Mach Translat (2015) 29:225–265
DOI 10.1007/s10590-015-9177-0

Labeling hierarchical phrase-based models without linguistic resources

Gideon Maillette de Buy Wenniger[1] · **Khalil Sima'an**[1]

Received: 16 September 2015 / Accepted: 16 December 2015 / Published online: 9 January 2016

Abstract Long-range word order differences are a well-known problem for machine translation. Unlike the standard phrase-based models which work with sequential and local phrase reordering, the hierarchical phrase-based model (Hiero) embeds the reordering of phrases within pairs of lexicalized context-free rules. This allows the model to handle long range reordering recursively. However, the Hiero grammar works with a single nonterminal label, which means that the rules are combined together into derivations independently and without reference to context outside the rules themselves. Follow-up work explored remedies involving nonterminal labels obtained from monolingual parsers and taggers. As of yet, no labeling mechanisms exist for the many languages for which there are no good quality parsers or taggers. In this paper we contribute a novel approach for acquiring reordering labels for Hiero grammars directly from the word-aligned parallel training corpus, without use of any taggers or parsers. The new labels represent types of alignment patterns in which a phrase pair is embedded within larger phrase pairs. In order to obtain alignment patterns that generalize well, we propose to decompose word alignments into trees over phrase pairs. Beside this labeling approach, we contribute coarse and sparse features for learning soft, weighted label-substitution as opposed to standard substitution. We report extensive experiments comparing our model to two baselines: Hiero and the known syntax augmented machine translation (SAMT) variant, which labels Hiero rules with nonterminals extracted from monolingual syntactic parses. We also test a simplified labeling scheme based on inversion transduction grammar (ITG). For the Chinese–

✉ Gideon Maillette de Buy Wenniger
gemdbw@gmail.com

Khalil Sima'an
k.simaan@uva.nl

[1] Institute for Logic, Language and Computation, University of Amsterdam, Amsterdam, The Netherlands

English task we obtain performance improvement up to 1 BLEU point, whereas for the German–English task, where morphology is an issue, a minor (but statistically significant) improvement of 0.2 BLEU points is reported over SAMT. While ITG labeling does give a performance improvement, it remains sometimes suboptimal relative to our proposed labeling scheme.

Keywords Hierarchical statistical machine translation · Reordering · Reordering labels · Soft constraints

1 Introduction

Word order differences between languages constitute a major challenge in machine translation (MT). The statistical machine translation (SMT) literature has produced a range of models aimed at predicting how the word order of the source sentence is transformed into a plausible target word order. Generally speaking, the existing reordering approaches that are integrated within translation (i.e. during decoding) can be grouped into the sequential (Tillmann 2004; Galley and Manning 2008) and the hierarchical (Chiang 2005; Zollmann and Venugopal 2006). While the sequential approach considers the reordering process as a finite-state process over word or phrase positions, the hierarchical approach (Hiero) works with a synchronous context-free grammar (SCFG). For a decade now, the hierarchical approach (Chiang 2005) shows improved performance for language pairs with long-range reordering such as Chinese–English and Japanese-English (Chiang 2005; Zollmann and Venugopal 2006). The present work falls squarely within the hierarchical approach to reordering.

Hiero SCFG rules are extracted from a word-aligned parallel corpus. Like other phrase-based models (Och and Ney 2004), the word alignment defines the set of translation rules that can be extracted from the parallel corpus. Hiero's rules are labeled with a single nonterminal label X, beside the start symbol of the SCFG. Hiero's reordering patterns (straight/inverted) are embedded together with lexical context within synchronous rules, which makes local reordering within a rule sensitive to direct context. However, during decoding every rule may substitute on every nonterminal X, and thus it is independent of any other rule given the source string. This may result in suboptimal reordering as we now explain.

Figure 1 shows a toy training parallel corpus of three word-aligned sentence pairs, decomposed into Hiero rules (hierarchical phrase pairs); the boxed Ri indices at the nodes stand for rule identities placed on the left-hand side of every rule. For example, in Fig. 1b we find rule $R5$

$$X \rightarrow \langle \text{eine rege und effective, a lively and effective} \rangle$$

and in Fig. 1a, rule $R3$

$$X \rightarrow \langle \text{zu erreichen, to achieve} \rangle$$

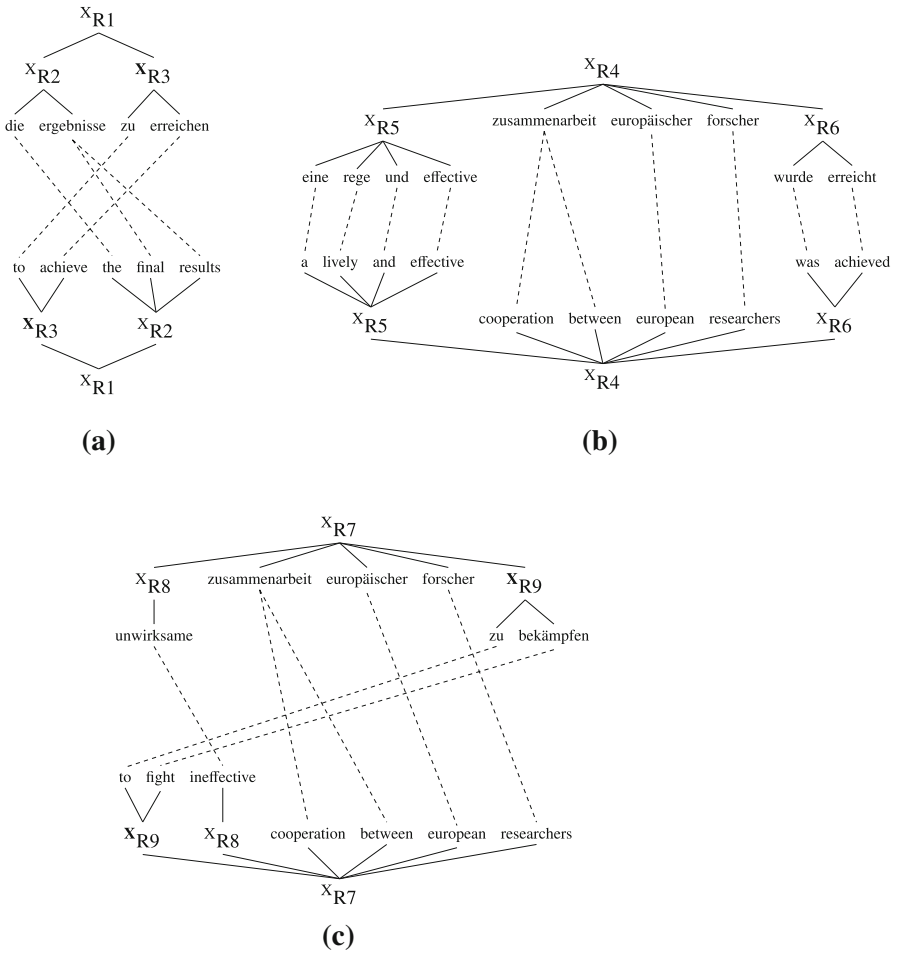

Fig. 1 Training examples, where the labeled and indexed nodes represent (some of the) phrase pairs that can be extracted from the aligned sentence pairs. **a** (*Left-binding*) Inverted training example 1. **b** Monotone training example. **c** (*Left-binding*) Inverted training example 2

By cutting out some of the embedded phrase pairs, we obtain Hiero rules with gaps. As an example, from the phrase pair at the root of the aligned sentence pair in Fig. 1b, the hierarchical rule $R4$

$$X \rightarrow \langle \mathrm{X}_{\boxed{1}} \quad \text{zusammenarbeit europäischer forscher} \quad X_{\boxed{2}} \,, \\ X_{\boxed{1}} \quad \text{cooperation between european researchers} \quad X_{\boxed{2}} \rangle$$

can be extracted by cutting out the two embedded phrase pairs $R5$ and $R6$ as gaps labeled X. Similarly, we obtain rule $R7$

$$X \rightarrow \langle \mathrm{X}_{\boxed{1}} \quad \text{zusammenarbeit europäischer forscher} \quad X_{\boxed{2}} \,, \\ X_{\boxed{2}} \quad X_{\boxed{1}} \quad \text{cooperation between european researchers} \rangle$$

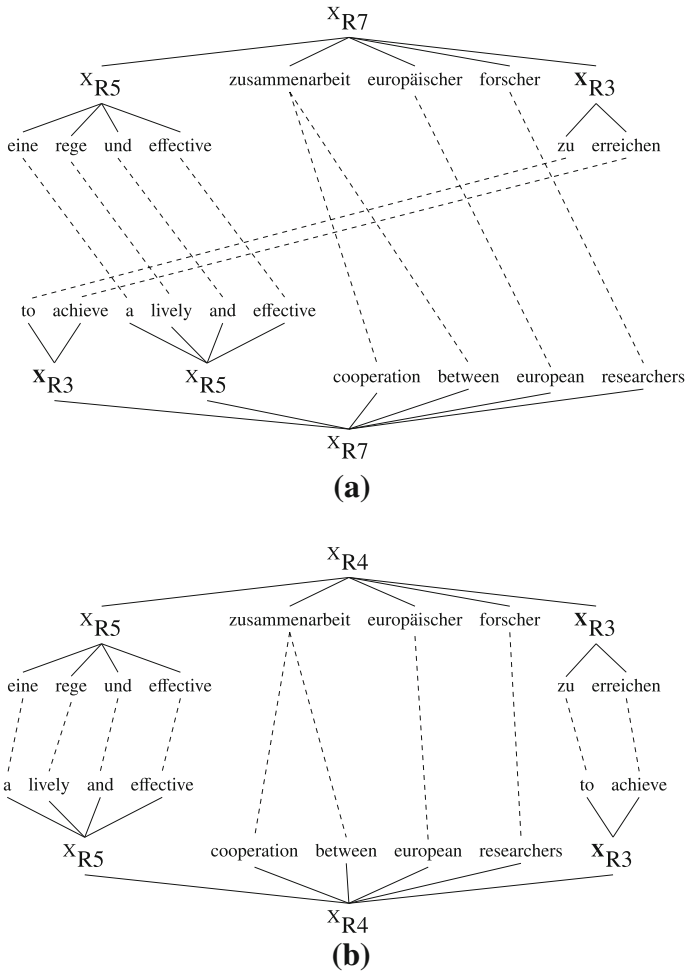

Fig. 2 Translations of the new sentence "eine rege und effektive zusammenarbeit europäischer forscher zu erreichen". **a** Correct translation for new sentence that produces the right word order. **b** Wrong alternative translation that can be produced by Hiero

from the root phrase pair in Fig. 1c. Note how the training examples for the English verbs "to fight" in Fig. 1c and "was achieved" in Fig. 1b are embedded within, respectively, monotone and inverted reordering patterns when translated into German.[1]

We now exemplify how Hiero risks missing the correct word order and how labels from the surrounding word alignment context may help. In Fig. 2a, translation rule $R7$ is combined with rule $R5$ and rule $R3$ to translate the new sentence "eine rege und effektive zusammenarbeit europäischer forscher zu erreichen". Here starting from translation rule $R7$ and then substituting $R5$ and $R3$ on the two X nonterminals, the cor-

[1] In the German-English examples in this section, we use fully lowercased versions of all words including proper nouns, as is also done in the experiments.

rect word order can be obtained. However, the rules extracted during training also permit a different translation of this sentence that produces the wrong word order, shown in Fig. 2b. This translation is formed by combining $R4$ with $R5$ and $R3$. Both Hiero derivations are eligible, and the independence assumptions between rules suggest that there is no reason why Hiero's synchronous grammar should be able to select the correct word order. The independence assumptions between the rules suggest also that the burden of selecting the correct reordering is left over to the target language model.

