Syntax-Based Alignment: Supervised or Unsupervised?

Hao Zhang and Daniel Gildea
Computer Science Department
University of Rochester
Rochester, NY 14627

Abstract

Tree-based approaches to alignment model translation as a sequence of probabilistic operations transforming the syntactic parse tree of a sentence in one language into that of the other. The trees may be learned directly from parallel corpora (Wu, 1997), or provided by a parser trained on hand-annotated treebanks (Yamada and Knight, 2001). In this paper, we compare these approaches on Chinese-English and French-English datasets, and find that automatically derived trees result in better agreement with human-annotated word-level alignments for unseen test data.

1 Introduction

Statistical approaches to machine translation, pioneered by Brown et al. (1990), estimate parameters for a probabilistic model of word-to-word correspondences and word re-orderings directly from large corpora of parallel bilingual text. In recent years, a number of syntactically motivated approaches to statistical machine translation have been proposed. These approaches assign a parallel tree structure to the two sides of each sentence pair, and model the translation process with reordering operations defined on the tree structure. The tree-based approach allows us to represent the fact that syntactic constituents tend to move as unit, as well as systematic differences in word order in the grammars of the two languages. Furthermore, the tree structure allows us to make probabilistic independence assumptions that result in polynomial time algorithms for estimating a translation model from parallel training data, and for finding the highest probability translation given a new sentence.

Wu (1997) modeled the reordering process with binary branching trees, where each production could be either in the same or in reverse order going from source to target language. The trees of Wu's Inversion Transduction Grammar were derived by synchronously parsing a parallel corpus, using a grammar with lexical translation probabilities at the leaves and a simple grammar with a single nonterminal providing the tree structure. While this grammar did not represent traditional syntactic categories such as verb phrases and noun phrases, it served to restrict the word-level alignments considered by the system to those allowable by reordering operations on binary trees. This restriction corresponds to intuitions about the alignments that could be produced by systematic differences between the two language's grammars, and allows for a polynomial time algorithm for finding the highest-probability alignment, and for re-estimation of the lexical translation and grammar probabilities using the Expectation Maximization algorithm.

Yamada and Knight (2001) present an algorithm for estimating probabilistic parameters for a similar model which represents translation as a sequence of re-ordering operations over children of nodes in a syntactic tree, using automatic parser output for the initial tree structures. This gives the translation model more information about the structure of the source language, and further constrains the reorderings to match not just a possible bracketing as in Wu (1997), but the specific bracketing of the parse tree provided.

In this paper, we make a direct comparison of a *syntactically unsupervised* alignment model, based on Wu (1997), with a *syntactically supervised* model, based on Yamada and Knight (2001). We use the term *syntactically supervised* to indicate that the syntactic structure in one language is given to the training procedure. It is important to note, however, that both algorithms are unsupervised in that they are not provided any hand-aligned training data. Rather, they both use Expectation Maximization to find an alignment model by iteratively improving the likelihood assigned to unaligned parallel sentences. Our evaluation is in terms of agreement with word-level alignments created by bilingual human annotators. We describe each of the models used in more detail in the next two sections, including the clone operation of Gildea (2003). The reader who is familiar with these models may proceed directly to our experiments in Section 4, and

further discussion in Section 5.

2 The Inversion Transduction Grammar

The Inversion Transduction Grammar of Wu (1997) can be thought as a a generative process which simultaneously produces strings in both languages through a series of synchronous context-free grammar productions. The grammar is restricted to binary rules, which can have the symbols in the right hand side appear in the same order in both languages, represented with square brackets:

$$X \rightarrow [YZ]$$

or the symbols may appear in reverse order in the two languages, indicated by angle brackets:

$$X \rightarrow \langle YZ \rangle$$

Individual lexical translations between English words e and French words f take place at the leaves of the tree, generated by grammar rules with a single right hand side symbol in each language:

$$X \rightarrow e/f$$

Given a bilingual sentence pair, a synchronous parse can be built using a two-dimensional extension of chart parsing, where chart items are indexed by their nonterminal Y and beginning and ending positions l, m in the source language string, and beginning and ending positions i, j in the target language string. For Expectation Maximization training, we compute inside probabilities $\beta(Y, l, m, i, j)$ from the bottom up as outlined below:

