SMT: syntax imposes in fact constraints that are reminiscent of RBMT, resulting in lesser coverage of possible translation-equivalent expressions. For instance, the equivalence between *tout particulièrement* and *special* in Example 10 might be overlooked by syntax-based models.

In current-day Neural Machine Translation models (Sutskever et al., 2014; Bahdanau et al., 2015; Vaswani et al., 2017), the relationship between source sentences and their translations becomes hidden, while the need to explicitly state their reordering disappears. The self-learning abilities of neural networks have opened the way to the syntactic interpretation of encoder representations (Shi et al., 2016; Tang et al., 2018; Chang and Rafferty, 2020). In relation to the linguistic motivations of TAG for instance, Wilcox et al. (2018) show that RNN language models can learn to identify *wh*-filler-gap dependencies and some of the syntactic-island constraints associated with them. Meanwhile, the interest for guiding data-based translation with syntax is undiminished (Sennrich and Haddow, 2016; Bastings et al., 2017; Chen et al., 2017; Li et al., 2017; Currey and Heafield, 2019; Bugliarello and Okazaki, 2020), and word alignments have been proposed to improve neural attention (Cohn et al., 2016; Alkhouli and Ney, 2017; Zhang et al., 2017). So while this work deals with Statistical MT rather than Neural MT, we trust that the more linguistically oriented results presented here will still be relevant for neural models.

1.2 Objective

The main objective of this thesis is to investigate how modelling adjunction contributes to compositional, phrase-based SMT:

> How does adjunction, an elementary syntactic recursion tool supporting semantic relevance, fit in asyntactic, compositional models of translation?

Concretely, this work aims at utilizing adjunction as a source of linguistic information to improve hierarchical phrase-based SMT models. Adjunction provides a different, more coarse sort of information than the syntactic labels which have been generally used for enriching Hiero models, so one can ask *how* to use adjunction in these models. This work primarily looks at adjuncts as a source of information for guiding recursion, but as part of this, we also look at the effect of modelling adjunct optionality and of adjunct labelling in Hiero. The utility of adjunction for translation modelling in Hiero is investigated in Chapter 4. In

Chapter 5, we utilize adjuncts to improve reordering models for phrase-based preordering.

More generally, the objective of this thesis is to contribute an answer to the translation-modelling problem outlined earlier in section 1.1.3: *what kind of linguistic knowledge, or representation level should be encoded in translation models?*

In SMT, phrase-based models (Koehn et al., 2003; Chiang, 2005) on one hand, and syntax-based models (Yamada and Knight, 2001; Galley et al., 2004) on the other hand, stand against each other in their approach to syntax and non-syntactic phrases, but generally agree that the truth is in the middle. As we will see in Chapter 2, Phrase-based SMT has been looking at linguistic information as a means to guide models, while Syntax-based SMT has had to surmount inherent syntactic constraints. This work inscribes itself in research to guide phrase-based models with syntax, with an interest in the respective values of syntactic constraints and asyntactic phrases.

Within phrase-based SMT, phrase-based models have superseded early word-based approaches (Brown et al., 1993), and hierarchical models like Hiero (Chiang, 2005) have shown useful for reordering-intensive language pairs. In this perspective, this work aims at answering whether modelling adjunction provides benefits over generic recursion. To this end, experiments in this work compare the effects of adjuncts to those of arguments on one hand, and constituents on the other hand.

1.3 Contribution

The main contribution of this thesis is the application of adjunction to hierarchical phrase-based models. Adjunction has been used before in SMT, in syntax-based models (Nesson et al., 2006; DeNeefe and Knight, 2009; Liu et al., 2011). Syntax-based models lend themselves naturally to the *formal* application of adjunction as in Synchronous TAG (Shieber and Schabes, 1990), as they perform a transduction between the syntactic parse tree of a sentence and the surface form of its translation or between two syntactic trees. Distinguishing adjunction from substitution rules allows these models to generalize over syntactic modification and to learn more compact grammars. Applying adjunction in hierarchical phrase-based models may seem counterintuitive in this respect: hierarchical phrase-based models perform a transduction between surface forms through an SCFG extracted from word-aligned data, leaving little room for a formal application of adjunction. But we are considering here adjunction from a more general perspective—that also motivates STAG in fact (Shieber and Schabes, 1990; Shieber, 2007)—of a syntactic recursion operation with a potential to apply *synchronously*, in parallel across a bilingual bitext.

The question of the degree to which adjunction applies synchronously in translation data has received little attention, despite its relevance for STAG. This question is addressed in Chapter 3, where we measure to what degree adjuncts align to adjuncts through hierarchical word alignments. This chapter contributes a corpus study of adjunct equivalence in translation data, as well as an evaluation of the effect of experimental conditions on adjunct equivalence. Besides, we compare the translation equivalence of adjuncts to that of arguments, to determine whether adjuncts and arguments behave differently in terms of translation equivalence. Our corpus study shows that both adjuncts and arguments are synchronous to a high degree in French-English, suggesting that it is semantic relevance in the broad sense that favors synchronous behaviour, while adjuncts and arguments do tend to preserve their role in translation. Comparing empirical measures of synchronous adjunction and complementization, we find that synchronous adjunction is indicative of translation compositionality, and synchronous complementization primarily of syntactic similarity.

Chapter 4 tackles the contribution of adjunction to translation modelling in Hierarchical Phrase-based SMT. The contribution of this chapter is threefold:

- we propose an extension to Hiero that leverages on adjunction in two ways: first by exploiting the long-distance dependencies introduced by adjuncts to relax phrase-span extraction constraints; and secondly by exploiting adjunct optionality to extract rules by factoring out adjuncts. Our first extension gives beneficial results for English-Chinese and English-Japanese, showing the utility of long-range rules, also for English-Chinese. The second extension provides promising results for English-Japanese.

- we analyze the relative contributions of span-length and adjunct-crossing constraints in our first extension. This analysis shows that the performance of the model actually rests on adjunct-constrained rules, which efficiently filter useless short-range rules.

- we compare adjunct-based constraints to constraints based on arguments or constituents. This analysis shows that argument-based constraints tend to be as beneficial as adjunct-based constraints, while constituent-based constraints perform best. In other words, it is a closer adherance to linguistic recursion through constituency(-crossing) constraints that benefits Hiero most.

Finally, we look at the role that adjunction plays in reordering in Chapter 5. In this work, we use adjuncts to refine the latent reordering grammar of Stanojević and Sima'an (2015), to be used for preordering reordering-intensive language pairs. We show that adjuncts are informative for reordering in English-Japanese and English-Chinese, providing large gains over fully latent reordering for English-Japanese. When comparing adjuncts to arguments and constituents, we find that constituency benefits English-Japanese the most, while adjuncts carry most of the improvement in performance. Further refinements of the reordering grammars show that adjuncts also differ from arguments in reordering, as the information they bring appears to be local to reordering rules.

1.4 Outlook

Chapter 2 introduces SMT, starting with word-alignment models. We then present phrase-based models, their hierarchical variant Hiero and linguistic enrichment in Hiero. We end with a short presentation of syntax-based models and of applications of adjunction and STAG in Syntax-Based SMT.

Chapter 3 deals with the following question: *to what extent do adjuncts translate as adjuncts, and how do adjuncts compare to arguments in this respect?* We are interested in answering this question both in general, and in experimental conditions, where adjuncts are identified from syntactic parses, and translation equivalence is established through word alignments. We present first a corpus study of adjunct alignment in a manually annotated French-English subset of the Europarl corpus, then show how automatic tools (word alignments, parsing) affect adjunct alignment in experimental conditions. We end with measures of adjunct alignment in experimental conditions.
The work in this chapter is based on the following publication:

- Sophie Arnoult and Khalil Sima'an. *How synchronous are adjuncts in translation data?*. In SSST 2014.

Chapter 4 addresses the following question: *how does adjunction fit in the compositional, phrase-based model provided by Hiero, and does adjunction constitute a useful guide for this model?* We start by extending the Hiero grammar with long-range rules, selected to respect adjunct boundaries. We then present a second extension, leveraging on adjunct optionality to extract more rules by excising adjuncts. In the rest of the chapter, we further analyze the first extension, to evaluate the respective contributions of asyntactic, short-range Hiero rules and of the adjunct-based, long-range rules added by our model. We end by comparing adjunct-based constraints to constraints based on arguments or constituents.
The work in this chapter is based on the following publications:

- Sophie Arnoult and Khalil Sima'an. *Modelling the Adjunct/Argument distinction in Hierarchical Phrase-Based SMT*. In DMTW 2015.

- Sophie Arnoult and Khalil Sima'an. *Factoring Adjunction in Hierarchical Phrase-Based SMT*. In DMTW 2016.

Chapter 5 focuses on the reordering properties of adjunction for Machine Translation: *are adjuncts informative of translation reordering?* We present an extension to the latent PCFG reordering grammar of Stanojević and Sima'an (2015), where adjuncts are used to split reordering nonterminals prior to latent splitting. The grammar can then be used to preorder source sentences for a phrase-based system. We present experiments with this grammar on English-Chinese, English-Japanese, English-German and German-English, showing large gains for English-Japanese. We complement this work with label-refinement experiments, using either hard or soft constraints, and with a comparison of adjuncts to arguments and constituents in reordering.

Chapter 2

Background: Statistical Machine Translation and Adjunction

Statistical Machine Translation (SMT, Brown et al., 1988, 1990, 1993) is grounded in the double idea that translation can be seen as a decoding process, and that translation equivalence can be specified at the word level. Section 2.1 introduces the translation model of Brown et al. (1988, 1990). This model is based on a *noisy-channel* approach: sentences to be translated are seen as the output of a noisy channel, and translating amounts to decoding original sentences.

The translation model is parametrized at the word level by a series of alignment models, the *IBM models* (Brown et al., 1993), introduced in section 2.2. Each successive model builds on the preceding one to account for lexical collocations and reordering. Alignment is unidirectional however, and the translation model cannot account for interdependencies on the target translation side.

The SMT models that developed thereafter build upon the word alignments of Brown et al. (1993), but capture translation equivalence beyond the word level, and provide richer reordering models.

Phrase-Based models capture translation equivalence at the phrase level, by extracting aligned phrase pairs from symmetrized word alignments. Phrase-Based SMT (Koehn et al., 2003), presented in section 2.3, assembles phrase pairs sequentially at decoding, using a reordering model to score target reorderings. Hierarchical Phrase-Based SMT (Hiero, Chiang, 2005, 2007), presented in section 2.4, provides a compositional view of translation, by recasting Phrase-Based models as Synchronous Context-Free Grammars. Reordering is then directly modelled by the translation grammar. Hiero grammars are too generic however. Section 2.5 presents approaches to enrich Hiero with linguistic information.

Syntax-Based models (Yamada and Knight, 2001; Galley et al., 2004) base translation equivalence on the alignment of a sentence string to a syntactic parse of its translation. The syntactic structure on either side directly provides a compositional model for translation equivalence and reordering. Syntax-Based models are characterized as *tree-to-string*, *string-to-tree* or *tree-to-tree*, depending on the syntactic nature of the source sentence and its translation.

Section 2.6 introduces the GHKM model of rule extraction (Galley et al., 2004), and its string-to-tree instantiation by Galley et al. (2006). The linguistic constraints imposed on Syntax-Based models limit their coverage. One way of improving coverage consists in factoring out adjunction, as proposed in Tree-Adjoining Grammar (TAG, Joshi et al., 1975).

Section 2.7 introduces TAG, its synchronous variant STAG (Shieber and Schabes, 1990), and their applications to SMT by Nesson et al. (2006), DeNeefe and Knight (2009) and Liu et al. (2011).

2.1 Statistical Machine Translation

Statistical Machine Translation (SMT; Brown et al., 1993) followed on Example-Based MT (EBMT; Nagao, 1984) as a fully data-based approach to Machine Translation. In contrast to EBMT, SMT provides a word-level decomposition of parallel data, coupled with a statistical decision mechanism for candidate translations.

2.1.1 The Noisy-Channel approach

Brown et al. (1990) propose to model translation as a generative, noisy-channel process, where sentences to translate are seen as the encodings of source sentences. Translating then amounts to decoding sentences into their 'source' sentences. Given a sentence f of language F, decoding aims at finding the sentence $\hat{e}$ of goal language E that maximizes their joint probability:

$$\hat{e} = \arg\max_{e \in \mathcal{E}} P(e, f) = \arg\max_{e \in \mathcal{E}} P(f|e) \cdot P(e) \tag{2.1}$$

where $P(e)$ is the probability assigned to e by a language model and $P(f|e)$ is the translation-model probability. The translation space $\mathcal{E}$ is determined in practice by the translation model's learned translation units and their compositionality (through the sequential or hierarchical nature of the model).

Following (Och et al., 1999; Och and Ney, 2002), the generative formulation of the translation problem gave way to discriminative models. These models allow for more control over the output and the translation process, while still combining a language and a translation model; whereas the translation model ensures that translations are adequate, the language model ensures their fluency.

2.1.2 Language model

The language model generally assumed in SMT is n-gram language model, where the probability of a sentence is decomposed into word probabilities conditioned on a limited history:

$$P(e = w_1^n) = \prod_{i=1}^{n} P(w_i | w_{i-k}^{i-1}) \tag{2.2}$$

where k is the order of the language model. Values of three to five are typical, depending on the amount of training data. Unseen and rare events are accommodated through smoothing (Chen and Goodman, 1998), and in particular modified Kneser-Ney smoothing (Kneser and Ney, 1995).

2.1.3 Translation model

The translation model aims at explaining how sentences in E generate sentences in F while allowing for efficient estimation. This requires a decomposition of the sentence pair $\langle e, f \rangle$ into smaller units of translation, the nature of which depends on the model.

While the IBM models of Brown et al. (1990, 1993) presented next propose a word-pair decomposition, Phrase-Based SMT (Koehn et al., 2003), presented in section 2.3, regroups word alignments to perform a phrase-pair decomposition. Hierarchical Phrase-Based SMT (Chiang, 2005), presented in section 2.4, performs the same phrase-pair decomposition as Phrase-Based models, but provides a hierarchical composition mechanism over them. In Syntax-Based SMT (Yamada and Knight, 2001; Galley et al., 2004), presented in section 2.6, decomposition is constrained by syntax, typically on one side of the data only, leading to a tree-to-string or string-to-tree decomposition as in the GHKM model (Galley et al., 2004, 2006).

While early generative models proposed alternatives to the word-alignment IBM models (Yamada and Knight, 2001; Marcu and Wong, 2002 inter alia), most subsequent models have built upon lexical alignment through word-alignment models.

2.2 Word alignments

Word alignment models allow to decompose translation data at the word level. In the IBM models of Brown et al. (1993) presented in section 2.2.1, word alignments are treated as hidden variables in a generative model: English words are seen as generating French words following some hidden word alignment. The resulting alignments are unsatisfactory for modelling translation equivalence: they are unidirectional, allowing to align several English source words to a same French word, but not the other way around; untranslated French words can be explained

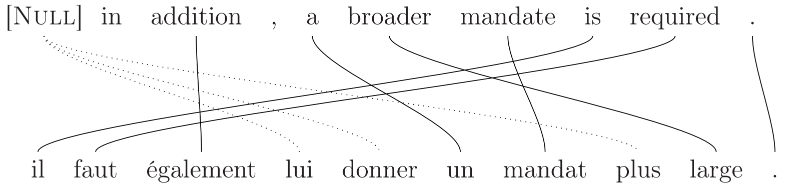


Figure 2.1: Example of English/French word alignment. Each French word is generated either by an English word or by the NULL token.

by a *null* English word, but tend to be aligned to non-equivalent uncommon words in practice. These models are also unsatisfactory for translation proper, as they provide no means to recover untranslated English words. Symmetrization (Och et al., 1999; Och and Ney, 2003) and symmetrized models (Liang et al., 2006), presented in section 2.2.2, provide multiple word equivalence in both directions while reducing noisy alignments. They form the basis for phrase-based and syntax-based translation models.

2.2.1 The IBM models

Brown et al. (1993) propose a series of generative models over words to model the relationship between an English source sentence $\mathbf{e}$ and its French translation $\mathbf{f}$. These models assume that French words are generated from words in the source sentence, a NULL token being reserved for French words without an English equivalent. As Figure 2.1 shows, each French word is aligned once and only once, while English words may remain unaligned or align to distinct French words. The word alignments between source and target words form hidden parameters, that are refined by each successive model. Model parameters are estimated with the EM algorithm (Dempster et al., 1977).

IBM1 and IBM2

The first two models (IBM1 and IBM2) assume that French words are generated independently of each other, allowing for exact parameter estimation. Both models follow a generative story in three steps, given a source sentence $\mathbf{e}$:

1. select a length m for the French sentence
2. for each position j in the French sentence, select an English word at position a_j to generate the aligned French word
3. for each j, and aligned position a_j, select a French word f_j

Both models make the same simplifying assumptions:

- sentence length m is generated with a uniform probability ϵ, assume some finite maximum for m;
- lexical translation is conditioned on the English word e_{a_j} a French word is aligned to:
$$P(f_j|a_1^j, f_1^{j-1}, \mathbf{e}) = t(f_j|e_{a_j})$$

The models differ in their assumptions for the alignment model: Model 1 assumes a uniform probability distribution on aligned positions, while Model 2 conditions aligned positions on the position in the French sentence and on the lengths of the French and English sentences.

Consequently, both models are parametrized by lexical translation probabilities $t(f|e)$, and Model 2 is additionally parametrized by alignment probabilities $a(i|j, l, m)$, where i and j stand for English and French positions, and l and m stand for English and French sentence lengths. In training, the lexical translation probabilities are first estimated under Model 1. They are then used to initialize parameter estimation for Model 2.

The HMM model of word alignment

The alignment parameter of Model 2 allows to learn alignment distributions for French words occurring in different sentence positions. In English-to-French alignments, the model would learn that French positions tend to align to similar positions in English; in English-to-German alignments, the model would reflect differences in verb position, but word order freedom in German is also likely to result in flatter distributions. In this context, the independence assumptions between the generation of French words are unfortunate. In the example of Figure 2.1, the alignment estimates for the French-English positions 1-7 and 2-8 (the positions of the word pairs *il/is* and *faut/required*) are likely to be low, but the independence assumptions further hide the relationship between these alignments.

Consequently, Model 2 is generally replaced by the HMM model of Vogel et al. (1996). They model word alignments as a Markov process, with aligned English positions a_j as hidden states. Aligned positions have then a 1-order dependence on the aligned position of the precedent French word, resulting in smoother alignments.

The HMM is then parametrized by transition probabilities between aligned positions, and by lexical translation probabilities for its emissions. Like Model 2, the HMM model is initialized with lexical translation probabilities estimated with Model 1.

Models 3 and 4

Model 3 accounts for one-to-many alignments between the English and French sides, by means of a *fertility* parameter ϕ_e, which controls how many words each

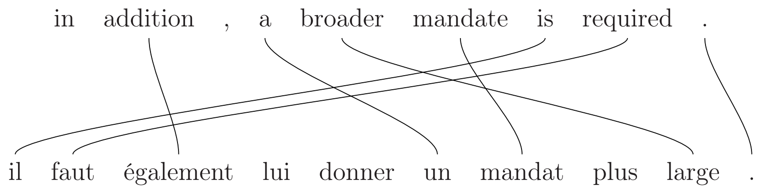

in addition , a broader mandate is required .
il faut également lui donner un mandat plus large .

Figure 2.2: Example of symmetric English/French word alignment. Unaligned words result from alignments to null or disagreement during symmetrization.

English word generates. The generative story followed by Model 3 now consists in selecting a fertility for each English word; selecting French words to associate to each English word; and selecting positions for these French words.

Model 4 accounts for phrasal movement by reparametrizing French positions, conditioning them on word classes of the aligned French word and of the previously aligned English word.

The dependence between French words introduced by the fertility parameter prevents exact estimation, and parameters for Model 3 and Model 4 are thus estimated using hard EM: the model is initialized with Model 2 Viterbi alignments, and only the space of neighbouring alignments is explored to collect expected counts for the parameters.

2.2.2 Symmetrization

Notwithstanding the relaxed independence assumptions of the later IBM models, word alignments remain unidirectional, and as such unable to capture many-to-many alignments. Besides, unaligned French words are not always handled appropriately in these models; uncommon unaligned words in particular tend to be misaligned, a phenomenon referred to as *garbage collection* (Liang et al., 2006).

Och et al. (1999) propose to derive symmetric word alignments by training word alignments in both translation directions, and selecting alignment links in the union of both alignments. As word alignments only provide one-to-many alignments, taking the intersection of both word alignments would block the capture of many-to-many alignments. On the other hand, the union of both alignments would retain noisy alignments (Moore, 2004) in both alignments. To reach a middle ground, Och et al. propose an iterative procedure that starts from the intersection of word alignments, and includes alignment links from the union that can be reached by adjacency from links in the intersection.

Alternatively, Liang et al. (2006) propose to train IBM1 and HMM word alignments in both directions jointly, by constraining both models to agree on alignment links. The resulting alignments tend to be bijective and monotonic, while being less sparse than intersected alignments. The agreement constraint effec-

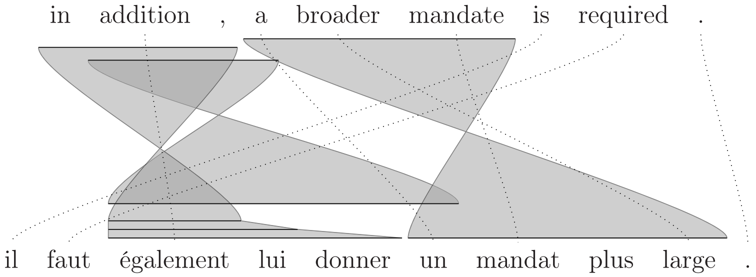


Figure 2.3: Word-aligned sentence pair with extractable phrases. Only phrase pairs with a length of three tokens on the English side are represented here. Only phrases containing alignment links can be extracted, and a length limit is imposed in practice.

tively filters out the errors made by a single model, allowing the corresponding words to be better aligned, while the HMM models also disfavors gaps in the alignments.

2.3 Phrase-Based SMT

Phrase-Based SMT (PBMT, Koehn et al., 2003) builds upon the IBM models by using word alignments to capture phrasal translations. The model closely succeeded to the Alignment Template approach (Och et al., 1999; Och and Ney, 2002), taking over most of its methods while moving from *abstract* collocational units of translation ('alignment templates') to *lexical* fragments ('phrase pairs'). Besides changing the units of translation from words to phrases, PBMT differs from the IBM models by using a discriminative model, allowing for the incorporation of features beyond the translation model proper and the language model.

2.3.1 From words to phrases

Symmetrized word alignments provide the basis for the extraction of phrase pairs in Phrase-Based SMT. The model extracts phrasal equivalents that are *consistent* with word alignments: phrase pairs must contain at least one alignment link, and words on either side of the phrase pair must be either unaligned or aligned to and only to words on the other side. Figure 2.3 represents some of the phrase pairs that can be extracted based on the symmetrized alignment of Figure 2.2.

2.3.2 Model

The model is defined as a log-linear model over features h_i:

$$\log P(\mathbf{e}|\mathbf{f},\mathbf{s}) = \sum_i w_i \log h_i(\mathbf{e},\mathbf{f},\mathbf{s}) \tag{2.3}$$

where $\mathbf{s}$ stands for a segmentation of the sentence pair ($\mathbf{e}$,$\mathbf{f}$) into phrase pairs.

Segmentation into phrase pairs is not modelled explicitely in Phrase-Based SMT, the segmentation leading to the best model score being selected at decoding.

The model uses the following features:

- **translation model (TM)**: these consist of phrase-translation and lexical weights in both translation directions. Phrase-translation features provide estimates for the translation of full phrases, while lexical weights use the underlying word alignments to provide word-based translation scores for the words in a phrase. Feature values for each phrase pair are computed during training by taking relative-frequency estimates.
- **language model (LM)**: this is typically an n-gram language model scored at the sentence level.
- **word penalty (WP)**: this feature controls the length of output translations.
- **phrase penalty**: this feature controls the number of phrase pairs for segmenting sentence pairs.
- **unknown word penalty**: unknown words are copied over at decoding. This feature allows to control segmentation in the case of rare words—as rare words may find a translation when they are part of phrase pairs but not in isolation.
- **distortion**: this feature controls how much reordering may take place at decoding. Distortion values record the distance on the source side between adjacent phrases on the target side.

Feature weights are optimized against the BLEU score (Papineni et al., 2002) through Minimum Error Rate Training (MERT, Och, 2003).

2.3.3 Decoding

Decoding aims at finding the best translation $\hat{e}$ for a source sentence f. Decoding proceeds by building translations from left to right, allowing for language model scoring. Candidate translations are built sequentially, until all tokens in the source sentence have been covered.

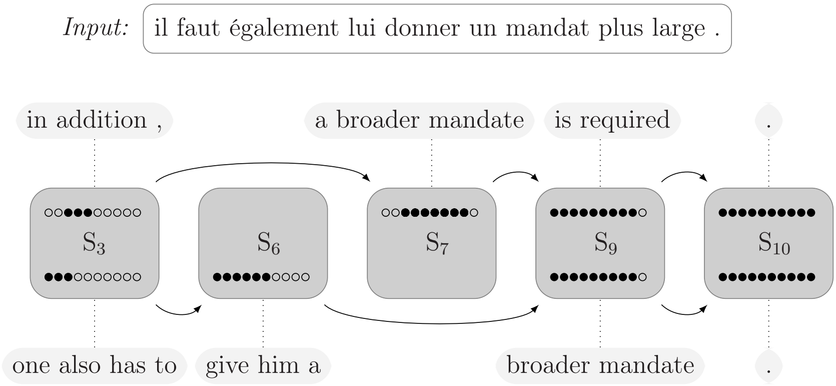


Figure 2.4: Two PBMT candidate translations with decoding states. The stacks S_i keep hypotheses covering i input words. The hypotheses shown in stacks S_9 and S_{10} cover the same input words, but differ by their last translated source words (for S_9) and target words.

Decoding Phrase-Based models is NP complete (Knight, 1999): on one hand, the exponential complexity of translation reorderings make decoding a Travelling Salesman problem, and on the other hand, the combinatorial complexity of searching for the best input sentence segmentation make decoding a Set Covering problem.

Phrase-Based models consequently use approximate inference through beam search Koehn et al. (2003). Beam search keeps distinct stacks to group hypotheses that cover a same number of source words. The number of hypotheses in each stack is limited by hypothesis recombination and pruning. Hypothesis recombination looks for hypotheses with a same search state (which regroups the information need to compute further model costs: covered input words, last covered source words and last translated words) and discards all but the best hypothesis. Pruning is based on a combination of hypothesis model cost and a future cost estimate. This estimate approximates model cost by parameters that can be precomputed, namely phrase-pair translation probability estimates and language model estimates for candidate phrase pairs.

Figure 2.4 illustrates decoding for the input, French sentence with two candidate translations: our example English sentence, and a more literal translation. Producing the first translation incurs a high reordering cost for the model, because of the token distance between the phrases *il faut* and *un mandat plus large* on one hand, and *un mandat plus large* and the period '.'.

2.3.4 Reordering

Reordering in PBMT is addressed by two models, a distortion model and a lexical orientation model.

The distortion model simply counts the number of tokens between the source sides of succeeding phrase pairs in candidate translations at decoding. The resulting feature captures how much reordering can be expected for a given language pair and data, but cannot distinguish good from bad reorderings; ultimately, reordering decisions are sanctionned by the language model, which itself has a limited scope.

To improve reordering decisions for PBMT, Tillmann (2004) and Axelrod et al. (2005) have proposed lexical orientation models: at training, phrase-pair orientation counts are collected to estimate how often phrase pairs either follow directly on preceding phrase pairs; are swapped on the source side with regard to preceding phrase pair; or are discontinuous with regard to the preceding phrase pair.

The lexical orientation model allows PBMT to model reordering preferences of phrase pairs based on their lexical content. Besides, these reorderings are local as they only consider neighbouring phrases. Hierarchical Phrase-Based SMT (Chiang, 2005) learns hierarchical rules that capture the lexical context of reorderings, allowing the model to perform better on complex, mid-range reorderings.

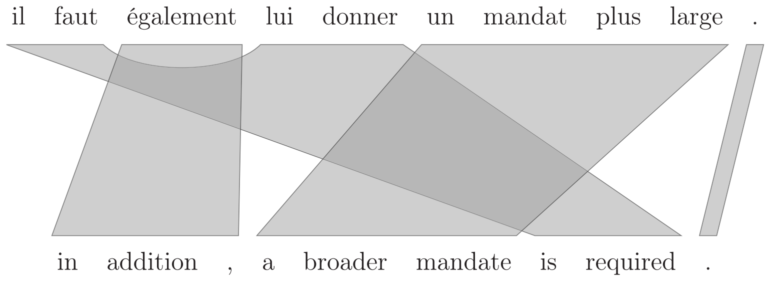
2.4 Hierarchical Phrase-Based SMT

Phrase-Based models capture idiomatic, phrasal translations, but they are limited to *continuous* translation-equivalent phrase pairs. In our example sentence pair, PBMT cannot capture the equivalence between *if faut ... lui donner* and *is required*, and would associate the unaligned phrase *lui donner* to either *également* or *un mandat*. Neither option is satisfying from a compositional point of view.

Hierarchical Phrase-Based SMT (HPBMT, or Hiero, Chiang, 2005) allows to capture discontinuous translation equivalences by casting Phrase-Based SMT into Synchronous Context-Free Grammar (SCFG). Like PBMT, Hiero extracts phrase pairs from the training data, but it then derives SCFG rules by abstracting phrase pairs embedded in larger fragments.

2.4.1 Rule extraction

Like Phrase-Based SMT, Hiero extracts phrase pairs from parallel data, but further derives SCFG rules from them. These consist of *lexical* rules on one hand, which directly rewrite to a phrase pair, and *hierarchical* rules on the other hand, which are obtained by excising embedded phrase pairs and which rewrite to a


il faut également lui donner un mandat plus large .
in addition , a broader mandate is required .

r$_0$: $S \rightarrow \langle\ X \mid X\ \rangle$

r$_1$: $X \rightarrow \langle\ X\ .\ \mid X\ .\ \rangle$

r$_2$: $X \rightarrow \langle$ il faut X^1 lui donner X^2 | X^1 X^2 is required $\rangle$

r$_3$: $X \rightarrow \langle$ également | in addition , $\rangle$

r$_4$: $X \rightarrow \langle$ un mandat plus large | a broader mandate $\rangle$

Figure 2.5: Example sentence pair with a possible phrasal decomposition and Hiero rules. The sentence pair can be derived by successive applications of the glue rule r_0, the hierarchical rules r_1 and r_2 and the lexical rules r_3 and r_4.

string of source/target tokens and synchronous nonterminals, as shown in Figure 2.5. The grammar is completed by abstract *glue* rules for sentence-level rewritings.