How could use of word alignment context help produce preference for correct reordering in Hiero? Contrast the reordering structures in Fig. 1a, c to the structure in Fig. 1b. In the first two the verb units, "to achieve" and "to fight" (labeled with a bold **X**), are inverted with respect to the embedding context, whereas in the latter example, the verb "was achieved" is monotone with respect to the embedding context. In this simple example, two types of verbs can be discriminated using word-alignment types from the embedding rules, which can be used as Hiero labels. Such labeling can be obtained during Hiero rule extraction from the word-aligned training-sentence pairs without need for other resources. By extracting such *reordering labels*, the incorrect substitution in Fig. 2b could be either prevented or made far less likely than the correct alternative.[2] Phrases induce a certain reordering pattern with respect to their sub-phrases and with respect to the parent phrase that embeds them. We note that in a sentence-aligned, word-aligned parallel corpus, it turns out that there are more such reordering patterns than the binary choice of monotone/inverted.

The core idea in this work is to extract phrase labels from word alignments by first decomposing them recursively into their sub-component alignments. The decomposition we are interested in proceeds in the same way that word alignments decompose recursively into phrase pairs (Zhang et al. 2008). Such decomposition results in trees in which the nodes dominate phrase pairs. But the decomposition in this work maintains on every node also the alignment relation (called *node operator* or simply *operator*) which expresses how the sibling phrase pairs under that node compose together at the target side relative to the source side. Subsequently we bucket the resulting node operators into classes and use these classes as labels for Hiero rules. The ITG orientations (straight and inverted) (Wu 1997) turn out to be special cases of this general scheme, and in our experiments we show that limiting the choice to ITG, although beneficial, could be suboptimal sometimes.

Traditional grammar nonterminal labels signify hard categories, and substituting a rule with a left-hand side label X may take place only on the same nonterminal X. Like earlier labeling approaches (e.g., (Zhang et al. 2008)), we also find that exact match substitution for nonterminals is suboptimal. Following Chiang (2010), we devise a set of feature weights that allow any nonterminal label Y to substitute on any other label X with some cost determined by tuning the feature weights associated with the substitution. We call this approach *elastic-substitution decoding*, because during

[2] One might wonder about the frequency of verbs that show such preferences for reordering: in the filtered test grammar (see Sect. 5) there are more than 27,000 phrase pairs, each with 2 words on both sides, that show such a preference for inversion relative to their embedding context. A large fraction of these phrase pairs corresponds to such verbal constructs. This itself is just a part of one of many types of reordering phenomena, selected for this example.

decoding the label substitution of Y on X with some cost can be seen as if the labels stretch during decoding to allow for as wide a set of translation hypotheses as needed.

After summarizing the Hiero model and discussing related work in some more detail, we propose a simple extension of normalized decomposition trees (NDTs) (Zhang et al. 2008) with transduction operators that represent target-source phrase many-to-many mappings, including non-contiguous translation equivalents. Based on this extension, this paper contributes:

- A novel labeling approach for Hiero, which exploits tree decompositions of word alignments, together with an effective proposal for features for *elastic-substitution decoding*,
- Extensive experiments on German–English and Chinese–English showing the superiority of this proposed labeling relative to Hiero and SAMT,
- Analysis of the experimental results showing the type and source of improved performance.

2 Hierarchical models and closely related work

Hiero SCFGs (Chiang 2005, 2007) allow only up to two (pairs of) nonterminals on the right-hand-side (RHS) of synchronous rules. The types of permissible Hiero rules are:

$$X \rightarrow \langle \alpha,\ \delta \rangle \tag{1}$$

$$X \rightarrow \langle \alpha\ X_{\boxed{1}}\ \gamma,\ \delta\ X_{\boxed{1}}\ \eta \rangle \tag{2}$$

$$X \rightarrow \langle \alpha\ X_{\boxed{1}}\ \beta\ X_{\boxed{2}}\ \gamma\ ,\ \delta\ X_{\boxed{1}}\ \zeta\ X_{\boxed{2}}\ \eta\ \rangle \tag{3}$$

$$X \rightarrow \langle \alpha\ X_{\boxed{1}}\ \beta\ X_{\boxed{2}}\ \gamma\ ,\ \delta\ X_{\boxed{2}}\ \zeta\ X_{\boxed{1}}\ \eta\ \rangle \tag{4}$$

Here $\alpha, \beta, \gamma, \delta, \zeta, \eta$ are terminal sequences. These sequences can be empty, except for β, since hierarchical phrase-based translation, as first proposed by Chiang (2005) (HIERO) prohibits rules with nonterminals that are adjacent on the source side. It also requires all rules to have at least one pair of aligned words. These extra constraints are intended to reduce the amount of spurious ambiguity. Equation (1) corresponds to a normal phrase pair, (2) to a rule with one gap and (3) and (4) to the monotone and inverting rules, respectively.

Given a Hiero SCFG G, a source sentence $\mathbf{s}$ is translated into a target sentence $\mathbf{t}$ by one or more synchronous derivations $\mathbf{d}$, each of which is a finite sequence of well-formed substitutions of synchronous productions from G, see (Chiang 2006, 2007). The goal of finding the most likely translation is then replaced by the somewhat simpler problem of finding the most likely derivation $\mathbf{d}$, as in (5):

$$\arg\max_{\mathbf{d} \in G} P(\mathbf{t}, \mathbf{d} \mid \mathbf{s}) \tag{5}$$

We parse $\mathbf{s}$ with G so we limit the space of derivations to those that are licensed by G for $\mathbf{s}$, and so we have $P(\mathbf{t}, \mathbf{d} \mid \mathbf{s}) = P(\mathbf{d})$ ($\mathbf{t}$ is the sequence of target terminals generated by $\mathbf{d}$). Following Och and Ney (2002), a log-linear model over derivation $\mathbf{d}$ computes the probability of a derivation as a product of weighted features ϕ_i for

that derivation. Apart from the language model feature ϕ_{LM}, every other feature ϕ_i is defined as a product over a function applied at the individual rule level. The total derivation probability is then computed by multiplying the weighted language model probability $P_{LM}(e)^{\lambda_{LM}}$ with the product over the other features, weighted by their feature weight λ_i, as in (6):

$$\begin{aligned} P(\mathbf{d}) &\propto P_{LM}(\mathbf{t})^{\lambda_{LM}} \cdot \prod_{i \neq LM} \prod_{(X \rightarrow \langle \alpha, \delta \rangle) \in \mathbf{d}} \phi_i(X \rightarrow \langle \alpha, \delta \rangle)^{\lambda_i} \\ &= P_{LM}(\mathbf{t})^{\lambda_{LM}} \cdot \prod_{(X \rightarrow \langle \alpha, \delta \rangle) \in \mathbf{d}} \prod_{i \neq LM} \phi_i(X \rightarrow \langle \alpha, \delta \rangle)^{\lambda_i} \end{aligned} \tag{6}$$

By rearranging the two products, we obtain a product ranging over individual rule features. Apart from the language model feature, all other weighted features can be multiplied together for every rule separately, giving individual rule weights which are computed efficiently. Unfortunately, the computation of $P(\mathbf{d})$ demands multiplication with the language model probability $P_{LM}(e)$, which is not defined in terms of individual rules. This adds considerable complexity to the decoding process, and for this reason approximation is necessary in the form of beam-search with pruning, e.g., *cube-pruning* (Huang and Chiang 2007; Chiang 2007).

In the next two subsections we will discuss work that is closely related to our work, followed by an overview of our contributions. Other distantly related work will be discussed in Sect. 7.

2.1 Lexicalized orientation models

We first look at work that distils reordering information from word alignments, sharing a general intuition with this work. Xiao et al. (2011) add a lexicalized orientation model to Hiero (akin to Tillmann (2004)), and achieve significant gains. Nguyen and Vogel (2013) extend this idea by integrating a phrase-based (non-hierarchical) lexicalized orientation model as well as a distance-based reordering model into Hiero. This involves adapting the decoder, so that rule chart items are extended to keep the first and last phrase pair for their lexical spans. Huck et al. (2013) overcome the technical limitations of both Xiao et al. (2011) and Nguyen and Vogel (2013) by including a *hierarchical* lexicalized orientation model into Hiero. This requires making even more drastic changes to the decoder, such as delayed (re-)scoring at hypernodes up in the derivation of nodes lower in the chart whose orientations are affected by them. Although sharing a similar intuition to our work, phrase-orientation models are not equivalent to Hiero/SCFG labeling mechanisms because formally they require extensions to SCFGs (which demand drastic changes in the decoder).

2.2 Soft constraints

Our approach towards soft constraints is based on Chiang (2010). Chiang's work uses labels similar to Zollmann and Venugopal (2006) with syntax on both sides. It

applies Boolean features for rule-label and substituted-label combinations and uses discriminative training (MIRA: (Cherry and Foster 2012)) to learn which substitution combinations are associated with better translations. Their work also explores the usage of further rule extraction heuristics to extract a set of only non-crossing[3] rules, selected in order of relative linguistic robustness of the (partial) constituents for the left-hand-sides of the extracted rules. This yields a grammar that is even smaller than Hiero itself, while still giving similar results. In our case, without access to linguistic labels, this type of selection is not directly applicable and is therefore not used. Other related work on soft constraints will be discussed in Sect. 7.

2.3 Innovations of the proposed method

This work is an extended version of an SSST 2014 workshop paper (Maillette de Buy Wenniger and Sima'an 2014a) and differs substantially as follows. We provide a thorough motivation for our kind of labeling and explain Hierarchical Alignment Trees in Sect. 3.1 (absent in the SSST paper). We provide full detail of the label extraction approach (which was not discussed in detail in the short paper). Beside Hiero, we also report experiments for a new baseline, the syntactically-labeled SAMT (shortly discussed in Sect. 5.1), both on German–English and Chinese–English. Comparing to a syntactically-labeled baseline gives a better feel for the performance differences to our approach. We discuss label-substitution features, our implementation of soft (label) matching constraints, in Sect. 4. Beside the basic label-substitution features found in SSST 2014, here we add a sparse label-substitution feature set, plus extensive additional experiments using this expanded feature set, which show how it further improves the results for German–English and Chinese–English translation. Finally, we provide qualitative analysis of the behaviour of our model in terms of reordering and the role of the language model.

The labeling approach presented next differs from existing approaches. It is inspired by work on *elastic-substitution decoding* (Chiang 2010) that relaxes the label matching constraints during decoding, but employs novel, non-linguistic bilingual labels. Furthermore, it shares the bilingual intuition with phrase orientation models but is based on a new approach to SCFG labeling, thereby remaining within the confines of Hiero SCFG, avoiding the need to make changes inside the decoder.[4] Our approach is, to the best of our knowledge, the first to exploit labels extracted from decompositions of word alignments.

[3] Two extracted rules r_1 and r_2 *cross* when their associated source (and target) spans in the training data overlap, e.g. if r_1 spans source and target words 0–3, and r_2 spans words 3–4, these rules are crossing.