for all l, m, n such that $1 \leq l < m < n < N_s$ **do**
  for all i, j, k such that $1 < i < j < k < N_t$ **do**
    for all rules $X \rightarrow YZ \in G$ **do**
      $\beta(X, l, n, i, k)$+=
      $P([YZ]|X)\beta(Y, l, m, i, j)\beta(Z, m, n, j, k)$
      $\beta(X, l, n, i, k)$+=
      $P(\langle YZ \rangle|X)\beta(Y, m, n, i, j)\beta(Z, l, m, j, k)$
    end for
  end for
end for

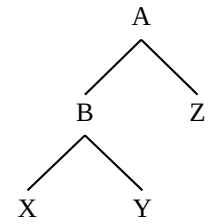
A similar recursion is used to compute outside probabilities for each chart item, and the inside and outside probabilities are combined to derive expected counts for occurrence of each grammar rule, including the rules corresponding to individual lexical translations. In our experiments we use a grammar with a start symbol S, a single preterminal C, and two nonterminals A and B used to ensure that only one parse can generate any given word-level alignment (ignoring insertions and deletions) (Wu, 1997; Zens and Ney, 2003). The individual lexical translations produced by the grammar may include a NULL word on either side, in order to represent insertions and deletions.

3 The Tree-To-String Model

The model of Yamada and Knight (2001) can be thought of as a generative process taking a tree in one language as input and producing a string in the other through a sequence of probabilistic operations. If we follow the process of an English sentence's transformation into French, the English sentence is first given a syntactic tree representation by a statistical parser (Collins, 1999). As the first step in the translation process, the children of each node in the tree can be re-ordered. For any node with m children, $m!$ re-orderings are possible, each of which is assigned a probability P_{order} conditioned on the syntactic categories of the parent node and its children. As the second step, French words can be inserted at each node of the parse tree. Insertions are modeled in two steps, the first predicting whether an insertion to the left, an insertion to the right, or no insertion takes place with probability P_{ins}, conditioned on the syntactic category of the node and that of its parent. The second step is the choice of the inserted word $P_t(f|\mathrm{NULL})$, which is predicted without any conditioning information. The final step, a French translation of each original English word, at the leaves of the tree, is chosen according to a distribution $P_t(f|e)$. The French word is predicted conditioned only on the English word, and each English word can generate at most one French word, or can generate a NULL symbol, representing deletion. Given the original tree, the re-ordering, insertion, and translation probabilities at each node are independent of the choices at any other node. These independence relations are analogous to those of a stochastic context-free grammar, and allow for efficient parameter estimation by an inside-outside Expectation Maximization algorithm. The computation of inside probabilities β, outlined below, considers possible reorderings of nodes in the original tree in a bottom-up manner:

for all nodes ε_i in input tree T **do**
  for all k, l such that $1 < k < l < N$ **do**
    for all orderings ρ of the children $\varepsilon_1 ... \varepsilon_m$ of ε_i **do**
      for all partitions of span k, l into $k_1, l_1 ... k_m, l_m$ **do**
        $\beta(\varepsilon_i, k, l)$+=
        $P_{order}(\rho|\varepsilon_i) \prod_{j=1}^{m} \beta(\varepsilon_j, k_j, l_j)$
      end for
    end for
  end for
end for

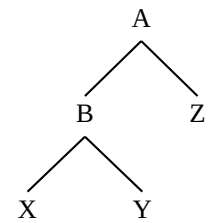
As with Inversion Transduction Grammar, many alignments between source and target sentences are not allowed. As a minimal example, take the tree:



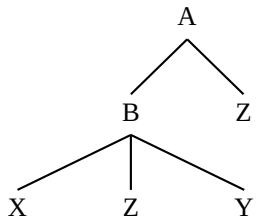
Of the six possible re-orderings of the three terminals, the two which would involve crossing the bracketing of the original tree (XZY and YZX) are not allowed. While this constraint gives us a way of using syntactic information in translation, it may in many cases be too rigid. In part to deal with this problem, Yamada and Knight (2001) flatten the trees in a pre-processing step by collapsing nodes with the same lexical head-word. This allows, for example, an English subject-verb-object (SVO) structure, which is analyzed as having a VP node spanning the verb and object, to be re-ordered as VSO in a language such as Arabic. Larger syntactic divergences between the two trees may require further relaxation of this constraint, and in practice we expect such divergences to be frequent. For example, a nominal modifier in one language may show up as an adverbial in the other, or, due to choices such as which information is represented by a main verb, the syntactic correspondence between the two sentences may break down completely. While having flatter trees can make more reorderings possible than with the binary Inversion Transduction Grammar trees, fixing the tree in one language generally has a much stronger opposite effect, dramatically restricting the number of permissible alignments.