The model uses only two nonterminal symbols, S for the top symbol, and a single symbol X for hierarchical, word-alignment driven rewritings. As rule r_2 in Figure 2.5 shows, Hiero rules capture discontinous translation equivalences, but also local reorderings. This allows Hiero to perform better than Phrase-Based SMT in language pairs that exhibit complex reordering patterns, like Chinese-English.

2.4.2 Features

Notwithstanding the reordering model, Hiero uses similar features as Phrase-Based SMT. Derivations are scored in a log-linear model over a language-model for the target side of the derivation and translation features.

$$\log P(\mathbf{d}|\mathbf{f}) = w_{LM} \log P_{LM}(\mathbf{e}(\mathbf{d})) + \sum_i w_i \sum_{r \in d} \log P_{TM_i}(r) \tag{2.4}$$

Hiero uses essentially the same translation features as PBMT, owing to the lexical character of rules. Translation features (translation probability estimates conditioned on source and target, as well as lexical weighting features) and the unknown-word penalty features carry over to Hiero, as does the phrase penalty, which becomes a rule-application penalty.

2.4.3 Decoding

Decoding SCFG grammars like Hiero involves parsing the source sentence while searching for the best or n-best synchronous derivations.

Parsing is based on the CYK+ (Chappelier and Rajman, 1998) algorithm, an Early-parsing (Earley, 1970) variant of the Cocke-Younger-Kasami (CYK) algorithm for parsing non-binary input. To combine and score hypotheses in the parsing chart with the language model requires in principle splitting parsing states by target-side n-grams. This contributes a high-order polynomial function of the target-language vocabulary size to decoding complexity; for a n^{th}-order language model and a rank-2 grammar, one needs to store $(n-1)$-grams on each side of two nonterminals to compute the language-model term exactly. The cube-pruning algorithm (Chiang, 2007) addresses this problem by approximating the language-model score of partial hypotheses with unigram scores. The algorithm further conducts beam search over parsing states, by deriving the best partial hypotheses from lower states in the parsing chart and pruning lower-scoring hypothesis.

2.5 Linguistic enrichment in Hierarchical Phrase-Based SMT

The Hiero grammar is too generic. In a CFG for monolingual parsing, syntactic nonterminal labels control rewritings and limit overgeneration. These labels are chosen to be linguistically adequate (Chomsky, 1957), and can further be refined for statistical parsing (Klein and Manning, 2003; Petrov et al., 2006). In a SCFG like Hiero's, rewritings are based on word alignments and do not reflect linguistic syntax. The asyntactic character of Hiero is reflected in the use of a single rewriting label X. The target-side language model consequently plays a large role in validating candidate translations.

Besides, as the grammar is driven by word alignments, rewritings are not lead by syntactical government as they would in syntax. The phrase pair ⟨ *un mandat plus large* | *a broader mandate* ⟩ can be equally parsed with any of the following rules, among others:

(13) $X \rightarrow \langle$ un mandat plus large | a broader mandate $\rangle$
$X \rightarrow \langle$ X mandat plus large | X broader mandate $\rangle$; $X \rightarrow \langle$ un | a $\rangle$
$X \rightarrow \langle$ un mandat X | a X mandate $\rangle$; $X \rightarrow \langle$ plus large | broader $\rangle$
$X \rightarrow \langle$ un X plus large | a broader X $\rangle$; $X \rightarrow \langle$ mandat | mandate $\rangle$

Table 2.1: SAMT labels are assigned to phrase spans depending on their syntactic nature: full constituent of type Y; an incomplete constituent missing a constituent of type Z to form a constituent of type Y, either to the right (Y/Z) or to the left ($Y\backslash Z$); a pair of constituents $Y + Z$. Remaining phrase spans receive a default label 'X'.

label type	label instance	examples
Y	ADV	in addition
	NN	mandate
Y/Z	NP/NN	a broader
Y\Z	NP/DT	broader mandate
Y+Z	ADV+,	in addition ,
'*X*'	*(default)*	addition , a broader

This *spurious ambiguity* (Chiang, 2005) generates many alternative derivations for a same translation. This is palliated by constraints on the form of rules: phrases (for rule extraction) may not be bordered by unaligned words; nonterminals may not be adjacent on the target side; and the (token) scope of the top nonterminal S is limited heuristically.

These constraints limit the expressivity of the grammar, and its ability to model long-range reorderings. Syntax-based refinement of nonterminals allows to limit overgeneration, and to relax rule constraints and/or to refine long-range reordering rules.

2.5.1 Syntax-Augmented Machine Translation

Zollmann and Venugopal (2006) proposed to use a syntax-based label set inspired from Combinatorial Categorical Grammar (Steedman, 2000) to refine Hiero nonterminals. Their model, Syntax-Augmented Machine Translation (SAMT) formed the starting point for the utilisation of different types of syntactic information for the enrichment of hierarchical PBSMT models.

Applying syntax in Hiero requires a way of labelling phrases and phrase pairs that do not in principle have a clear syntactic status. SAMT (Zollmann and Venugopal, 2006) assigns CCG-like labels to phrase pairs based on phrase-structure parses of the target side of the data. Given a target parse, every span of the target string is assigned a phrasal label as given in Table 2.1.

Going back to the example sentences of Figure 2.4, SAMT labels allow to differentiate between the two translations of *également*, the sentence-initial *in addition ,* and the pre-verbal *also.*

(14) $X \rightarrow \langle$ il faut ADV+, lui donner NP | ADV+, NP is required $\rangle$
$ADV+, \rightarrow \langle$ également | in addition , $\rangle$

$X \rightarrow \langle$ il faut ADV lui donner NP | one ADV has to give him NP $\rangle$
$ADV \rightarrow \langle$ également | also $\rangle$

The distinction prevents the generation of the infelicitous translations:

(15) * also a broader mandate is required
* one in addition , has to give him a broader mandate

2.5.2 Source-side disambiguation

Phrasal labels can also be derived from source-side syntax, and follow a different labelling scheme. For instance, Li et al. (2012) propose to use source-side syntax for Chinese-English, concatenating the POS tags of independent syntactic heads in each source phrase. Applying source-side syntax allows to refine the selection of target translations. This allows for instance to distinguish between two French constructions employing *il faut*:

(16) $X \rightarrow \langle$ il faut NP | NP is required $\rangle$
$X \rightarrow \langle$ il faut VP | one has to VP $\rangle$

Li et al. (2012) further extend Hiero with abstract, nonterminal reordering rules. These model the rewriting of syntactically-labelled nonterminals to pairs of monotonically aligned or inverted nonterminals.

2.5.3 Limitations of syntactic label refinement

Phrase-labelling schemes produce rich nonterminal vocabularies. This adds to the complexity of decoding, but also induces data sparsity for translation probability estimates. Hanneman and Lavie (2013) coarsen for instance the label set by merging target labels with similar distributions over source-side labels.

Besides, syntactic phrase labels impose hard rewriting constraints, which harm translation as they become more fragmented. Soft constraint features can replace hard labels: Marton and Resnik (2008) use features to mark whether phrases match or cross the boundaries of a core set of syntactic labels; Chiang (2010) applies soft tree-matching features for both the source and target sides. Huang et al. (2010) match phrases against part-of-speech sequences to induce a distribution over latent label vectors, and use similarity between vectors as a decoding feature.

Other grammatical formalisms have been proposed instead of phrase-structure or dependency grammar, as being more adapted to syntactic phrase labelling or to translation. Almaghout et al. (2011) advocate the use of CCG labels, as they are syntactically adapted to labelling incomplete syntactic phrases. Li et al. (2013) employ again a single nonterminal, but apply source-side syntax as a means to constrain rule extraction, forcing rewritings to constituents or sequences thereof. Besides, they enrich the reordering model with predicate-argument structure reordering features, as do Xiong et al. (2012).

2.6 Syntax-Based SMT

Syntax-Based SMT Yamada and Knight (2001); Galley et al. (2004); Huang et al. (2006) takes a syntactic parse of the source or target sentence as a ground to establish translation equivalence. As such, Syntax-Based SMT has opposite qualities to Hierarchical Phrase-Based SMT: the grammar is unambiguous both in terms of rules and rewritings, but dependence on syntax limits data coverage.

2.6.1 The GHKM model

Yamada and Knight (2001) proposed a generative model with operations transforming a source tree into the observed translation sentence. Galley et al. (2004, 2006) use instead word alignments to extract **xRs** tree-transducer rules (Graehl and Knight, 2004; Graehl et al., 2008) aligning a syntactic subtree to a CFG-translation string. The resulting model, called GHKM after (Galley et al., 2004), extracts rules as shown in Figure 2.6. Minimal rules are completed with composed rules, which admit a predefined number of internal nodes, and allow to capture common constructions. Finally, rule variants are introduced to reflect possible attachments of unaligned words on the asyntactic, source side of the data.

Galley et al. (2006) apply the extracted rules in a string-to-tree setting, where the best translation $\hat{\mathbf{e}}$ of an input sentence $\mathbf{f}$ is the one that maximises the joint probability over $\mathbf{f}$ and $\mathbf{e}$, and where the translation probability $P(\mathbf{f}|\mathbf{e})$ is summed over tree-conditional probabilities:

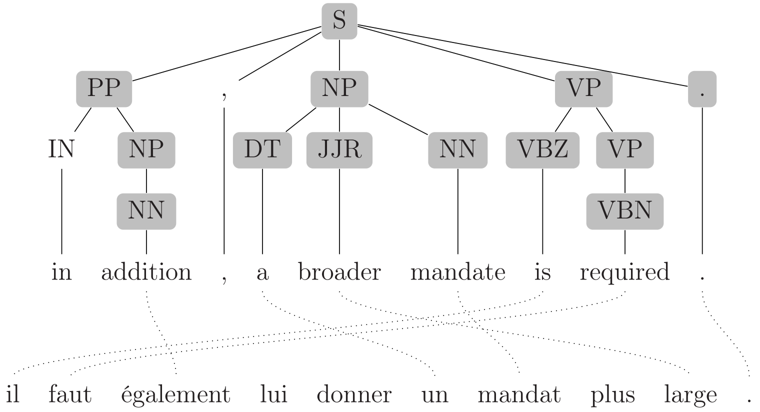
$$\hat{\mathbf{e}} = \arg\max_{\mathbf{e}\in\mathcal{E}} P(\mathbf{e}) \cdot \sum_{\pi\in\mathcal{T}(\mathbf{e})} P(\mathbf{f}|\pi) \cdot P(\pi|\mathbf{e}) \tag{2.5}$$

Translation model probabilities $P(\mathbf{f}|\pi)$ are decomposed over translation rules. Translation-rule probabilities are normalized over nodes (conditioned on left-hand-side symbol and span) and are estimated with EM over possible derivations. Normalizing over nodes gives better estimates, notably as this limits the overall weight of rules with unaligned source words.

Decoding is performed with the CKY algorithm, using binarized rules (Zhang et al., 2006). Binarization decisions may be heuristic (left or right binarization), or syntax driven: with head binarization, binarization is performed to the left or the right depending on the position of the syntactic head (Wang et al., 2007). Binarization decisions can be optimized per syntactic node with EM (Wang et al., 2007, 2010).

2.6.2 Increasing coverage

Parallel-data coverage is limited in Syntax-Based SMT compared to Hierarchical Phrase-Based SMT, even if rule composition allows to increase coverage compared


S
PP
,
NP
VP
.
IN
NP
DT
JJR
NN
VBZ
VP
NN
VBN
in addition , a broader mandate is required .
il faut également lui donner un mandat plus large .

Minimal rules

S(x_0:PP ,(,) x_1:NP x_2:VP x_3:.) → x_0 , x_1 x_2 x_3
PP(IN(in) x_0:NP) → x_0
NP(x_0:NN) → x_0
NN(addition) → également
NP(x_0:DT x_1:JJR x_2:NN) → x_0 x_2 x_1
DT(a) → un
JJR(broader) → large

NN(mandate) → mandat
VP(x_0:VBZ x_1:VP) → x_0 x_1
VBZ(is) → il
VP(x_0:VBN) → x_0
VBN(required) → faut

Composed rules

PP(IN(in) NP(NN(addition))) → également
VP(VBZ(is) VP(VBN(required))) → il faut
NP(DT(a) x_0:JJR x_1:NN) → un x_1 x_0

Accounting for unaligned words

NP(DT(a) x_1:JJR x_2:NN) → un x_1 plus x_0
JJR(broader) → plus large
NN(mandate) → mandat plus

Figure 2.6: GHKM rule extraction. The GHKM model identifies *frontier* nodes as syntactic nodes for which the yield forms a constituent phrase pair (boxed nodes). A *minimal* set of rules is then extracted, each frontier node leading to the extraction of one single rule. These rules are completed by composed nodes admitting a predefined number of internal nodes. Additionally, unaligned words on the source, asyntactic side are reattached to produce rule variants.

to a model employing only minimal rules.

One way to increase coverage consists in relaxing extraction constraints. Liu et al. (2007) notably propose to extract rules spanning over constituent sequences rather than single constituents, and use auxiliary rules to integrate them into a tree-to-string setting.

An alternative approach leverages on adjunction to extract more general rules (DeNeefe and Knight, 2009; Liu et al., 2011).

2.7 Adjunction in Syntax-Based SMT

Adjunction has been proposed as a means to model linguistic recursion beyond substitution. The Tree-Adjoining Grammar formalism introduced by Joshi et al. (1975) provides a greater generative power than Context-Free Grammar, namely that of mildly context-sensitive grammars. For linguistic applications, TAG is interesting because it allows to capture dependencies locally, as recursion is factored away. Synchronous TAG (Shieber and Schabes, 1990) allows to apply TAG properties to Machine Translation. Since Nesson et al. (2006) introduced Synchronous Tree Insertion Grammar to reduce the complexity attached to TAG, DeNeefe and Knight (2009) and Liu et al. (2011) have shown that adjunction can be succesfully modelled for Syntax-Based SMT.

2.7.1 Tree-Adjoining Grammar

Tree-Adjoining Grammar (TAG, Joshi et al., 1975; Joshi and Schabes, 1997) models linguistic recursion through adjunction. The grammar distinguishes *initial* trees and *auxiliary* trees, where central trees represent complete, *elementary* sentences, and adjunct trees represent modifiers. The operation of adjunction allows to derive more complex trees and sentences by combining auxiliary trees and initial, or derived trees. TAG is more powerful than CFG, allowing to model mildly context-sensitive languages (Joshi, 1985).

TAGs are generally lexicalized (Schabes et al., 1988), to reflect the linguistic assumption that syntactic behaviour is encoded in the lexicon, as in Lexical Functional Grammar (Kaplan and Bresnan, 1982) or Categorial Combinatorial Grammar (Steedman, 2000). The syntactic arguments of a given lexical item are then abstracted through substitution sites. Besides, TAGs allow to specify constraints on adjunction, allowing to specify nodes for which adjunction is forbidden, compulsory, or allowed selectively for a subset of auxiliary trees.

Two properties characterize the expressive power of TAG: the *Extended Domain of Locality* (EDL), and *Factoring Recursion from the domain of Dependencies* (FRD). The EDL refers to the ability of TAG trees to represent all arguments of a lexical item in a same rule, in contrary to phrase-structure grammars, where positing verb predicates (VP) for instance means that subjects and objects of a verb are rewritten in separate steps (Joshi and Schabes, 1997). The FRD directly refers to the distinction between auxiliary trees (for recursion) and initial trees (for dependencies). These properties make TAG interesting for the modelling of subcategorizations and filler-gap dependencies, like in *wh*-movement (Kroch and Joshi, 1985), but also idiomatic expressions (Abeillé and Schabes, 1989).

2.7.2 Synchronous Tree-Adjoining Grammar

Synchronous Tree-Adjoining Grammar (STAG, Shieber and Schabes, 1990) was introduced to pair trees with their semantic interpretation, or alternatively, equivalent trees in a target language. STAG allows to represent translation-equivalent but syntactically-divergent forms for Machine Translation (Abeillé et al., 1990), as shown in Figure 2.7.2.

Shieber (2007) further motivates the application of STAG for Machine Translation with bilingual dictionaries, in which entries are given without possible modifiers—modification is thus viewed as applying synchronously to both sides of the data. This is also the case with the modifiers represented by the auxiliary trees in Figure 2.7.2.

2.7.3 Applications in SMT

Parsing complexity of STAG grammars is prohibitevely high: monolingual TAG parsing complexity is in $O(n^6)$, and STAG parsing in $O(n^{12})$. Nesson et al. (2006) therefore propose to use Synchronous Tree Insertion Grammar for Machine Translation instead. Tree-Insertion Grammar (TIG, Schabes and Waters, 1995) restricts TAG by disallowing *wrapping adjunction*, where auxiliary trees adjoin elements on both sides. Restricting adjunction to left or right adjunction makes TIG equivalent to CFG in terms of generative power, while preserving the TAG properties of extended domain of locality and factored recursion.

Nesson et al. (2006) associate each lexical item in the source and target vocabulary with a set of abstract auxiliary trees, allowing for different adjoining orientations on source and target side, but using a single nonterminal X. The grammar is completed by abstract initial trees, with adjoining sites paired to the abstract auxiliary trees, and allowing to rewrite to the empty string on one side for unaligned words. EM is used to estimate auxiliary-tree probabilities for each aligned word pair. Nesson et al. show that STIG can perform better than Phrase-Based SMT for German-English, although their evaluation is restricted to short sentences and a small training and evaluation set.

DeNeefe and Knight (2009) also use STIG, but their grammar represents linguistic syntax on the English, target side of the data. Further, trees may be unlexicalized, and adjunction sites and their orientation are marked for each node;

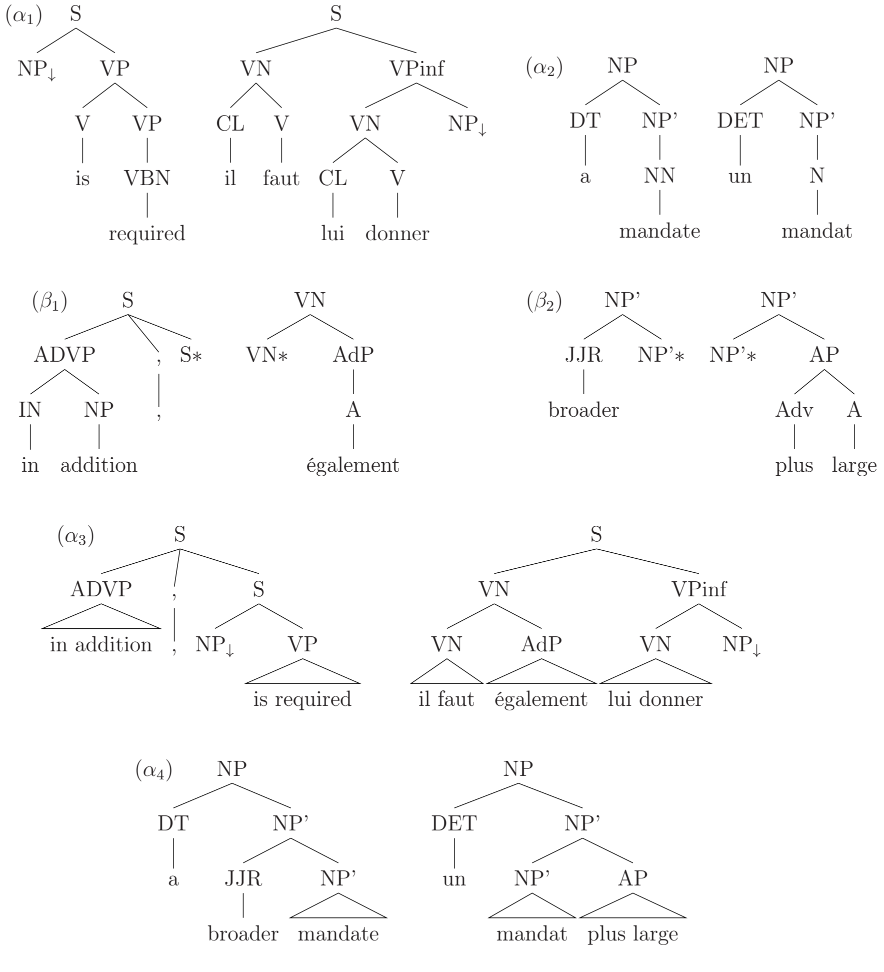


Figure 2.7: Initial, auxiliary and derived STAG trees. The derived trees α_3 and α_4 are obtained by adjoining the auxiliary tree β_1 to α_1 and β_2 and α_2, respectively. Substitution nodes in initial and derived trees are marked with a downarrow; auxiliary trees adjoin to initial or derived trees by their foot node, marked with an asterisk. Roots, substitution sites and foot nodes are linked synchronously in each paired tree.

nodes may have several adjunction sites, but adjunction sites cannot accept more than one auxiliary tree. Rule extraction starts by parsing the English side of the data, and identifying heads, arguments and adjuncts following (Collins, 2003) to binarize the trees and extract TIG trees, following (Chiang, 2000). Synchronous rules are extracted bottom-up, by aligning elementary trees to their source side, and recursively extracting substitution and adjunction rules with their adjunction sites.

Derivations are scored over substitution and adjunction rules with the probability distributions P_{sub} and P_{adj}, respectively, and an auxiliary distribution P_{ifadj}, which models adjunction decisions at each node (in both initial and auxiliary trees, as the latter may also carry adjunction sites):

$$\begin{aligned} P(d) = &\prod_{r_s} P_{sub}(r_s|\text{labels}(\text{root}(r_s))) \cdot \prod_{\eta\in r_s} P_{ifadj}(\eta) \\ &\cdot \prod_{r_a} P_{adj}(r_a|\text{dir}(r_a), \text{labels}(\text{root}(r_a))) \cdot \prod_{\eta\in r_a} P_{ifadj}(\eta) \end{aligned}$$

These distributions are estimated by counting. DeNeefe and Knight propose two models for adjunction decisions: a joint model, where adjunction decisions at a given node η are estimated jointly for all sites at the node; and an independent model, where adjunction decisions are estimated independently for each site at η.

The full translation model is a log-linear model that further incorporates Phrase-Based SMT lexical and phrasal features. DeNeefe and Knight evaluate their model on Arabic-English, and show that their model benefits primarily from the ability to generate new adjunction patterns.

Liu et al. (2011) also apply STIG to Machine Translation, but in a tree-to-string setting. Rule extraction starts with GHKM minimal rule extraction (Galley et al., 2004), to obtain *substitution rules*; those that exhibit adjunction patterns further lead to *adjunction rules* (these are substitution rules with an adjoining site at their root), and *auxiliary rules* that take the shape of TAG auxiliary trees. Finally, composed rules are derived by combining the substitution, adjunction and auxiliary rules, but allowing for a limited tree depth and source token count. Adjunction patterns are detected by identifying type-preserving rewritings.

The probabilistic model follows that of Resnik (1992) for Probabilistic TAG parsing (Chiang, 2000):

$$\begin{aligned} \sum_{\alpha} P_i(\alpha) &= 1 \\ \forall\eta, \sum_{\alpha} P_s(\alpha|\eta) &= 1 \\ \forall\eta, \sum_{\alpha} P_a(\alpha|\eta) + P_a(NONE|\eta) &= 1 \end{aligned}$$

Where $P_i(\bullet)$, $P_s(\bullet|\eta)$ and $P_a(\bullet|\eta)$ are probability distributions over initial trees, substitution at node η, and adjunction at η, respectively. Parameters are estimated by counting, and incorporated into a log-linear model (Liu et al., 2006). At decoding, auxiliary rules and adjoining rules are recombined to form STSG rules allowing for forest-based parsing (Mi et al., 2008).

Liu et al. evaluate their model on Chinese-English, and show improved performance with regard to a STSG baseline (Liu et al., 2006). The model performs on-par or better than Hiero, depending on the test set.

Chapter 3

How synchronous is adjunction in translation data?

The argument-adjunct distinction is central to most syntactic and semantic theories. As optional elements that refine the meaning of a phrase, adjuncts are important for recursive, compositional accounts of syntax, semantics and translation. In formal accounts of machine translation, adjuncts are often treated as modifiers applying *synchronously* in source and target derivations. But how well can the assumption of *synchronous adjunction* explain translation equivalence in actual parallel data? To address this question, we first study synchronous adjunction in a manually annotated and aligned subset of standard machine translation data. We then study the impact of automatic word alignments and adjunct labelling heuristics on synchronous adjunction, and show that synchronous adjunction remains high in experimental conditions for French-English. Measures of empirical synchronous adjunction in different parallel corpora suggest it is indicative of translation compositionality.

The work in this chapter is based on the following publication: Sophie Arnoult and Khalil Sima'an. *How Synchronous are Adjuncts in Translation Data?*. In SSST 2014.

While this chapter investigates the same question as the publication, its content has been significantly revised: manual annotations are better motivated and more consistent than previously; they include argument annotations, allowing for a comparison of adjuncts and arguments. These annotations now form a significant part of this chapter, as they provide the material for a qualitative study of translation equivalence of adjuncts and arguments, as well as a reference for measuring the effect of automatic word alignments and adjunct identification heuristics.

I implemented the code and conducted the research for both versions of the work myself, under the guidance of Khalil Sima'an.

Chapter Highlights

Problem Statement

- Synchronous Tree Adjoining Grammar (STAG) models adjunction as a synchronous process in parallel data, but the extent to which adjuncts behave synchronously in translation data is unknown. Assessing the degree of alignment of adjuncts is important for linguistic enrichment of translation models that rely on projecting annotations from one side of the data.

Research Questions

- To what extent is adjunction synchronous in translation data?
- How informative are empirical measurements of synchronous adjunction for translation equivalence?
- Do adjuncts behave differently than arguments in terms of alignment?

Research Contributions

- A corpus study of adjunct and argument alignment in English-French translation data, showing notably that 80% of adjuncts are synchronous in these conditions.
- An analysis of the effect of automatic word alignments and of parse-based annotations on adjunct and argument alignment, from which we conclude that the hypothesis of synchronous adjunction also holds in experimental conditions.
- Measures of adjunct and argument alignment in experimental conditions in several English-French datasets, showing that empirical synchronous adjunction is informative of translation compositionality.

3.1 Introduction

Most syntactic and semantic theories agree on the argument/adjunct distinction, although they vary on the specifics of this distinction. Common to these theories is that adjunction is a central device for language recursion: adjunction contributes to semantic compositionality as it allows to modify initial but syntactically complete phrases by adding optional phrases. Shieber and Schabes (1990) transfer the role of adjuncts from monolingual syntax (Joshi et al., 1975) to translation through Synchronous Tree Adjoining Grammars (STAG), and propose to view adjunction as a synchronous operation for *recursive*, *compositional* translation. STAG therefore relies substantially on what Hwa et al. (2002) call the Direct Correspondence Assumption, the notion that semantic or syntactic relations directly correspond across a bitext; with a notion of translation equivalence based on word alignments, direct correspondence boils down to bijective projection through the word alignments. We know from various works—notably by Hwa et al. (2002) and Fox (2002) for dependency relations, Arnoult and Sima'an (2012) for adjuncts, and Padó and Lapata (2009) and Wu and Fung (2009) for semantic roles—that the Direct Correspondence Assumption does not always hold, as linguistic structures may diverge between languages (Dorr, 1994).

A question that has not received much attention is the degree to which the assumption of synchronous adjunction is supported in human translation data. This is crucial for the succesful application of linguistically-motivated STAG, but attempts at answering this question empirically are hampered by a variety of difficulties: translations may be more or less literal; annotation resources may be inaccurate; and translation equivalence in SMT models is based on automatic word alignments, which can be noisy. Consider for example the sentence pair of Figure 3.1. This sentence pair is representative of standard translation data in that both sentences are translation equivalent even though the equivalence may be non-literal locally, as with *cette attitude* and *it*; in this case, word alignments capture this relationship only partially. The sentence pair contains a single, synchronous adjunct pair, *considerably/grandement*, which we can identify by the dependency modifier labels *amod/mod*. But how reliable are these heuristics, especially considering the difficulty of disambiguating adjuncts from arguments (Manning, 2003), notably for prepositional-phrase attachment (Merlo, 2003; Abend and Rappoport, 2010; Greenberg, 2014; de Kok et al., 2017)?

This chapter presents a corpus study of adjunct alignment in French-English translation data, and relates gold measures of synchronous adjunction to empirical measures based on word alignments and parse-based adjunct-identification heuristics. To put these measures into context, we further compare adjuncts to arguments in the corpus study, and adjuncts to syntactic complements for the empirical measures.

Section 3.2 opens this work with a small corpus study of adjunct and argument alignment in French-English translation data. We present annotation guidelines for a corpus of adjunct/argument annotations and alignments, and analyze adjunct and argument alignment in that corpus. We show that about 80% of adjuncts and arguments are synchronous in French-English; the main difference between adjuncts and arguments lies in the ability of adjuncts to form synchronous pairs in dissimilar syntactic contexts.

Section 3.3 considers the relationship between word alignments and manual adjunct/argument alignments, showing that word alignments largely agree with

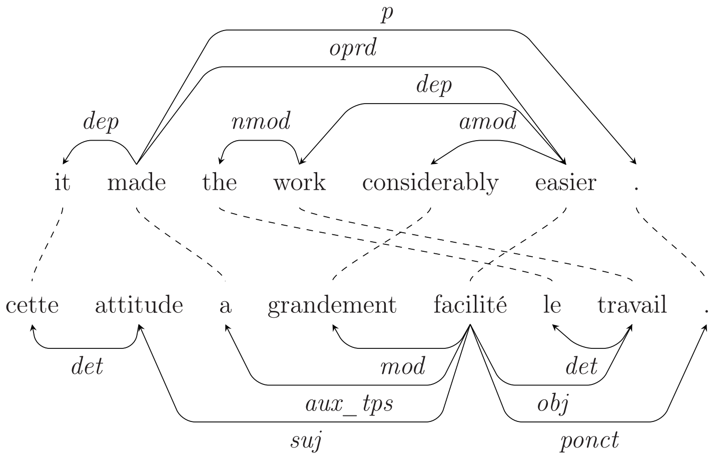


Figure 3.1: Aligning dependencies through word alignments.

manual alignments, while failures to align can be imputed to differences in syntactic category and misaligned function words.