[4] Elastic-substitution decoding can be easily implemented without adapting the decoder, through a smart application of "label bridging" unary rules. This is done by adding a set of unary rules—one rule for any combination of nonterminals—in combination with adding a marker to left-hand-side and right-hand-side nonterminals in order to avoid unary rule chains. In practice, however, adapting the decoder turns out to be computationally more efficient, so we use this solution in our experiments.

2.4 Some notes on terminology and definitions of basic concepts

A methodology of central importance in this paper is the earlier mentioned approach proposed by Chiang (2010), whereby the matching constraint is softened so that nonterminals can be substituted to other nonterminals with possibly different, mismatching labels. However, besides softening the matching constraint, a second crucial component of this approach is the use of dedicated features—so-called *label-substitution features*—that enable learning preferences over different types of label substitutions. Without addition of such features, labels would in fact be meaningless in a setting where strict matching of labels is not enforced. Unfortunately, no well-established name exists in the literature for Chiang (2010)'s approach. In this paper we have chosen to use the term *elastic-substitution decoding* to refer to this approach. Some of the other names sometimes used in the literature for this approach are: *soft matching*, *fuzzy matching*, *soft labeling*, and *soft matching constraints*. Finally, note that in earlier work (Maillette de Buy Wenniger and Sima'an 2014a), we have in fact used multiple different terms for this concept. Here we have tried to improve this, by using only a single term—*elastic-substitution decoding*—which implies both (i) the softening of the (label) matching constraint during decoding, and (ii) the usage of some set of label-substitution features.

The concepts *binarizable/non-binarizable* word alignment and *non-decomposable* phrase pair used in this work are based on the definition of phrase pair. Informally, a phrase pair corresponds to contiguous spans on the source and target side, so that each of the positions in the source span is only aligned to positions in the target span, and each of the positions in the target span is only aligned to positions in the source span. A *non-decomposable* phrase pair is a phrase pair that contains no other phrase pairs. A binarizable word alignment is then a word alignment which induces only phrase pairs that are either themselves non-decomposable or else can be decomposed (split) into just two smaller phrase pairs. A non-binarizable word alignment contains one or more phrase pairs that are not non-decomposable but that cannot be decomposed (split) into just two smaller phrase pairs.

3 Bilingual reordering labels by alignment decomposition

In the following we describe how reordering labels are formed. In particular, the rules we extract are identical to Hiero rules (Chiang 2007) (see Sect. 2) except for their labels. Following Zhang et al. (2008), we view Hiero SCFG rule extraction from the hierarchical perspective of word alignment decomposition as a two-step process. Initially, every word alignment in the training corpus is decomposed recursively into a canonical normalized decomposition tree (NDT). This results in a kind of training treebank of NDTs. Subsequently, the Hiero rules are extracted from these NDTs as in Zhang et al. (2008).

It is useful here to exemplify the decomposition of word alignments into NDTs because it helps understand how we extend NDTs and the extracted rules with the bilingual reordering labels. Figure 3 shows an alignment from Europarl German–English (Koehn 2005) along with a maximally decomposed phrase pair tree structure.

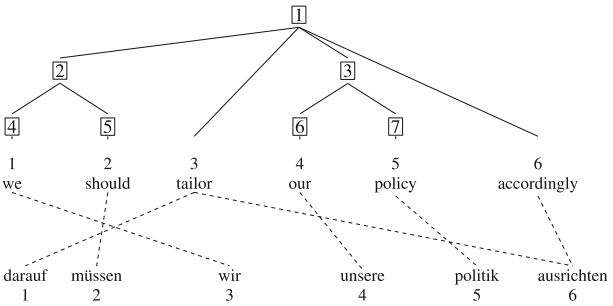

Fig. 3 Example alignment from Europarl

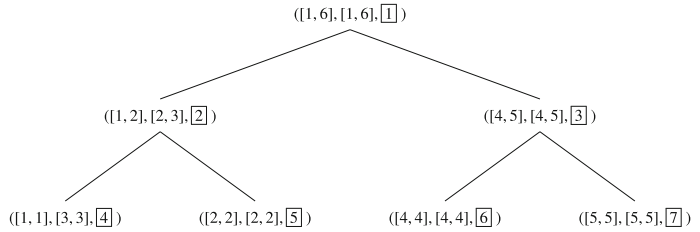

Fig. 4 Normalized decomposition tree (Zhang et al. 2008) extended with pointers to original alignment structure from Fig. 3

Figure 4 shows the NDT for Fig. 3 extended with pointers (boxed integers) to the original phrase pair tree in Fig. 3. The boxed integers indicate how the phrase pairs in the two representations correspond. In an NDT, the root node of every subtree represents a phrase pair with spans indicated by the ranges of the two pairs of integers that decorate that root node. Every composite phrase pair is recursively split up into a minimum number (two or greater) of contiguous parts. In Fig. 4 the root node covers the source and target span from words [1, 6], and it embeds two phrase pairs: the first covers the source-target spans ([1, 2], [2, 3]), and the second covers source-target spans ([4, 5], [4, 5]). From the source-target ranges that decorate the NDT nodes it is easy to compute bijective phrase permutation information: the two children of the root node in Fig. 4 have ranges ([1, 2], [2, 3]) and ([4, 5], [4, 5]), respectively, which shows that they are ordered in binary straight orientation. Note, however, that together these two phrase pairs in the example NDT do not explicitly show the build-up of their entire parent phrase-pair ([1, 6], [1, 6]) because of a discontinuous translation equivalence involving `tailor...accordingly/ darauf...ausrichten`. The NDT does not explicitly show this discontinuity, nor does it show the internal word alignment within. In short, the NDT shows how phrase pairs maximally decompose into other phrase pairs and how these permute at each tree level, but NDTs abstract away from aspects of word alignments that are important for representing cases of discontiguous translation equivalents and other non-bijective alignments (many-to-many or unaligned words) internal to phrase pairs. This should not be an issue as long as phrase and rule extraction is the sole goal. However, for extracting labels capturing the types of reordering occurring inside or around phrase pairs, we propose that other alignment information is also needed at the NDT nodes. For this purpose we

next present hierarchical alignment trees (HATs), which are decompositions of word alignments that retain all alignment information at the tree nodes.

3.1 From NDTs to HATs: explicit node operators

HATs are best understood by noting that they extend not just NDTs but also *permutation trees* (PETs) (Gildea et al. 2006). In Fig. 5a a non-binarizable word alignment with

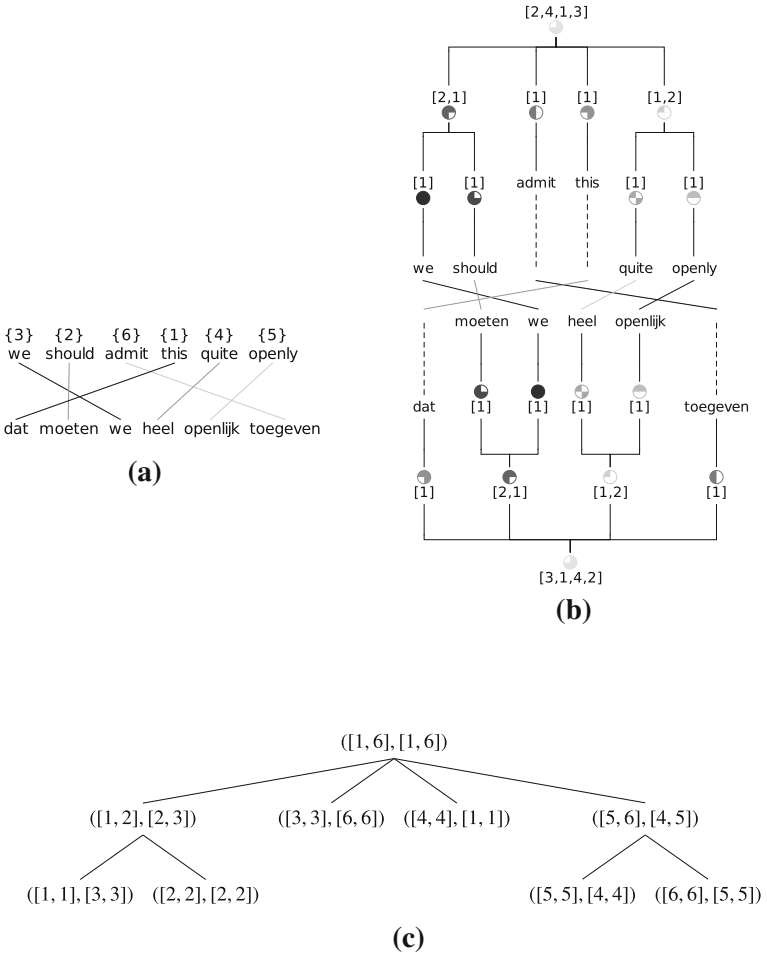

Fig. 5 A word alignment (**a**), with non-binarizable bijective word mappings (permutation) and its corresponding permutation tree (PET) (**b**) and normalized decomposition tree (NDT) (**c**). In (**b**) permutation labels such as [2,4,1,3] denote the local relative reordering mapping at every node. *Circles* with *different fillings* and *shades* are used to indicate matching translation equivalents on the source and target side of the PET. Note that the NDT representation (**c**) does not explicitly state the mapping relations, in contrast to the PET representation. It instead specifies pairs of source/target span ranges, such as ([1,6], [1,6]), that are translation-equivalent. While for PETs the mapping relations are in principle still retrievable from the NDT by reasoning, in the case of NDTs for general non-bijective (discontiguous) word alignments, even this reconstruction is no longer possible and information about the mapping is lost

bijective word mappings is shown. Figure 5b shows its corresponding permutation tree and Fig. 5c its corresponding NDT. A PET is a recursive hierarchical representation of a maximal decomposition (also called factorization) of a permutation. Word alignments with bijective word mappings can be represented as permutations, and therefore as permutation trees. The permutation labels on the tree nodes in a permutation tree describe exactly the recursive hierarchical reordering of the word alignment, see the example in Fig. 5b. Starting from the permutation label [**2**, 4, 1, 3] at the top, and expanding the first child **2** with the child permutation [2, 1], shifting the numbers as necessary, we arrive at the (intermediate) permutation [3, 2, 5, 1, **4**]. Finally expanding the last child **4** in the intermediate permutation (originally 3 before expansion) with the child permutation [1, 2] and shifting the numbers again we retrieve the original permutation [3, 2, 6, 1, 4, 5]. This illustrates an important property of PETs: they completely retain all information of the original permutation, so that by incremental evaluation of the node operators that original permutation is easily reconstructed. In contrast, NDTs represent only the set of phrase pairs and their subsumption relations, but not the information about the reordering. Figure 5c illustrates this. Thus, NDTs do not contain the reordering information and therefore are not equivalent to the original word alignments.

The capability of PETs to completely represent hierarchical reordering suggests that they are a good start for representing hierarchical translation equivalence. The only problem with permutation trees is that they are limited to alignments with bijective word mappings. HATs overcome this limitation by generalizing permutation trees to general alignments with many-to-many mappings. In doing so HATs extend NDTs by retaining exact alignment information on every node represented using a notation called *set permutation*. A set-permutation operator represents a given word alignment as a mapping from source-to-target units (or vice versa): every source unit is mapped into a target unit, where a unit can be a *set of multiple positions* (possibly discontinuous) that map to the same positions on the other side. Set permutations are exactly the generalization of permutations that is necessary to allow generalization of PETs into HATs. In Fig. 6 we show the HAT corresponding to the earlier example in Fig. 3 and the associated NDT in Fig. 4. Notice how for example the source-to-target mapping between subsumed words and phrases under the root node is explicitly represented by the set-permutation operator [2,{1,4},3,4]: The first English position maps into the second German position, and the second English position maps to two German positions {1, 4} and the third and fourth English positions map to third and fourth German positions, respectively.