3.1 Tree-to-String With Cloning

In order to provide more flexibility in alignments, a cloning operation was introduced for tree-to-string alignment by Gildea (2003). The model is modified to allow for a copy of a (translated) subtree from the English sentences to occur, with some cost, at any point in the resulting French sentence. For example, in the case of the input tree



a clone operation making a copy of node 3 as a new child of B would produce the tree:



This operation, combined with the deletion of the original node Z, produces the alignment (XZY) that was disallowed by the original tree reordering model.

The probability of adding a clone of original node ε_i as a child of node ε_j is calculated in two steps: first, the choice of whether to insert a clone under ε_j, with probability $P_{ins}(\text{clone}|\varepsilon_j)$, and the choice of which original node to copy, with probability

$$P_{clone}(\varepsilon_i|\text{clone} = 1) = \frac{P_{makeclone}(\varepsilon_i)}{\sum_k P_{makeclone}(\varepsilon_k)}$$

where $P_{makeclone}$ is the probability of an original node producing a copy. In our implementation, $P_{ins}(\text{clone})$ is estimated by the Expectation Maximization algorithm conditioned on the label of the parent node ε_j, and $P_{makeclone}$ is a constant, meaning that the node to be copied is chosen from all the nodes in the original tree with uniform probability.

4 Experiments

We trained our translation models on a parallel corpus of Chinese-English newswire text. We restricted ourselves to sentences of no more than 25 words in either language, resulting in a training corpus of 18,773 sentence pairs with a total of 276,113 Chinese words and 315,415 English words. The Chinese data were automatically segmented into tokens, and English capitalization was retained. We replace words occurring only once with an unknown word token, resulting in a Chinese vocabulary of 23,783 words and an English vocabulary of 27,075 words. Our hand-aligned data consisted of 48 sentence pairs also with less than 25 words in either language, for a total of 788 English words and 580 Chinese words. A separate development set of 49 sentence pairs was used to control overfitting. These sets were the data used by Hwa et al. (2002). The hand aligned test data consisted of 745 individual aligned word pairs. Words could be aligned one-to-many in either direction. This limits the performance achievable by our models; the IBM models allow one-to-many alignments in one direction only, while the tree-based models allow only one-to-one alignment unless the cloning operation is used.

Our French-English experiments were based on data from the Canadian Hansards made available by

Ulrich German. We used as training data 20,000 sentence pairs of no more than 25 words in either language. Our test data consisted of 447 sentence pairs of no more than 30 words, hand aligned by Och and Ney (2000). A separate development set of 37 sentences was used to control overfitting. We used of vocabulary of words occurring at least 10 times in the entire Hansard corpus, resulting in 19,304 English words and 22,906 French words. Our test set is that used in the alignment evaluation organized by Mihalcea and Pederson (2003), though we retained sentence-initial capitalization, used a closed vocabulary, and restricted ourselves to a smaller training corpus. We parsed the English side of the data with the Collins parser. As an artifact of the parser's probability model, it outputs sentence-final punctuation attached at the lowest level of the tree. We raised sentence-final punctuation to be a daughter of the tree's root before training our parse-based model. As our Chinese-English test data did not include sentence-final punctuation, we also removed it from our French-English test set.