Section 3.4 deals with the relationship between parse-based labelling heuristics and manual annotations. Abstracting away from differences in attachment of adjunct-like constituents, we measure the agreement of parse-based heuristics for adjuncts with gold annotations for three different parsers for English and one French parser. While unlabelled F_1 is high for adjuncts, reaching above 90%, labelled F_1 is lower, reflecting difficult cases of adjunct/argument disambiguation.

Finally, section 3.5 considers the relationship between experimental and gold measures of synchronous adjunction. Adjunct/argument ambiguity plays here too, as we find that about two thirds of adjuncts in extractable phrase pairs correspond to gold synchronous adjunct pairs, while about 80% correspond to gold synchronous pairs in general. Parsing both sides of the data further provides us with a measure of synchronous adjunction in experimental conditions, which is informative of gold synchronous adjunction, and appears to be indicative of translation compositionality for translation corpora.

3.2 A corpus study of adjunct alignment

We start addressing the question of how synchronous adjunction is with a small-scale study of adjunct and argument alignment in manually annotated and aligned data. The main goal of this study is to answer how synchronous adjunction is *in general*, and so to provide comparison material for measures of adjunct alignment in experimental conditions. Besides, this study also aims at assessing whether adjuncts and arguments differ in their alignment.

The adjunct/argument distinction is one that is notoriously hard to make: telling adjuncts from arguments demands in fact determining whether a constituent is part of a construction or whether it is modifying it in a systematic fashion. Adjuncts and arguments can be seen as both ends of a continuum (Manning, 2003) in this respect, and models for prepositional-phrase attachment (Pantel and Lin, 2000; Olteanu and Moldovan, 2005; Greenberg, 2014; de Kok et al., 2017; Madhyastha et al., 2017) generally quantify the interaction between a head and a dependent to assess whether the latter is selected by its head (an argument) or selects it (an adjunct). Linguistic accounts also address the ambiguity of the adjunct/argument distinction while proposing criteria for categorization (Dowty, 2000; Partee and Borschev, 2003; Kay, 2005).

Section 3.2.1 presents annotation criteria for adjuncts and arguments based on (Dowty, 2000), as well for as criteria for their alignment. These criteria are applied on a subset of the French-English Europarl corpus, for which section 3.2.2 provides basic statistics. Finally, section 3.2.3 provides measures of adjunct and argument alignment in this dataset, and an analysis thereof.

3.2.1 Annotation criteria

Telling adjuncts from arguments

For this study, adjuncts in verbal context are identified following (Dowty, 2000): a modifier Y is an adjunct of a phrase X if X has the same meaning on its own and in the modified phrase [XY]; adjuncts modify the meaning of phrases in the *same* way. For instance in Example 17, *on young people* is annotated as an argument and *for young people* as an adjunct: the first gives rise to a telic, goal-oriented reading of the verb, while the second preserves its generic meaning.

(17) working [*on young people*]$_{arg}$ and [*for young people*]$_{adj}$

In noun phrases, the adjunct/argument distinction is based on syntactic optionality. Essentially, all noun-phrase modifiers are identified as adjuncts, except for genitives and constituents of a fixed expression. Fixed expressions include:

- multi-word prepositional expressions: *on the basis [of . . .]*, *in order [to . . .]*;
- quantifying expressions: *a plethora [of . . .]*, *a lot [of . . .]*;
- proverbial expressions: *les bras [ballants]*[1]

Multi-word proper nouns, like *Saudi Arabia*, *South Korea*, *Proinsias De Rossa*, are seen as a single term, and their constituents are left unannotated.

[1] lit., *with hanging arms*, i.e., *helplessly.*

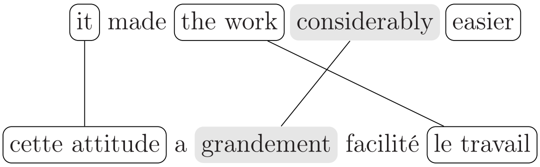


Figure 3.2: Aligning adjunct (shaded) and argument annotations. Alignment between phrases is perform based on reference or event sharing.

Table 3.1: Statistics for the manually annotated corpus.

	sentence length			annotated constituents		
lang	min	max	avg	total	adj	arg
en	2	88	35.07	1160	626	534
fr	2	113	39.50	1216	662	552

Purely syntactic complements are also excluded from the annotations. This concerns:

- determiners
- impersonal pronouns
- coordinators
- complements of prepositions or conjunctions
- verb auxiliaries

Aligning adjuncts and arguments

Adjuncts or arguments are regarded as translation equivalent and aligned if: they denote the same referent (for noun phrases), or event (for predicates); they modify translation-equivalent phrases in a similar manner. For instance, in Figure 3.2, *it* and *cette attitude* are aligned because they share the same referent, and *considerably* and *grandement* because they modify a translation-equivalent expression in a similar fashion.

3.2.2 Statistics

The corpus for this study consists of 100 sentences extracted from the French-English Europarl corpus. About a third of tokens are annotated as heads of an adjunct or argument. Other statistics are reported in Table 3.1.

Figure 3.3 shows the distributions of the syntactic types of adjuncts and arguments for each side of the corpus, and of the syntactic type of modified phrases. These confirm known syntactic patterns in English and French: English employs more noun phrases and less prepositional phrases than French, for both adjuncts and arguments. Besides, English employs more adjectival phrases and less prepositional phrases than French for adjuncts. Looking at the type of modified phrases, we find slightly more nominal constructions in French, and more verbal constructions in English.

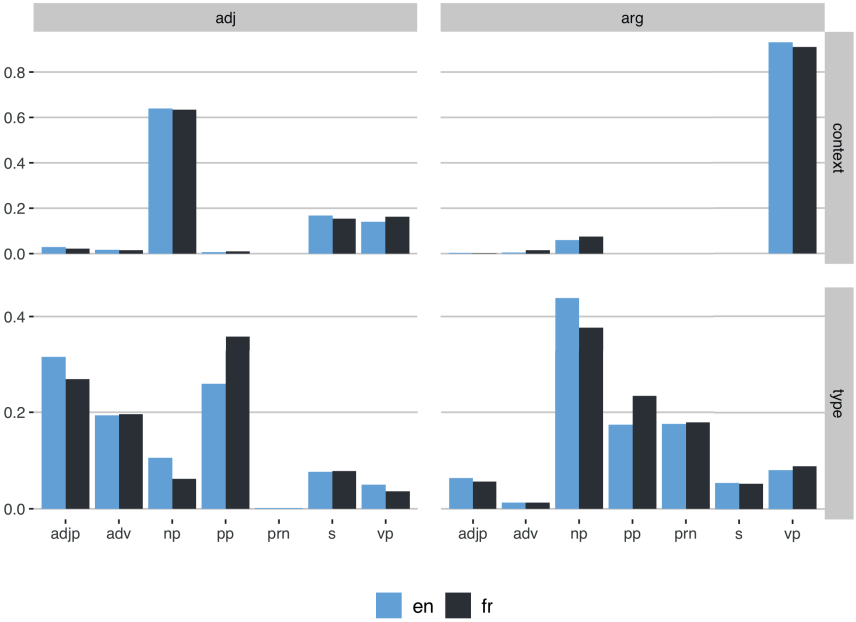


Figure 3.3: Distributions of syntactic types and modified-phrase types (context) for adjuncts and arguments in English and French.

3.2.3 Adjunct and argument alignment in translation data

Table 3.2 provides counts of adjunct and argument alignment in the corpus. About 80% of adjuncts and arguments are synchronous in the corpus, arguments following adjuncts by only a few points.

The rest of this section details the different types of alignment.

Table 3.2: Adjunct and argument alignment in the manually aligned corpus. Four types of alignment are considered: *synchronous* for synchronous adjuncts/arguments; *switch* for adjuncts and arguments that switch role to argument or adjunct; *n-m* for one-to-many or many-to-one alignments; *unaligned* for remaining cases.

		freq.	alignment			
			synchronous	switch	n-m	unaligned
adj	en → fr	626	81.6	1.9	2.6	13.9
	fr → en	662	77.2	3.9	4.7	14.2
arg	en → fr	534	79.0	4.9	0.6	15.5
	fr → en	552	76.2	2.2	0.7	20.7

Role switching

The corpus counts 38 cases of role switching, 12 English adjuncts and 26 arguments switching to the opposite role in French. Most cases of role switching concern NP adjuncts and VP arguments (18 of 38 cases); 11 cases occur between VP adjuncts and VP arguments, and the remaining 9 cases between NP adjuncts and arguments.

English role switching in an NP-VP contetx results mostly from attributive/predicative alternation, and French role switching from nominalized constructions:

(18) this is a *necessary* operation / cette opération est *nécessaire*

(19) la désindustrialisation *d'une région entière* / *an entire region* was desindustrialized

Cases of role switching concern VP modifiers and arguments result mostly from active/passive alternation and/or differences in topic choice:

(20) the distinction is not made *according to ...*
une telle distinction ne se base pas *sur ...*

(21) *the conference* is not meant to be addressing this issue
ce sujet ne doit pas être abordé *au cours de la CIG*

Most cases of NP-NP role switching concern English genitives switching to French prepositional modifiers:

(22) *the European Parliament's* committee
la commission *du Parlement européen*

Table 3.3: Top-three categories (by type and modified-phrase type) of unaligned adjuncts and arguments, and their proportion.

		freq.	Top-three categories and proportions		
adj	en → fr	87	adj^np - 0.29	adv^vp - 0.16	pp^np - 0.11
	fr → en	94	adj^np - 0.28	adv^vp - 0.19	pp^np - 0.15
arg	en → fr	83	prn^vp - 0.33	np^vp - 0.23	adj^vp - 0.12
	fr → en	114	prn^np - 0.25	np^vp - 0.24	pp^np - 0.16

N-M alignments

The corpus contains sixteen 1-2 and two 2-1 en-fr alignments. Thirteen of the 1-2 alignments concern the use of double negation in French (Example 23); the three other 1-2 alignments concern coordinated prepositional phrases (Example 24) and show a preference for having coordination within the scope of prepositions in English, and outside of prepositions for French. The two 2-1 en-fr alignments correspond to a negated adjective (Example 25) and a paraphrase (Example 26).

(23) I see *no* threats / je *ne* vois *aucune* menace

(24) *on culture and education* / *de la culture* et *de l'éducation*

(25) it is *not acceptable* / il est *inacceptable*

(26) the *beaf* and *veal* market / le marché *de la viande*

Unaligned adjuncts/arguments

Table 3.3 lists the top three categories of unaligned adjuncts and arguments for both language directions.

Unaligned adjuncts are likely to be untranslated (Example 27), to switch to the head of the translation-equivalent construction (Example 28), or to merge lexically with their governor (Example 29):

(27) to give *just* one simple example / prenons un exemple simple

(28) for his *sterling* work / pour la *qualité* de son travail

(29) *trade* union / syndicat

Unaligned pronominal arguments may result from a personal/impersonal alternation or from a change of construction:

(30) *we* still have / *il* reste

Table 3.4: Proportion of synchronous adjuncts and arguments according to syntactic similarity. Proportions are conditionned on constituents with an adjunct/argument translation equivalent, i.e. synchronous and role-switching adjuncts and arguments. For syntactic similarity, we distinguish similarity in syntactic type and in the type of the modified phrase (context). Counts correspond to the syntactic-similarity types.

			same context		different context	
			same type	diff. type	same type	diff. type
adj	en → fr	counts	394	81	35	12
		P(synch)	0.990	0.988	0.857	0.846
	fr → en	counts	392	88	32	25
		P(synch)	0.995	0.909	0.938	0.440
arg	en → fr	counts	362	63	6	17
		P(synch)	0.994	0.873	0.667	0.176
	fr → en	counts	364	56	9	5
		P(synch)	0.989	0.982	0.444	0.6

(31) if *they* see a conflict / en cas de conflit

Like adjuncts, unaligned arguments are likely to merge with their governor in the translation-equivalent construction:

(32) *get* our programme *up to strength* / *renforcer* notre programme

(33) took *place* / perpetué

Synchronous adjuncts and arguments

We have seen that about 80% of adjuncts and arguments are synchronous. How do these figures relate to syntactic similarity between French and English? To address this question, we look at adjuncts and arguments that have a translation-equivalent adjunct or argument (whether they are synchronous or switch role), and assess how their degree of synchronous alignment is affected by syntactic similarity.

We use the syntactic type of aligned adjuncts and arguments, and the type of their parent phrase to delimitate four categories of syntactic similarity. Table 3.4 presents counts for each category, and the proportion of synchronous adjuncts/arguments over all aligned adjuncts/arguments.

We find that the proportion of synchronous adjuncts is relatively invariant to syntactic similarity. Arguments in contrary are unlikely to be synchronous in dissimilar contexts.

3.2.4 Summary

In this section, we have studied adjunct alignment using manual annotations and alignments in a small subset of the French-English Europarl corpus. We have compared alignment of adjuncts and arguments, and considered the relationship between adjunct/argument synchronous alignments and syntactic similarity.

These are our main findings:

- Our annotations confirm some patterns of difference between English and French, like a preference for nominalized constructions in French, or for (prenominal) adjectival or nominal modifiers in English.

- Measures of adjunct/argument alignment do not show a striking difference between adjuncts and arguments. We find that about 80% of adjuncts are synchronous, against just a few points less for arguments. An apparent difference between adjuncts and arguments concerns unaligned cases, where we observe different causes of unalignment: discourse markers form likely categories for unaligned adjuncts and pronouns for arguments; both categories can be united as semantically *light*, and therefore more likely to be untranslated.

- This confirms in the context of translation that adjuncts are more loosely connected than arguments to the phrases they modify.

3.3 Synchronous adjunction and word alignments

In our corpus study, the alignment between adjuncts and arguments was established manually. To what degree do word alignments agree with adjunct alignments? And to what extent can synchronous adjuncts be extracted for Phrase-Based SMT or Hiero?

Section 3.3.1 presents the method used for relating adjunct alignments and word alignments. Section 3.3.2 analyzes the relationship between manual annotations and word alignment, and section 3.3.3 provides measures of the extractability of synchronous adjuncts.

3.3.1 Matching word alignments and manual annotations

Word alignments for the annotated corpus are trained with the Berkeley Aligner (Liang et al., 2006) on the full Europarl corpus, for sentence lengths up to 80 tokens. Permutation Trees (Zhang and Gildea, 2007; Zhang et al., 2008) are used to represent the word alignments hierarchically and facilitate the search of matching alignment spans. Figure 3.4 presents the example of Figure 3.2 with its word alignment and permutation tree.

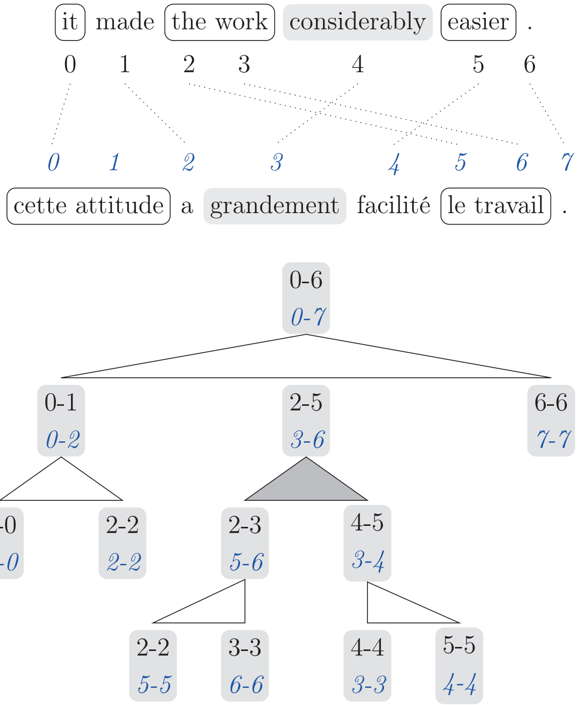


Figure 3.4: Projecting adjuncts and arguments through permutation trees. Permutations may be monotone (white), inverted (gray) or complex. Sequences of monotone or inverted alignments are flattened.

Matching adjunct/argument alignments proceeds in two steps, by first comparing the source span of a given constituent to the source side of the permutation tree; if this source span corresponds to a permutation node or a sequence of sibling nodes, the span of the aligned constituent is compared to the target span of the matched node.

Matching source spans to the permutation tree takes four forms:

tight the source span exactly matches the span of a permutation node or of a sequence of sister nodes;

loose the source span covers a permutation node or a sequence thereof, and includes bordering unaligned source positions;

null the source span covers unaligned source positions only;

fail the source span crosses permutation node boundaries or covers permutation nodes partially.

Given a source-matching permutation node or node sequence and an aligned annotation, the same possibilities excepted `null` arise on the target side. This leads to the following types of agreement between synchronous adjuncts or arguments and word alignments:

tight both sides of the synchronous adjunct/argument match word alignments tightly

loose both sides agree with the word alignments, but only loosely on one or both sides

src-fail the source side of the synchronous pair disagrees with the word alignments (*null* or *fail*)

trg-fail the source side of the synchronous pair agrees with the word alignments, but not the target side

The distinction between tight and loose synchrounous-pair agreement is relevant for SMT: whereas Phrase-Based SMT (Koehn et al., 2003) allows for the extraction of loose phrase pairs, Hiero (Chiang, 2005) only allows to tight phrase pairs. In the example of Figure 3.4, *it/cette attitude* forms a loose phrase pair, *the work/le travail* and *considerably/grandement* tight phrase pairs. The non-synchronous adjunct *easier* is tight for word alignments, and would lead to the tight phrase pair *easier/facilité*.

3.3.2 Adjunct agreement with word alignments

We start by considering single-sided agreement with word alignments, comparing adjuncts to arguments as before, but also synchronous pairs and non-synchronous adjuncts and arguments. Table 3.5 shows that source-side agreement of adjuncts and arguments is high in general, while correlating with synchronous alignment. Overall, adjuncts agree more often tightly, and arguments loosely. The most notable difference between adjuncts and arguments concerns non-synchronous constituents that disagree with word alignments: adjuncts show a higher prevalence of null-alignments, while arguments tend to agree only partially with word alignments.

As Figure 3.5 shows, synchronous adjuncts that disagree with word alignments on their source side are predominantly prepositional phrases, in which the preposition is likely to be misaligned. For synchronous arguments, the high proportions of noun-phrases that fail to agree tightly may be caused by misaligned determiners. Misalignment of function words across English and French generally explains failures to align, for all categories aside of pronouns, adjectival and adverbial phrases.

Finally, cases of non-synchronous adjuncts that do agree with word alignments represent adjuncts that switch role in translation, to arguments or heads, as in

Table 3.5: Source-side agreement of adjuncts/arguments with word alignments. Non-synchronous alignments include role-switching and n-m alignments.

	synchronous role	source role	source-side agreement (%) tight	loose	null	fail
en→fr	yes	adj	84.9	4.9	0.2	10.0
		arg	77.3	8.3	0.5	14.0
	no	adj	32.8	14.7	28.4	24.1
		arg	31.2	18.8	14.3	35.7
fr→en	yes	adj	76.3	10.0	0.6	13.1
		arg	72.5	13.7	0.7	13.0
	no	adj	19.9	15.9	23.8	40.4
		arg	24.6	13.1	16.9	45.4

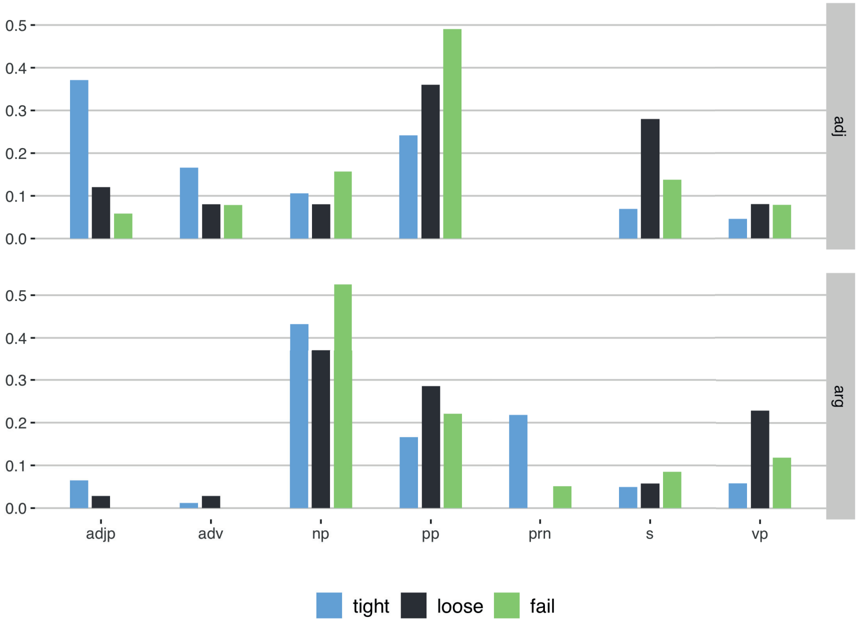


Figure 3.5: Distribution of English-side agreement types over syntactic categories, for synchronous pairs.

the example of Figure 3.4, where the adjunct *easier* is translated as the verbal head *facilité*.

3.3.3 Extractability of synchronous adjuncts

Turning to the relation between synchronous adjuncts/arguments and phrase-pair extractability, we find that single-side agreement from one side is echoed on the other side, as shown in Table 3.6: synchronous loose agreements cumulate loose agreements on the French and English sides; failures to align on the target side echo source-side failures while being lower.

Table 3.6: Agreement of synchronous adjuncts/arguments with word alignments

		tight	loose	src-fail	trg-fail
en→fr	adj	68.9	11.4	10.2	9.6
	arg	61.4	17.1	14.5	7.1
fr→en	adj	68.9	11.4	13.7	6.1
	arg	61.8	17.3	13.7	7.8

In total, 70% of synchronous adjuncts form tight phrase pairs, against 60% of arguments. Including loose alignments increases the proportion of extractable synchronous adjuncts and arguments to 80%. This reflects the tendency of arguments to agree only loosely with word alignments on either side.

3.3.4 Summary

We have evaluated in this section how word alignments affect measures of synchronous adjunction. We find that word alignments agree largely with adjunct and argument alignments, and that about 80% of synchronous adjuncts and arguments can be extracted by phrase-based models; for Hiero and models relying on tight agreement with alignments, phrase-extractability figures decrease to 70% for adjuncts and 60% for arguments. Comparing adjuncts and arguments, we find that arguments are less likely to agree tightly with word alignments. This can be explained by the fact that arguments, more than adjuncts, are expressed with categories containing function words, which are more likely to misalign.

3.4 Parsed-based adjunct heuristics

For linguistic enrichment of SMT models, including applications of adjunction to SMT (DeNeefe and Knight, 2009; Liu et al., 2011), it is customary to apply heuristics on parse labels. But to what degree do such heuristics agree with adjuncts and arguments?

Section 3.4.1 presents mapping rules for adjuncts and syntactic complements for a French dependency converter and three English parsers. Section 3.4.2 provides agreement scores between parse-based labels and gold adjunct/argument annotations.

3.4.1 Parse-based adjunct/complement mapping rules

Parsers allow for a simple distinction between syntactic modifiers and complements: in phrase-structure parses, modifiers are constituents that have a sister head and a mother of the same syntactic category, which requires knowing the head of each constituent; dependency parsers employ dedicated modifier labels; in CCG parses, modifiers are interpreted as functors that modify a category X and return a constituent of the same category.

In this work, we identify adjuncts with syntactic modifiers. Modifier dependency labels are adapted however, to avoid counting determiners and function words as adjuncts. All non-adjuncts are defaulted to complements.

French dependency-based rules

French adjunct/complement labels are derived from the dependency converter of Candito et al. (2010), which converts French phrase-structure parses from the Berkeley Parser (Petrov et al., 2006) to dependency parses.

The modifier labels *mod* and *mod_rel* are mapped to adjuncts, as well as the default label *dep* for noun-phrase modifiers. Closed-class dependents are interpreted as arguments. This concerns leaf dependents with one of the following part-of-speech tags: `C`, `CL`, `P`, `P+D`, `D`, `PRO`, `PONCT`. Remaining dependents are interpreted as arguments.

English dependency-based rules

We use two different parsers to obtain dependency-based labels.

The first is the Pennconverter (Johansson and Nugues, 2007), which allows to convert constituency to dependency parses with fine-grained annotations. The Pennconverter is applied to the output of the Berkeley Parser as for the French data. The labels `ADV`, `APPO`, `AMOD` and `NMOD` are mapped to adjuncts, excluding closed-class leaf dependents with the part-of-speech: `CC`, `CD`, `DT`, `EX`, `IN`, `MD`, `POS`, `PRP`, `PRP$`, `RP`, `SYM`, `TO`, `WDT`, `WP`, `WP$`, or `WRB`. Remaining dependents are interpreted as arguments.

The second is the Turbo parser[2]. The modifier labels `AMOD`, `NMOD` and `VMOD` are mapped to adjuncts, excluding the same closed-class leaf dependents as for the Pennconverter parses.

[2] `http://github.com/andre-martins/TurboParser`

English CCG-based rules

EasyCCG (Lewis and Steedman, 2014) was used to obtain CCG parses for the English data. The CCG parses are then converted to a dependency format (as only adjuncts and arguments are to be annotated) by identifying in order: absorbtion rules (punctuation and conjunction tags become arguments), application rules, and composition rules. Tags with the pattern X/X are treated as adjuncts (but not tags with the pattern X[y]/X). The CCG syntax is preserved except for determiners, which are functors in CCG, but which are mapped here to arguments instead of heads of noun phrases.

3.4.2 Agreement between parse-based labels and gold annotations

We use two criteria to match the spans of parse trees and gold annotations: *strict* and *relaxed*. The latter accounts for differences in attachment of adjuncts (prepositional phrases or other), conjunctions and punctuation. Consequently, the following spans are equivalent in the *relaxed* case:

(34) *wherever they are based* (Penn/Turbo)
wherever they are based . (CCG)
, wherever they are based (manual)

Agreement between parse-based and gold adjunct/argument annotations is measured by iterating over parse trees and determining whether:

- the tree span matches that of a manual annotation, by either *strict* or *relaxed* match. The comparison between both spans is symmetrical, taking both tree and manual spans as reference. Parse labels are used to identify the adjuncts, punctuation and conjuncts to abstract away for *relaxed* matches. When a span approximately matches several manual annotations, preference is given to the one with a matching head.

- the parse label matches the role of a matched annotation; complement labels are matched to gold arguments, even though the two categories are not entirely comparable, since complements include arguments and syntactically motivated complements.

Agreement measures for both matching criteria and sets of parse-based labels are summarized in Table 3.7. We find that:

- abstracting over adjuncts, punctuation and conjuncts for span matching has a large positive effect on agreement, increasing not only precision but also recall;

- parse-based heuristics tend to interpret as adjuncts what our manual annotations interpret as arguments, as shown by lower labelled recall scores for arguments, and the decrease between unlabelled and labelled precision scores for adjuncts.

Table 3.7: Agreement of parsed adjuncts and complements with manual adjunct/argument annotations. Unlabelled recall is computed jointly for all adjunct/argument annotations. Parsed complements naturally have low precision with regard to manual argument annotations, as they also cover syntactically motivated complements.

			unlabelled			labelled		
parser	match	role	p	r	F_1	p	r	F_1
penn	strict	adj	76.1	77.4	76.7	60.8	64.7	62.7
		cmp/arg	22.5	77.4	34.9	18.0	58.3	27.5
	relaxed	adj	92.4	94.9	93.7	74.7	79.0	76.8
		cmp/arg	28.7	94.9	44.0	22.0	71.6	33.7
cand	strict	adj	71.4	77.1	74.2	59.4	68.9	63.8
		cmp/arg	20.8	77.1	32.8	19.0	64.2	29.4
	relaxed	adj	89.0	94.2	91.5	74.3	84.6	79.1
		cmp/arg	26.2	94.2	41.0	22.1	74.1	34.1
turbo	strict	adj	67.0	75.8	71.1	50.9	72.0	59.6
		cmp/arg	18.9	75.8	30.2	17.2	48.6	25.4
	relaxed	adj	81.4	92.2	86.5	62.2	87.5	72.7
		cmp/arg	23.8	92.2	37.8	21.1	59.4	31.2
ccg	strict	adj	44.5	50.0	47.1	43.1	56.5	48.9
		cmp/arg	13.5	50.0	21.3	12.9	38.1	19.3
	relaxed	adj	77.3	90.3	83.3	66.2	86.7	75.1
		cmp/arg	26.2	90.3	40.6	24.5	72.5	36.7

The manual annotations presented in section 3.2 only annotate arguments that can be compared to adjuncts as part of the adjunct/argument distinction. In the heuristics presented above, complements however regroup all non-adjuncts, resulting in an awkward comparison between the sets of parse-based complements and manually annotated arguments. As shown in Table 3.8, agreement figures for complements and arguments improve if we exclude closed-class leaf dependents and complements of syntactic heads, such as complements of prepositions. The increase in precision does come at the price of a large drop in recall. Only for the French set of labels do we obtain fair agreement measures for arguments.

Table 3.8: Agreement of parsed complements with manual argument annotations with relaxed span matching, comparing the effect of labelling all non-adjuncts as complements (`non-adj`), to a stricter definition of arguments excluding function words and arguments thereof (`non-func`).

		unlabelled			labelled		
		p	r	F_1	p	r	F_1
penn	non-adj	28.7	94.9	44.0	22.0	71.6	33.7
	non-func	45.9	79.8	58.3	35.2	45.4	39.6
cand	non-adj	26.2	94.2	41.0	22.1	74.1	34.1
	non-func	73.9	80.8	77.2	66.5	50.6	57.5
turbo	non-adj	23.8	92.2	37.8	21.1	59.4	31.2
	non-func	42.5	79.3	55.4	41.1	37.2	39.1

3.4.3 Summary

This section presented parse-based mapping rules for French and English adjuncts and complements, and agreement measures between adjunct/complement labels and gold adjunct/argument annotations. We proposed to match parse-based labels and gold annotations based on span similarity, abstracting away differences in attachment of surrounding punctuation, adjuncts and conjuncts. The unlabelled precision of parsed adjuncts is high in this case, reaching about 90% for the Pennconverter and the French converter. Labelling scores decrease for adjuncts, reflecting a tendency of parsers to interpret manual argument annotations as modifiers, which is likely to point to difficult cases for the adjunct/argument distinction.