In fact, the node operators can be computed by minor additional book-keeping *while decomposing* the word alignment into NDTs. Following the algorithm of Zhang et al. (2008), this book-keeping involves keeping track of where each source subtree child of a node maps to on the target side according to the original word alignment. This demands representing decomposed word alignments as set-permutation node operators, which extend beyond the bijective case (permutations). The details of this extension of NDTs can be found together with the phrase decomposition algorithm in Sima'an and Maillette de Buy Wenniger (2013).

For Hiero rule extraction it is possible to avoid enumerating the exponentially many possible HATs for the same word alignments by straightforwardly representing them

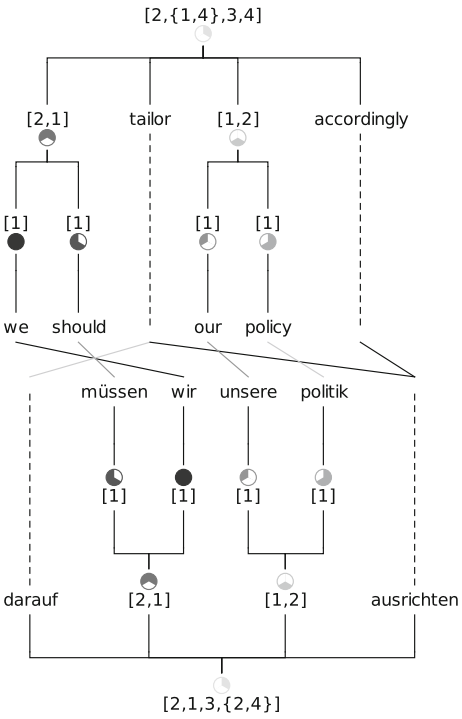

Fig. 6 Hierarchical alignment tree (HAT) corresponding to the example of Figs. 3 and 4. Note that while in this case there is only one HAT for the alignment, in general a set of alternate HATs is induced, corresponding to alternate maximal decompositions for an alignment and encoded as a chart (*packed forest*)

in a $O(n^3)$ parse forest in a CYK-style parsing algorithm (Sima'an and Maillette de Buy Wenniger 2013).[5]

3.2 Nonterminal labels by bucketing node operators

The node operators on HAT nodes encode decomposed word-alignment information. The HAT representation exposes the shared operators between different word alignments across a large corpus. In this work we propose to bucket these operators and employ the buckets as labels for the Hiero rules while extracting them. The bucketing is technically needed for various reasons. Firstly, it results in a manageable number of labels and avoids problems with sparsity. In a strict-matching decoding setup for example, having more labels leads to more spurious derivations and splitting of the probability mass. Similarly, when working with elastic-substitution decoding, just one labeled version per Hiero rule type is used (see *canonical labeled rules* at the end of Sect. 4). While necessary to keep the approach efficient and coherent, however, keeping just one labeled version does introduce uncertainty. Therefore the number of labels should be restricted, to avoid spreading out the probability mass over many different

[5] Our software for decomposing word alignments into HATs and for graphical visualization of these HATs is described in Maillette de Buy Wenniger and Sima'an (2014b) and is available for download from https://bitbucket.org/teamwildtreechase/hatparsing. The software is licensed under the terms of the GNU Lesser General Public License.

alternatives, making the selected rule versions and thereby the labels in general ultimately less reliable. Thirdly, the most common and well-known operators monotone ([0,1]) and inverted ([1,0]) have only one variant, while there are many variants of more complex operators for permutation and discontinuous reorderings. To avoid having the simpler but more frequent operators be obscured by a heap of complex but rare distinct operators, we bucket them to limit the total number of operators. Finally, in a elastic-substitution decoding setting, reducing the number of labels helps to keep the number of features down (while also reducing complexity and problems with search errors) and is altogether important to keep the soft constraints learnable by the tuner.

In what follows we define two approaches for bucketing the HAT operators. The first approach simply uses the identity of the bucket of the operator on the current node itself (hence 0th order), whereas the second approach employs a bucketing of the operator on the parent of the current node (1st order).[6]

Phrase-centric (0th-order) labels are based on the view of looking inside a phrase pair to see how it decomposes into sub-phrase pairs. The operator signifying how the sub-phrase pairs are reordered (target relative to source) is bucketed into a number of "permutation complexity" categories. As a baseline labeling approach, we can start out by using the two well-known cases of inversion transduction grammars (ITG) {*Monotone*, *Inverted*} and label everything[7] that falls outside these two categories with a default label "*X*" (leaving some Hiero nodes unlabeled). This leads to the following *coarse* phrase-centric labeling scheme, which we name $0^{th}{}_{ITG+}$:

1. *Monotonic (Mono)*: binarizable, fully monotone plus non-decomposable phrase pairs.
2. *Inverted (Inv)*: binarizable, fully inverted.
3. *X*: decomposable phrase pairs that are not binarizable.

A clear limitation of the above ITG-like labeling approach is that all phrase pairs that decompose into complex non-binarizable reordering patterns are not further distinguished. Furthermore, non-decomposable phrase pairs are lumped together with decomposable monotone phrase pairs, although they are in fact quite different. To overcome these problems we extend ITG in a way that further distinguishes the non-binarizable phrase pairs and also distinguishes non-decomposable phrase pairs from the rest. This gives a labeling scheme we will call simply 0*th*-order labeling, abbreviated 0^{th}, consisting of a more fine-grained set of five cases, ordered by increasing complexity (see examples in Fig. 7):

[6] We think of our labels as implementing a Markov approach to SCFG labeling. The first (0th order) labeling approach just describes the reordering information at the phrase pairs themselves, analogous to the way syntactic labels describe the syntactic category for the source and/or target side of phrase pairs in syntactic hierarchical SMT. The second (1st order) labeling approach describes the reordering relative to an embedding parent phrase pair, thereby looking not at the local reordering but at the reordering context of the parent.

[7] Non-decomposable phrase pairs (an example is the "Atomic" phrase pair in Fig. 7) will still be grouped together with Monotone phrase pairs (an example is the "Monotone" phrase pair in Fig. 7), since they are more similar to this category than to the catchall "*X*" category.

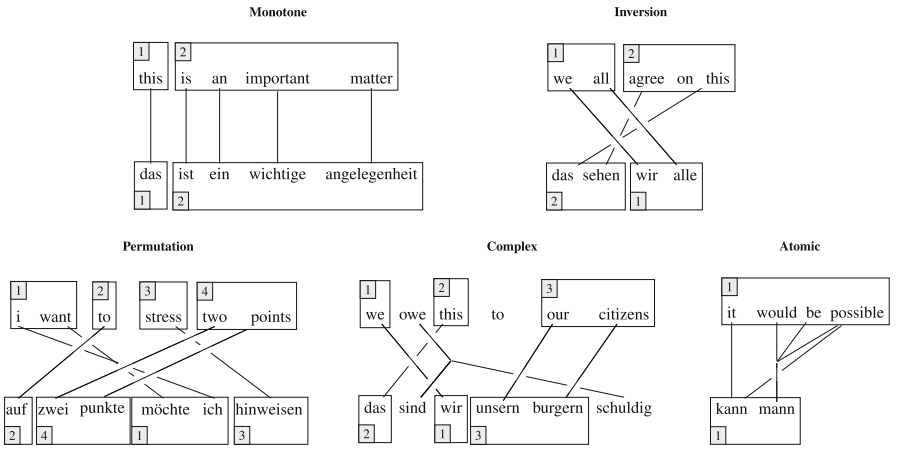

Fig. 7 Different types of phrase-centric alignment labels

1. *Atomic*: non-decomposable phrase pairs.
2. *Monotonic (Mono)*: binarizable, fully monotone.
3. *Inverted (Inv)*: binarizable, fully inverted.
4. *Permutation (Perm)*: decomposes into a permutation of four or more sub-phrases.[8]
5. *Complex (Comp)*: does not decompose into a permutation and contains at least one embedded phrase pair.

In Fig. 8, we show a phrase-complexity labeled derivation for the example in Fig. 3. Observe how the phrase-centric labels reflect the relative reordering at the node. For example, the *Inverted* label of node-pair 2 corresponds to the inversion in the alignment of ⟨we should, müssen wir⟩; in contrast, node-pair 1 is complex and discontinuous and the label is *Complex*.

Parent-relative (1*st-order*) *labels* capture the reordering that a phrase undergoes relative to an embedding parent phrase pair. This can be seen as a first-order view on reordering (if the phrase-centric type is considered a zero-order).

1. For a binarizable parent phrase pair with orientation $X_o \in \{Mono, Inv\}$, the source side of the phrase pair itself can either group to the left only *Left-Binding-*X_o, right only *Right-Binding-*X_o, or with both sides (*(Embedded) Fully-*X_o) of the source side of the embedding parent phrase pair.
2. *(Embedded) Fully-Discontinuous*: any phrase pair within a non-binarizable permutation or complex alignment containing discontinuity.
3. *Top*: phrase pairs that span the entire aligned sentence pair.

In cases where multiple labels are applicable, the simplest applicable label is chosen according to the following preference order: {*Fully-Monotone*, *Left/Right-Binding-Monotone*, *Fully-Inverted*, *Left/Right-Binding-Inverted*, *Fully-Discontinuous*, *TOP*}.

[8] A permutation of length 3 can always be decomposed into a set of simpler nested permutations of length 2. As an example, the permutation [3,1,2] can be decomposed as the simpler nested permutation [2,[1,2]]. Equally, any SCFG of rank 3 can always be converted into a SCFG of rank 2, but not all SCFGs with rank ≥ 3 are binarizable.

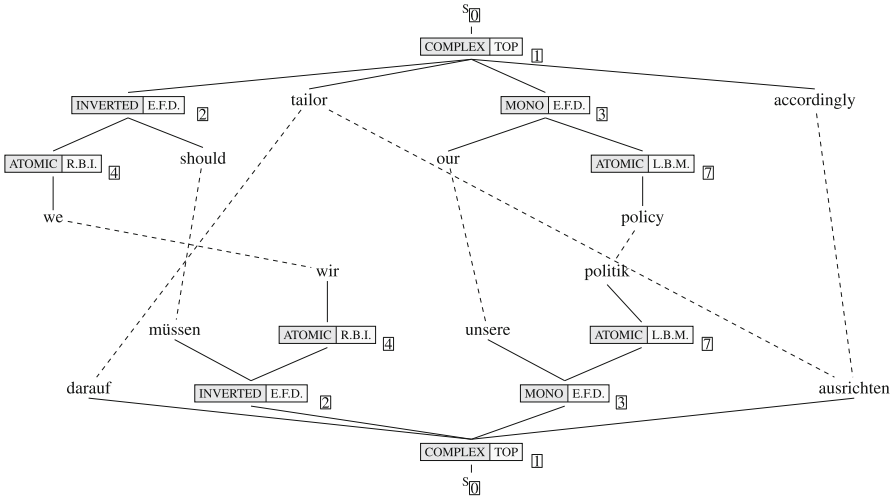

Fig. 8 Synchronous trees (implicit derivations) based on differently labeled Hiero grammars. The figure shows alternative labeling for every node: *Phrase-Centric* (0*th-order)* (*gray*) and *Parent-Relative* (1*st-order)* (*very light gray*). The abbreviations for the Parent-Relative labels are: *E.F.D.* embedded fully discontinuous, *R.B.I* right-binding inverted, *L.B.M.* left-binding monotone

In Fig. 8 the parent-relative labels in the derivation reflect the reordering taking place at the phrase pairs with respect to their parent node. Node 4 has a parent node that inverts the order and the sibling node it binds is on the right on the source side, so it is labeled "right-binding inverted" (R.B.I.); E.F.D. and L.B.M. are similar abbreviations for "(embedded) fully discontinuous" and "left-binding monotone", respectively. As yet another example node 7 in Fig. 8 is labeled "left-binding monotone" (L.B.M.) since it is monotone, but the alignment allows it only to bind to the left at the parent node, as opposed to only to the right or to both sides, whose cases would have yielded "right-binding monotone" R.B.M. and "(embedded) fully monotone" (E.F.M.) parent-relative reordering labels, respectively.