We evaluate our translation models in terms of agreement with human-annotated word-level alignments between the sentence pairs. For scoring the viterbi alignments of each system against gold-standard annotated alignments, we use the alignment error rate (AER) of Och and Ney (2000), which measures agreement at the level of pairs of words:

$$AER = 1 - \frac{|A \cap G_P| + |A \cap G_S|}{|A| + |G_S|}$$

where A is the set of word pairs aligned by the automatic system, G_S is the set marked in the gold standard as "sure", and G_P is the set marked as "possible" (including the "sure" pairs). In our Chinese-English data, only one type of alignment was marked, meaning that $G_P = G_S$. For a better understanding of how the models differ, we break this figure down into precision:

$$P = \frac{|A \cap G_P|}{|A|}$$

and recall:

$$R = \frac{|A \cap G_S|}{|G_S|}$$

Since none of the systems presented in this comparison make use of hand-aligned data, they may differ in the overall proportion of words that are aligned, rather than inserted or deleted. This affects the precision/recall tradeoff; better results with respect to human alignments may be possible by adjusting an overall insertion probability in order to optimize AER.

Table 1 provides a comparison of results using the tree-based models with the word-level IBM models. IBM Models 1 and 4 refer to Brown et al. (1993). We used the GIZA++ package, including the HMM model of Och and Ney (2000). We ran Model 1 for three iterations, then the HMM model for three iterations, and finally Model 4 for two iterations, training each model until AER began to increase on our held-out cross validation data. "Inversion Transduction Grammar" (ITG) is the model of Wu (1997), "Tree-to-String" is the model of Yamada and Knight (2001), and "Tree-to-String, Clone" allows the node cloning operation described above. Our tree-based models were initialized from uniform distributions for both the lexical translation probabilities and the tree reordering operations, and were trained until AER began to rise on our held-out cross-validation data, which turned out to be four iterations for the tree-to-string models and three for the Inversion Transduction Grammar. French-English results are shown in Table 2. Here, IBM Model 1 was trained for 12 iterations, then the HMM model for 5 iterations and Model 4 for 5 iterations. The ITG and tree-to-string models were both trained for 5 iterations. A learning curve for the Inversion Transduction Grammar, is shown in Figure 1, showing both perplexity on held-out data and alignment error rate. In general we found that while all models would increase in AER if trained for too many iterations, the increases were of only a few percent.

5 Discussion

The Inversion Transduction Grammar significantly outperforms the syntactically supervised tree-to-string model of Yamada and Knight (2001). The tree-to-string and IBM models are roughly equivalent. Adding the cloning operation improves tree-to-string results by 2% precision and recall. It is particularly significant that the ITG gets higher recall than the other models, when it is the only model entirely limited to one-to-one alignments, bounding the maximum recall it can achieve.

Our French-English experiments show only small differences between the various systems. Overall, performance on French-English is much better than for Chinese-English. French-English has less reordering overall, as shown by the percentage of productions in the viterbi ITG parses that are inverted: 14% for French-English in comparison to 23% for Chinese-English.

One possible explanation for our results is parser error. While we describe our system as "syntacti-

	Precision	*Recall*	*Alignment Error Rate*
IBM Model 1	.56	.42	.52
IBM Model 4	.67	.43	.47
Inversion Transduction Grammar	.68	.52	.40
Tree-to-String w/ Clone	.65	.43	.48
Tree-to-String w/o Clone	.63	.41	.50

Table 1: Alignment results on Chinese-English corpus. Higher precision and recall correspond to lower alignment error rate.

	Precision	*Recall*	*Alignment Error Rate*
IBM Model 1	.63	.71	.34
IBM Model 4	.83	.83	.17
Inversion Transduction Grammar	.82	.87	.16
Tree-to-String w/ Clone	.84	.85	.15

Table 2: French-English results.

cally supervised", in fact this supervision comes in the form of the annotation of the Wall Street Journal treebank on which the parser is trained, rather than parses for our parallel training corpus. In particular, the text we are parsing has a different vocabulary and style of prose from the WSJ treebank, and often the fluency of the English translations leaves something to be desired. While both corpora consist of newswire text, a typical WSJ sentence

> Pierre Vinken, 61 years old, will join the board as a nonexecutive director Nov. 29.

contrasts dramatically with

> In the past when education on opposing Communists and on resisting Russia was stressed, retaking the mainland and unifying China became a slogan for the authoritarian system, which made the unification under the martial law a tool for oppressing the Taiwan people.

a typical sentence from our corpus.