Comparing syntactic complements to arguments is awkward as complements include both arguments and syntactically motivated complements. Refining mapping heuristics for complements to better align to the argument-annotation criteria of section 3.2.1 only partially improves agreement, as the increase in precision is countered by a large decrease in recall. In the rest of this dissertation, we leave aside the relationship between syntactic complements and semantically-motivated arguments, and keep a distinction between adjuncts/modifiers on one hand and complements on the other hand.

3.5 Synchronous adjunction in experimental conditions

Knowing that word alignments and parse-based labels agree only partially with synchronous adjuncts, to what degree can we expect parsed adjuncts in phrase pairs to translate as adjuncts? And how informative are measurements of synchronous adjunction based on these word alignments and parse-based labels?

We address the first question in section 3.5.1, where we project parse-based adjuncts and complements through word alignments and measure how often the resulting phrase pairs correspond to gold synchronous adjuncts or arguments. We find that two thirds of parsed adjuncts in extractable phrase pairs correspond to gold synchronous adjuncts, and around 80% to synchronous pairs in general. Section 3.5.2 considers the relationship between word-aligned, parse-based synchronous alignments and gold synchronous alignments. We find that empirical and gold synchronous alignments are mutually informative for adjuncts, albeit modestly. Still, 80% to 90% of empirically synchronous adjuncts correspond to gold synchronous adjuncts. Finally, section 3.5.3 provides experimental measures of synchronous adjunct/complement alignment in a number of French-English Machine Translation corpora. We show that empirical synchronous adjunction is indicative of translation compositionality, and synchronous complementization of syntactic similarity.

3.5.1 Projecting source-side annotations

Given parse-based labels for identifying adjuncts, how often can we expect phrase pairs with an adjunct source to correspond to a gold synchronous adjunct? This is relevant for the work presented in Chapter 4, as it relies on the Direct Correspondance Assumption (Hwa et al., 2002) to project source-side adjunct annotations through synchronous, SCFG rules. For comparison, we perform the same measures for parsed complements and synchronous arguments.

To this end, we select parse-based adjuncts and complements that are part of extractable phrase pairs, and count how many correspond, first to a manual annotation, and secondly to an aligned annotation. For phrase pairs, we consider tight agreement with alignments on the source side, and we use the relaxed span-matching criterion of section 3.4 to match parsed adjuncts/complements and manual annotations on either side. Results are presented in Table 3.9. To compensate for the low precision of complements with regard to argument annotations, we also condition the counts of adjuncts and complements corresponding to gold synchronous pairs on those that match the span of a manual source annotation.

In extractable phrase pairs, two thirds of parsed adjuncts correspond to a gold synchronous pair (62.8% to 73.6%, depending on the parser). Isolating complements that match argument spans result in a comparable degree of correspondance with synchronous pairs.

Besides, 10 to 15% of adjuncts correspond to a gold aligned pair in the unlabelled case. These cases point again to adjuncts that are difficult to distinguish from arguments, and that correspond to synchronous arguments in our annotations. If we abstract away from labelling differences, around 80% (73.9% to

Table 3.9: Projecting parse adjunct/complement labels through word alignments. Figures report the proportion of parsed constituents in extractable phrase pairs with: a matching manual annotation ($\texttt{src}$) and a matching gold alignment ($\texttt{pair}$) for both unlabelled and labelled cases; $\texttt{pair}_l$ corresponds to gold synchronous adjuncts/arguments. $\texttt{pair}_l|\texttt{src}_u$ conditions synchronous matches on parsed constituents with a matching source-annotation span.

		unlabelled		labelled		
		src$_u$	pair$_u$	src$_l$	pair$_l$	pair$_l$|src$_u$
penn	ADJ	94.8	87.2	80.0	73.6	77.6
	CMP	35.3	32.1	26.8	24.4	69.0
turbo	ADJ	83.4	77.0	68.1	62.8	75.3
	CMP	30.3	26.7	27.0	24.1	79.4
ccg	ADJ	80.5	73.9	71.0	65.0	80.7
	CMP	31.7	28.7	29.9	27.2	85.9
cand	ADJ	89.9	81.8	77.0	70.6	78.5
	CMP	31.8	28.6	26.4	23.9	75.0

87.2%, depending on the parser) of parsed adjuncts correspond to a synchronous adjunct or argument pair.

3.5.2 Experimental and gold synchronous alignments

Until now we have only applied parsed labels on one side of the data, and compared them to manual annotations. It is interesting however to consider the relation between source and target labels, and the experimental measures of synchronous alignments one can derive from them, as they do not depend on manual annotations and can thus be exploited on unannotated data.

In this section, we parse both sides of the data to relate experimental, parse-based measures of synchronous adjunction with gold synchronous alignments. We consider again only parsed adjuncts or complements on the source side of extractable phrase pairs, and count first, how many form synchronous pairs based on the role of their target counterpart; and secondly, how many correspond to gold synchronous pairs with the same role. Measures are reported in Table 3.10. 80% to 90% of phrase pairs with source and target adjunct labels correspond to manual annotations, but mutual information between parse-based and gold synchronous adjuncts is low. In a relatively high number of cases in fact, extractable parsed adjuncts do correspond to gold synchronous adjuncts, but the target parser fails to see their aligned counterparts as adjuncts. This again can reflect difficult cases for the adjunct/argument distinction, that are cumulated over both sides of the

Table 3.10: Relationship between label identity L across extractable parsed adjuncts/complements and their correspondance G to a gold synchronous pair: mutual information (in bits) and conditional probability.

	role	$\#l, g$	$\#l, \neg g$	$\#\neg l, g$	$\#\neg l, \neg g$	$I(L;G)$	$Pr(G\|L)$
penn → cand	ADJ	257	45	100	83	0.079	0.85
	CMP	172	482	73	279	0.003	0.26
turbo → cand	ADJ	281	60	113	173	0.145	0.82
	CMP	148	371	64	297	0.011	0.29
ccg → cand	ADJ	280	45	101	160	0.183	0.86
	CMP	153	377	90	274	0.002	0.29
cand → penn	ADJ	251	42	78	95	0.132	0.86
	CMP	173	478	55	249	0.006	0.27
cand → turbo	ADJ	273	52	56	85	0.138	0.84
	CMP	144	361	84	366	0.010	0.29
cand → ccg	ADJ	189	17	140	120	0.137	0.92
	CMP	112	231	116	496	0.017	0.33

data.

Measures for complements are expectedly low, and only a quarter to a third of synchronous complements correspond to gold synchronous arguments. We interpret the remaining cases as indicative of syntactic similarity between French and English.

3.5.3 Synchronous adjunction in translation data

We now apply our measures of empirical synchronous adjunction and complementization to different corpora. Are there any differences between corpora relating to these measures?

To answer this question, we measure synchronous adjunction for three different corpora: the Europarl corpus at large[3]; a News Commentary dataset[4]; and the Canadian Hansards[5]. Word alignments are trained for each dataset with the Berkeley Aligner (Liang et al., 2006). Measures are taken for all three English parse-based label sets, in order to see if there are consistent differences between corpora across the different label sets. Figures are reported in Table 3.11, showing

[3] release v7, http://www.statmt.org/europarl/

[4] http://www.statmt.org/wmt08/shared-task.html

[5] House and Senate training data, https://www.isi.edu/natural-language/download/hansard/

Table 3.11: Parse-based synchronous pairing in French-English corpora.

		manual	Europarl	News Com.	Hansards
penn → cand	ADJ	61.2	65.7	63.3	59.3
	CMP	69.8	67.7	64.2	66.7
cand → penn	ADJ	62.6	63.1	64.6	54.2
	CMP	72.8	70.8	69.7	70.8
turbo → cand	ADJ	52.9	53.7	52.8	47.8
	CMP	65.6	65.5	62.1	67.9
cand → turbo	ADJ	69.4	66.7	70.2	62.1
	CMP	60.5	60.7	59.4	61.9
ccg → cand	ADJ	54.2	55.1	54.1	49.4
	CMP	65.7	64.6	61.2	66.1
cand → ccg	ADJ	43.9	42.1	45.9	38.7
	CMP	45.3	46.4	45.7	47.2

that this is in fact the case for the Canadian Hansards, where adjunct alignment figures are consistently lower than for the Europarl or News-Commentary dataset.

Factoring the effect of sentence length on synchronous adjunction (see Figure 3.6) shows that the difference between the Hansards and the Europarl and News Commentary corpora plays for all sentence lengths, but is most pronounced in short to average-length sentences. The lesser difference between corpora in long sentences can be explained by the larger degree of compositionality in these sentences, while shorter sentences are more likely to contain, and be affected by non-compositional constructions, like idiomatic expressions. The lower degree of synchronous adjunction in the Hansards suggests that translations are less litteral in that data set. Conversely, we can assume that data from legal or technical domains would display a higher degree of adjunct alignment. This suggests that parse-based synchronous adjunction could be used for data selection to select more regular, compositional translations for training translation systems.

3.5.4 Summary

This section considered the relation between synchronous adjunction in experimental conditions—with parse-label mappings to identify adjuncts and complements, and word-alignment based translation equivalence—and gold synchronous adjunction, with manually identified and aligned adjuncts and arguments.

About two thirds of parsed adjuncts on the source side of extractable phrase pairs correspond to gold synchronous adjuncts. Figures increase by 10 to 15% if we do not require labels to match with the gold annotations, pointing to adjuncts

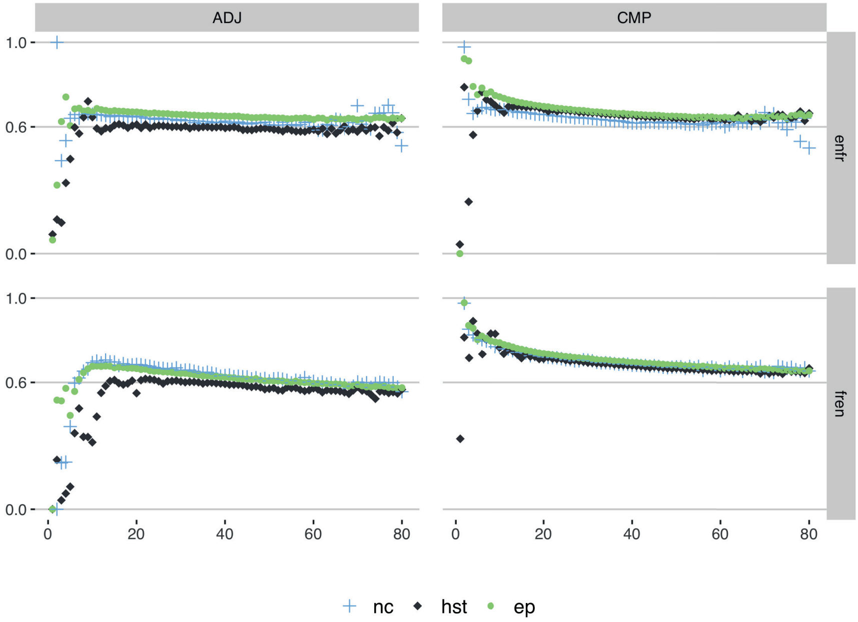


Figure 3.6: Parse-based synchronous adjunction/complementization as a function of sentence length, for the Pennconverter/Candito label mappings.

that are annotated as arguments in our corpus, and that are likely to constitute difficult cases for the adjunct/argument distinction.

Equating source-side and target side labels in extractable adjuncts provides us with a measure of synchronous adjunction in experimental conditions. This measure is informative of gold synchronous adjunction, as 80% to 90% of empirical synchronous adjuncts correspond to gold synchronous adjuncts. This is the case for only a quarter of complements, as they regroup semantically-motivated arguments and syntactically motivated complements.

When comparing synchronous adjunction and complementization in different French-English corpora, we find that synchronous complementization remains stable across corpora, while synchronous adjunction is noticeably lower for one of the corpora, the Canadian Hansards. Differences in synchronous adjunction with other corpora play mostly in short to average-length sentences, pointing to a larger degree of compositionality in longer sentences in general. We conclude that empirical synchronous complementization is mostly indicative of syntactic similarity, and synchronous adjunction of translation compositionality.

3.6 Conclusion

This chapter explored synchronous adjunction from a linguistic and empirical perspective. We studied synchronous adjunction in a manually annotated corpus, evaluated the effect of word alignments and parse-based labelling heuristics on measures of synchronous adjunction, and ended with a comparison of empirical synchronous adjunction in French-English translation corpora.

Our corpus study is based on French-English parliament proceedings from the Europarl corpus. Unaligned adjuncts and arguments reflect the nature of parliamentary language, which allows for locutions and a certain degree of freedom in translations. Even then, the translation data in this corpus are highly compositional in terms of adjunction, as 80% of adjuncts translate to adjuncts. We find that arguments behave similarly to adjuncts in translation data, differing only in the flexibility of adjuncts in switching contexts. Evaluating parse-label heuristics showed that parsers interpret many of our annotated arguments as modifiers. As arguments and adjuncts are synchronous to the same degree, and as they are affected by word alignments in the same way, one can conclude that the adjunct/argument distinction is not relevant for translation and for these modifiers. We will consequently interpret these modifiers as adjuncts in the rest of this work.

Word alignments and parse-based labelling heuristics affect experimental measures of synchronous adjunction. If one abstracts from labelling differences however, around 80% of parsed adjuncts in extractable phrase pairs correspond to a synchronous pair. We conclude from this that we can rely on the Direct Correspondence Assumption when applying adjunction in Hierarchical Phrase-Based SMT, as we will in the next chapter.

Experimental measures of synchronous adjunction show that 80% to 90% of synchronous modifiers correspond to gold synchronous adjuncts, abstracting from differences in labelling and adjunct attachment. These measures further appear to be indicative of translation compositionality, as the variations we observe between corpora are consistent across parse-based labelling heuristics, and affect mostly short and average-length sentences. In comparison, synchronous complementization appears to capture syntactic similarity, as it corresponds to only a quarter to gold synchronous arguments, and is stable across corpora.

Chapter 4

Adjunction for Hierarchical Phrase-Based SMT

Hierarchical translation models adopt a compositional view of translation as they model sentences and their translations through synchronous grammars. And while the grammar is constrained to the syntax of one side of the corpus in Syntax-Based models, Hierarchical Phrase-Based models are in principle only constrained by phrasal constraints on word alignments. Knowing how translation compositionality relates to monolingual syntax is important for structuring the search space in these models: linguistic information allows for better constraints and better performance. In this chapter, we look at adjunction and the role it plays in translation compositionality for Hierarchical Phrase-Based models. We show that adjunction forms a useful basis for extraction constraints in hierarchical models, not only because it extends their long-range capacity, but also because it effectively filters short-range phrase pairs. We show however that it is a closer adherance to linguistic recursion through constituency(-crossing) constraints that benefits Hiero most.

The work in this chapter is based on the following publications:

- Sophie Arnoult and Khalil Sima'an. *Modelling the Adjunct/Argument distinction in Hierarchical Phrase-Based SMT*. In DMTW 2015.
- Sophie Arnoult and Khalil Sima'an. *Factoring Adjunction in Hierarchical Phrase-Based SMT*. In DMTW 2016.

The work in (Arnoult and Sima'an, 2015) represents a first attempt to utilize the adjunct/argument distinction for Hierarchical Phrase-Based SMT. In that work, we used adjunct/argument labels on both sides of a French-English bitext to derive synchronous labels. This provided little or no improvement over Hiero. In (Arnoult and Sima'an, 2016), we proposed to utilize adjunction to selectively add long-range rules to the Hiero grammar and to generalize rules by factoring out adjuncts. This chapter complements that work with an analysis of the respective effects of standard Hiero constraints against adjunct-driven constraints, and

with a comparison of adjunct-based constraints to constraints based on syntactic complements or constituents.
I conducted the research for both works myself, under the guidance of Khalil Sima'an. Experiments were carried with an in-house grammar extracter built by Gideon Maillette de Buy Wenniger, and which I extended for the implementation of my own models.

Chapter Highlights

Problem Statement

- Hiero (Chiang, 2005) offers a compositional view of translation while being agnostic about linguistic notions guiding compositionality. The latter can however help guiding this model towards better translations.

Research Questions

- How does adjunction fit in asyntactic, compositional models of translation?
- Is adjunction a useful source of information for hierarchical phrase-based models?
- How does adjunction compares to argumenthood or constituency in this respect?

Research Contributions

- We propose an extension to Hiero that leverages on two properties of adjunction, long-distance dependencies and optionality, to extend the grammar with adjunct-based long-range rules and with rules gained by factoring out adjuncts.
- We analyze the effect of Hiero span-length constraints against adjunct-crossing constraints, and show that the latter not only extend the long-range expressiveness of the model but also effectively filter short-range phrase-based rules.
- We further analyze the effect of adjunct-based constraints against constraints based on syntactic complements or constituency, and show that it is a better modelling of recursion through constituency constraints that benefits the model the most.

4.1 Introduction

Hierarchical models of translation integrate reordering into their lexical translation model, and this imposes constraints on their expressivity as the complexity of reordering is exponential for SMT (Knight, 1999; Bisazza and Federico, 2016). In Hiero (Chiang, 2005), the reordering space is constrained by a limit on the decoding span, which is accompanied by a constraint on extraction spans. Longer fragments have to be concatenated monotonically by the application of so-called 'glue rules', unless reordering extensions are provided (Mylonakis and Sima'an, 2011; Huck et al., 2012).

This is problematic for complex reorderings spanning long distances. Consider for example the sentence pair in Figure 4.1. On the English side, the relative phrase *who have . . . industry* separates the subject *those workers* from the verb form *have shown*. As Hiero limits the length of extracted phrase pairs, this dependency cannot normally be captured, and the translation model may fail to translate *have shown* as a third person plural. A second issue is at play here, that of complex reordering. The limited extraction capacity of Hiero is not problematic when alignments between source and target are simple, as for the second adjunct (*to the solar power industry*) in Figure 4.1; Hiero can then capture and recombine fragments monotonically. When alignments are complex however, as with the permutation on the left in Figure 4.1—note also that this complex permutation is caused in part by the different placement of the relative phrase in the English and Dutch sentences—they can only be captured if their length falls within the extraction length limit.

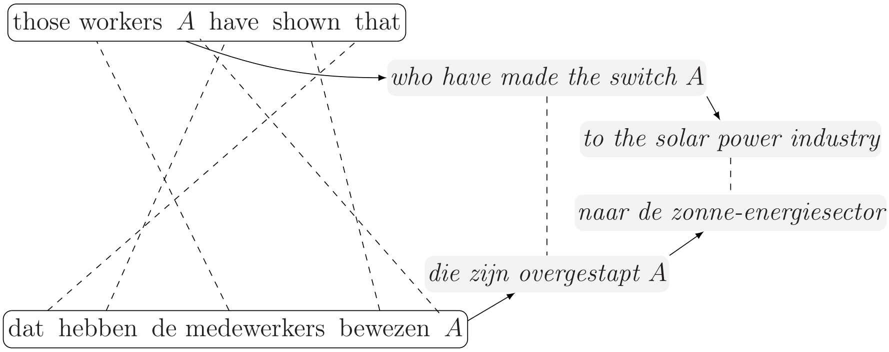

those workers *A* have shown that
who have made the switch A
to the solar power industry
naar de zonne-energiesector
die zijn overgestapt A
dat hebben de medewerkers bewezen *A*

Figure 4.1: Example sentence; adjuncts introduce long-distance dependencies and complex reorderings (here a 2-4-1-3 permutation).

Relaxing extraction constraints for phrase pairs that contain adjuncts would allow us to achieve two goals: capturing lexical (syntactic or semantic) dependencies that are obscured by intervening adjuncts; and capturing phrase pairs with complex underlying alignments that cannot be reconstructed otherwise by Hiero. In the example of Figure 4.1, this would mean that one can extract rules for each of the three fragments shown in the figure, and that the two rules rewriting to adjuncts in that example could be applied to other adjuncts, regardless of their length.

The notion that adjuncts introduce long-distance dependencies forms the linguistic motivation for Tree-Adjoining Grammars (Joshi and Schabes, 1997). Formally, adjunction was introduced by Joshi et al. (1975) in Tree-Adjoining Grammars (TAG), as an operation to model recursion. Factoring recursion allows for a compact grammar of initial trees where dependencies are kept local (Joshi and Schabes, 1997). TAG grammars are able to generate unseen adjunction patterns (Chiang, 2000). Shieber and Schabes (1990) introduced a synchronous variant of TAG, Synchronous Tree-Adjoining Grammar, that has been applied in Syntax-Based SMT (DeNeefe and Knight, 2009; Liu et al., 2011). We propose here to leverage on the role played by adjunction in recursion in the context of Hierarchical Phrase-Based SMT. By selectively relaxing extraction constraints around adjuncts, we leverage on some of the long-distance dependencies introduced by adjunction. Besides, we experiment with enriching the grammar by factoring out adjuncts to leverage on adjunct optionality.

The adjunct/argument distinction forms an interesting notion for linguistic enrichment of Hiero, as it is generic and pertains to the syntax-semantics interface. Syntactic enrichment of Hiero with phrase-structure labels (Zollmann and Venugopal, 2006) is in fact directed at improving syntactic cohesion on the target-side, regardless of synchronous validity. More semantically motivated sources of syntactic enrichment, with dependencies (Li et al., 2012) or CCG (Almaghout et al., 2011), still create large grammars with little or no guarantee of translation compositionality. Generic refinement methods like that of Maillette de Buy Wenniger and Sima'an (2013) directly target the synchronous nature of the data but are not linguistically informative. Semantically oriented annotations, such as predicate-argument structures (Xiong et al., 2012; Li et al., 2013) provide better guarantees of translation compositionality and have proved useful for English-Chinese. In particular, Li et al. (2013) combine a Predicate-Argument-Structure (PAS) reordering model, which they derive from projecting source-side PAS to the target side, with generic syntactic constituency constraints, which allow them to lift the standard length limit on extracted phrase pairs. In this work, we also use generic syntactic constraints to lift heuristic span-length constraints, but our focus is on adjunction and its generic syntactic/semantic value.

This chapter is organized as follows. Section 4.2 presents the experimental setup. The adjunct-based extraction model is presented in section 4.3; we show that the added rules improve performance for English-Japanese, and for

English-Chinese given further refinement with adjunct labels. Section 4.4 extends this model by leveraging on adjunct optionality to extract more rules, again showing gains for English-Japanese. Section 4.5 provides additional experiments for the analysis of adjunct-driven extraction, by balancing Hiero and syntactic constraints, and comparing adjunct to constituency constraints; we find that adjunct-based extraction constraints effectively filter uninformative short-range rules, while constituency-based constraints further improve performance for English-Japanese and English-Chinese. Section 4.6 concludes this chapter.

4.2 Experimental set-up

Adjunct annotations are obtained from dependency labels, essentially by mapping modifier labels to adjuncts, as explained in section 4.2.1. The data for experiments are described in section 4.2.2, experimental settings in section 4.2.3, and evaluation metrics in section 4.2.4.

4.2.1 Identifying adjuncts

Adjuncts are identified heuristically, based on dependency labels—we use the Turbo parser[1] for this work.

Dependents with a modifier label AMOD, NMOD or VMOD are mapped to adjuncts in principle, as well as dependents of enumerations and conjunctions. This follows from a dependency analysis that treats one of the conjuncts as the head, and conjunctions and other conjuncts as its dependents. To this end, the head-final representation employed by the Turbo parser is replaced by a nested one, as shown in Figure 4.2.1; to support the adjunct reading of enumerations and conjunctions, dependents with a punctuation label P are therefore also mapped to adjuncts.

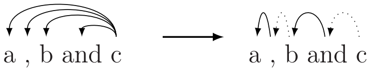


Figure 4.2: Modifying the representation of enumerations and conjunctions. Dotted lines represent adjuncts.

Function words are excluded even if they carry an adjunct-like label (AMOD, NMOD, VMOD and P). Leaves in the dependency tree with one of the following POS tags are identified as function words: DT, EX, IN, POS, MD, PRP, PRP$, RP, SYM, TO, WDT, WP, WP$, WRB, ..
Possessors in genitive constructions (dependents to the immediate left of a dependent with a POS part of speech) are also reinterpreted as arguments.

[1] http://www.cs.cmu.edu/~ark/TurboParser/

4.2.2 Data

Models were tested on four language pairs: English-Chinese, English-Japanese, English-Dutch and English-French.

The English-Chinese data were taken from the MultiUN corpus (Eisele and Chen, 2010), limited to sentences of up to 40 tokens. The development and test set consist each of 2000 sentences, drawn randomly without replacement from the corpus (after having removed English-side duplicates). Word alignments were trained on the rest of the corpus (ca. 5.6M sentence pairs). The language model was trained on the Xinhua section of the Chinese Gigaword corpus (LDC2003T09).

The English-Japanese data were taken from the NTCIR-8 patent translation task, using sentences between 4 and 50 tokens. The NTCIR-7 development sets were used for tuning, and the NTCIR-9 test sets (both English-Japanese and Japanese-English) for testing.

The English-Dutch data were taken from the Europarl corpus (v7). Development and test sets of 2000 sentence pairs each were extracted following the same method as for the English-Chinese data.

The English-French data were taken from the Europarl corpus (v7), limited to sentences of up to 40 tokens. We used the Europarl 2006 development and test sets.

For all language pairs except English-Chinese, word alignments were trained on the training corpus, the language model was trained on the target side of the training corpus. For all language pairs, the translation models were trained on 500k random sentence pairs from the training corpora. For English-Chinese, translation models were additionally trained on 2M sentence pairs, randomly selected from the training corpus. Table 4.1 summarizes the sizes and average sentence length of the different data sets.

Table 4.1: Data-set sizes

		train		dev	test
fr	sentences	500k		2.0k	2.0k
	avg. tokens	20.6		29.0	29.7
nl	sentences	500k		2.0k	2.0k
	avg. tokens	27.4		27.6	27.1
zh	sentences	500k	2M	2.0k	2.0k
	avg. tokens	22.5	22.5	22.7	22.6
ja	sentences	500k		2.1k	2.9k
	avg. tokens	26.9		26.3	26.5

4.2.3 Experimental settings

Word alignments were trained with GIZA++ using 'grow-diag-final-and' symmetrization (Och and Ney, 2003) for all language pairs except English-Japanese, for which the Berkeley aligner was used (Liang et al., 2006).

The language model is a 4-gram language model with modified Kneser-Ney smoothing, trained with KenLM (Heafield et al., 2013).

All models use an extended set of dense features (not counting adjunction features), following Maillette de Buy Wenniger and Sima'an (2013). Feature weights are tuned with MIRA (Cherry and Foster, 2012), for 20 iterations. Tuning was performed three times to compensate for tuner instability (Clark et al., 2011).

Decoding was performed with Joshua (Li et al., 2009), relaxing the decoding span to 100 tokens for all models except the Hiero baseline. This allows to apply hierarchical rules that may span the entire sentence in the case of the extended models.

4.2.4 Evaluation

Evaluation was performed using Beer[2]. We report BLEU (Papineni et al., 2002), BEER (Stanojević and Sima'an, 2014) and TER (Snover et al., 2006) scores computed over three optimization runs with p values computed by approximate randomization Clark et al. (2011). We add a reordering-oriented metric in the form of mean LR-KB1 values (Birch and Osborne, 2011), giving equal weight to the Kendall τ and BLEU-1 components.

The grammar sizes that are reported in a number of experiments concern grammars filtered by the test set.

4.3 Adjunct-driven extraction

Hiero limits phrase spans for rule extraction through a *max-phrase-length* constraint (of typically 10 tokens). This limit is needed to restrict the number of extractable phrases, that may grow exponentially with sentence length. But this limit also rests on an assumption that lexical dependencies and reorderings are sufficiently captured by short-range translation rules.

We show in this section how one can use adjuncts to selectively relax extraction constraints, and that the rules gained in this manner provide new, useful information for English-Japanese, English-Chinese, but also for English-French to a lesser extent.

The extraction constraints are presented in section 4.3.1. The resulting model can be augmented with minimal labelling, as explained in section 4.3.2, and/or

[2] `https://github.com/stanojevic/beer`

with extraction features, which are presented in section 4.3.3. Experiments are presented in section 4.3.4.

4.3.1 Extraction constraints

We use adjunction as a guide for extending rule extraction for larger phrases. As with Hiero, extraction and rewriting are unconstrained for all phrases under *max-phrase-length*. For larger phrases, extraction is subjected to three constraints: *max-effective-length*, *non-adjunct-crossing*, and *max-target-symbols*. Rewriting in larger phrases is also constrained to respect adjunct boundaries, by applying the *non-adjunct-crossing* constraint and a *non-adjunct-masked* constraint. For the application of extraction constraints, contiguous adjunct sequences are regrouped and treated as single adjuncts.

max-effective-length

Hiero applies a *max-phrase-length* limit on extraction spans. Let us replace this constraint by *max-effective-length*, where the effective length is defined as the phrase length *less* the token count of the adjuncts contained in it (disregarding adjuncts embedded in other adjuncts). Let a phrase ϕ, that contains $\alpha_0..\alpha_n$ top-level adjunct subphrases, its effective length $\lambda(\phi)$ is:

$$\lambda(\phi) = \text{len}(\phi) - \sum_{i=0}^{n} \text{len}(\alpha_i)$$

Note that adjunct phrases have zero effective length under this definition (they are always extractable). In practice, *max-effective-length* is set to the same value as *max-phrase-length*.

non-adjunct-crossing

This constraint prevents the extraction of larger phrases that cross adjuncts, or groups of adjuncts. This forces rewriting to an adjunct group as a whole. When rewriting from an adjunct group, one only forbids adjunct crossings, allowing rewriting to sub-groups.

non-adjunct-masked

This rewriting constraint prevents rewriting to phrases that are contained by adjuncts in a given phrase pair: combined with the *non-adjunct-crossing* constraint, this forces rewriting to phrase pairs that contain or are disjoint from adjuncts contained in a phrase pair.

max-target-symbols

As span length is relaxed, the number of right-hand-side symbols on the target side may increase even if the number of source-side symbols is restricted. This happens in particular with idiomatic expressions, but can also be a sign of noisy data. To keep the number of target-side symbols close to the number of source-side symbols, we restrict the number of target-side symbols to *max-phrase-length.*

Table 4.2 shows a possible derivation for the example of Figure 4.1. Extracting rules from larger phrases allows to capture long-range dependencies and reorderings inaccessible to Hiero. While rule r_1 is likely in fact to be learned by Hiero in a different context, rule r_2 displays a pattern (extraposed modifier in the Dutch sentence but not in the English sentence) that is only likely to occur with a long modifier.