There is some similarity between the information gained in parent-relative reordering labels (by distinguishing left- and right side binding directions) with the information gained in lexicalized orientation models that keep track of orientation in both left-to-right and right-to-left direction, i.e. Galley and Manning (2008), Huck et al. (2013). For these models, determining the orientation in both directions slightly improves performance. Because in lexicalized orientation models keeping orientation in two directions helped, and since the binding direction for our monotone and inverted labels has similarity with it, we expected this binding direction to be also helpful for improving word order. Nevertheless, more fine-grained labels also increase sparsity and consequently make the learning problem more difficult. For this reason, the net effect of distinguishing binding direction remained hard to predict and could still have been negative. Accordingly, we also formed a set of *coarse* parent-relative labels ("$1^{st}{}_{Coarse}$") by collapsing the label pairs *Left/Right-Binding-Mono* and *Left/Right-Binding-Inverted* into single labels *One-Side-Binding-Mono* and *One-*

Side-Binding-Inv.[9] This coarse variant was tested in all settings, but gave in general comparable or lower results than the original, more fine-grained version, and is therefore left out to increase readability of the reported result tables.[10]

4 Features for elastic-substitution decoding

Labels used in hierarchical SMT are typically adapted from external resources such as taggers and parsers. As in our case, these labels are typically not fitted to the training data, with very few exceptions, e.g. Mylonakis and Sima'an (2011), Mylonakis (2012) and Hanneman and Lavie (2013). Unfortunately this means that the labels will either overfit or underfit, and when they are used as strict constraints on SCFG derivations they are likely to underperform. Experience with mismatch between syntactic labels and the data is abundant, and using elastic-substitution decoding with suitable label substitution features or a similar approach has been shown to be an effective solution (Venugopal et al. 2009; Chiang 2010; Marton et al. 2012). The intuition behind elastic-substitution decoding is that even though heuristic labels are not perfectly tailored to the data, they do provide useful information provided that the model is "allowed to learn" to use them only in as far as they can improve the final evaluation metric (usually BLEU, (Papineni et al. 2002)). Next we introduce the set of label-substitution features used in our experiments.

Basic label-substitution features Consist of a unique feature for every pair of labels $\langle L_\alpha, L_\beta \rangle$ in the grammar, signifying a rule with left-hand-side label L_β substituting on a gap labeled L_α. These features are combined with two more coarse features, "*Match*" and "*Nomatch*", indicating whether the substitution involves labels that match or not.

Figure 9 illustrates schematically the concept of label-substitution features. In this figure the substituting rule is substituted onto two gaps in the chart, which induces two label-substitution features indicated by the two ellipses. The situation is analogous for rules with just one gap. To make things concrete, assume that both the first nonterminal of the rule *N1* as well as the first gap it is substituted onto (*GAP1*) have the label *MONO*. Furthermore, assume that the second nonterminal *N2* has the label *COMPLEX* while the label of the gap *GAP2* it substitutes onto is *INV*. This situation results in the following two specific label-substitution features:

- subst(*MONO*,*MONO*),
- subst(*INV*,*COMPLEX*).

Sparse label-substitution features Every applied rule, abstracted by its orientation plus reordering label signature, is enriched with information regarding the nature of

[9] We could also further coarsen the 1^{st} labels by removing entirely all sub-distinctions of binding-type for the binarizable cases, but that would make the labeling essentially equal to the earlier mentioned $0^{th}{}_{ITG+}$ except for looking at the reordering occurring at the parent rather than inside the phrase pair itself. We did not explore this variant in this work, as the high similarity to the already explored $0^{th}{}_{ITG+}$ variant made it seem unlikely to add much extra information.

[10] The coarse version does sometimes perform better in combination with sparse features. We attribute this to the fact that sparse features can lead to overfitting, but only to a lesser degree with a coarser (and thus smaller) label set, since the number of sparse features is a polynomial function of the number of labels.

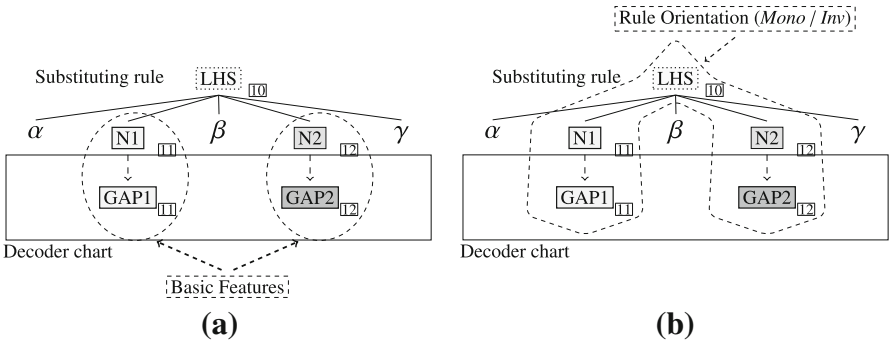

Fig. 9 Schematic view of label-substitution features. Labels/Gaps with the same filling in the figures correspond to the situation of a nonterminal/gap whose labels correspond (for N1/GAP1). *Fillings* of *different shades* (as for N2/GAP2 on the rigth in the two figures) indicate the situation where the label of the nonterminal and the gap is different. **a** Basic label-substitution features. **b** Sparse label-substitution feature

the labeled gaps it is substituting onto. This information is encoded as sparse features defined as follows: For every non-empty ordered set of gaps denoted $gaps$ and rule substituting on it $rule$ with left-hand-side $LHS(rule)$ and nonterminals $N(rule)$, binary features are added for the specific combinations of four tuples: $\langle LHS(rule), N(rule), L(gaps), O(rule)\rangle$, where $O(rule)$ is reordering orientation (inverted/monotone) internal to $rule$ and $L(gaps)$ the ordered set of labels belonging to $gaps$ in the derivation. This is illustrated in Fig. 9b by the dashed curve, which indicates these elements defining the sparse label-substitution feature for this rule substitution. Assuming again the label assignment mentioned before—$N1 = MONO$, $L(GAP1) = MONO$, $N2 = COMPLEX$, $L(GAP1) = INV$—and furthermore assuming the left-hand-side of the rule is $MONO$ and the orientation of the rule is monotone, we would obtain the following sparse label-substitution feature:

– $\langle MONO, \{MONO, COMPLEX\}, \{MONO, INV\}, monotone\rangle$.

Canonical labeled rules Typically when labeling Hiero rules there can be many different labeled variants of every original Hiero rule. With elastic-substitution decoding this leads to prohibitive computational cost. This also has the effect of making tuning the features more difficult. In practice, elastic-substitution decoding usually exploits a single labeled version per Hiero rule, which we call the "canonical labeled rule". Following Chiang (2010), this canonical form is the most frequent labeled variant.

5 Experiments

We choose to evaluate our models on two different language pairs: German–English and Chinese–English. The choice of these two pairs is driven by the knowledge that while in German word order is often tied with morphological changes at the word level, this is not the case for Chinese. [11]

[11] As one illustration of this, in German, some sentences can be written with different word orders and morphological case markings, while expressing the same meaning. Various examples are given in

Table 1 Size statistics for the training, development and testing datasets used in the experiments

Data set	# Sentence pairs	# Source words	Mean/Std source sentence length	# Target words	Mean/Std target sentence length
	German–English				
Training	994,861	20,358,970	20.46 ± 8.88	21,449,488	21.56 ± 9.14
Development	2000	55,526	27.76 ± 15.93	59,425	29.71 ± 16.99
Testing	2000	55,580	27.79 ± 15.67	59,153	29.58 ± 16.85
	Chinese–English				
Training	7,340,000	147,609,854	20.11 ± 9.25	159,567,205	21.74 ± 9.91
Development	1812	46,864	25.86 ± 13.02	54,665.5	30.17 ± 15.89
Testing	1912	48,250	25.24 ± 12.8	58,844.0	30.78 ± 16.69

For the Chinese–English dataset, there are 4 references for the target side of the development set and testing set. Hence, for these datasets the number of target words (#target words) is a mean taken over the four references, and the mean/std target sentence length is computed from the four references combined into one

Hence, we may expect different behaviour on the two language pairs, which could provide insight into the limitations of our approach (reordering approaches in general) for dealing with languages where word order and morphology are tied together.

All data is lowercased as a last pre-processing step. In all experiments we use our own grammar extractor for the generation of all grammars, including the baseline Hiero grammars. This enables us to use the same features (as far as applicable given the grammar formalism) and ensures that the grammars under comparison are identical in terms of using exactly the same set of extracted rules (differing only in labels and associated label features).

German–English The training data for our German–English experiments is derived from parliament proceedings sourced from the Europarl corpus (Koehn 2005), with WMT-07 development and test data. We used a maximum sentence length of 40 for filtering the training data. We employ 995,909 sentence pairs for training, 2000 for development and 2000 for testing (single reference per source sentence). An overview of these and other statistics about the training, development and testing dataset is shown in Table 1. Both source and target of all datasets are tokenized using the Moses (Koehn et al. 2007) tokenization script. We do not use compound splitting as part of the data preparation.[12] For these experiments both the baseline and our methods use a

Footnote 10 continued
(Müller 2002). As one example, the sentence "dass **der** Astronaut **den** Planeten entdeckt hat" and "dass **den** Planeten **der** Astronaut entdeckt hat" both have the same meaning: "that the astronaut discovered the planet". The morphological markings of "**der** Astronaut" and "**den** Planeten" disambiguate their function as subject and object, respectively, in both sentences.

[12] Although compound-splitting could be important for building the best possible system, this was not the goal in our experiments. Our goal was to create an experimental setup that allows for a fair, replicable comparison of our systems against Hiero and SAMT. As we believe that the potential disadvantage of omitting compound-splitting should affect all compared systems equally, given our goal, we judged that for the sake of simplicity it was reasonable to do so.

Table 2 Language model training corpora sizes

Language pair	#Sentences	#Words	Mean/Std sentence length
German–English	994,861	21,449,488	21.56 ± 9.14
Chinese–English	5,427,696	135,635,561	24.99 ± 15.99

4-g language model with Kneser-Ney smoothing (Chen and Goodman 1999) trained on the target side of the full original training set (995,909 sentences). Statistics about the data used to train the language models is shown in Table 2.