While we did not have human-annotated gold-standard parses for our training data, we did have human annotated parses for the Chinese side of our test data, which was taken from the Penn Chinese Treebank (Xue et al., 2002). We trained a second tree-to-string model in the opposite direction, using Chinese trees and English strings. The Chinese training data was parsed with the Bikel (2002) parser, and used the Chinese Treebank parses for our test data. Results are shown in Table 3. Because the ITG is a symmetric, generative model, the ITG results in Table 3 are identical to those in Table 1. While the experiment does not show a significant improvement, it is possible that better parses for the training data might be equally important.

Even when the automatic parser output is correct, the tree structure of the two languages may not correspond. Dorr (1994) categorizes sources of syntactic divergence between languages, and Fox (2002) analyzed a parallel French-English corpus, quantifying how often parse dependencies cross when projecting an English tree onto a French string. Even in this closely related language pair with generally similar word order, crossed dependencies were caused by such common occurrences as adverb modification of a verb, or the correspondence of "not" to "ne pas". Galley et al. (2004) extract translation rules from a large parsed parallel corpus that extend in scope to tree fragments beyond a single node; we believe that adding such larger-scale operations to the translation model is likely to significantly improve the performance of syntactically supervised alignment.

The syntactically supervised model has been found to outperform the IBM word-level alignment models of Brown et al. (1993) for translation by Yamada and Knight (2002). An evaluation for the alignment task, measuring agreement with human judges, also found the syntax-based model to outperform the IBM models. However, a relatively small corpus was used to train both models (2121 Japanese-English sentence pairs), and the evaluations were performed on the same data for training, meaning that one or both models might be significantly overfitting.

Zens and Ney (2003) provide a thorough analysis of alignment constraints from the perspective of decoding algorithms. They train the models of Wu

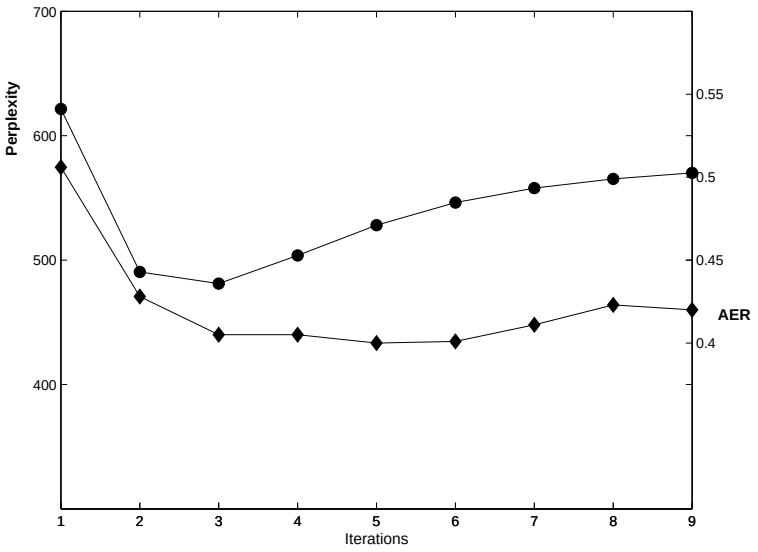


Figure 1: Training curve for ITG model, showing perplexity on cross-validation data, and alignment error rate on a separate hand-aligned dataset.

	Precision	*Recall*	*Alignment Error Rate*
Inversion Transduction Grammar	.68	.52	.40
Tree-to-String, automatic parses	.61	.48	.46
Tree-to-String, gold parses	.61	.52	.44

Table 3: Chinese Tree to English String

(1997) as well as Brown et al. (1993). Decoding, meaning exact computation of the highest probability translation given a foreign sentence, is not possible in polynomial time for the IBM models, and in practice decoders search through the space of hypothesis translations using a set of additional, hard alignment constraints. Zens and Ney (2003) compute the viterbi alignments for German-English and French-English sentences pairs using IBM Model 5, and then measure how many of the resulting alignments fall within the hard constraints of both Wu (1997) and Berger et al. (1996). They find higher coverage for an extended version of ITG than for the IBM decoding constraint for both language pairs, with the unmodified ITG implementation covering about the same amount of German-English data as IBM, and significantly less French-English data. These results show promise for ITG as a basis for efficient decoding, but do not address which model best aligns the original training data, as IBM-derived alignments were taken as the gold standard, rather than human alignments. We believe that our results show that syntactically-motivated models are a promising general approach to training translation models as well to searching through the resulting probability space.