Table 4.2: Example rules for the sentence pair of Figure 4.1

r_1	X → ⟨ X that , dat X ⟩
r_2	X → ⟨ those workers $A^{[1]}$ have shown , hebben de medewerkers bewezen $A^{[1]}$ ⟩
r_3	A → ⟨ $X^{[1]}$ made the switch $A^{[2]}$, $X^{[1]}$ overgestapt $A^{[2]}$⟩
r_4	X → ⟨ who have , die zijn ⟩
r_5	A → ⟨ to the solar power industry , naar de zonne-energiesector ⟩

4.3.2 Labelling

To further guide the model, one can apply labelling to distinguish adjuncts from other phrases. We experimented with two schemes. In the first scheme, rules and rule gaps receive a label `A` when they correspond to a source adjunct, and `A`x when they correspond to adjunct sequences of size x; a default label is used in other cases (see Table 4.2).

In the second scheme, adjunct-sequence labels are simplified to `A`, while their size x appears in the following feature:

$$f_x = e^{1-x} \tag{4.1}$$

For other rules (adjuncts and other phrase pairs), f_x is taken to be 1.

4.3.3 Features

The model uses two rule features to distinguish larger phrase pairs from Hiero-extractable phrase pairs: a *long-distance* feature, corresponding to the probability estimate that a rule was extracted from a larger phrase pair (exceeding Hiero's *max-phrase-length*); and an *adjunct-crossing* feature corresponding to the

Table 4.3: Effect of max decoding span d_{max} on Hiero

	d_{max}	BLEU	BEER	TER	LR-KB1
en-ja	10	31.0	12.3	57.1	0.287
	100	**33.5**	**14.1**	**54.8**	0.305
en-zh	10	21.8	11.2	63.8	0.265
	100	21.7	11.2	64.4	0.262
en-zh (2M)	10	23.3	12.6	61.8	0.273
	100	23.4	12.6	62.0	0.268
en-fr	10	32.9	22.2	53.9	0.390
	100	32.7	22.0	54.3	0.388
en-nl	10	27.5	16.4	59.5	0.300
	100	27.4	16.3	59.7	0.300

Boldface marks significant improvement with $p \leq 0.05$

probability that a rule was extracted from a (shorter) phrase pair violating the *non-adjunct-crossing* constraint.

4.3.4 Experiments

To see the effect of extracting long-range rules, one needs to also relax the decoding span, especially for rules that involve complex reorderings. To allow for a fair comparison with a Hiero baseline, we therefore relax the decoding span for both Hiero and our adjunct-based extraction model.

We first evaluate the effect of relaxing the *decoding span* for Hiero, and show that this in itself improves English-Japanese translation. We then present experimental results on adjunct-based extraction, and complement these experiments with labelling experiments.

Relaxing the decoding span

Hiero limits the search space for the decoder by limiting decoding span. This limit must be relaxed to allow for the application of the long-range extraction rules gained by adjunct-based extraction. Rather than following the same approach for decoding as for extraction—by relaxing decoding span selectively based on adjuncts—, we opt here for relaxing the decoding span to a large value for all models.

Table 4.3 compares system performance for Hiero with a relaxed decoding span of 100 against the default decoding span of 10. Relaxing the decoding span has a large beneficial effect for English-Japanese.For this language pair, where

Table 4.4: Effect of extraction constraints: standard Hiero constraints vs. adjunct-based constraints (`H/adjs`)

		BLEU	BEER	TER	LR-KB1
en-ja	`Hiero`	33.5	14.1	54.8	0.305
	`H/adjs`	**34.2**	**14.8**	**53.6**	0.309
en-zh	`Hiero`	21.7	11.2	64.4	0.262
	`H/adjs`	**21.9**	11.4	**64.1**	0.261
en-zh (2M)	`Hiero`	23.4	12.6	62.0	0.268
	`H/adjs`	23.7	12.7	62.0	0.271
en-fr	`Hiero`	32.7	22.0	54.3	0.388
	`H/adjs`	**33.0**	22.1	**53.9**	0.390
en-nl	`Hiero`	27.4	16.3	59.7	0.300
	`H/adjs`	27.5	16.3	59.7	0.299

Boldface marks significant improvement with $p \leq 0.05$

verb-object order is notably inverted, relaxing the decoding span is necessary to correctly translate longer sentences; we see here that locally-learned rules and the reorderings they capture are applicable to long-range reorderings. For the other language pairs, locally-learned rules do not generalize well to larger contexts, or at least not enough to outweigh the increase in errors due to misapplied rules.

Decoding span is relaxed for all models and language pairs in the remainder of this chapter, allowing us to focus on extracted rules.

Adjunct-driven extraction constraints

We now look at the effect of selectively relaxing extraction spans around adjuncts. As Table 4.4 shows, adjunct-based extraction provides a further boost to English-Japanese performance, and more modest improvements for the other language pairs.

While English-French and English-Dutch are challenging language pairs for Hiero, as their reorderings are largely monotonic or hard to capture in the case of English-Dutch (split) verb reorderings, the modest improvements for English-Chinese are perhaps more surprising. We see next that this language pair benefits most from further guidance.

Guiding the model with features and labelling

We now look at the effect of the features and labelling schemes presented in sections 4.3.3 and 4.3.2. As Table 4.5 shows, English-Chinese benefits most from

Table 4.5: Effect of features (*f*) and/or labelling (*A* or *Ax*) on a model with adjunct-driven extraction constraints (`H/adjs`)

		BLEU	BEER	TER	LR-KB1
en-ja	`H/adjs`	34.2	14.8	53.6	0.309
	+*f*	**34.4**	14.8	**53.4**	0.309
	+*Ax*	34.2	*14.6*	53.7	0.309
	+*f*,*A*	34.3	*14.6*	53.4	0.310
en-zh	`H/adjs`	21.9	11.4	64.1	0.261
	+*f*	22.0	*11.3*	64.1	0.260
	+*Ax*	**22.1**	11.4	64.3	0.261
	+*f*,*Ax*	**22.3**	11.4	63.9	0.263
	+*f*,*A*	**22.3**	11.4	64.3	0.261
en-zh (2M)	`H/adjs`	23.7	12.7	62.0	0.271
	+*f*,*A*	23.9	12.8	62.2	0.270
en-fr	`H/adjs`	33.0	22.1	53.9	0.390
	+*f*,*A*	*32.7*	*21.9*	*54.2*	0.389
en-nl	`H/adjs`	27.5	16.3	59.7	0.299
	+*f*,*A*	27.3	16.2	59.7	0.299

Boldface and italics mark significant positive, respectively negative difference with $p \leq 0.05$

helping the model discriminate adjunct rewritings from other rewritings and rules.

Examples

Inspection of output translations shows several cases of improved lexical selection and reordering for French. For instance in Table 4.6, adjunct-based extraction allows to capture the dependency between *enthusiasm* and *wane* in the first example, and the dependency between *outpost* (*retranchement*) and *of europe* in the second example. In these cases however, one also sees that Hiero is led astray by the increased decoding span.

Summary

Using adjuncts to extract long-range rules for Hiero is beneficial for reordering-intensive language pairs like English-Japanese and English-Chinese.

Adjunct-based constraints are particularly useful for English-Japanese. While long-range inversion in that language pair can be addressed by relaxing the decoding span, the long-range rules gained by adjunct-based extraction further benefit the model.

Table 4.6: Example translations

src	the problem is that , if you set a date , there is a danger that the *enthusiasm* for reform in these countries will *wane* .
Hiero	le problème est que , si vous **wane** fixer une date , il y a un risque que l' *enthousiasme* de réforme dans ces pays .
H/adjs	le problème est que , si vous fixer une date , il y a un risque que l' *enthousiasme* de réforme dans ces pays *diminue* .
src	because of its geopolitical position as the last *outpost of europe* , at the crossroads with the middle east and north africa , the importance of malta goes far beyond its geographical size and its small population.
Hiero	en raison de sa position en tant que dernier *retranchement* géopolitique , au carrefour avec le moyen-orient et l' afrique du nord , l' importance de malte va bien au-delà de sa taille et sa petite population géographique **de l' europe** .
H/adjs	en raison de sa position en tant que dernier *retranchement* géopolitique *de l' europe* , à la croisée des chemins avec le moyen-orient et l' afrique du nord , l' importance de malte va bien au-delà de sa taille et sa petite population géographique .

Long-range rules gained by adjunct-based extraction also benefit English-Chinese translation, albeit modestly. Labelling and features improve performance, suggesting that short-range and long-range rules are not directly distinguishable for the system.

Finally, adjunct-based extraction yields some improvement for English-French, but mostly because it allows to correct mistakes made by Hiero when increasing the decoding span.

4.4 Factoring out adjuncts

The grammars extracted with adjunct-based constraints are only loosely related to adjunction: rewriting is not constrained beyond adjunct-boundary violations (rewriting is phrase-based), and adjuncts are not excised to extract initial and auxiliary trees (Chiang, 2000). In this section, we consider adjunct optionality as a source of generalization for the grammar: we assume that source-side adjuncts are optional, and that they align to optional adjuncts on the target side, and we derive new extraction phrase pairs by excising these adjunct pairs. This follows (Arnoult and Sima'an, 2012), where this idea is applied to a (non hierarchical) phrase-based model. We leverage the hierarchical nature of Hiero by applying substitution in these generalized phrase pairs.

Compared to STAG, our grammar is thus not more compact, instead it is augmented with *derived* rules that are obtained by generalizing adjunction patterns in the training data.

4.4.1 Model

The adjunct-factoring model builds upon the adjunct-based extraction model of section 4.3. For each extracted phrase pair in the training data, one first extracts rules by standard substitution. Besides, for each adjunct contained in the phrase pair, one instantiates a copy of the extraction phrase where the adjunct is *blind*: the adjunct blocks the extraction of overlapping gaps, and its yield is excised from the rule. Rules are then extracted by phrase substitution from this copy.

The combinations of adjuncts that can be excised from a phrase grow exponentially with the number of adjuncts in the phrase. Even if this number remains small in general, adjunct factorization is applied to all phrases, in an extraction space that is already increased by extending extraction-phrase spans. Grammar size increase is contained accordingly by excising only one adjunct at a time in adjunct-group phrases, and one adjunct group (or stand-alone adjunct) at a time in other phrases. Besides, we modify the source parses for enumerations by nesting enumeration tails, as explained in section 4.2.1.

Table 4.7 shows some of the resulting rules for the example of Figure 4.1.

Table 4.7: Rules added by adjunct factorization

r_6	X	→	$\langle$ those workers have shown , hebben de medewerkers bewezen $\rangle$
r_7	X	→	$\langle$ those $X^{[1]}$ have shown , hebben de $X^{[1]}$ bewezen $\rangle$
r_8	X	→	$\langle$ those workers have $X^{[1]}$, hebben de medewerkers $X^{[1]}$ $\rangle$
r_9	X	→	$\langle$ $X^{[1]}$ have $X^{[2]}$, hebben $X^{[1]}$ $X^{[2]}$ $\rangle$

Adjunct factorization allows to create new lexical rules, like r_6, and new hierarchical rules by substitution, like rules r_7 to r_9. Substitution is slightly restricted with regard to Hiero, as it is allowed only to the left or right of blind adjuncts.

4.4.2 Experiments

Table 4.8 presents results for the adjunct factorization model. Next to the usual evaluation metrics, we report the size of the grammar filtered by the test set: $\mathcal{G}_{src}$ refers to the number of source-side rule types, and $\mathcal{G}$ to the number of rule types. Factorizing out adjuncts appears useful for English-Japanese, but not so for English-Chinese, where the added rules generally decrease performance. In English-Chinese, factoring out adjuncts increases the number of source rules applicable to the test set ($\mathcal{G}_{src}$) by only 5% (from 0.8M to 0.86M rules for the smaller training set, and from 1.15M to 1.2M rules for the larger one), against 16% for English-Japanese. Adjuncts may align less well across English and Chinese (possibly because of unaligned tokens) than they do between English and Japanese,

Table 4.8: Experimental results for the adjunct-factorization model

		$\|\mathcal{G}_{src}\|$	$\|\mathcal{G}\|$	BLEU	BEER	TER	LR-KB1
en-ja, 500k	`H/adjs`	1.00	7.25	34.2	14.8	53.6	0.309
	+*opt*	1.16	12.8	**34.5**	14.9	**53.2**	0.311
	+*opt,f,A*	2.82	27.9	**34.6**	**15.0**	**53.2**	0.312
en-zh, 500k	`H/adjs`	0.80	7.86	21.9	11.4	64.1	0.261
	+*opt*	0.86	10.6	*21.3*	*11.0*	*64.5*	0.259
	+*opt,f,A*	2.15	23.5	*21.5*	*11.1*	**63.7**	0.262
en-zh, 2M	`H/adjs`	1.15	23.9	23.7	12.7	62.0	0.271
	+*opt*	1.20	31.9	*23.4*	12.6	61.7	0.272
	+*opt,f,A*	3.18	70.5	23.6	12.6	**61.5**	0.273

Boldface and italics mark significant positive, respectively negative difference w.r.t `H/adjs` with $p \leq 0.05$. Grammar sizes are in million rules.

resulting in less usable options on the source side, and accordingly less usable options on the target side.

4.4.3 Summary and future work

Excising adjuncts and their aligned counterpart amounts to creating rules that rewrite to the empty string, which pushes the independence assumption taken by SCFG to the extreme. While this assumption yields direct improvement for English-Japanese, results for English-Chinese are negative. This may be due to a lesser alignment of adjuncts (or to noisy word alignments in general) in the English-Chinese data, leading to relatively more ungrammatical target rules.

An interesting alternative for modelling adjunct optionality would be to employ abstract adjunction rules, rewriting to an adjunct nonterminal and a default X nonterminal, allowing for left and right reorderings on the source and target side. The limited number of adjuncts in the data, possibly combined with a dedicated adjunction-site nonterminal, should largely limit the risk of spurious ambiguity normally associated with abstract rules in Hiero, while splitting adjunct nonterminals into leftwise and rightwise target-side reorderings, following the lexicalized reordering SCFG of Mylonakis and Sima'an (2010), would capture adjunct reordering preferences. Such a model would not only allow to model adjunct optionality, but adjunction as a formal operation, similarly to applications of STAG in Syntax-Based models (DeNeefe and Knight, 2009; Liu et al., 2011).

4.5 Balancing constraints

Adding long-range rules that abstract over adjuncts is beneficial for reordering-intensive language pairs like English-Chinese and English-Japanese. In this section, we consider two questions raised by this improvement. First, what is the relative contribution of the added long-range rules to the grammar and the extraction constraints that underlie them? Does performance simply improves from a larger grammar? To answer this question, we study in section 4.5.2 the effect of constraining extraction to adjunct-based rules only, and the effect of extracting short-range adjunct-based rules only. We find that adjunct-based constraints are central to performance in English-Japanese and to a lesser extent in English-Chinese, but that English-French also benefits from short-range adjunct-based constraints.

Secondly, we ask in section 4.5.3 whether adjuncts provide a specific advantage over constituents for extraction constraints. We find that this is in fact the case for English-French, but not for English-Japanese and English-Chinese, where constituency-based constraints provide the best performance.

Before going into these questions, we first introduce simpler adjunct constraints in section 4.5.1.

4.5.1 Simplifying extraction constraints

The models proposed above employ a complex set of rewriting constraints, as specific constraints apply to adjunct groups. We remove adjunct-group constraints here and replace them by simple adjunct-boundary constraints. Extracted phrase pairs in the simplified `H/adj` model satisfy *at least one* of the following constraints:

- the source phrase spans up to the length allowed by Hiero (*max-phrase-length*);

- the source phrase does not cross adjunct boundaries, and its *effective length* is within limit of *max-phrase-length*.

As shown in Table 4.9, the simplified extraction constraints lead to a slight increase in grammar size and while improving translation performance[3].

Simplifying adjunct-rewriting constraints provides a modest improvement for English-Chinese, and a significant one for English-Japanese. We use these simplified constraints in the remainder of this chapter.

[3] Implementing new extraction constraints exposed a bug that affected the limit on the number of target-side symbols, reducing it for hierarchical rules. This results in differences in scores for `Hiero` and `H/adjs` with regard to sections 4.3 and 4.4, the only notable difference being a BLEU jump from 34.2 to 34.6 after correction.

Table 4.9: Adjunct-based extraction with adjunct-group constraints (`H/adjs`) vs. simplified constraints (`H/adj`)

		$\lvert\mathcal{G}$	BLEU	BEER	TER	LR-KB1
en-ja	`Hiero`	6.71	33.4	14.1	54.7	0.303
	`H/adjs`	8.42	**34.6**	**15.0**	53.5	0.311
	`H/adj`	9.28	**35.0***	**15.2***	53.2	0.314
en-zh	`Hiero`	6.99	21.8	11.1	64.5	0.261
	`H/adjs`	8.12	21.9	11.2	64.5	0.260
	`H/adj`	9.48	**22.0**	11.4	**64.2**	0.262
en-fr	`Hiero`	22.2	32.8	22.0	54.1	0.389
	`H/adjs`	27.1	32.9	22.0	54.1	0.389
	`H/adj`	33.1	32.9	21.9	54.2	0.388

Boldface marks significant difference with $p \leq 0.05$ w.r.t. `Hiero`, and an asterisk significant difference w.r.t. `H/adjs`.
Grammar sizes are reported in millions.

4.5.2 Contribution of Hiero and adjunct-constrained rules

We now turn to a comparative analysis of span-length and adjunct constraints, so as to evaluate the relative contribution of syntactically constrained, short or long-range rules compared to syntactically unconstrained, short-range (Hiero) rules. We consequently compare four model variants:

- `Hiero`: constrained span length
- `H/adj`: either span-length or adjunct constraint
- `adj`: adjunct constraint only
- `H+adj`: both span-length and adjunct constraints

Table 4.10 reports grammar size (filtered by the test set) and model performance for these variants. Comparing `H+adj` to `Hiero` shows that adjunct constraints are useful in the short range: removing rules that violate adjunct boundaries decreases grammar size by 30% to 60%, while bringing about a significant improvement in performance for English-Japanese, and a modest one for English-Chinese and English-French.

Comparing `adj` to `H+adj` shows that adjunct constraints are useful in the long range too, at least for English-Japanese and English-Chinese: lifting the span constraint increases grammar sizes by 30% to 50%, and provides another significant improvement for English-Japanese, and a modest improvement for English-Chinese. Performance decreases slightly for English-French, as the lesser degree of reordering limits the effect of the added long-range rules.

Table 4.10: Contribution of constrained vs. unconstrained and short-range vs. long-range rules

		$\lvert\mathcal{G}\rvert$	BLEU	BEER	TER	LR-KB1
en-ja	`Hiero`	6.71	33.4	14.1	54.7	0.303
	`H+adj`	4.79	**34.1**	**14.4**	**54.1**	0.308
	`adj`	6.15	**34.9**	**15.1**	**53.0**	0.315
	`H/adj`	9.28	**35.0**	**15.2**	**53.2**	0.314
en-zh	`Hiero`	6.99	21.8	11.1	64.5	0.261
	`H+adj`	4.17	21.8	11.2	**63.7**	0.263
	`adj`	5.72	**22.0**	11.3	**63.6**	0.263
	`H/adj`	9.48	**22.0**	11.4	**64.2**	0.262
en-fr	`Hiero`	22.2	32.8	22.0	54.1	0.389
	`H+adj`	8.79	**33.0**	21.9	**53.9**	0.389
	`adj`	13.3	32.9	21.9	54.1	0.388
	`H/adj`	33.1	32.9	21.9	54.2	0.390

Boldface marks significant difference with $p \leq 0.05$ w.r.t Hiero. Grammar sizes are in millions.

Comparing `H/adj` to `adj` confirms that short-range rules that violate adjunct constraints are not useful on the whole: they increase grammar size by 50% to 150%, while hardly affecting performance.

The improvement brought by the `H/adj` model therefore does indeed result from the addition of long-range rules, but we also see that syntactically unconstrained, short-range rules actually bring little benefit. For language pairs with little reordering such as English-French, it is sufficient to extract short-range, adjunct-based rules; for reordering-intensive language pairs like English-Japanese and English-Japanese, there is additional benefit to relaxing span constraints to capture long-range rules.

4.5.3 Relevance of adjunct-based constraints

Relaxing extraction constraints around adjuncts is based on the assumption that adjuncts introduce long-distance dependencies. We are not considering linguistic extraction phenomena however, but plain syntactic modification. In the long-range, we may find that arguments, and in general syntactic complements, are also involved in relevant dependencies and reorderings. We can therefore ask how constraints based on syntactic complements, or more generally constituents, compare to adjunct-based constraints.

To this effect, we compare models with adjunct-based, complement-based and constituent-based extraction, `adj`, `cmp` and `con` respectively. We obtain

Table 4.11: Comparing adjunct, complement and constituency constraints

		$\|\mathcal{G}\|$	BLEU	BEER	TER	LR-KB1
en-ja	`adj`	6.15	34.9	15.1	53.0	0.315
	`cmp`	5.88	35.0	15.2	53.0	0.314
	`con`	4.71	**35.2**	**15.3**	**52.8**	0.316
en-zh	`adj`	5.72	22.0	11.3	63.6	0.263
	`cmp`	5.58	22.0	11.4	63.5	0.266
	`con`	4.12	22.2	11.4	**63.0**	0.267
en-fr	`adj`	13.3	32.9	21.9	54.1	0.389
	`cmp`	12.7	**32.7**	21.8	**54.4**	0.388
	`con`	7.42	**32.7**	21.9	54.1	0.388

Boldface marks significant difference w.r.t. `adj` ($p \leq 0.05$). Grammar sizes are in millions.

constituent and complement annotations from dependency parses, labelling all non-adjuncts as complements. Table 4.11 shows that constituent-based extraction performs better for reordering-intensive language pairs (English-Chinese and English-Japanese), while adjunct-based extraction is still preferable for English-French. While this supports the idea that abstracting out adjuncts allows to capture more interesting lexical dependencies than abstracting out complements, complement-based constraints may also block relevant phrasal chunks . In the example of Figure 4.1, complement-based constraints forbid in fact the extraction of phrasal rules like *who have X*, as the verb phrase *have made the switch ...* is seen as a constituent by the parser.

4.6 Conclusion

In this chapter, we have exploited adjunct properties for Hiero in two ways: to guide the extraction of long-range rules, and to derive more rules by factoring out adjuncts.

We first showed that adjuncts are useful for guiding the extraction of long-range rules, increasing the performance of Hiero for reordering-intensive language pairs like English-Chinese and English-Japanese, and also English-French to a lesser extent.

Pushing adjunct optionality through to derive new phrase pairs for rule extraction however proved useful for English-Japanese, but not for English-Chinese, as the model creates too many useless target rules. Abstract adjunction rules would likely be more efficient for modelling optionality: while abstract rules are a source of spurious ambiguity in Hiero, a limited number of source phrases can form adjuncts, preventing spurious ambiguity to a large degree. One could simply augment the Hiero grammar with adjunction rules and adjunct labels to account for possible reorderings, and possibly an adjunction-site label to prevent spurious ambiguity from leftmost or rightmost right-hand-side rewritings. Such a grammar would constitute a full-fledged Synchronous Phrase-Based Adjunction Grammar (SPAG).

Separating the relative contributions of syntactic and length constraints reveals that adjunct-based extraction constraints not only contribute useful long-range rules for reordering-intensive language pairs, but also that they effectively filter short-range rules, and in some cases can advantageously replace span-length constraints. We found in fact that short-range, hiero-extractable rules that violate adjunct boundaries tend to either harm or provide no benefit for system performance. These results are consistent with the work of Cherry (2008) on phrase cohesion. As Li et al. (2013) also observe, the negative results obtained by Koehn et al. (2003) with syntactic constraints concern extremely strict constraints, as phrase pairs were restricted to full constituents. Our results confirm that well-chosen syntactic constraints are in fact important in Hierarchical Phrase-Based SMT.

Comparing adjuncts to syntactic complements or constituents in general further showed that constituency-based constraints are best for structuring long-range reorderings for Hiero, with little or no difference between adjuncts and complements. We found that adjunct-based constraints were best for English-French, as enforcing other constituency-based constraints blocks useful phrasal chunks.

Comparing lexical metrics like BLEU and reordering-oriented metrics like BEER and LR-KB1 shows that our models improve both reordering and lexical selection for English-Japanese, while improvement in English-Chinese seems to be mostly lexical. In the next chapter, we turn to reordering models and to the role that adjuncts play in reordering.

Chapter 5

The role of adjunction in reordering

Reordering is integrated in the translation model in Hierarchical Phrase-Based SMT, and it is guided lexically as rewriting rules combine lexical terminals and abstract nonterminals. Reordering models in Phrase-Based SMT also condition on source and target lexical information. In contrast, reordering models used for preordering are trained to reorder based on source-lexical information and word alignments alone. In this work, we show that core source-syntactic information, in the form of adjuncts or constituents, can be advantageously used to improve reordering models. To elicit the role played by adjuncts as a category in reordering proper, we use adjuncts to guide the latent reordering grammar of Stanojević and Sima'an (2015). We combine latent reordering nonterminal splits with adjunct-based splits, and show that this improves preordering for English-Japanese, and English-Chinese to a lesser extent. Compared to syntactic arguments, adjuncts appear to be directly informative to reordering, and their role appears to be local to rules.

I performed the research myself, under the guidance of Khalil Sima'an. The code for this work is based on the code of Miloš Stanojević. I implemented the extensions for this work, and performed all experiments myself.

Chapter Highlights

Problem Statement

- Reordering is a crucial aspect of SMT. For linguistic models of reordering, the challenge lies in determining which linguistic aspects of a source sentence explain its reordering in a target sentence.

Research Questions

- Are adjuncts informative for reordering models?
- What is the part played by adjuncts in reordering compared to other constituents?

Research Contributions

- We extend the latent reordering grammar of Stanojević and Sima'an (2015) with adjunct-based splits to make explicit the relation between adjunction and reordering.
- We present experiments with refined hard and soft splits, where dependency information is used as an external signal to learning.
- We show that adjuncts are informative as a whole for reordering, while their contribution to reordering appears to be local to rules.

5.1 Introduction

We have seen in the previous chapter that adjunction and constituency form useful guides for translation modelling in hierarchical phrase-based models, notably as they allow to capture long-range dependencies and increase the model's expressiveness in the long-range. We turn in this chapter to the role played by adjuncts in reordering proper. We hypothesize that adjuncts contribute largely to reordering, as they tend to be preserved in translation, and as they engage in specific reordering patterns at different levels in a sentence. As Figure 5.1 shows for instance, adjectives like *asylum* precede nouns in English as in Chinese, while larger adnominal modifiers like *to asylum procedures* follow the noun in English but precede it in Chinese.

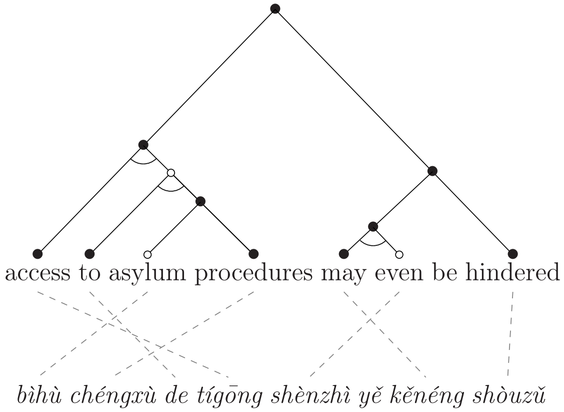

access to asylum procedures may even be hindered
bìhù chéngxù de tígōng shènzhì yě kěnéng shòuzǔ

Figure 5.1: A simple reordering between English and (Mandarin) Chinese. Permutations in this example are either monotone or inverted (arcs). Adjuncts (white nodes) engage in specific reordering patterns and provide core syntactic information for reordering.

Constructions involving adjunction concern about half of the word-order features listed in the World Atlas of Language Structures (Dryer and Haspelmath, 2013). As shown in Table 5.1, adjuncts appear to contribute the most to reordering between English on one hand and Chinese and Japanese on the other hand, either because they are reordered with regard to their head, or because they are reordered internally. In particular, oblique objects and relative phrases can span long distances, making them informative for the reordering of large fragments.

This makes adjuncts an interesting source of information for preordering models that only rely on source information and word-aligned data to perform reordering. For models that use source syntax (Xia and McCord, 2004; Genzel, 2010; Lerner and Petrov, 2013), it is interesting to determine which aspects of syntax are informative for reordering, also because rule-based preordering models (Collins et al., 2005; Isozaki et al., 2010b) can be hard to beat. Zhou et al. (2019) for instance use the head-finalization rules of Isozaki et al. (2010b) to reorder English sentences to Japanese and create additional training data for low-resource Neural MT. Preordering models that do not use syntax (Tromble and Eisner, 2009; DeNero and Uszkoreit, 2011; Neubig et al., 2012; Stanojević and Sima'an, 2015) also do not depend on syntactic resources and are not hampered by their limitations—as Khalilov and Sima'an (2012) show with their study on the effect of oracle preorderings on back-end translation, source parse trees severely limit the benefit of preordering. These models are however charged with inducing from lexical information the syntax that is relevant for reordering. We show here that core syntactic information, in the form of adjuncts or constituents, can be

Table 5.1: Main word-order features from the WALS database and their values in English, German, Mandarin Chinese and Japanese. We distinguish features that involve an adjunct and its governor, from features that involve the dependent and head of an adjunct (adjunct internal). Boldface marks a difference in word order with regard to English, whereas dashes mark an absence of dominant word order.

Feature	English	German	Chinese	Japanese
order between adjunct dependent and governor				
oblique, object and verb	VOX	**-**	**XVO**	**XOV**
genitive and noun	-	**N-gen**	**gen-N**	**gen-N**
relative clause and noun	N-rel	N-rel	**rel-N**	**rel-N**
adjective and noun	adj-N	adj-N	adj-N	adj-N
numeral and noun	num-N	num-N	num-N	num-N
degree word and adjective	deg-A	deg-A	deg-A	deg-A
adjunct-internal order				
adposition and noun phrase[a]	ad-N	ad-N	**-**	**N-ad**
adverbial subordinator and clause	initial	initial	n.a.	**final**
other features				
subject and verb	SV	SV	SV	SV
object and verb	VO	**-**	VO	**OV**
demonstrative and noun	dem-N	dem-N	dem-N	dem-N
position of polar question particles	n.a.	n.a.	final	final
position of interrogative phrases	initial	initial	**not init**	**not init**
negative morpheme and verb	neg-V	neg-V	neg-V	**V-neg**

[a] Assuming that phrases marked by adpositions generally function as adjuncts. This is not always the case: Japanese notably marks objects with an adposition.

informative for unsupervised models.