Chinese–English The training data for our Chinese–English experiments is formed by combining the full sentence-aligned *MultiUN* (Eisele and Chen 2010; Tiedemann 2012)[13] parallel corpus with the full sentence-aligned *Hong Kong Parallel Text*[14] parallel corpus from the Linguistic Data Consortium.[15] The *Hong Kong Parallel Text* data is in *traditional Chinese* and is thus first converted to *simplified Chinese* to be compatible with the rest of the data.[16] We used a maximum sentence length of 40 for filtering the training data. The combined dataset has 7,340,000 sentence pairs. The *MultiUN* dataset contains translated documents from the United Nations, similar in genre to the parliament domain. The *Hong Kong Parallel Text* in contrast contains a richer mix of domains, namely Hansards, Laws and News. For the development and test sets we use the *Multiple-Translation Chinese* datasets from LDC, parts 1–4,[17] which contain sentences from the News domain. We combined parts 2 and 3 to form the development set (1813 sentence pairs), and parts 1 and 4 to form the test set (1912 sentence pairs). For both development and testing we use 4 references. The Chinese source side of all datasets is segmented using the Stanford Segmenter (Chang et al. 2008).[18] The English target side of all datasets is tokenized using the Moses tokenization script.

For these experiments both the baseline and our method use a 4-g language model with Kneser–Ney smoothing trained on 5,427,696 sentences of *domain-specific*[19] news data taken from the "Xinhua" subcorpus of the English Gigaword corpus of LDC.[20]

[13] Freely available and downloaded from http://opus.lingfil.uu.se/

[14] The *Hong Kong Parallel Text* corpus contains a significant amount of duplicate sentence pairs. We removed these duplicates and kept only one copy per unique sentence pair.

[15] The LDC catalog number of this dataset is LDC2004T08.

[16] Using a simple conversion script downloaded from http://www.mandarintools.com/zhcode.html

[17] LDC catalog numbers: LDC2002T01, DC2003T17, LDC2004T07 and LDC2004T07.

[18] Downloaded from http://nlp.stanford.edu/software/segmenter.shtml

[19] For Chinese–English translation the different domain of the train data (mainly parliament) and development/test data (news) requires usage of a domain-specific language model to obtain optimal results. For German–English, all data is from the parliament domain, so a language model trained on the (translation model) training data is already domain-specific.

[20] The LDC catalog number of this dataset is LDC2003T05.

Table 3 An overview of our labeling schemes, their system names and the components they exploit

System name	Label order	Label granularity	Matching type	Label substitution features set
Hiero-$0^{th}{}_{ITG+}$	0^{th} order	Coarse	Strict	None
Hiero-0^{th}	0^{th} order	Fine	Strict	None
Hiero-1^{st}	1^{th} order	Fine	Strict	None
Hiero-$0^{th}{}_{ITG+}$-Sft$_B$	0^{th} order	Coarse	Soft	Basic
Hiero-0^{th}-Sft$_B$	0^{th} order	Fine	Soft	Basic
Hiero-1^{st}-Sft$_B$	1^{th} order	Fine	Soft	Basic
Hiero-$0^{th}{}_{ITG+}$-Sft$_{B+S}$	0^{th} order	Coarse	Soft	Basic + Sparse
Hiero-0^{th}-Sft$_{B+S}$	0^{th} order	Fine	Soft	Basic + Sparse
Hiero-1^{st}-Sft$_{B+S}$	1^{th} order	Fine	Soft	Basic + Sparse

The suffixes in the system names in the first table column are abbreviations, directly corresponding to the system dimensions in the other columns: label order $\{0^{th}, 1^{th}\}$, label granularity ("$ITG+$" indicating the coarse variant of 0^{th} order labels), matching type (default is strict,"Sft" denotes elastic-substitution decoding), label substitution type ("B" denoting basic- and "$B+S$" basic + sparse label substitution features)

5.1 Experimental structure

We compare our reordering-labeled systems against two baseline systems: the (unlabeled) Hiero and the target-language syntax-labeled variant known as SAMT. In our experiments we explore the influence of three dimensions of bilingual reordering labels on translation accuracy. These dimensions are:

- *label order*: the type/order of the labeling $\{0th, 1st\}$,
- *label granularity*: granularity of the labeling {Coarse,Fine},
- *matching type*: the type of label matching performed during decoding {Strict,Soft},
- *label substitution feature set*: the type of label substitution features that is used during decoding, if any.

An overview of the naming of our reordering labeled systems is given in Table 3.

SAMT We use the original label extraction scheme as described in Zollmann and Venugopal (2006). In particular we allow the binary "\", "/" and "+" operators. These operators are based on combinatory categorial grammar (Steedman 2000). Using $NT1$ and $NT2$ to represent syntactic constituents, these operators informally denote the following:

- $NT1 + NT2$: A partition into two sub-spans that both correspond to constituents,
- $NT1/NT2$: $NT1$ missing a $NT2$ on the right,
- $NT2\backslash NT1$: $NT1$ missing a $NT2$ on the left.[21]

[21] Combinatory categorial grammar (CCG) (Steedman 2000) uses $NT1\backslash NT2$ in place of $NT2\backslash NT1$, to indicate that $NT1$ misses $NT2$ on the left. The different notation used by SAMT, which places the argument itself to the left in this case can be confusing to people that are used to CCG notation.

To keep the grammar size manageable, we do not allow "double plus" (A+B+C) type labels, and we do not allow non-lexicalized rules. The choice not to allow non-lexicalized rules was made to keep SAMT (like our systems) comparable to Hiero, apart from the labels. This avoids giving SAMT additional reordering capacity (through abstract rules) which Hiero lacks, and thereby also keeps decoding times more workable.[22] Finally, we use SAMT, as in the original work, with strict matching.[23]

Training and decoding details Our experiments use Joshua (Ganitkevitch et al. 2012) with Viterbi best derivation. Baseline experiments use normal decoding, whereas elastic-substitution decoding experiments relax the label-matching constraints while adding label-substitution features to facilitate learning of label-substitution preferences.

For training we use standard Hiero grammar extraction constraints (Chiang 2007) (phrase pairs with source spans up to 10 words; abstract rules are forbidden). During decoding a maximum span of 10 words on the source side is maintained.

Following common practice, we use relative frequency estimates for phrase probabilities, lexical probabilities and generative rule probability. We additionally add common binary indicator features for glue rules, only terminals/nonterminals on the right hand side, terminals on the source but not the target side, terminals on the target but not the source side, and monotonic rules. We furthermore add the common rule application count, word penalty and rarity penalty features,[24] see e.g. Zollmann and Venugopal (2006) for details.

With Hiero and the labeled systems in soft-matching setups, we use the Hiero phrase probabilities in both directions (Chiang 2007), making the labeled systems weakly equivalent to Hiero apart from their label-substitution features. For the labeled systems in the strict matching systems, we follow Zollmann and Venugopal (2006) in using the phrase probabilities that use the labels as well as all smoothed versions of these phrase probabilities.[25] Smoothing is done by removing the labels on source and/or target side in all combinations. In what follows, we abbreviate source as src, target as tgt, and use un to indicate the labels are removed. The phrase probability features for the labeled systems in the strict matching setting are:

[22] For example, Li et al. (2012) show that such abstract rules can by themselves provide performance gains on top of improvements from the labels used in normal Hiero rules.

[23] We spent major effort at implementing elastic-substitution decoding for SAMT as in Chiang (2010) but faced huge scalability issues due to the number of labels which give problems for the implementation of dot items in Joshua (Ganitkevitch et al. 2012).

[24] Rule application count and word penalty simply count the number of rules and words, respectively. This allows the model to learn preferences for longer or shorter derivations, and longer or shorter translations. The rarity penalty for a rule r is defined as $\theta_{rare} = exp(\frac{1}{\#(r)})$ with $\#(r)$ the number of times a rule has been seen during training. This allows penalization of derivations using rare rules.

[25] We use only the Hiero phrase-probability features for the labeled systems in the elastic-substitution decoding setting, to keep them as close as possible to Hiero, so that the effect of the label-substitution features can be measured purely. However, for the systems in the strict matching setting, we use the phrase probability features and phrase probability smoothing features that involve the labels. In this setting we involve the labels to allow them to influence the translation decisions through the phrase probabilities. However, when using the labels in the phrase probabilities, the smoothed variants are necessary to avoid sparsity problems, particularly with the sparse SAMT labels (Zollmann 2011).

– $\hat{p}(tgt(r)|src(r))$: Phrase probability target side given source side,
– $\hat{p}(src(r)|tgt(r))$: Phrase probability source side given target side.

which are reinforced by the following phrase-probability smoothing features:

– $\hat{p}(tgt(r)|un(src(r)))$ and $\hat{p}(un(src(r))|tgt(r))$: labels removed on source side,
– $\hat{p}(un(tgt(r))|src(r))$ and $\hat{p}(src(r)|un(tgt(r)))$: labels removed on target side,
– $\hat{p}(un(tgt(r))|un(src(r)))$, $\hat{p}(un(src(r))|un(tgt(r)))$: labels removed on both sides.

When labels on both sides are removed we obtain the original Hiero phrase probabilities.

We train our systems using (batch k-best) MIRA (Cherry and Foster 2012) as borrowed by Joshua from the Moses codebase, allowing up to 30 tuning iterations. Following standard practice, we tune on BLEU, and after tuning we use the configuration with the highest scores on the development set with actual (corpus-level) BLEU evaluation. We report lowercase BLEU (Papineni et al. 2002), METEOR (Denkowski and Lavie 2011), BEER (Stanojević and Sima'an 2014) and TER (Snover et al. 2006) scores for the test set. We also report average translation length as a percentage of the reference length for all systems. This is useful for analysis of possible overfitting. In our experiments we repeated each experiment three times to counter unreliable conclusions due to optimizer variance, so the scores are averages over three runs of tuning plus testing. We use MultEval version 0.5.1.[26] for computing these metrics. We also use MultEval's implementation of statistical significance testing between systems, which is based on multiple optimizer runs and approximate randomization. Differences that are statistically significant with respect to the HIERO baseline and correspond to improvement/worsening are marked with $\triangle H/\triangledown H$ at the $p \leq .05$ level and $\blacktriangle H/\blacktriangledown H$ at the $p \leq .01$ level. For average translation length either higher or lower may be better, depending on whether the baseline length was too low or too high. We therefore use $\square H/\blacksquare H$ in case of length to mark significant *change* with respect to the baseline at the $p \leq .05$ / $p \leq .01$ level. Apart from computing the statistical significance of differences with respect to the HIERO baseline, we also computed statistical significance of differences with respect to the SAMT baseline. The significance for these differences are analogously marked $\triangle S/\triangledown S/\square S$ at the $p \leq .05$ level and $\blacktriangle S/\blacktriangledown S/\blacksquare S$ at the $p \leq .01$ level. We also report the Kendall Reordering Score (KRS), which is the reordering-only variant of the LR-score (Birch et al. 2010; Birch and Osborne 2010) (without the optional interpolation with BLEU) and which is a sentence-level score. For the computation of statistical significance of this metric we use our own implementation of the *sign test*[27] (Dixon and Mood 1946), as also described in (Koehn 2010). For every experiment we use boldface to accentuate the highest score across systems for all metrics except TER. Since TER is an error metric,

[26] https://github.com/jhclark/multeval

[27] To make optimal usage of the 3 runs we computed equally-weighted improvement/worsening counts for all possible 3×3 baseline output / system output pairs and use those weighted counts in the sign test. While traditionally the procedure of dealing with ties in the sign test is discarding them, there is in fact no real consensus with respect to their correct treatment. However, recent literature explains that it may sometimes be better to equally divide the ties between two systems (Rayner and Best 1999). This is intuitively a more 'conservative' approach which we adopted in our experiments.

lower values are better, so we instead mark the lowest value with boldface for it. For length, neither higher or lower is necessarily better. What is best is a length closest to the reference length. Therefore, in case of the length metric, we boldface the value that is closest to 100, in absolute terms.