Computational complexity is an issue for the tree-based models presented here. While training the IBM models with the GIZA++ software takes minutes, the tree-based EM takes hours. With our C implementation, one iteration of the syntactically supervised model takes 50 CPU hours, which can be parallelized across machines. Our tree-based models are estimated with complete EM, while the training procedure for the IBM models samples from a number of likely alignments when accumulating expected counts. Because not every alignment is legal with the tree-based models, the technique of sampling by choosing likely alignments according to a simpler model is not straightforward. Nonetheless, we feel that training times can be improved with the right pruning and sampling techniques, as will be necessary to train on the much larger amounts data now available, and on longer sentences.

6 Conclusion

We present a side-by-side comparison of syntactically supervised and unsupervised tree-based alignment, along with the non tree-based IBM Model 4. For Chinese-English, using trees helps the align-

ment task, but a data-derived tree structure gives better results than projecting automatic English parser output onto the Chinese string. The French-English task is easier overall, and exhibits smaller differences between the systems.

Acknowledgments We are very grateful to Rebecca Hwa for assistance with the Chinese-English data, and to everyone who helped make the resources we used available to the research community. This work was partially supported by NSF ITR IIS-09325646.

References

Adam Berger, Peter Brown, Stephen Della Pietra, Vincent Della Pietra, J. R. Fillett, Andrew Kehler, and Robert Mercer. 1996. Language translation apparatus and method of using context-based tanslation models. United States patent 5,510,981.

Daniel M. Bikel. 2002. Design of a multi-lingual, parallel-processing statistical parsing engine. In *Proceedings ARPA Workshop on Human Language Technology*.

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

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

Michael John Collins. 1999. *Head-driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania, Philadelphia.

Bonnie J. Dorr. 1994. Machine translation divergences: A formal description and proposed solution. *Computational Linguistics*, 20(4):597–633.

Heidi J. Fox. 2002. Phrasal cohesion and statistical machine translation. In *In Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 304–311.

Michel Galley, Mark Hopkins, Kevin Knight, and Daniel Marcu. 2004. What's in a translation rule? In *Proceedings of the Human Language Technology Conference/North American Chapter of the Association for Computational Linguistics (HLT/NAACL)*.

Daniel Gildea. 2003. Loosely tree-based alignment for machine translation. In *Proceedings of the 41th Annual Conference of the Association for Computational Linguistics (ACL-03)*, pages 80–87, Sapporo, Japan.

Rebecca Hwa, Philip Resnik, Amy Weinberg, and Okan Kolak. 2002. Evaluating translational correspondence using annotation projection. In *Proceedings of the 40th Annual Conference of the Association for Computational Linguistics (ACL-02)*.

Rada Mihalcea and Ted Pederson. 2003. An evaluation exercise for word alignment. In *HLT-NAACL 2003 Workshop on Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, pages 1–10, Edmonton, Alberta.

Franz Josef Och and Hermann Ney. 2000. Improved statistical alignment models. In *Proceedings of the 38th Annual Conference of the Association for Computational Linguistics (ACL-00)*, pages 440–447, Hong Kong, October.

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

Nianwen Xue, Fu-Dong Chiou, and Martha Palmer. 2002. Building a large-scale annotated chinese corpus. In *Proceedings of the 19th. International Conference on Computational Linguistics (COLING 2002)*, Taipei, Taiwan.

Kenji Yamada and Kevin Knight. 2001. A syntax-based statistical translation model. In *Proceedings of the 39th Annual Conference of the Association for Computational Linguistics (ACL-01)*, Toulouse, France.

Kenji Yamada and Kevin Knight. 2002. A decoder for syntax-based statistical MT. In *Proceedings of the 40th Annual Conference of the Association for Computational Linguistics (ACL-02)*, Philadelphia, PA.

Richard Zens and Hermann Ney. 2003. A comparative study on reordering constraints in statistical machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Sapporo, Japan.