We take the latent PCFG reordering grammar of Stanojević and Sima'an (2015) as a basis for refinement with adjunct annotations. This model allows to cover the full space of reorderings, while lending itself naturally to syntactic enrichment through labelling. We combine adjunct-based nonterminal splits with latent nonterminal splits to evaluate the effect of adjuncts in reordering, and show that this leads to large gains for English-Japanese reordering. Splitting adjuncts and non-adjuncts is a very simple refinement, which can be complemented with additional information such as dependency labels. We use dependencies to refine adjunct-driven and constituency-driven grammars with either hard splits, or using dependency labels as additional observations during learning, and show that adjunct-driven grammars benefit most from local enrichment, and constituency-driven grammars from richer contextual information in the form of first-order dependency labels.

The reordering model of Stanojević and Sima'an (2015) is introduced in section 5.2. Section 5.3 presents extensions to this grammar using adjunct, constituent and complement splits. Grammar refinements with dependency labels are presented in section 5.4. Section 5.5 complements this work with an analysis of the grammars.

5.2 Background: a generative PCFG model for reordering

We first present the latent reordering grammar of Stanojević and Sima'an (2015), as it forms the basis for our study of adjunction in reordering.

5.2.1 Model

Stanojević and Sima'an (2015) propose to model reordering with a PCFG grammar having the source vocabulary as terminals and permutation operators as nonterminals. The grammar is induced from source sentences s and their reorderings s'[1]. Reorderings are obtained by projecting source words to target positions given by word alignments, that are modified heuristically to ensure one-to-one projections.

The alignment between source sentences and their reorderings can then be represented as a tree of elementary permutations: monotone, inverted, and non-binarizable permutations of four or more elements (Wu, 1997). Besides, two separate monotone nonterminals, P_{01} and P_{10} are used for unaligned words and their right/left neighbour. The reordering grammar nonterminal vocabulary is completed by a start symbol S, and a set of preterminal symbols, one for every word type. Table 5.2 lists the set of inner nonterminals.

Table 5.2: Nonterminal vocabulary of the reordering grammar, preterminals not included. Subscript indices correspond to reordered positions.

arity	nonterminals
2	P_{12}, P_{21}, P_{01}, P_{10}, S
4	P_{2413}, P_{3142}
5	P_{24153}, P_{25314}, P_{31524}, P_{35142}, P_{41352}, P_{42513}

[1] I use here the term 'reordering' for full sentences, and the term 'permutation' for any sequence of tokens or nonterminals; a reordering is thus a permutation of a full sentence.

5.2.2 Learning

Nonterminals are split once prior to training following Isozaki et al. (2010a), while rule probabilities are modified with a small amount of noise to break symmetry. Nonterminals are not merged afterwards. The unary trick is applied to prevent the blow up of the grammar (with l latent splits, a nonterminal of arity r can rewrite through $(|N|l + |T|)^r$ rules, where N is the set of nonterminals and T the set of preterminals). The grammar is then modified to let nonterminals rewrite in two steps: each nonterminal (except the start symbol) first rewrites deterministically to positional variants; each positional variant then rewrites to a nonterminal through a unary rule. The grammar is trained with expectation-maximisation (Dempster et al., 1977), applying the inside-outside algorithm (Lari and Young, 1990) on the permutation forests defined by source sentences and their reorderings (as consecutive monotone or inverted alignments give rise to alternative branchings).

5.2.3 Parsing

Parsing uses the CYK+ algorithm (Chappelier and Rajman, 1998) to construct possible derivations of a sentence s into a reordering s'. Minimum Bayes Risk (MBR) is then used to search for the reordering $\hat{\pi}$ that minimizes the expected Kendall-tau loss in a set Π of reorderings:

$$\hat{\pi} = \arg\min_{\pi} \sum_{\pi' \in \Pi} \text{Loss}_{\tau}(\pi, \pi') P(\pi') \tag{5.1}$$

where the Kendall-tau loss between two reorderings π and π' counts the proportion of index pairs that are reordered differently in π and π':

$$\text{Loss}_{\tau}(\pi, \pi') = \frac{\sum_{i,j \in |\pi| : i<j} z_{ij}}{n(n-1)/2} \tag{5.2}$$

with

$$z_{ij} = \begin{cases} 1 & \text{if } \text{sign}(\pi(j) - \pi(i)) \neq \text{sign}(\pi'(j) - \pi'(i)) \\ 0 & \text{otherwise} \end{cases} \tag{5.3}$$

and the set Π of reorderings is obtained by Monte-Carlo sampling.

5.3 Making adjuncts explicit in reordering

To study the effect of adjunct annotations in the unsupervised reordering grammar of Stanojević and Sima'an (2015), we start simply by splitting permutation nonterminals into adjuncts and non-adjuncts, complementing these hard splits by latent splits. We also experiment with constituent and complement splits for comparison. Experiments show that adjuncts are beneficial for reordering in English-Japanese, and although constituent splits provide additional benifts, adjuncts play the largest role in constituent reordering.

Section 5.3.1 presents the adjunct-driven reordering model, section 5.3.2 the experimental setup for experiments in this section and the rest of this chapter, and section 5.3.3 ends this section with experimental results.

5.3.1 Driving reordering with adjuncts

Combining hard and latent labels

To account for adjuncts, we split nonterminal labels into two separate labels for each type of permutation: one for nonterminals that correspond to an adjunct, and one for non-adjuncts, which then regroups both non-adjunct constituents and nonconstituents[2], as shown in Figure 5.2. Preterminal labels are left undifferentiated, and are common to adjuncts and non-adjuncts. Adjunct splits are then completed by latent splits. This is a simple scheme, and one that allows to compare grammars with this manual split to latent-only grammars: in the experiments that follow, grammars with an adjunct split further use twice less latent splits than latent-only grammars.

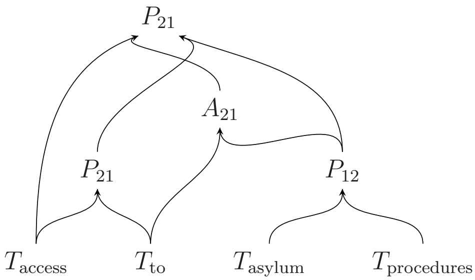


Figure 5.2: Permutation forest with adjunct/non-adjunct permutations, prior to latent splitting. P_{12} stands for a non-adjunct monotonic binary permutation, P_{21} and A_{21} stand for non-adjunct and adjunct binary inverted permutations, and T_{term} for a preterminal.

Training

Training utilizes source adjunct annotations alongside with source sentences and their reorderings. These annotations are obtained from dependency parses: mod-

[2] One could also distinguish nonterminals based on the nature of their children. This however would lead to 2^r labels for each permutation nonterminal of arity r, resulting in overspecific complex permutation labels.

ifier labels, excluding function words, are mapped to adjuncts; remaining dependents are mapped to complements.

After mapping permutations in the training data to either adjuncts or non-adjunct permutations, permutations are split automatically, as with the fully-latent grammar.

Parsing

Source sentences are parsed directly with the reordering grammar, without constraining them with adjunct annotations. Permutation trees are sampled as with the fully-latent grammar, and all trees yielding the same reordering are aggregated to find the best permutation, regardless of adjunct/non-adjunct labelling differences.

5.3.2 Experimental setup

Data sets

Experiments were performed on four language pairs: English-Japanese (en-ja), English-Chinese (en-zh), English-German (en-de) and German-English (de-en). The English-Japanese data consist of: 500k sentence pairs between 4 and 50 words from the NTCIR-8 patent translation task for training; the NTCIR-7 development sets (2101 sentence pairs) for tuning; the NTCIR-9 English-Japanese and Japanese-English test sets (2947 sentence pairs) for testing. The English data are lowercased and tokenized with a modified version of the Moses tokenizer to facilitate parsing.

The English-Chinese data are taken from the MultiUN corpus and consist of: 500k sentence pairs with a unique source side of 40 words at most for training; 2000 distinct sentence pairs for tuning and 2000 more distinct sentence pairs for testing. The English data are lowercased and tokenized as above.

The English-German and German-English data consist of 500k sentence pairs with up to 40 words from the Europarl corpus (v7), and of the WMT08 development and test sets, filtered to keep sentences with at most 50 source words; the development sets have 1790 sentence pairs for English-German, and 1850 for German-English; the test sets have 1795 sentence pairs for English-German and 1836 for German-English. The data are true-cased, and tokenized as above.

Dependency annotations

The English data are parsed with the Turbo parser[3], and the German data with Parzu[4]. Punctuation marks that enclose dependents with a modifier label are

[3] http://www.cs.cmu.edu/~ark/TurboParser/

[4] https://github.com/rsennrich/ParZu

identified as parenthetical markers; they are reattached under the head of the parenthetical, and relabelled with a dedicated dependency label. Parses are projectivized (by reattaching crossing dependents to the head of the subtree that dominates them), and multiple roots (for Parzu) are resolved by imposing a precedence: verbal > nominal > other roots; secondary roots are relabeled by concatenating the POS tag of the head to the ROOT label.

Training and parsing

Following Stanojević and Sima'an (2015), all reordering grammars are trained with 10 iterations of expectation-maximiza-tion. Parsing is performed without pruning, using 10000 sampled trees for MBR. Besides, non-binarizable permutations over more than five elements are filtered out at training, and unaligned words are attached by default to their right neighbour.

In most experiments, the baseline reordering grammar uses 16 latent splits for binary nonterminals, and 2 for complex nonterminals; grammars with an adjunct or other linguistically motivated hard split use 8 latent splits for binary nonterminals, and do not split complex nonterminals further.

Evaluation

The reordering models are evaluated intrinsically by comparing the reordered input sentence s' against a gold reordering, and extrinsically by comparing translations of preordered sentences against a baseline translation system, following Stanojević and Sima'an (2015).

For intrinsic evaluation, gold reorderings are obtained by applying a pretrained word-alignment model on the source sentences, and accordingly projecting the source words through the word alignments. In the case of one-to-many alignments, target positions are disambiguated by taking the mean. The reordering of null-aligned source words is disambiguated by attaching unaligned words to their right neighbour—or to their left neighbour in the absence of a right neighbour. Reordering quality is evaluated by taking the Kendall τ score (Birch and Osborne, 2011) of the reordered source (or the source in the case of the phrase-based baseline) with regard to the gold reordering.

For extrinsic evaluation, the baseline translation system is a phrase-based model trained with Moses using standard settings, and no lexicalized reordering model—the distortion limit is 6 for tuning/decoding. Training data are aligned with the Berkeley Aligner (Liang et al., 2006). The language model is a third-order model with modified Kneser-Ney smoothing, trained with KenLM (Heafield et al., 2013). The translation model used to evaluate the reordered data is trained with the gold reorderings as source input. The model is trained once for all reordering models, so that parsing with the reordering grammars only needs to be applied to the development and test sets. Tuning is performed with batch Mira

(Cherry and Foster, 2012), taking the best parameters out of 25 iterations. Tuning is performed three times to compensate for tuning instability (Clark et al., 2011). Translations are evaluated with BLEU (Papineni et al., 2002), BEER (Stanojević and Sima'an, 2014), and TER (Snover et al., 2006). Significance testing is performed following Clark et al. (2011) through BEER[5]. For reference, Table 5.3 provides the BLEU scores for the baseline translation system, and for the preordered model used to evaluate the reordering grammars. Results show that English-Japanese and English-Chinese have the highest potential for improvement with preordering.

Table 5.3: Baseline translation results. PB-dist is a phrase-based baseline with a distance-based reordering model; the Oracle results are obtained with a phrase-based model trained and evaluated on preordered data (gold source reorderings).

	en-ja		en-zh		en-de		de-en	
	dev	test	dev	test	dev	test	dev	test
PB-dist	24.4	27.3	25.2	24.8	17.8	18.1	24.5	24.4
Oracle	34.9	39.2	34.8	34.3	20.6	21.0	27.5	27.5

5.3.3 Comparing adjuncts, complements and constituents

We start by evaluating adjunct-driven reordering against a fully latent grammar and against constituent-driven reordering. In a second experiment, we look into the relative contribution of adjuncts, complements and constituents to reordering in English-Japanese.

Contribution of adjuncts and constituents to reordering

Table 5.4 reports experiments comparing adjunct-informed and constituent-informed reordering grammars to fully latent grammars and a Phrase-Based baseline. The Kendall τ score measures the similarity between the word orderings of target sentences and: source sentences for the Phrase-Based baseline; predicted reorderings for the reordering grammars. Experiments are carried out on four language pairs: English-Japanese, English-Chinese, English-German and German-English.

For English-Japanese, Kendall τ scores show that all reordering grammars produce reorderings that are closer to target word order. For the fully latent reordering grammar however, the predicted reorderings are not informative enough to improve downstream translation scores compared to the Phrase-Based baseline. This is different for the linguistically informed grammars, which get closer

[5] https://github.com/stanojevic/beer

Table 5.4: Contribution of adjuncts and constituents to reordering. PB-dist is a phrase-based model with a distance-based reordering model; $\mathrm{RG}_{16,2}$ is a latent reordering grammar with 16 latent splits for binary permutations and 2 splits for higher-order permutations; $\mathrm{RG\text{-}A}_{8,1}$ and $\mathrm{RG\text{-}C}_{8,1}$ are latent reordering grammars with an adjunct, respectively constituent split, completed with 8 latent splits for binary permutations and no complex-permutation latent splits.

		τ	BLEU	BEER	TER
en-ja	PB-dist	0.683	27.3	10.6	60.9
	$\mathrm{RG}_{16,2}$	0.715	27.3	10.6	60.6*
	$\mathrm{RG\text{-}A}_{8,1}$	0.751	**29.4***	**11.6***	**58.5***
	$\mathrm{RG\text{-}C}_{8,1}$	0.756	**29.8***	**12.0***	**57.8***
en-zh	PB-dist	0.797	24.8	12.9	61.0
	$\mathrm{RG}_{16,2}$	0.804	23.3	11.9	62.2
	$\mathrm{RG\text{-}A}_{8,1}$	0.810	23.2	**12.1**	**61.8**
	$\mathrm{RG\text{-}C}_{8,1}$	0.807	23.4	12.0	62.0
en-de	PB-dist	0.941	18.1	14.4	67.0
	$\mathrm{RG}_{16,2}$	0.939	17.5	14.5	68.3
	$\mathrm{RG\text{-}A}_{8,1}$	0.939	17.2	14.3	68.6
	$\mathrm{RG\text{-}C}_{8,1}$	0.939	17.5	14.6	68.2
de-en	PB-dist	0.946	24.4	17.2	60.2
	$\mathrm{RG}_{16,2}$	0.944	24.1	17.0	60.6
	$\mathrm{RG\text{-}A}_{8,1}$	0.943	24.1	17.0	60.7
	$\mathrm{RG\text{-}C}_{8,1}$	0.941	**24.4**	17.1	**60.3**

Boldface marks a significant improvement with $p = 0.05$ with regard to the baseline reordering grammar $\mathrm{RG}_{16,2}$, and asterisks a significant improvement with $p = 0.05$ with regard to PB-dist

than the fully latent grammar to target word orders, and for which the predicted reorderings succesfully leverage on the preordered phrase-based model. Constituent-driven grammars perform better than adjunct-driven grammars, for both intrinsic and extrinsic evaluation.

In English-Chinese, word-order differences are lesser than in English-Japanese, as shown by the higher Kendall τ score between source and target (reported for PB-dist). The reordering grammars learn to reorder closer to target word orders, but to a lesser extent than for English-Japanese. The predicted reorderings are also not informative for translation, compared to the PB-dist baseline. Linguistically informed grammars, in particular the adjunct-informed grammar, still perform slightly better than the fully latent baseline, both intrinsically and extrinsically.

For English-German and German-English, the reordering grammars reorder away from the target orderings, and the predicted reorderings accordingly underperform on the translation task. The constituent-informed grammar does however perform better than the fully latent grammar on translation, in spite of a slightly lower Kendall τ score.

Contribution of complements to reordering

To complement the results with adjunct-informed and constituent-informed grammars for English-Japanese, we experiment with complement splits. Table 5.5 reports English-Japanese results for the reordering grammars of Table 5.4, for a complement-informed grammar using a complement/noncomplement split, and for a grammar with a three-way adjunct/complement/nonconstituent split.

Table 5.5: Contribution of adjuncts and complements in English-Japanese reordering. RG-Cmp$_{8,1}$ uses a complement/noncomplement split, RG-A/Cmp$_{8,1}$ a three-way adjunct/complement/nonconstituent split, RG-A$_{8,1}$ an adjunct/non-adjunct split, and RG-C$_{8,1}$ a constituent/nonconstituent split. All reordering grammars are refined with 8 latent splits for binary nonterminals, and no further split for complex nonterminals.

		dev		test			
	$\|\mathcal{G}\|$	τ	BLEU	τ	BLEU	BEER	TER
RG$_{16,2}$	2.19M	0.718	24.5	0.715	27.3	10.6	60.6
RG-Cmp$_{8,1}$	1.48M	0.739	25.6	0.745	28.4	11.1	59.5
RG-A/Cmp$_{8,1}$	1.75M	0.751	26.2	0.759	29.3	11.5	58.0
RG-A$_{8,1}$	1.39M	0.750	26.2	0.751	29.4	11.6	58.5
RG-C$_{8,1}$	1.55M	0.747	26.8	0.756	29.8	12.0	57.8

Complement/noncomplement splits are informative for reordering, but less so than adjunct splits, and this shows especially in the translation results. Combining information about complements and adjuncts through a three-way adjunct/complement/nonconstituent split increases the Kendall τ score compared to adjunct/non-adjunct splits, but not the downstream translation results.

In conclusion, both adjuncts and complements inform the model about constituent reordering, but adjuncts perform better as a category, both because they lead to better results than complements on their own, and because the adjunct/nonconstituent split operated in the three-way split model improves results for the complement/noncomplement split model (in contrary to the complement/nonconstituent split for the adjunct/non-adjunct split model). The lower results obtained by adjunct-informed grammars compared with constituent-informed grammars shows however that there is something to be gained from complements. The three-way split appears too restrictive in this respect, and it is likely both that only part of the complements are informative for reordering, and that the informative complements should be merged with adjuncts.

Summary

In summary, we see that adjuncts are useful for guiding reordering, at least for English-Japanese. While constituency in general leads to better reordering and downstream translation than adjunction, further splitting non-adjuncts into complements and non-constituents in the adjunct-informed grammar does not suffice to bridge the gap. This is likely due to complements being less informative on the whole as a reordering category than adjuncts, and to informative complements behaving like adjuncts for reordering, so that these should be merged in a same category.

For the other language pairs, the reordering grammars are not able to learn sufficiently useful reorderings. Only for English-Chinese do we see some improvement of adjunct-informed grammars compared to the fully-latent grammars. This improvement is still not able to perform equally to the phrase-based baseline, and that while oracle scores show similar potential for improvement to English-Japanese. The next section addresses this issue with a number of refinements to the grammars.

5.4 Grammar refinements

The poor reordering results of English-Chinese are surprising given oracle scores and the number of adjunct-related reordering phenomena in this language pair. In this section, we consider different grammar refinements to try and improve English-Chinese preordering. We start by increasing the number of latent splits in section 5.4.1, we refine adjunct hard splits with dependency annotations in section 5.4.2, and we add dependency annotations as an external signal for learning in section 5.4.3.

5.4.1 Increasing the number of latent splits

We consider first the effect of increasing the number of latent nonterminal splits, both for fully-latent and adjunct-driven reordering grammars. As Table 5.6 shows, doubling the number of nonterminals through latent splits in the baseline reordering grammar is beneficial both for English-Japanese and English-Chinese. For adjunct-based grammars, further splitting also benefits English-Chinese, but not English-Japanese, where excessive splitting removes the benefits of the linguistic, adjunct bias. Increasing the number of nonterminals also has its limits for fully-latent grammars, as can be observed for English-Japanese, where 24 latent splits provide better performance than 32. We can further observe that intrinsic

Kendall τ scores and extrinsic MT scores appear not to correlate for English-Chinese, in contrary to English-Japanese: increasing the number of latent splits increases MT scores but not the Kendall τ scores, while adjunct splits have the opposite effect.

Table 5.6: Increasing the number of splits in latent-only and adjunct grammars. Adjunct splits double the number of nonterminal labels, so that, e.g., $\mathrm{RG}_{16,2}$ and $\mathrm{RG\text{-}A}_{8,1}$ have the same nonterminal vocabulary size. Grammar sizes are in million rules.

		$\lvert\mathcal{G}\rvert$	τ	BLEU	BEER	TER
en-ja	$\mathrm{RG}_{16,2}$	2.19	0.715	27.3	10.6	60.6
	$\mathrm{RG}_{24,2}$	3.24	0.736	28.8	11.1	59.1
	$\mathrm{RG}_{32,2}$	4.29	0.733	28.6	11.2	59.3
	$\mathrm{RG\text{-}A}_{8,1}$	1.39	0.751	29.4	11.6	58.5
	$\mathrm{RG\text{-}A}_{16,1}$	2.71	0.730	28.2	10.9	59.6
en-zh	$\mathrm{RG}_{16,2}$	2.65	0.804	23.3	11.9	62.2
	$\mathrm{RG}_{32,2}$	5.18	0.804	23.7	12.1	61.5
	$\mathrm{RG\text{-}A}_{8,1}$	1.74	0.810	23.2	12.1	61.8
	$\mathrm{RG\text{-}A}_{16,1}$	3.39	0.811	23.5	12.1	61.6

5.4.2 Refining hard adjunct splits

We turn to the effect of refining adjunct splits with dependency labels, to assess whether the nature of syntactic dependencies is informative for reordering.

For adjunct-driven grammars, we split adjunct nonterminals by their dependency label: adverbial (VMOD); adnominal (NMOD) or ad-adjectival (AMOD). Each type of adjunct binary nonterminal is further split in 3 latent labels, so that the resulting grammar uses almost the same number of splits as the reference adjunct grammar. Non-adjunct binary nonterminals are split like the reference adjunct grammar, in 8 latent splits. As before, complex nonterminals are not refined further than the adjunct/non-adjunct distinction.

The same procedure is applied to constituent-driven grammars. Constituents are now split into 13 labels: 12 Turbo-parser labels (ROOT, SUB, OBJ, PRD, VC, SBAR, DEP, PMOD, NMOD, VMOD, AMOD, P) extended with a parenthetical, apposition marker. Constituent labels are not split any further. Nonconstituent binary labels are split into 8 latent labels as for the reference constituent-driven grammar.

Results are presented in Table 5.7, showing that the adjunct refinement in particular is beneficial for English-Chinese, with an improvement of 0.4 BLEU.

Table 5.7: Refining reordering grammars with dependency information. RG-Ar$_{8,1}$ uses three adjunct dependency-based labels with three latent splits for binary labels, and eight latent splits for non-adjunct binary labels; RG-Ar$_{16,1}$ splits non-adjunct binary labels further into sixteen latent splits; RG-Cr$_{8,1}$ splits constituent labels by their dependency label, and nonconstituent labels by eight latent splits for binary labels. Grammar sizes are in million rules.

	$\|\mathcal{G}\|$	τ	BLEU	BEER	TER
RG-A$_{8,1}$	1.74	0.810	23.2	12.1	61.8
RG-Ar$_{8,1}$	1.55	0.810	**23.6**	12.3	**61.4**
RG-C$_{8,1}$	1.91	0.807	23.4	12.0	62.0
RG-Cr$_{8,1}$	1.51	0.808	23.5	12.1	61.8

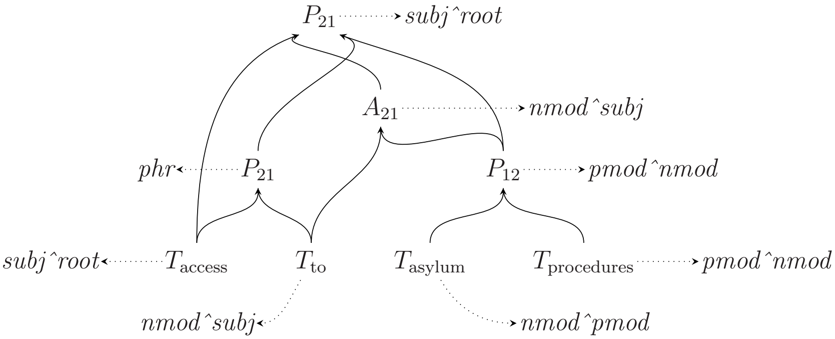


Figure 5.3: Permutation forest with first-order dependency emissions. Spans that do not match a syntactic dependent receive a default label *phr*.

5.4.3 Adding observations

We have seen that adjuncts are informative for English-Chinese reordering, even though the improvement they bring is modest, and that complementary linguistic splitting through dependency labels further improves reordering. Further refining nonterminals increases however the risk of rule sparsity. When combining hard and latent labels, it also becomes difficult to determine how many latent splits each hard label should take, and it would then be advantageous to merge labels after splitting, following Petrov et al. (2006). To further refine nonterminals, we use syntactic annotations as additional observations to guide learning, only keeping a hard distinction between adjuncts and non-adjuncts, or between constituents and nonconstituents, as shown in Figure 5.3.

Training and parsing

The emission rules form a new set of parameters for the model, that can be simply integrated in the inside-outside algorithm. The dependency observations are determined for the chart prior to computation, and are used only for parameter estimation; no syntactic constraints are imposed during parsing.

Results

Table 5.8 reports results for the adjunct-driven and constituent-driven grammars with zero-order and first-order dependency-label emissions. We find that zero-order emissions slightly improve translation performance for the adjunct-driven grammar, but not for the constituent-driven grammar—this is similar to the results observed with refined label splits reported in Table 5.7. In contrast, first-order emissions provide no improvement for the adjunct-driven grammar, but improve both reordering and translation performance for the constituent-driven grammar.

Table 5.8: Evaluation of Reordering Grammars with label emission for English-Chinese. RG-A/C$^{x}_{8,1}$ is an adjunct-split, respectively constituent-split latent reordering grammar with x-order dependency-label emissions.

	Prec.	τ	BLEU	BEER	TER
RG-A$_{8,1}$		0.810	23.2	12.1	61.8
RG-A$^{0}_{8,1}$	0.700	0.810	**23.4**	12.2	**61.6**
RG-A$^{1}_{8,1}$	0.624	0.811	23.2	11.9	61.9
RG-C$_{8,1}$		0.807	23.4	12.0	62.0
RG-C$^{0}_{8,1}$	0.765	0.807	23.4	12.0	62.1
RG-C$^{1}_{8,1}$	0.672	0.811	**23.8**	12.3	**61.6**

Boldface mark significant improvements (with $p \leq 0.05$) over the reference grammar without soft constraints

Table 5.8 also gives precision figures for the prediction of dependency labels when parsing the test set. Labelling precision is computed during sampling, by sampling a dependency label from the distribution of syntactic labels for each sampled permutation label. For each sampled tree, labelling precision counts the proportion of correctly predicted syntactic labels given the reference dependency parse; for nonconstituent nodes, the correct syntactic label is the default *phr* label. The reported figures are averaged over the 1-best permutation trees after MBR reranking.

Labelling-precision scores are naturally lower for first-order labels than zero-order labels, but they are also lower for adjunct-driven grammars than for constituent-driven grammars. In constituent-driven grammars, the hard split between constituents and nonconstituents forms in fact a natural fit for the prediction of syntactic labels, as only constituent permutation labels should match dependencies. First-order emissions appear particularly informative for guiding constituent permutations. In adjunct-driven grammars in contrast, non-adjunct permutation labels correspond both to non-adjunct constituents and to nonconstituents. This makes it harder to predict dependency labels accurately, and diminishes their contribution to guiding permutations. That zero-order emissions benefit adjunct-driven grammars more than first-order emissions may at the same time result from adjunct reordering being *more local* than complement reordering: the distinction between, e.g., adnominal and adverbial modifiers is more informative for reordering than the distinction between adnominal modifiers of subject or object noun phrases. These results are confirmed by experiments with English-Japanese, as shown in Figure 5.9. In contrary to English-Chinese however, the reordering model does not benefit from the added observations.

Table 5.9: Extrinsic evaluation of reordering grammars with label emission for English-Japanese.

	BLEU	BEER	TER
RG-A$_{8,1}$	29.4	11.6	58.5
RG-A$^0_{8,1}$	29.1	11.7	58.7
RG-A$^1_{8,1}$	27.9	11.0	59.6
RG-C$_{8,1}$	29.8	12.0	57.8
RG-C$^0_{8,1}$	28.8	11.6	58.4
RG-C$^1_{8,1}$	29.5	11.9	57.8

Summary

Refining reordering grammars with dependency information provides additional benefits to the adjunct/non-adjunct or constituent/nonconstituent distinction for English-Chinese. Adjunct-driven reordering benefits from a distinction between different adjunct types (adnominal, adverbial or ad-adjectival), while constituent-driven reordering benefits from higher-order syntactic information. The locality of improvements for adjunct-driven grammars agrees with the optional character of adjuncts, so that adjuncts affect reordering, but only locally. We added dependency information in two ways, first through hard splits, and secondly by taking dependency labels as additional observations for learning. The first method is limited by the risk of rule sparsity, and would benefit from merging labels after splitting, following Petrov et al. (2006). The second method depends on the adequacy between permutations and dependencies, and favors grammars with a constituent/nonconstituent split. And while this method provides benefits for

English-Chinese, it also leads the model astray for English-Japanese. Feature-based learning (Berg-Kirkpatrick et al., 2010) would provide a safer way to enrich the model.

5.5 Analysis

Refining the grammars with dependency information improves English-Chinese reordering, but only modestly. This section provides an analysis of results for this language pair. We show in section 5.5.1 that the reordering grammars primarily learn monotone reorderings for English-Chinese. This is confirmed by inspection of rule distributions in section 5.5.2, where we see that adjuncts are useful to delimitate linguistic contexts, but are no match against the domination of monotonic rewritings.

5.5.1 Learning

Given the minor improvement in Kendall τ scores observed with adjunct-driven grammars, one could wonder whether the reordering grammars perform any reordering at all. To this end, we compare Kendall τ scores measured with regard to the gold reorderings s' ($\tau(\hat{s'}, s')$), to scores measured with regard to the source word order ($\tau(\hat{s'}, s')$). Results are shown in Table 5.10 for a number of reordering grammars, for English-Chinese and English-Japanese.