5.2 Primary results: soft bilingual constraints and basic+sparse label-substitution features

Table 4 shows the primary results of our full labeling scheme which uses elastic-substitution decoding both with basic and sparse label-substitution features. *Hiero* is the Hiero baseline, beneath it are shown the systems that use elastic-substitution decoding (Sft): *Hiero*-$0^{th}{}_{ITG+}$-Sft and *Hiero*-0^{th}-Sft using $0th$-order labels. *Hiero* -1^{st}-Sftcorresponds to the system with $1st$-order, parent-relative labels.

German–English Hiero-0^{th}-Sft_{B+S} (with BLEU score of 28.57) slightly outperforms both Hiero and SAMT baselines by almost 0.2 BLEU points, which is statistically significant. We remind the reader that German–English is rather difficult because word order in German is tied with morphological variations at the word level, for which our model, like other models of reordering, does not have a proper solution. This goes to highlight the limitations of these kind of models in general.

Chinese–English Hiero-1^{st}-Sft_{B+S} has the highest score for BLEU for all tested systems on Chinese–English translation, outperforming Hiero and SAMT by approximately 0.8–1.1 BLEU points. For metric TER, all labeled systems, including SAMT, suffer performance loss in comparison to Hiero. We found out that the length ratio for the output of the Hiero-1^{st}-Sft_{B+S} system to the reference is 0.99 whereas the ratio for Hiero's output is 0.97, i.e. it seems that TER is penalizing more heavily longer output even if it is closer in length to the reference (cf. (He and Way 2009)). This turns out largely due to the fact that the 4-g LM tuned weight for the labeled systems is always far lower than for Hiero, suggesting that the 4-g LM has a smaller contribution during tuning for BLEU. Tuning for BLEU is not guaranteed to give improved performance on all metrics, as noted by He and Way (2009), but we do see here improved performance for three out of four metrics.

In Table 5 we show the absolute and relative sizes of the grammars for the different label types. The reported sizes are for grammars that are filtered for the test set and that are taken from the systems that use the labels in a strict matching setting. Note that for the systems that use the bilingual reordering labels as soft bilingual constraints, the grammar size is always equal to that of Hiero. The reason for this is that, as mentioned earlier, with elastic-substitution decoding we use only one canonical labeled rule per Hiero rule. Looking now at the grammar sizes in the table, we see that the size of the grammar for SAMT is on average more than a factor of 4 bigger than the one used by Hiero and the reordering labeled systems in the elastic-substitution decoding setting. At the same time, the improvement over both SAMT and Hiero by the reordering labeled systems is considerable, especially for Chinese–English. Even in the strict matching setting, the reordering labeled systems still have grammar sizes that are

Table 4 Primary mean results bilingual labels with elastic-substitution decoding and basic plus sparse label-substitution features

System name	BLEU ↑	METEOR ↑	BEER ↑	TER ↓	KRS ↑	Length
	German–English					
Hiero	28.39	32.94	19.01	58.01 ▼S	67.44	100.60 ■S
SAMT	28.32	32.88	18.81	57.70 ▲H	**67.63**	**100.07** ■H
Hiero-$0^{th}{}_{ITG+}$-Sft$_{B+S}$	28.48 △H △S	32.93	18.97	57.69 ▲H	67.37	100.08 ■H
Hiero-0^{th}-Sft$_{B+S}$	**28.57** ▲H ▲S	32.92	18.99	**57.65** ▲H	67.41	100.16 ■H
Hiero-1^{st}-Sft$_{B+S}$	28.47	**33.03** ▲H ▲S	**19.07**	57.77 ▲H	67.45	100.59 ■S
	Chinese–English					
Hiero	31.63 ▽S	30.56	13.15	59.28 ▲S	58.03 ▼S	97.15 ■S
SAMT	31.87 △H	30.61	13.38	59.97 ▼H	59.94 ▲H	98.46 ■H
Hiero-$0^{th}{}_{ITG+}$-Sft$_{B+S}$	32.02 ▲H	30.66 ▲H	13.20	**59.12** △H ▲S	58.66 ▲H ▼S	97.41 ■H ■S
Hiero-0^{th}-Sft$_{B+S}$	32.43 ▲H ▲S	30.96 ▲H ▲S	13.54	60.33 ▼H ▼S	**60.17** ▲H △S	**99.35** ■H ■S
Hiero-1^{st}-Sft$_{B+S}$	**32.69** ▲H ▲S	**31.01** ▲H ▲S	**13.65** △H	60.02 ▼H	59.99 ▲H	99.10 ■H ■S

Statistical significance is dependent on the variance of resampled scores, and hence sometimes different for same mean scores across different systems

In this and the following result tables, statistically significant improvement/worsening/change with respect to the HIERO baseline is marked with △H/▽H/□H/ at the $p \leq .05$ level and ▲H/▼H/■H at the $p \leq .01$ level. Analogously, statistically significant improvement/worsening/change with respect to the SAMT baseline is marked with △S/▽S/□S/ and ▲S/▼S/■S

Table 5 Filtered test grammar sizes for different label types and different language pairs

Label type	German–English		Chinese–English	
	Absolute size	Relative size	Absolute size	Relative size
Hiero	17.2	1	33.4	1
SAMT	74.9	4.35	154.7	4.63
Hiero-$0^{th}{}_{ITG+}$	19.1	1.11	38.4	1.15
Hiero-0^{th}	28.7	1.67	55.7	1.67
Hiero-1^{st}	23.6	1.37	48.9	1.46

This means that, in contrast to the elastic-substitution decoding system that allows only one canonically labeled rule per Hiero (unlabeled) rule type, there can be multiple alternative labeled rule versions per Hiero rule type

Absolute sizes are in millions of rules. Relative sizes are with respect to the Hiero (baseline) grammar, or equivalently with respect to the grammars used in the elastic-substitution decoding experiments, which are equal in size to Hiero. Grammars are taken from the strict matching systems for the label types

much smaller than SAMT, at most 1.67 times the size of the baseline Hiero grammar. Furthermore, in what follows we will see that also for these systems the reordering labeled systems are performing as well as SAMT and Hiero or better.

Next we perform ablation experiments where we isolate the effects of using sparse features on top of the basic ones, and after that the using elastic-substitution decoding vs. the traditional mere strict label matching.

5.3 Experiments with elastic-substitution decoding with basic label-substitution features only

Now we isolate out the sparse features and use only the basic label-substitution features with elastic-substitution decoding. The results are shown in Table 6. For brevity, we show only the baseline results and results for Hiero-1^{st}-Sft_B, which scores overall the best amongst the reordering labeled systems using the basic feature set.

German–English There are only minor improvements for BLEU and METEOR over the Hiero and SAMT baselines, with somewhat bigger improvements for TER. However, SAMT has the highest improvements for TER and KRS over Hiero on this language pair.

Chinese–English the improvements are considerable: +0.98 BLEU improvement over the Hiero baseline for Hiero-1^{st}-Sft as well as +0.42 METEOR and +1.81 KRS. TER is worse by +0.85 for this system. For Chinese–English the *Fine* version of the labels gives overall superior results for both $0th$-order and $1st$-order labels.

Compared with the primary system (basic+sparse label-substitution features) results in Table 4, we see that the added nuances of sparse label-substitution features can make a difference, strongly so for German–English and to a lesser degree also for Chinese–English.

Table 6 Mean results bilingual labels with elastic-substitution decoding and only basic label-substitution features (see Footnote 18)

System name	BLEU ↑	METEOR ↑	BEER ↑	TER ↓	KRS ↑	Length
	German–English					
Hiero	28.39	32.94	**19.01**	58.01 ▼S	67.44	100.60 ■S
SAMT	28.32	32.88	18.81	**57.70** ▲H	**67.63**	**100.07** ■H
Hiero-1st-Sft$_B$	**28.45**	**33.00** △H ▲S	**19.01**	57.79 ▲H	67.45	100.52 ■S
	Chinese–English					
Hiero	31.63 ▽S	30.56	13.15	**59.28** ▲S	58.03 ▼S	97.15 ■S
SAMT	31.87 △H	30.61	13.38	59.97 ▼H	**59.94** ▲H	98.46 ■H
Hiero-1st-Sft$_B$	**32.61** ▲H ▲S	**30.98** ▲H ▲S	**13.58** △H	60.19 ▼H ▽S	59.84 ■H	**99.03** ■H ■S

Table 7 Mean results bilingual labels with strict matching (see Footnote 18)

System name	BLEU ↑	METEOR ↑	BEER ↑	TER ↓	KRS ↑	Length
	Chinese–English					
Hiero	31.63 ▽*S*	30.56	13.15	**59.28** ▲*S*	58.03 ▼*S*	97.15 ■*S*
SAMT	31.87 △*H*	30.61	13.38	59.97 ▼*H*	**59.94** ▲*H*	98.46 ■*H*
Hiero-$0^{th}{}_{ITG+}$	**31.94** ▲*H*	**30.84** ▲*H* ▲*S*	13.37	60.76 ▼*H* ▼*S*	59.45 ▲*H*	**99.13** ■*H* ■*S*
Hiero-0^{th}	31.90 ▲*H*	30.79 ▲*H* ▲*S*	**13.45**	60.11 ▼*H*	59.68 ▲*H*	98.65 ■*H*
Hiero-1^{st}	31.77	30.62	13.20	60.13 ▼*H*	59.89 ▲*H*	98.47 ■*H*

5.4 Experiments with strict label matching: no added softeners

Now we explore the added value of soft label matching features by excluding them and using the labels as traditional grammar labels (hard constraints). In contrast to the elastic-substitution decoding experiments where only canonical labeled rules are included in the grammar, in this setting all labeled rule variants are used. The motivation for this difference is that in a strict label-matching setting, coverage-loss problems arise during translation. Using all labeled rule variants, as common in strict labeling approaches (e.g., (Zollmann and Venugopal 2006)), does not solve these problems but at least reduces them as much as possible in this setting.

The results are shown in Table 7. Here we only show results for Chinese–English. For brevity, we omit the results for German–English as there are no clear improvements over the baseline for this language pair, or at least not for BLEU, the tuning metric used. For the computation of SAMT for Chinese–English we initially had problems with grammar extraction due to the enormous size of even the filtered grammar. We finally overcame this by extracting the grammar in parts and merging them. To be exact, this does make some of the feature values which involve normalization potentially slightly different from what they would have been if they were directly computed for the full grammar in one go. However, due to the inherently highly heuristic nature of these features, this is assumed to not have a real effect on the actual results. All systems in the table use the default decoding with strict label matching.

Chinese–English overall Hiero-$0^{th}{}_{ITG+}$ shows the biggest improvements, namely significant improvements of +0.31 BLEU, +0.28 METEOR and +1.42 KRS. TER is the only metric that worsens, and considerably so with +1.48 points. This system is also superior to SAMT for BLEU and METEOR, but not for TER and KRS. SAMT achieves the highest improvement in KRS, namely 1.91 points higher than the Hiero baseline. Just like the reordering labeled systems, SAMT also loses performance on TER over Hiero.