Table 5.10: Evaluating reorderings with regard to source sentences s and gold reorderings s'. RG$_{16,2}$ and RG$_{32,2}$ are latent reordering grammars, RG-C8, 1 a grammar with constituent splits, RG-A$_{8,1}$ a grammar with adjunct splits, and RG-A$^0_{8,1}$ an adjunct-split reordering grammar with 0-order dependency-label emissions.

		RG$_{16,2}$	RG$_{32,2}$	RG-C$_{8,1}$	RG-A$_{8,1}$	RG-A$^0_{8,1}$
en-zh	$\tau(\hat{s}', s)$	0.982	0.978	0.964	0.960	0.956
	$\tau(\hat{s}', s')$	0.808	0.809	0.812	0.815	0.815
en-ja	$\tau(\hat{s}', s)$	0.922	0.892	0.743	0.844	0.821
	$\tau(\hat{s}', s')$	0.718	0.733	0.747	0.750	0.749

We find for English-Chinese that the predicted reorderings stay extremely close to the source orderings, and are affected only a little by syntactic information. With adjunct-split grammars, reordered sentences for Chinese still share 96% of index-pair orderings with the original source sentences. In contrast, different settings for the English-Japanese grammars have a strong effect on reordering. One can then also observe that not all permutations of the source are equally succesful at approaching gold reorderings. Adjunct-split grammars

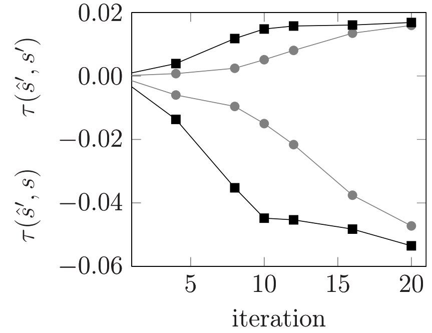


Figure 5.4: Variation of Kendall τ scores with the number of EM iterations for the RG-A$_{8,1}$ (squares) and the RG$_{16,2}$ grammars (dots) for English-Chinese. Predicted reorderings move towards gold reorderings (positive differences) and away from source orderings (negative differences).

appear in this respect to perform more targeted permutations than constituent-split grammars: while the constituent-split grammar RG-C$_{8,1}$ alters the source ordering more than the adjunct-split grammar RG-A$_{8,1}$ (with a Kendall τ score of 0.743 against 0.844), both have very similar reordering scores with regard to the gold reorderings (0.747 and 0.750).

The lack of performance of the English-Chinese reordering grammars cannot be blamed to premature ending of learning. As Figure 5.4 shows, Kendall τ scores can still increase after 10 iterations of EM, but in fact they hardly do so for adjunct-based grammars.

5.5.2 Rule distributions

The rule distributions learned by adjunct-driven reordering grammars also show a prevalence of monotonic rule rewritings. Comparing for instance the rewritings for non-adjunct monotone nonterminals P_{12} for English-Chinese and English-Japanese in Table 5.11, one finds that the English-Chinese grammar learns at best to distinguish between low and high rewritings: variant 3 specialized towards low rewritings to terminals and permutations involving unaligned tokens; and variants 0, 1 and 4 towards higher rewritings. Only variant 3 rewrites preferably to an inverted permutation through its rightmost child, but it does so with a probability of 0.077—the four next monotonic rewritings receive together 0.16 of the probabibility mass for this nonterminal variant.

The English-Japanese distributions for P_{12} also show variants that specialize in low rewritings (4 and 6 on the left, 0, 1 and 3 on the right) or higher rewritings (all except 4 and 6 on the left). Besides, the higher leftmost rewritings specialize in monotonic rewritings (2, 3, 5 and 7) or in inverted rewritings (0 and 3). Fur-

Table 5.11: Top-8 P_{12} rewritings per latent split and positional variant for English-Chinese and English-Japanese, for the adjunct-driven grammar $A_{8,1}$. Nonterminals T_{term} are represented by their corresponding terminal *term*. Latent nonterminal variants are indicated by upper indices. P_{12} rewritings are grayed out for readability.

	left	right
English-Chinese		
0	P^7_{12} P^4_{12} P^6_{12} P^5_{12} P^2_{12} P^1_{12} P^3_{12} P^4_{21}	. P^0_{12} P^1_{12} P^4_{12} P^7_{12} the P^6_{12} P^5_{12}
1	P^7_{12} P^6_{12} P^5_{12} P^3_{12} P^2_{12} P^4_{12} P^1_{12} P^6_{01}	P^0_{12} . P^7_{12} P^1_{12} P^4_{12} P^6_{12} P^5_{12} P^2_{12}
2	P^7_{12} P^3_{12} P^6_{01} P^6_{12} P^5_{12} P^2_{12} and P^1_{01}	P^2_{12} P^7_{12} P^6_{12} to P^5_{12} P^3_{12} P^4_{12} and
3	the P^6_{01} P^1_{01} P^4_{01} and P^7_{12} P^3_{12} P^0_{01}	P^5_{21} P^3_{12} P^2_{12} P^5_{12} P^6_{12} P^4_{21} P^4_{12} P^7_{12}
4	P^7_{12} P^6_{12} P^5_{12} P^3_{12} P^2_{12} P^4_{12} P^1_{12} P^6_{21}	P^0_{12} P^7_{12} P^4_{12} P^1_{12} P^6_{12} P^2_{12} P^5_{12} .
5	, P^7_{12} and P^6_{01} the P^6_{12} P^5_{12} P^1_{01}	P^2_{12} P^7_{12} P^6_{12} P^5_{12} P^3_{12} P^4_{12} P^1_{12} P^0_{12}
6	, P^7_{12} P^6_{12} P^3_{12} P^5_{12} a P^2_{12} and	P^7_{12} P^6_{12} P^2_{12} P^4_{12} P^3_{12} P^0_{12} P^5_{12} P^1_{12}
7	, P^7_{12} P^6_{12} (P^5_{12} P^3_{12} P^2_{12} P^4_{12}	P^7_{12} and P^6_{12} P^0_{12} P^4_{12} P^1_{12} P^5_{12} P^2_{12}
English-Japanese		
0	P^6_{12} P^3_{21} P^5_{21} A^7_{12} fig P^1_{21} P^2_{21} P^4_{12}	P^3_{12} P^0_{12} a P^4_{01} the P^5_{12} P^5_{21} P^1_{21}
1	P^6_{12} P^1_{12} P^2_{12} P^7_{12} P^3_{01} P^4_{12} control P^5_{12}	to P^7_{12} circuit signal P^1_{12} is P^2_{12} P^5_{12}
2	P^2_{12} P^1_{12} P^6_{12} P^7_{12} , P^3_{01} P^4_{12} P^5_{12}	P^7_{12} and P^2_{12} P^5_{12} P^1_{12} is P^3_{12} P^1_{21}
3	P^5_{21} P^1_{21} is P^3_{21} P^2_{21} P^0_{12} P_{3142} P^6_{12}	. P^3_{12} P^0_{12} P^4_{01} P^0_{01} formed P^5_{21} P^1_{21}
4	the P^6_{01} P^6_{12} P^3_{01} in P^1_{01} P^2_{01} P^1_{12}	P^7_{12} the P^2_{12} P^1_{12} P^5_{12} P^4_{12} P^6_{12} a
5	P^6_{12} P^7_{12} P^1_{12} P^2_{12} P^5_{12} P^4_{12} P^3_{01} P^3_{21}	and P^7_{12}) is P^5_{12} P^3_{12} P^2_{12} film
6	the a P^6_{01} P^3_{01} P^2_{01} P^6_{12} P^1_{01} to	P^7_{12} P^2_{12} P^1_{12} P^5_{12} P^4_{12} be P^6_{12} P^7_{21}
7	P^1_{12} P^2_{12} P^6_{12} P^5_{12} P^7_{12} P^4_{12} (and	P^7_{12} P^5_{12} 1 P^1_{12} P^2_{12} 2) 10

thermore, the distribution for the rightmost rewritings of variant 7 shows some lexical specialization, namely into figures.

We find similar effects in the distributions of other permutation nonterminals. Overall, the English-Japanese grammars are able to specialize lexically and to generate inverted permutations, in contrary to the English-Chinese grammars that remain largely undifferentiated. This is reflected by entropy values for rule rewritings, for which summary statistics are presented in Table 5.12. Entropy is lower on average for English-Japanese than for English-Chinese. The larger standard-deviation values for English-Japanese reflect that some nonterminal-variant distributions specialize more than others. An extreme case of specialization is given by two variants of A_{12} right-child rewritings that specialize in punctuation preterminals, both rewriting to a comma preterminal with a probability higher than 0.96.

Rule entropy is also lower for adjunct nonterminals, as they capture specific linguistic information. For inverted adjunct permutations for instance, as shown

Table 5.12: Rule entropy statistics for the binary nonterminals P_{12}, A_{12}, P_{21} and A_{21} of the adjunct-driven grammar $A_{8,1}$, averaged over latent nonterminal variants.

	en-zh				en-ja			
	left		right		left		right	
	mean	σ	mean	σ	mean	σ	mean	σ
P_{12}	5.07	0.52	4.83	0.92	4.70	0.83	4.57	1.43
A_{12}	3.12	0.56	4.44	0.70	3.05	0.44	3.41	1.97
P_{21}	4.59	0.38	4.88	0.48	4.05	0.20	4.16	0.32
A_{21}	3.23	0.36	4.18	0.49	2.64	0.77	2.53	0.68

in Table 5.13, English-Chinese latent nonterminals specialize either in lexical rewritings to prepositions, or to abstract rewritings, where more room is given to inversions and complex permutations. In particular, variant 5 is the only nonterminal (of all binary permutation types) with a top rewriting to an inversion nonterminal (with a probability of 0.24). Lexical specialization, here too, is more advanced for English-Japanese, where all but two variants specialize to prepositional rewritings, and one lexically oriented variant (5) also captures non-generic, temporal prepositions.

Table 5.13: Top-10 A_{21} left-child rewritings per latent split for English-Japanese and English-Chinese, for the adjunct-driven grammar $A_{8,1}$.

en-zh	0	P^0_{12} P^3_{12} P^1_{21} P^6_{01} of P^4_{01} in P_{2413} P^6_{12} P^0_{21}
	1	P^3_{12} P^4_{01} P^0_{12} P^6_{12} P^2_{12} P^1_{12} P^4_{12} P^1_{21} P^3_{21} P^0_{21}
	2	P^4_{01} P^3_{12} P^6_{01} P^1_{21} P^0_{12} P^2_{12} P^0_{21} P^6_{12} P^3_{21} P^0_{01}
	3	P^0_{12} of P^3_{12} P^4_{01} in P_{2413} P^6_{01} P^6_{12} to P^1_{21}
	4	of P^0_{12} to P^3_{12} P^6_{12} P^4_{01} P^2_{12} mr. P_{2413} P_{25314}
	5	P^1_{21} P^3_{12} P^0_{21} P^6_{01} P^0_{12} P^4_{01} P^2_{12} P^6_{12} P^3_{21} P^7_{21}
	6	of in for by under P^0_{12} between within with on
	7	of in on for between P^0_{12} to among by from
en-ja	0	of in from on by between via for to at
	1	of in from to on for between through by at
	2	of in from as to by on with P^4_{21} at
	3	to by at via of through in into from on
	4	P^4_{12} P^2_{21} P^4_{21} P^6_{21} P^6_{01} P^0_{12} P^7_{21} P^6_{12} P^3_{21} P^0_{21}
	5	of in by from to between after on during as
	6	as P^4_{12} in to P^4_{21} from P^0_{12} P^6_{01} on P^6_{12}
	7	of in from by to on at when between through

5.5.3 Summary

Adjuncts are useful guides for reordering in English-Chinese and English-Japanese, as they delimitate relevant linguistic contexts, allowing to reorder faster and/or better than with latent splits only. The informational benefit of adjuncts is however no match against the predominance of monotonic rewritings in English-Chinese. In particular, the adjunct nonterminals that capture inversion patterns or complex permutations stand little chance of being rewritten to. The dominance of monotonic rewritings in English-Chinese extends to lexical specialization, which is much less advanced than for English-Japanese.

Reordering patterns are more limited in English-Chinese than in English-Japanese, as shown previously in Table 5.1. Oblique arguments excepted, most inversions concern noun phrases and their modifiers. As inversion patterns are more isolated than in English-Japanese, they are harder for the generative PCFG model to pick up.

5.6 Conclusion

In this chapter, we have studied the contribution of adjuncts to reordering, by using them to guide the latent PCFG reordering grammar of Stanojević and Sima'an (2015), and applying the resulting grammars to preordering for language pairs with varying degrees of reordering.

Adjuncts and constituents are particularly informative for English-Japanese reordering, yielding a gain of 2.5 BLEU points over the phrase-based and fully-latent baselines. While constituent-driven grammars yield the best results, adjuncts play the larger part in their performance. Complements appear to include both informative and non-informative elements; semantically-motivated arguments are likely to be the missing elements, and to be as relevant as adjuncts in reordering.

English-Chinese preordering also benefits from information from adjuncts and constituents, but to a much lower extent than English-Japanese, and that while oracle scores show a similar potential for preordering in both language pairs. Refining linguistic splits provides some improvement for English-Chinese: adjunct-driven grammars notably benefit from rule-local information, and constituent-driven grammars from higher-order information, in the form of first-order dependency-label emissions. While the first result points to a local effect of adjunction on reordering, the second points to the complexity of reordering in this language pair. While local optimization is problematic in this case, the relatively small gains obtained with adjunct splits, combined with the local character of adjunction for reordering, also suggest that adjuncts only play a minor role in reordering for this language pair.

Chapter 6

Conclusion

6.1 Summary

The objective of this dissertation was to investigate adjunction as a source of linguistic information for hierarchical phrase-based translation. We took syntactic modification as the main type of adjunction, similarly to applications of STAG (Shieber and Schabes, 1990) in Syntax-Based SMT (DeNeefe and Knight, 2009; Liu et al., 2011). Unlike STAG however, we did not consider adjunction as a formal operation, but considered the operands of adjunction—adjuncts, or syntactic modifiers—as strings. Like STAG, we hypothesized that adjunction has a semantic basis and that it operates synchronously in translation. We further hypothesized that adjunction through syntactic modification explains most of recursion in translation data, and tested this hypothesis in hierarchical phrase-based translation modelling, whereby we also exploited adjunct optionality. Similarly, we hypothesized that adjuncts explain most of reordering in translation data, and tested this hypothesis in a latent reordering grammar for translation preordering.

Synchronous Adjunction in translation data
In Chapter 3, we evaluated the degree to which adjuncts translate as adjuncts in translation data. To factor the effect of word alignments and parse-based adjunct-labelling heuristics on synchronous adjunction, we performed measures both in a corpus study with manually annotated and aligned adjuncts, and in experimental conditions. For the corpus study, we contrasted adjuncts to arguments to assess whether arguments entering in the adjunct/argument distinction behaved differently in translation. We found that adjuncts and arguments engage to a similar degree in synchronous alignments, and that adjuncts only differ by a higher flexibility in aligning in different syntactic contexts. We further found that the distinction between adjuncts and arguments does not have to be fine-grained for translation in experimental conditions, as synchronously aligned parsed modifiers also include annotated arguments. For experimental measures of adjunct align-

ment, we contrasted parsed adjuncts to complements. We found that synchronous adjunction is more sensitive to data than synchronous complementization, especially in the short to medium range. We conclude from this that synchronous adjunction is indicative of translation compositionality, and synchronous complementization primarily of syntactic similarity. All in all, we conclude that adjuncts and arguments are equally synchronous in translation, only parse modifier labels are more likely to correspond to synchronous adjuncts and (oblique) arguments, while non-modifier labels regroup both synchronous arguments (like subjects or objects), and non-synchronous arguments reflecting syntactic idiocracies.

Adjunction in Hierarchical Phrase-Based SMT
In Chapter 4, we assumed that adjuncts play a central role in recursion for translation, and extended Hiero (Chiang, 2005) by relaxing span-length constraints around adjuncts. We found that adjuncts are useful for driving recursion in hierarchical phrase-based models, as adjunct-crossing constraints not only allow for useful long-range rules, but also effectively filter short-range rules. Comparing adjuncts to complements and constituents showed however that constituents tend to be the better guide for synchronous recursion. This is consistent with the results of Chapter 3, where we saw that a quarter of complements in French-English correspond to synchronously-aligned arguments. We also leveraged on adjunct optionality to extend the grammar by excising adjuncts at rule extraction, obtaining promising results for English-Japanese. An interesting extension would consist in adding abstract adjunction rules to the model, thereby targetting the selection properties of adjunction more than optionality only, and implementing adjunction as a formal operation in Hierarchical Phrase-Based SMT.

Adjunction in reordering
In Chapter 5, we hypothesized that adjuncts collectively explain most of reordering in translation, and tested this hypothesis by utilizing adjuncts as a source of information in the latent reordering grammar of Stanojević and Sima'an (2015). We found that adjuncts explain most of reordering in English-Japanese *constituent-based* reordering, while constituents are more informative on the whole. This is consistent again with the results of Chapter 3 and Chapter 4. It would be interesting in this respect to regroup adjuncts and semantically motivated arguments. Through dependency-based refinements, we further showed that adjuncts play a local role in reordering. The modest improvements obtained for English-Chinese suggest that adjuncts play a minor role in reordering in that language pair.

6.2 Future work

Neural MT (Sutskever et al., 2014; Bahdanau et al., 2015; Vaswani et al., 2017) solves some of the worst defaults of SMT. The first is the dependence on (Viterbi) word alignments, that binds SMT models to a strict, word-level interpretation of translation equivalence. All SMT models consequently have had to account, *somehow*, for unaligned words. The second is the *direct* reliance on surface forms. In NMT, continuous vector-space representation of words allows for a natural handling of synonymy and morphology, while translation structures are and remain deep, with a more natural realisation of agreement as a result.

Interest for syntax and linguistic enrichment is undiminished in NMT. Numerous approaches have been proposed to use syntax to guide encoder representations (Sennrich and Haddow, 2016; Bastings et al., 2017; Chen et al., 2017; Li et al., 2017; Currey and Heafield, 2019; Bugliarello and Okazaki, 2020), or decoding (Eriguchi et al., 2017; Saunders et al., 2018). Handling long-distance dependencies notably remains hard for NMT models. Xu et al. (2020) show that phrasal representations improve long-distance subject-verb agreement for German in transformer models, while improving translation for English-French and English-German in general. We have shown in this work that adjuncts—and likely semantically motivated arguments—are informative in this respect.

Reordering is also still of interest for NMT. While attention has been seen as soft word alignment, Cohn et al. (2016), Alkhouli and Ney (2017) and Zhang et al. (2017) all have proposed to use word alignments to bias attention. As adjuncts inform reordering locally, they may also provide an interesting linguistic bias for attention-based NMT.

Bibliography

Anne Abeillé and Yves Schabes. Parsing idioms in lexicalized TAGs. In *Fourth Conference of the European Chapter of the Association for Computational Linguistics*, Manchester, England, April 1989. Association for Computational Linguistics.

Anne Abeillé, Yves Schabes, and Aravind K. Joshi. Using lexicalized tags for machine translation. In *Proceedings of the 13th Conference on Computational Linguistics - Volume 3*, COLING '90, pages 1–6, Stroudsburg, PA, USA, 1990. Association for Computational Linguistics. ISBN 952-90-2028-7. doi: 10.3115/991146.991147.

Omri Abend and Ari Rappoport. Fully unsupervised core-adjunct argument classification. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 226–236, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

Alfred V. Aho and Jeffrey D. Ullman. *The Theory of Parsing, Translation, and Compiling.* Prentice-Hall, Inc., USA, 1972. ISBN 0139145567.

Tamer Alkhouli and Hermann Ney. Biasing attention-based recurrent neural networks using external alignment information. In *Proceedings of the Second Conference on Machine Translation*, pages 108–117, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-4711.

Hala Almaghout, Jie Jiang, and Andy Way. CCG contextual labels in hierarchical phrase-based SMT. In *Proceedings of the 15th conference of the European Association for Machine Translation*, pages 281–288, 2011.

S. Arnoult and K. Sima'an. Modelling the Adjunct/Argument Distinction in Hierarchical Phrase-Based SMT. In *Proceedings of the 1st Deep Machine Transla-*

tion Workshop (DMTW 2015), pages 2–11, Praha, Czech Republic, September 2015.

Sophie Arnoult and Khalil Sima'an. Adjunct Alignment in Translation Data with an Application to Phrase-Based Statistical Machine Translation. In *Proceedings of the 16th Annual Conference of the European Association for Machine Translation*, pages 287–294, 2012.

Sophie Arnoult and Khalil Sima'an. Factoring Adjunction in Hierarchical Phrase-Based SMT. In *Proceedings of the 2nd Deep Machine Translation Workshop*, pages 11–20, Lisbon, Portugal, 2016. ISBN 978-80-88132-02-8.

Amittai Axelrod, Ra Birch Mayne, Chris Callison-burch, Miles Osborne, and David Talbot. Edinburgh system description for the 2005 iwslt speech translation evaluation. In *In Proc. International Workshop on Spoken Language Translation (IWSLT*, 2005.

Dzmitry Bahdanau, KyungHyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *Proceedings of ICLR*, 2015.

Jasmijn Bastings, Ivan Titov, Wilker Aziz, Diego Marcheggiani, and Khalil Sima'an. Graph convolutional encoders for syntax-aware neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1957–1967, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1209.

Taylor Berg-Kirkpatrick, Alexandre Bouchard-Côté, John DeNero, and Dan Klein. Painless unsupervised learning with features. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 582–590, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics. ISBN 1-932432-65-5.

Alexandra Birch and Miles Osborne. Reordering Metrics for MT. In *Proceedings of the Association for Computational Linguistics*, Portland, Oregon, USA, 2011.

Arianna Bisazza and Marcello Federico. A survey of word reordering in statistical machine translation: Computational models and language phenomena. *Computational Linguistics*, 42(2):163–205, 2016.

P. Brown, J. Cocke, S. Della Pietra, V. Della Pietra, F. Jelinek, R. Mercer, and P. Roossin. A statistical approach to language translation. In *Coling Budapest 1988 Volume 1: International Conference on Computational Linguistics*, 1988.

Peter F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Fredrick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. A statistical approach to machine translation. *Computational Linguistics*, 16(2): 79–85, 1990.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, 1993.

Emanuele Bugliarello and Naoaki Okazaki. Enhancing machine translation with dependency-aware self-attention. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1618–1627, Online, July 2020. Association for Computational Linguistics.

Marie Candito, Benoît Crabbé, and Pascal Denis. Statistical French dependency parsing: treebank conversion and first results. In *Proceedings of The seventh international conference on Language Resources and Evaluation (LREC)*, 2010.

Tyler A. Chang and Anna Rafferty. Encodings of source syntax: Similarities in NMT representations across target languages. In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 7–16, Online, July 2020. Association for Computational Linguistics.

Jean-Cédric Chappelier and Martin Rajman. A generalized CYK algorithm for parsing stochastic CFG. In *Proceedings of the 1st Workshop on Tabulation in Parsing and Deduction, TAPD'98, Paris, France, April 2-3, 1998*, pages 133–137, 1998.

Huadong Chen, Shujian Huang, David Chiang, and Jiajun Chen. Improved neural machine translation with a syntax-aware encoder and decoder. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1936–1945, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1177.

Stanley F. Chen and Joshua Goodman. An empirical study of smoothing techniques for language modeling. Technical Report TR-10-98, Computer Science Group, Harvard University, 1998.

Colin Cherry. Cohesive phrase-based decoding for statistical machine translation. In *In Proceedings of ACL-08: HLT*, pages 72–80, 2008.

Colin Cherry and George Foster. Batch Tuning Strategies for Statistical Machine Translation. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 427–436, Montréal, Canada, 2012.

David Chiang. Statistical Parsing with an Automatically-Extracted Tree Adjoining Grammar. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 456–463, 2000.

David Chiang. A Hierarchical Phrase-Based Model for Statistical Machine Translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 263–270, 2005.

David Chiang. Hierarchical phrase-based translation. *Computational Linguistics*, 33(2):201–228, June 2007. ISSN 0891-2017. doi: 10.1162/coli.2007.33.2.201.

David Chiang. Learning to Translate with Source and Target Syntax. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1443–1452, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

Noam Chomsky. *Syntactic Structures*. Mouton, The Hague, 1957.

Jonathan H Clark, Chris Dyer, Alon Lavie, and Noah A Smith. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 176–181. Association for Computational Linguistics, 2011.

Trevor Cohn, Cong Duy Vu Hoang, Ekaterina Vymolova, Kaisheng Yao, Chris Dyer, and Gholamreza Haffari. Incorporating structural alignment biases into an attentional neural translation model. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 876–885, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1102.

Michael Collins. Head-driven statistical models for natural language parsing. *Computational Linguistics*, 29(4):589–637, 2003.

Michael Collins, Philipp Koehn, and Ivona Kucerova. Clause restructuring for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 531–540, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. doi: 10.3115/1219840.1219906.

Anna Currey and Kenneth Heafield. Incorporating source syntax into transformer-based neural machine translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 24–33, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-5203.

Daniël de Kok, Jianqiang Ma, Corina Dima, and Erhard Hinrichs. PP attachment: Where do we stand? In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume*

2, Short Papers, pages 311–317, Valencia, Spain, April 2017. Association for Computational Linguistics.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977. ISSN 00359246.

Steve DeNeefe and Kevin Knight. Synchronous Tree Adjoining Machine Translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 727–736, 2009.

John DeNero and Jakob Uszkoreit. Inducing sentence structure from parallel corpora for reordering. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 193–203, Edinburgh, Scotland, UK., July 2011. Association for Computational Linguistics.

Bonnie J. Dorr. Machine Translation Divergences: A Formal Description and Proposed Solution. *Computational Linguistics*, 20(4):597–633, 1994.

David Dowty. The dual analysis of adjuncts/complements in categorial grammar. *ZAS Papers in Linguistics*, (17), 2000.

Matthew S. Dryer and Martin Haspelmath, editors. *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig, 2013.

Jay Earley. An efficient context-free parsing algorithm. *Commun. ACM*, 13(2): 94–102, February 1970. ISSN 0001-0782. doi: 10.1145/362007.362035.

Andreas Eisele and Yu Chen. MultiUN: A Multilingual Corpus from United Nation Documents. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, 2010. ISBN 2-9517408-6-7.

Akiko Eriguchi, Yoshimasa Tsuruoka, and Kyunghyun Cho. Learning to parse and translate improves neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 72–78, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-2012.

Charles J. Fillmore. Frame semantics and the nature of language. *Annals of the New York Academy of Sciences: Conference on the Origin and Development of Language and Speech*, 280(1):20–32, 1976.

Charles J. Fillmore. *Frame semantics*, pages 111–137. Hanshin Publishing Co., Seoul, South Korea, 1982.

Charles J. Fillmore. Frames and the semantics of understanding. *Quaderni di Semantica*, 6(2):222–254, 1985.

Charles J. Fillmore and Collin F. Baker. Frame semantics for text understanding. In *Proceedings of WordNet and Other Lexical Resources Workshop*, Pittsburgh, June 2001. NAACL, NAACL.

Heidi J. Fox. Phrasal cohesion and statistical machine translation. In *Proceedings of EMNLP-02*, pages 304–311, 2002.

Michel Galley, Mark Hopkins, Kevin Knight, and Daniel Marcu. What's in a translation rule? In Daniel Marcu Susan Dumais and Salim Roukos, editors, *HLT-NAACL 2004: Main Proceedings*, pages 273–280, Boston, Massachusetts, USA, May 2 - May 7 2004. Association for Computational Linguistics.

Michel Galley, Jonathan Graehl, Kevin Knight, Daniel Marcu, Steve DeNeefe, Wei Wang, and Ignacio Thayer. Scalable inference and training of context-rich syntactic translation models. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 961–968, Sydney, Australia, 2006. Association for Computational Linguistics. doi: 10.3115/1220175.1220296.

Dmitriy Genzel. Automatically learning source-side reordering rules for large scale machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 376–384, Beijing, China, August 2010. Coling 2010 Organizing Committee.

Jonathan Graehl and Kevin Knight. Training tree transducers. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 105–112, Boston, Massachusetts, USA, May 2 - May 7 2004. Association for Computational Linguistics.

Jonathan Graehl, Kevin Knight, and Jonathan May. Training tree transducers. *Computational Linguistics*, 34(3):391–427, 2008. doi: 10.1162/coli.2008.07-051-R2-03-57.

Clayton Greenberg. Disambiguating prepositional phrase attachment sites with sense information captured in contextualized distributional data. In *Proceedings of the ACL 2014 Student Research Workshop*, pages 71–77, Baltimore, Maryland, USA, June 2014. Association for Computational Linguistics.

Greg Hanneman and Alon Lavie. Improving syntax-augmented machine translation by coarsening the label set. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics:*

Human Language Technologies, pages 288–297, Atlanta, Georgia, June 2013. Association for Computational Linguistics.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. Scalable modified Kneser-Ney language model estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 690–696, Sofia, Bulgaria, 2013.

Liang Huang, Kevin Knight, and Aravind Joshi. A syntax-directed translator with extended domain of locality. In *Proceedings of the Workshop on Computationally Hard Problems and Joint Inference in Speech and Language Processing*, pages 1–8, New York City, New York, June 2006. Association for Computational Linguistics.

Zhongqiang Huang, Martin Cmejrek, and Bowen Zhou. Soft syntactic constraints for hierarchical phrase-based translation using latent syntactic distributions. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 138–147. Association for Computational Linguistics, 2010.

Matthias Huck, Stephan Peitz, Markus Freitag, and Hermann Ney. Discriminative Reordering Extensions for Hierarchical Phrase-Based Machine Translation. In *16th Annual Conference of the European Association for Machine Translation*, pages 313–320, Trento, Italy, 2012.

Rebecca Hwa, Philip Resnik, Amy Weinberg, and Okan Kolak. Evaluating Translational Correspondence Using Annotation Projection. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 392–399, 2002. doi: 10.3115/1073083.1073149.

Hideki Isozaki, Tsutomu Hirao, Kevin Duh, Katsuhito Sudoh, and Hajime Tsukada. Automatic evaluation of translation quality for distant language pairs. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 944–952, Cambridge, MA, October 2010a. Association for Computational Linguistics.

Hideki Isozaki, Katsuhito Sudoh, Hajime Tsukada, and Kevin Duh. Head finalization: A simple reordering rule for SOV languages. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, pages 244–251, Uppsala, Sweden, July 2010b. Association for Computational Linguistics.

Richard Johansson and Pierre Nugues. Extended Constituent-to-dependency Conversion for English. In *Proceedings of NODALIDA 2007*, pages 105–112, Tartu, Estonia, May 25-26 2007.

Aravind K. Joshi. *Natural Language Parsing*, chapter Tree Adjoining Grammars: How Much Context-Sensitivity Is Required to Provide Reasonable Structural Descriptions?, pages 206–250. Cambridge University Press, 1985.

Aravind K. Joshi and Yves Schabes. Tree-Adjoining Grammars. In G. Rosenberg and A. Salomaa, editors, *Handbook of Formal Languages*. Springer-Verlag, New York, NY, 1997.

Aravind K. Joshi, Leon S. Levy, and Masako Takahashi. Tree Adjunct Grammars. *Journal of Computer and System Sciences*, 10(1):136–163, 1975. ISSN 0022-0000. doi: 10.1016/S0022-0000(75)80019-5.

Ronald Kaplan and Joan Bresnan. Lexical-functional grammar: A formal system for grammatical representation. In *The Mental Representation of Grammatical Relations*, pages 173–281. MIT Press, Cambridge, MA, 1982.

Paul Kay. Argument structure constructions and the argument-adjunct distinction. In Mirjam Fried and Hans Christian Boas, editors, *Grammatical Constructions: Back to the Roots*. John Benjamins Publishing Company, 2005.

Maxim Khalilov and Khalil Sima'an. Statistical translation after source reordering: Oracles, context-aware models, and empirical analysis. *Natural Language Engineering*, 18:491–519, 9 2012. ISSN 1469-8110. doi: 10.1017/S1351324912000162.

Dan Klein and Christopher D. Manning. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 423–430, Sapporo, Japan, July 2003. Association for Computational Linguistics. doi: 10.3115/1075096.1075150.

Reinhard Kneser and Hermann Ney. Improved backing-off for m-gram language modeling. In *International Conference on Acoustics, Speech, and Signal Processing*, volume 1, 1995. doi: 10.1109/ICASSP.1995.479394.

Kevin Knight. Squibs and discussions: Decoding complexity in word-replacement translation models. *Computational Linguistics*, 25(4), 1999.

Philipp Koehn, Franz J. Och, and Daniel Marcu. Statistical phrase-based translation. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 127–133, 2003.

Anthony Kroch and Aravind Joshi. The Linguistic Relevance of Tree Adjoining Grammars. Technical Report MC CIS 85 18, Department of Computer and Information Science, University of Pennsylvania, 1985.

K. Lari and S. J. Young. The estimation of stochastic context-free grammars using the inside-outside algorithm. *Computer Speech and Language*, (4):35–56, 1990.

Uri Lerner and Slav Petrov. Source-side classifier preordering for machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 513–523, Seattle, Washington, USA, October 2013. Association for Computational Linguistics.

Mike Lewis and Mark Steedman. A* CCG Parsing with a Supertag-factored Model. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 990–1000. Association for Computational Linguistics, 10 2014.

P. M. Lewis and R. E. Stearns. Syntax-directed transduction. *J. ACM*, 15(3): 465?488, July 1968. ISSN 0004-5411. doi: 10.1145/321466.321477.

Junhui Li, Zhaopeng Tu, Guodong Zhou, and Josef van Genabith. Using Syntactic Head Information in Hierarchical Phrase-Based Translation. In *Proceedings of the 7th Workshop on Statistical Machine Translation*, pages 232–242, 2012.

Junhui Li, Philip Resnik, and Hal Daumé III. Modeling Syntactic and Semantic Structures in Hierarchical Phrase-based Translation. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 540–549, Atlanta, Georgia, 2013.

Junhui Li, Deyi Xiong, Zhaopeng Tu, Muhua Zhu, Min Zhang, and Guodong Zhou. Modeling source syntax for neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 688–697, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1064.

Zhifei Li, Chris Callison-Burch, Chris Dyer, Sanjeev Khudanpur, Lane Schwartz, Wren Thornton, Jonathan Weese, and Omar Zaidan. Joshua: An open source toolkit for parsing-based machine translation. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 135–139, Athens, Greece, 2009.

Percy Liang, Ben Taskar, and Dan Klein. Alignment by Agreement. In *Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, HLT-NAACL '06, pages 104–111, 2006. doi: 10.3115/1220835.1220849.

Yang Liu, Qun Liu, and Shouxun Lin. Tree-to-string alignment template for statistical machine translation. In *Proceedings of the 21st International Conference* *on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 609–616, Sydney, Australia, July 2006. Association for Computational Linguistics. doi: 10.3115/1220175.1220252.

Yang Liu, Yun Huang, Qun Liu, and Shouxun Lin. Forest-to-string statistical translation rules. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 704–711, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

Yang Liu, Qun Liu, and Yajuan Lü. Adjoining tree-to-string translation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 1278–1287, 2011. ISBN 978-1-932432-87-9.

Pranava Swaroop Madhyastha, Xavier Carreras, and Ariadna Quattoni. Prepositional phrase attachment over word embedding products. In *Proceedings of the 15th International Conference on Parsing Technologies*, pages 32–43, Pisa, Italy, September 2017. Association for Computational Linguistics.

Gideon Maillette de Buy Wenniger and Khalil Sima'an. Hierarchical Alignment Decomposition Labels for Hiero Grammar Rules. In *Proceedings of the Seventh Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 19–28, Atlanta, Georgia, 2013.

Christopher D. Manning. Probabilistic syntax. In J. Hay Rens Bod and S. Jannedy, editors, *Probabilistic Linguistics*, pages 289–341. MIT Press, Cambridge, MA, 2003.

Daniel Marcu and William Wong. A Phrase-Based, Joint Probability Model for Statistical Machine Translation. In *Proceedings of EMNLP*, 2002.

Yuval Marton and Philip Resnik. Soft syntactic constraints for hierarchical phrased-based translation. In *Proceedings of ACL-08: HLT*, pages 1003–1011, Columbus, Ohio, June 2008. Association for Computational Linguistics.

Paola Merlo. Generalised PP-attachment disambiguation using corpus-based linguistic diagnostics. In *10th Conference of the European Chapter of the Association for Computational Linguistics*, 2003.

Haitao Mi, Liang Huang, and Qun Liu. Forest-based translation. In *Proceedings of ACL-08: HLT*, pages 192–199, Columbus, Ohio, June 2008. Association for Computational Linguistics.

Robert C. Moore. Improving ibm word-alignment model 1. In *Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics. doi: 10.3115/1218955.1219021.

M.T. Rosetta. *Compositional translation.* The Kluwer international series in engineering and computer science. SECS. Kluwer Academic Publishers, Netherlands, 1994. ISBN 0-7923-9462-3. doi: 10.1007/978-94-015-8306-0. M.T. Rosetta is auteurscollectief.

Markos Mylonakis and Khalil Sima'an. Learning Probabilistic Synchronous CFGs for Phrase-based Translation. In *In Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 117–125, 2010.

Markos Mylonakis and Khalil Sima'an. Learning Hierarchical Translation Structure with Linguistic Annotation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 642–652, 2011.

Makoto Nagao. A Framework of a Mechanical Translation between Japanese and English by Analogy Principle. In *Proceedings of the international NATO symposium on Artificial and human intelligence*, pages 173–180. Elsevier North-Holland, Inc., 1984.

Rebecca Nesson, Stuart M. Shieber, and Alexander Rush. Induction of probabilistic synchronous tree-insertion grammars for machine translation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas (AMTA 2006)*, Boston, Massachusetts, 8–12 August 2006.

Graham Neubig, Taro Watanabe, and Shinsuke Mori. Inducing a discriminative parser to optimize machine translation reordering. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 843–853, Jeju Island, Korea, July 2012. Association for Computational Linguistics.

Dmitry Nikolaev, Ofir Arviv, Taelin Karidi, Neta Kenneth, Veronika Mitnik, Lilja Maria Saeboe, and Omri Abend. Fine-Grained Analysis of Cross-Linguistic Syntactic Divergences. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1159–1176, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.109.

Franz Josef Och. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167, Sapporo, Japan, July 2003. Association for Computational Linguistics. doi: 10.3115/1075096.1075117.

Franz Josef Och and Hermann Ney. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 295–302, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073133.

Franz Josef Och and Hermann Ney. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29:19–51, 2003.

Franz Josef Och, Christoph Tillmann, and Hermann Ney. Improved alignment models for statistical machine translation. In *1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 1999.

Marian Olteanu and Dan Moldovan. PP-attachment disambiguation using large context. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 273–280, Vancouver, British Columbia, Canada, October 2005. Association for Computational Linguistics.

Sebastian Padó and Mirella Lapata. Cross-lingual Annotation Projection for Semantic Roles. *Journal of Artificial Intelligence Research*, 36:307–340, 2009.

Martha Palmer, Dan Gildea, and Paul Kingsbury. The Proposition Bank: A Corpus Annotated with Semantic Roles. *Computational Linguistics Journal*, 31(1):71–105, 2005.

Patrick Pantel and Dekang Lin. An unsupervised approach to prepositional phrase attachment using contextually similar words. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 101–108, Hong Kong, October 2000. Association for Computational Linguistics. doi: 10.3115/1075218.1075232.

Kishore Papineni, Salim Roukos, Todd Ward, and Zhu Wei-Jing. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318, 2002.

Barbara H. Partee and Vladimir Borschev. Genitives, relational nouns, and argument-modifier ambiguity. In C. Maienborn E. Lang and C. Fabricius-Hansen, editors, *Modifying Adjuncts (Interface Explorations 4)*. Mouton de Gruyter, 2003.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL-44, pages 433–440, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics. doi: 10.3115/1220175.1220230.

Philip Resnik. Probabilistic tree-adjoining grammar as a framework for statistical natural language processing. In *COLING 1992 Volume 2: The 15th International Conference on Computational Linguistics*, 1992.

Danielle Saunders, Felix Stahlberg, Adrià de Gispert, and Bill Byrne. Multi-representation ensembles and delayed SGD updates improve syntax-based NMT. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 319–325, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2051.

Yves Schabes and Richard C. Waters. Tree insertion grammar: A cubic-time, parsable formalism that lexicalizes context-free grammar without changing the trees produced. *Computational Linguistics*, 21(4):479–513, December 1995. ISSN 0891-2017.

Yves Schabes, Anne Abeillé, and Aravind K. Joshi. Parsing strategies with 'lexicalized' grammars: Application to tree adjoining grammars. In *Coling Budapest 1988 Volume 2: International Conference on Computational Linguistics*, 1988.

Rico Sennrich and Barry Haddow. Linguistic input features improve neural machine translation. In *Proceedings of the First Conference on Machine Translation: Volume 1, Research Papers*, pages 83–91, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/W16-2209.

Xing Shi, Inkit Padhi, and Kevin Knight. Does string-based neural MT learn source syntax? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1526–1534, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1159.

Stuart Shieber and Yves Schabes. Synchronous Tree-Adjoining Grammars. In *Handbook of Formal Languages*, pages 69–123. Springer, 1990.

Stuart M. Shieber. Probabilistic Synchronous Tree-Adjoining Grammars for Machine Translation: The Argument from Bilingual Dictionaries. In *Proceedings of the Workshop on Syntax and Structure in Statistical Translation*, Rochester, New York, 2007.

Dan I. Slobin. The Many Ways to Search for a Frog: Linguistic Typology and the Expression of Motion Events. In *Relating events in narrative, Volume 2: Typological and contextual perspectives.*, pages 219–257. Lawrence Erlbaum Associates Publishers, Mahwah, NJ, US, 2004. ISBN 0-8058-4672-7 (Hardcover).

M. Snover, Bonnie J Dorr, R. Schwartz, L. Micciulla, and J. Makhoul. A study of translation edit rate with targeted human annotation. *Proceedings of Association for Machine Translation in the Americas*, pages 223 – 231, 2006/// 2006.

Miloš Stanojević and Khalil Sima'an. Fitting sentence level translation evaluation with many dense features. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 202–206, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1025.

Miloš Stanojević and Khalil Sima'an. Reordering Grammar Induction. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 44–54, Lisbon, Portugal, 2015.

Mark Steedman. *The Syntactic Process*. MIT Press, Cambridge, MA., 2000.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, 2014.

Leonard Talmy. Path to Realization: A Typology of Event Conflation. *Annual Meeting of the Berkeley Linguistics Society*, 17(1):480, July 1991. ISSN 2377-1666, 0363-2946. doi: 10.3765/bls.v17i0.1620.

Gongbo Tang, Mathias Müller, Annette Rios, and Rico Sennrich. Why self-attention? a targeted evaluation of neural machine translation architectures. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4263–4272, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1458.

Christoph Tillmann. A unigram orientation model for statistical machine translation. In *Proceedings of HLT-NAACL 2004: Short Papers*, pages 101–104, Boston, Massachusetts, USA, May 2 - May 7 2004. Association for Computational Linguistics.

Roy Tromble and Jason Eisner. Learning linear ordering problems for better translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1007–1016, Singapore, August 2009. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc., 2017.

Stephan Vogel, Hermann Ney, and Christoph Tillmann. Hmm-based word alignment in statistical translation. In *Proceedings of the 16th Conference on Computational Linguistics - Volume 2*, COLING '96, pages 836–841, Stroudsburg, PA, USA, 1996. Association for Computational Linguistics. doi: 10.3115/993268.993313.

Wei Wang, Kevin Knight, and Daniel Marcu. Binarizing syntax trees to improve syntax-based machine translation accuracy. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 746–754, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

Wei Wang, Jonathan May, Kevin Knight, and Daniel Marcu. Re-structuring, Re-labeling, and Re-aligning for Syntax-Based Machine Translation. *Computational Linguistics*, 36:247–278, 2010.

Ethan Wilcox, Roger Levy, Takashi Morita, and Richard Futrell. What do RNN language models learn about filler–gap dependencies? In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 211–221, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5423.

Dekai Wu. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3):377–404, 1997.

Dekai Wu and Pascale Fung. Semantic roles for smt: A hybrid two-pass model. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings (Short Papers)*, pages 13–16, 2009.

Fei Xia and Michael McCord. Improving a statistical mt system with automatically learned rewrite patterns. In *Proceedings of Coling 2004*, pages 508–514, Geneva, Switzerland, Aug 23–Aug 27 2004. COLING.

Deyi Xiong, Min Zhang, and Haizhou Li. Modeling the translation of predicate-argument structure for smt. In *ACL (1)*, pages 902–911, 2012.

Hongfei Xu, Josef van Genabith, Deyi Xiong, Qiuhui Liu, and Jingyi Zhang. Learning source phrase representations for neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 386–396, Online, July 2020. Association for Computational Linguistics.

Kenji Yamada and Kevin Knight. A syntax-based statistical translation model. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, ACL '01, pages 523–530, Stroudsburg, PA, USA, 2001. Association for Computational Linguistics.

Victor H Yngve. A framework for syntactic translation. *Mechanical Translation*, 4(3):59–65, December 1957.

Hao Zhang and Daniel Gildea. Factorization of synchronous context-free grammars in linear time. In *Proceedings of SSST, NAACL-HLT 2007 / AMTA Workshop on Syntax and Structure in Statistical Translation*, pages 25–32, Rochester, New York, April 2007. Association for Computational Linguistics.

Hao Zhang, Liang Huang, Daniel Gildea, and Kevin Knight. Synchronous binarization for machine translation. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 256–263, New York City, USA, June 2006. Association for Computational Linguistics.

Hao Zhang, Daniel Gildea, and David Chiang. Extracting synchronous grammar rules from word-level alignments in linear time. In *In Proceedings of the 22nd International Conference on Computational Linguistics (COLING-08)*, 2008.

Jinchao Zhang, Mingxuan Wang, Qun Liu, and Jie Zhou. Incorporating word reordering knowledge into attention-based neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1524–1534, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1140.

Chunting Zhou, Xuezhe Ma, Junjie Hu, and Graham Neubig. Handling syntactic divergence in low-resource machine translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1388–1394, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1143.

Andreas Zollmann and Ashish Venugopal. Syntax Augmented Machine Translation via Chart Parsing. In *Proceedings of NAACL 2006 - Workshop on statistical machine translation*, pages 138–141, 2006.

Samenvatting

Hiëriarchische frase-gebaseerde statistische vertalingsmodellen zijn compositioneel doordat ze steunen op formele, Synchrone Context-Vrije Grammatica. Er is echter geen garantie dat de geproduceerde vertalingen zelf compositioneel zijn. Linguïstische verrijkingsmetoden gebruiken vaak syntactische aanwijzingen om meer controle te bieden op doelherschrijvingen of om vertaalregels op hun bronzijde te selecteren, maar deze aanwijzingen beschrijven alleen een kant van de vertaaldata, en hebben op hun beurt weinig met compositionele vertalingsequivalentie te maken.

Hiërarchische frase-gebaseerde vertalingsmodellen steunen daarnaast op asyntactische, gelexicaliseerde regels. Dit bevordert het modelleren van idiomatische uitdrukkingen en complexe woordherschikkingen, maar leidt ook tot zeer grote vertaal grammatica's. Om dit te vermijden wordt de spanwijdte van frasen normaal beperkt, met de veronderstelling dat de meeste bruikbare vertalingsequivalenties lokaal zijn te vangen. De expressiviteit van hiërarchische modellen wordt hiermee natuurlijk ook beperkt, zodat het structureren van de herordeningsruimte een open vraag blijft voor deze modellen en statistiche machinevertaling (SMT) in het algemeen.

Dit proefschrift neemt adjunctie als bron van linguïstiche informatie voor hiërarchische frase-gebaseerde modellen. Adjunctie wordt gezien als sturend voor recursie in Tree-Adjoining Grammar (Joshi et al., 1975; Joshi and Schabes, 1997), waar het ingezet is om linguïstische fenomenen zoals *wh*-fronting te beschrijven en om syntactische modificering te abstraheren. Synchrone Tree-Adjoining Grammar stelt voor adjunctie simultaan toe te passen, en zo als drijvende kracht te zien voor recursie in vertaling. Terwijl synchrone adjunctie toepassing heeft gevonden in syntax-gebaseerde statistische machinevertaling, is het nooit benut geweest in hiërarchische frase-gebaseerde SMT, mogelijk door de schijnbare tegenstelling tussen het formele karakter van adjunctie en de asyntactische natuur van frasegebaseerde modellen.

Dit proefschrift beschouwt adjuncten, de constituënten die betrokken zijn bij adjunctie (syntactische modificeerders in het algemeen), and hun role in compositionele, frase-gebaseerde machinevertaling. Dit proefschrift contribuëert het volgende:

- een studie van adjunctequivalentie in frans-engels machinevertalingsdata, om te toetsen in hoeverre adjunctie synchroon verloopt in vertaaldata. Deze studie betracht referentiemetingen en empirische metingen van synchrone adjunctie in verhouding te brengen om het effect van automatische woord-alignments en parse-gebaseerde adjunct-identificeringsregels in kaart te brengen.

- een extensie van Hiero (Chiang, 2005), waar adjuncten ingezet worden om spanwijdte beperkingen te ontspannen, en adjunctoptionaliteit om het vertaalgrammatica te verrijken door het abstraheren van adjuncten.

- een extensie van de latente herordening PCFG grammatica van Stanojević and Sima'an (2015), waar adjuncten ingezet worden als linguïstische informatiebron voor herordening.

Onze corpusstudie bevestigt dat adjunctie tot hoge mate synchroon verloopt in vertaaldata. En terwijl deze studie alleen frans-engels data betreft, laat het ook zien dat synchrone adjunctie niet alleen berust of syntactische gelijkenis, maar dat het ook een semantische basis heeft en dat het zich tot vertaalcompositionaliteit verhoudt.

De voorgestelde extensie van Hiero laat zien dat adjuncten niet alleen nuttig zijn voor het sturen van recursie in hiërarchische frase-gebaseerde modellen door het gewin van langbereik vertaalregels, maar dat adjunctoverschrijdingsbeperkingen ook kortbereik regels effectief filteren. Experimenten met adjunct optionaliteit geven ook veelbelovende resultaten voor engels-japans, en laten zien dat het toepassen van adjunctie niet voorbehouden is aan syntax-gebaseerde SMT. Verdere analyse laat echter zien dat wat regelwinningbeperkingen betreft, constituentie in het algemeen een betere informatiebron vormt dan adjunctie.

Bij het inzetten van adjuncten in herordeningsmodellen ziet men weer dat adjuncten nuttig zijn, maar dat constituentie in het algemeen een betere informatiebron vormt. Wel ziet men dat adjuncten het grootste deel van herordening in engels-japans verklaren.

In het algemeen laat dit proefschrift zien dat adjunctie niet voorbehouden is aan syntax-gebaseerde vertalingsmodellen, omdat adjuncteigenschappen zoals optionaliteit even goed in asyntactische modellen ingezet kunnen worden. Het maakt tegelijkertijd ook een beroep op meer vertrouwen in linguïstische hoofdbeginselen van recursie, om te beginnen met constituentie.

Abstract

Hierarchical Phrase-Based SMT models are compositional by reliance on a formal, Synchronous Context-Free Grammar. There is however no guarantee that target-side translations are compositional. Linguistic enrichment methods often exploit syntactic cues for better target rewritings or source-side rule selection, but these cues generally describe one side of the data, and have in turn little bearing on compositional translation equivalence.

Besides, while reliance on lexicalized rules allows for modelling complex reorderings, unconstrained extraction of these rules potentially leads to very large grammars. This risk is normally circumvented by applying span-length constraints, under the assumption that short-range rules capture the most useful translation relations. These constraints naturally limit the expressiveness of hierarchical phrase-based models in the long range. Structuring the reordering space remains an open problem for these models and for SMT in general.

This dissertation takes adjunction as a source of linguistic information for hierarchical phrase-based models. Adjunction is regarded as the main driver of recursion in Tree-Adjoining Grammar (Joshi et al., 1975; Joshi and Schabes, 1997), allowing to explain linguistic phenomena like *wh*-fronting without resorting to transformations and, generally, to abstract syntactic modification away. Synchronous Tree-Adjoining Grammar proposes to view adjunction as applying synchronously, thus driving recursion in translation. While synchronous adjunction has been applied in Syntax-Based SMT, it has been left unexploited in Hierarchical Phrase-based SMT, possibly because of the apparent contradiction between the formal character of adjunction and the asyntactic nature of phrase-based models.

This dissertation considers the constituents involved in adjunction—adjuncts, in the broad sense of syntactic modifiers—and investigates which part they play in compositional, phrase-based translation. This dissertation makes the following contributions:

- a corpus study of adjunct alignment in French-English data, to assess the degree to which adjunction is synchronous in translation data. This study further relates gold measures of synchronous adjunction to empirical measures based on word alignments and parse-based adjunct-identification heuristics.

- an extension to Hiero (Chiang, 2005), where adjuncts are utilized to guide extraction, by fully relaxing span constraints around adjuncts. We further exploit adjunct optionality by factoring out adjuncts at rule extraction.
- an extension to the latent reordering PCFG grammar of Stanojević and Sima'an (2015) for translation preordering, where adjuncts are utilized to inform reordering.

Our corpus study confirms that adjuncts are synchronous to a high degree in translation data, at least for English-French. We show that synchronous adjunction is not solely indebted to syntactic similarity however, but also has a semantic basis and reflects translation compositionality.

The proposed extension to Hiero shows that adjuncts are useful for driving recursion in hierarchical phrase-based models, as adjunct-crossing constraints not only allow for useful long-range rules, but also effectively filter short-range rules. Experiments with adjunct optionality further provide promising results for English-Japanese, showing that adjunction can be applied outside of Syntax-Based SMT. Comparing adjuncts and constituents for extraction constraints reveals however that constituency tends to be the better guide.

Our reordering modelling experiments confirm this result, showing notably for English-Japanese that adjuncts explain most, but not all translation reordering when compared to constituents.

On the whole, this dissertation shows that adjunction need not be confined to syntax-based models of translation, as properties of adjunction like optionality can be exploited in asyntactic models as well. But it also calls for a more extensive reliance of data-based translation models on core linguistic principles of recursion—starting with constituency.

Titles in the ILLC Dissertation Series:

ILLC DS-2016-01: **Ivano A. Ciardelli**
Questions in Logic

ILLC DS-2016-02: **Zoé Christoff**
Dynamic Logics of Networks: Information Flow and the Spread of Opinion

ILLC DS-2016-03: **Fleur Leonie Bouwer**
What do we need to hear a beat? The influence of attention, musical abilities, and accents on the perception of metrical rhythm

ILLC DS-2016-04: **Johannes Marti**
Interpreting Linguistic Behavior with Possible World Models

ILLC DS-2016-05: **Phong Lê**
Learning Vector Representations for Sentences - The Recursive Deep Learning Approach

ILLC DS-2016-06: **Gideon Maillette de Buy Wenniger**
Aligning the Foundations of Hierarchical Statistical Machine Translation

ILLC DS-2016-07: **Andreas van Cranenburgh**
Rich Statistical Parsing and Literary Language

ILLC DS-2016-08: **Florian Speelman**
Position-based Quantum Cryptography and Catalytic Computation

ILLC DS-2016-09: **Teresa Piovesan**
Quantum entanglement: insights via graph parameters and conic optimization

ILLC DS-2016-10: **Paula Henk**
Nonstandard Provability for Peano Arithmetic. A Modal Perspective

ILLC DS-2017-01: **Paolo Galeazzi**
Play Without Regret

ILLC DS-2017-02: **Riccardo Pinosio**
The Logic of Kant's Temporal Continuum

ILLC DS-2017-03: **Matthijs Westera**
Exhaustivity and intonation: a unified theory

ILLC DS-2017-04: **Giovanni Cinà**
Categories for the working modal logician

ILLC DS-2017-05: **Shane Noah Steinert-Threlkeld**
Communication and Computation: New Questions About Compositionality

ILLC DS-2017-06: **Peter Hawke**
The Problem of Epistemic Relevance

ILLC DS-2017-07: **Aybüke Özgün**
Evidence in Epistemic Logic: A Topological Perspective

ILLC DS-2017-08: **Raquel Garrido Alhama**
Computational Modelling of Artificial Language Learning: Retention, Recognition & Recurrence

ILLC DS-2017-09: **Miloš Stanojević**
Permutation Forests for Modeling Word Order in Machine Translation

ILLC DS-2018-01: **Berit Janssen**
Retained or Lost in Transmission? Analyzing and Predicting Stability in Dutch Folk Songs

ILLC DS-2018-02: **Hugo Huurdeman**
Supporting the Complex Dynamics of the Information Seeking Process

ILLC DS-2018-03: **Corina Koolen**
Reading beyond the female: The relationship between perception of author gender and literary quality

ILLC DS-2018-04: **Jelle Bruineberg**
Anticipating Affordances: Intentionality in self-organizing brain-body-environment systems

ILLC DS-2018-05: **Joachim Daiber**
Typologically Robust Statistical Machine Translation: Understanding and Exploiting Differences and Similarities Between Languages in Machine Translation

ILLC DS-2018-06: **Thomas Brochhagen**
Signaling under Uncertainty

ILLC DS-2018-07: **Julian Schlöder**
Assertion and Rejection

ILLC DS-2018-08: **Srinivasan Arunachalam**
Quantum Algorithms and Learning Theory

ILLC DS-2018-09: **Hugo de Holanda Cunha Nobrega**
Games for functions: Baire classes, Weihrauch degrees, transfinite computations, and ranks

ILLC DS-2018-10: **Chenwei Shi**
Reason to Believe

ILLC DS-2018-11: **Malvin Gattinger**
New Directions in Model Checking Dynamic Epistemic Logic

ILLC DS-2018-12: **Julia Ilin**
Filtration Revisited: Lattices of Stable Non-Classical Logics

ILLC DS-2018-13: **Jeroen Zuiddam**
Algebraic complexity, asymptotic spectra and entanglement polytopes

ILLC DS-2019-01: **Carlos Vaquero**
What Makes A Performer Unique? Idiosyncrasies and commonalities in expressive music performance

ILLC DS-2019-02: **Jort Bergfeld**
Quantum logics for expressing and proving the correctness of quantum programs

ILLC DS-2019-03: **András Gilyén**
Quantum Singular Value Transformation & Its Algorithmic Applications

ILLC DS-2019-04: **Lorenzo Galeotti**
The theory of the generalised real numbers and other topics in logic

ILLC DS-2019-05: **Nadine Theiler**
Taking a unified perspective: Resolutions and highlighting in the semantics of attitudes and particles

ILLC DS-2019-06: **Peter T.S. van der Gulik**
Considerations in Evolutionary Biochemistry

ILLC DS-2019-07: **Frederik Möllerström Lauridsen**
Cuts and Completions: Algebraic aspects of structural proof theory

ILLC DS-2020-01: **Mostafa Dehghani**
Learning with Imperfect Supervision for Language Understanding

ILLC DS-2020-02: **Koen Groenland**
Quantum protocols for few-qubit devices

ILLC DS-2020-03: **Jouke Witteveen**
Parameterized Analysis of Complexity

ILLC DS-2020-04: **Joran van Apeldoorn**
A Quantum View on Convex Optimization

ILLC DS-2020-05: **Tom Bannink**
Quantum and stochastic processes

ILLC DS-2020-06: **Dieuwke Hupkes**
Hierarchy and interpretability in neural models of language processing

ILLC DS-2020-07: **Ana Lucia Vargas Sandoval**
On the Path to the Truth: Logical & Computational Aspects of Learning

ILLC DS-2020-08: **Philip Schulz**
Latent Variable Models for Machine Translation and How to Learn Them

ILLC DS-2020-09: **Jasmijn Bastings**
A Tale of Two Sequences: Interpretable and Linguistically-Informed Deep Learning for Natural Language Processing

ILLC DS-2020-10: **Arnold Kochari**
Perceiving and communicating magnitudes: Behavioral and electrophysiological studies

ILLC DS-2020-11: **Marco Del Tredici**
Linguistic Variation in Online Communities: A Computational Perspective

ILLC DS-2020-12: **Bastiaan van der Weij**
Experienced listeners: Modeling the influence of long-term musical exposure on rhythm perception

ILLC DS-2020-13: **Thom van Gessel**
Questions in Context

ILLC DS-2020-14: **Gianluca Grilletti**
Questions & Quantification: A study of first order inquisitive logic

ILLC DS-2020-15: **Tom Schoonen**
Tales of Similarity and Imagination. A modest epistemology of possibility

ILLC DS-2020-16: **Ilaria Canavotto**
Where Responsibility Takes You: Logics of Agency, Counterfactuals and Norms

Institute For Logic, Language and Computation