5.5 Summary of experimental findings

We may summarize the experiments with the following conclusions:

– Whereas for German–English the performance improvement is rather modest, for Chinese–English we see considerable improvements and overall the best results for

the combination of elastic-substitution decoding, with the *Fine* $1st$-order variant of the labeled systems using basic plus sparse label-substitution features (Hiero-1^{st}-Sft$_{B+S}$).
- Crucial for performance is the use of a soft-constraint approach to label matching, as opposed to strict-matching.
- Particularly interesting is the comparison to the ITG+ labeled variant of our scheme. While ITG+ labeling already obtains improved performance, we do see that a more elaborate labeling scheme (as simple as our bucketing) may bring about even further improvement.
- Finally, the different reordering labeled systems outperform SAMT on BLEU and METEOR and also for TER and/or KRS. Interestingly, the reordering-labeled grammars are comparable in size to Hiero's, i.e. less than one third of SAMT grammar size.

In conclusion, we find it encouraging that our automatic labeling approach, which does not demand additional (monolingual) resources beyond a parallel corpus,[28] gives comparable or better performance improvement to syntax-labeling approaches. We hypothesize that the two types of labeling capture complementary reordering information, particularly that target syntax in SAMT allows more fluent MT output, strengthening the target language model.

6 Analysis

In this section we will give a deeper analysis of the qualitative results obtained and discussed so far. We have seen how soft reordering constraints can significantly improve the results. These constraints are effectuated by using bilingual reordering labels in combination with elastic-substitution decoding and label-substitution features. The question is how exactly these constraints contribute to the performance. We will focus on two dimensions of analysis, with each dimension chosen to shed light on a different aspect of the effect of the reordering constraints. The dimensions we will look at are:

1. Interaction between reordering constraints and language model: to what extent does the function of soft reordering constraints overlap with the function of a strong language model, and to what extent does it add information that even a strong language model cannot capture?
2. Can we derive some qualitative understanding of where the quantitative improvements come from, and whether these improvements are valid?

Each of these dimensions of analysis will now be discussed in detail in the following subsections.

[28] As usual in contemporary SMT, our approach also needs an adequate target language model in the complete system in order to achieve state of the art performance. In particular, as is also the case for syntactic labeling approaches, we do not aim to replace the language model with our labels. We rather build upon the already reasonable translation afforded by a good language model, and strive to use this as a basis to improve performance further.

6.1 An experiment with a unigram language model: how good is the reordering model?

In this subsection we try to better understand the interaction between reordering model and language model. Here we contrast the experiments from the preceding section with new experiments with the same SCFG-based reordering models but intergrated with a unigram LM (instead of a 4-g LM). A unigram language model informs about prior lexical preferences but leaves the word order to the SCFG-based reordering model. This should provide some insight into two aspects: (1) the importance of the LM for final performance, and (2) the role of the bilingual labels in affecting final word order choice.

Table 8 shows the results for these experiments on German–English and Chinese–English and clearly the results drop substantially from the experiments in the preceding section with a 4-g LM. Some specific remarks per language pair are due.

GERMAN−ENGLISH: The relative improvement of the labeled systems over the Hiero baseline has increased to +0.65 BLEU points. Similar increase in improvement can also been seen for METEOR, BEER and KRS. The labeling seems to give a better reordering model, although the 4-g LM seems to catch up with it and reduce the margin of improvement.

CHINESE−ENGLISH: The relative improvements over the baseline are considerably smaller in the unigram LM experiments relative to the 4-g LM. Nevertheless, the best system still achieves approximately +0.3 BLEU improvement while also improving TER by approximately -0.2. Apparently, the labeled reordering model here is better than Hiero in reranking the hypotheses but there is a set of top-ranking hypotheses that cannot be differentiated well without a strong LM.

The drop for Chinese–English (-12 BLEU) is markedly larger than the drop for German–English (-5 BLEU), suggesting that the 4-g LM could be specifically important for discriminating between the top-ranking reorderings among the hypothesis translations for Chinese–English.

Table 8 Results for extra analysis translation experiments using only a Unigram Language Model (see Footnote 18)

System name	BLEU ↑	METEOR ↑	BEER ↑	TER ↓	KRS ↑	Length
	German–English					
Hiero	23.67	31.27	16.32	61.19	65.79	99.18
Hiero-1^{st}-Sft$_B$	24.15 ▲H	31.37 ▲H	16.67	61.09	**66.01**	99.87 ■H
Hiero-1^{st}-Sft$_{B+S}$	**24.32** ▲H	**31.39** ▲H	**16.78**	**61.02** △H	66.00	**99.94** ■H
	Chinese–English					
Hiero	20.23	27.88	9.50	66.57	58.56	98.60
Hiero-1^{st}-Sft$_B$	**20.51** ▲H	27.91	9.51	**66.36** ▲H	**58.82**	98.42 □H
Hiero-1^{st}-Sft$_{B+S}$	20.49 ▲H	**27.93**	**9.55**	66.81 ▼H	58.78	**98.78** □H

6.2 Qualitative analysis

A qualitative analysis can give some additional insight about what is going in the actual translations, which quantitative scores fail to provide. On the other hand qualitative analysis has the disadvantage of being biased and relying on a very small sample. While we cannot really alleviate the drawback of a small sample here, we do try to alleviate the problem of selection bias by looking at some of the most improved test sentences according to METEOR as well as some test sentences that gave the highest drop in METEOR score. We used METEOR as opposed to BLEU for the selection, because it is better as a sentence-level score, and has a more explicit reordering component. By considering an equal number of improved and worsened examples in this way, selected by a clear criterion (highest gain/drop in METEOR score), we hope to be able to form a somewhat more objective opinion based on the selected examples. Here we chose to look at Chinese–English examples, since Chinese–English is the language pair where we observe the greatest benefits from our method in terms of improving word order, as indicated by BLEU, METEOR and KRS improvements.

Looking first at the three improved examples in Table 9, we see that in all cases there is a clear improvement in word order (structure) as well as lexical selection. Looking next at the examples where performance worsens in Table 10, the errors seem to be mainly in lexical selection. A reason for this could be that the labeled system assigns a relatively lower weight to the language model, which may make it more susceptible to make such lexical selection errors. At the same time these examples do seem to support the expected increased capability in getting the global reordering structure right of the reordering labeled system.

While the drawbacks of qualitative analysis discourage us from drawing strong conclusions from this, the examples do seem to give some additional support for the thesis that in Chinese–English translation reordering labels help to improve word order and global sentence structure. This improvement seems to sometimes come at a price in the quality of lexical selection, possibly due to the relatively lower weight of the language model in comparison to the Hiero baseline.

6.3 Summary of findings from analysis

The conclusions from these different analyses can be summarized as follows:

- There is overlap between the work done by a strong language model and work done by the labeled reordering models. However, Hiero's performance depends to a larger extent on the 4-g language model than the labeled reordering systems, suggesting that the latter systems give more adequate reordering.
- The qualitative analysis of a small number of examples shows that also on the qualitative level there is evidence that the labels are effective in improving both reordering and lexical selection.

Table 9 Source sentence, reference, baseline and best system output for sentences 959, 1442 and 1239 from the Chinese–English testset

Sentence type	Sentence contents
Source sentence	泰 还 未 摆脱 危机
Reference1	France drawing up military withdrawal plan from bosnia-herzegovina
Reference2	France studying plan to withdraw its troops from bosnia-herzegovina
Reference3	France studies plan to withdraw troops from bosnia and herzegovina
Reference4	France considering troops withdrawal from bosnia and herzegovina
Hiero	Law research plan for withdrawal from bosnia
Hiero-1^{st}-Sft$_{B+S}$	Law is studying plans to withdraw its troops from bosnia and herzegovina
Source sentence	一 位 南韩 政府 官员 说 ， 昨天 南韩 已 向 北韩 发出 邀请 ， 请 他们 派 观察员 来 观摩 这 次 军事 演习 。
Reference1	A south korean government official said that south korea issued an invitation to north korea yesterday
	Asking them to send observers to watch this military exercise
Reference2	A south korea government official said south korea had already sent an invitation to north korea yesterday
	Asking them to send observers to view and learn from the military exercise
Reference3	A south korean government official said that south korea offered an invitation to north korea yesterday
	To send their observers to watch the military exercise
Reference4	A south korean official said that south korea sent invitation to north korea yesterday
	Asking it to send observers to the drills
Hiero	A south korean government official said that south korea has issued invitations to north korea
	Yesterday invited them to attend the military exercises as observers
Hiero-1^{st}-Sft$_{B+S}$	A south korean government official said that south korea has issued invitations to north korea yesterday
	Asking them to send observers to attend the meeting of the military exercise

Table 9 continued

Sentence type	Sentence contents
Source sentence	外交 人员 将 搭乘 第五 架 飞机 返国 。
Reference1	Diplomatic staff will take the fifth plane home
Reference2	Diplomatic staff would go home in a fifth plane
Reference3	Diplomats are to come back home aboard the fifth plane
Reference4	Diplomatic staff would be airlifted on a fifth plane
Hiero	Diplomats fifth aircraft will fly for repatriation
Hiero-1^{st}-Sft_{B+S}	Diplomatic personnel will travel on to the fifth aircraft for repatriation

These are amongst the test sentences with the highest improvement in METEOR (omitting some very short sentences and sentences with unknown words)

Table 10 Source sentence, reference, baseline and best system output for sentences 1870, 937 and 1833 from the Chinese–English testset

Sentence type	Sentence contents
Source sentence	埃及 航空 于 今年 1 月 正式 开通 首 条 开罗 至 北京 的 直飞 航线 。
Reference1	Egyptair officially opened the first direct flight route from cairo to beijing in january this year
Reference2	Egypt air set up the first direct flight between cairo and beijing in january this year
Reference3	The egyptian airline officially opened its first direct flight from cairo to beijing this january
Reference4	Air egypt formally opened the first direct flight line from cairo to beijing in january
Hiero	Egypt air in january this year was officially opened its first direct flight from cairo to beijing
Hiero-1^{st}-Sft$_{B+S}$	Egyptian aviation was officially opened in january this year. The first non-stop service from beijing to cairo
Source sentence	颁奖 仪式 在 菲律宾 文化 中心 隆重 举行 。
Reference1	The awarding ceremony was solemnly held at the philippines cultural center
Reference2	The award ceremony was solemnly held in the philippine cultural center
Reference3	The presentation ceremony was solemnly held at the philippine culture center
Reference4	The awarding ceremony was performed solemnly in philippine cultural center
Hiero	The award ceremony held in the philippines cultural center grand
Hiero-1^{st}-Sft$_{B+S}$	A ceremony held at the cultural center grand
Source Sentence	这些 国家 已经 停止 进口 巴拉圭 牛肉 。
Reference1	These countries have already stopped beef imports from paraguay
Reference2	These countries have suspended the import of paraguayan beef
Reference3	These countries have stopped importing beef from paraguay
Reference4	These countries have stopped importing beef from paraguay
Hiero	These countries have to stop importing beef in paraguay
Hiero-1^{st}-Sft$_{B+S}$	These countries have put an end to the paraguayan beef imports

These are amongst the test sentences with the highest performance loss in METEOR (omitting some very short sentences and sentences with unknown words)

7 Related work

7.1 Syntax-based labels

A range of (distantly) related work exploits syntax for Hiero models, e.g. Huang et al. (2006), Liu et al. (2006), Zollmann and Venugopal (2006), Mi et al. (2008), Mi and Huang (2008), Almaghout et al. (2010), Almaghout et al. (2012), Li et al. (2012). In terms of labeling Hiero rules, SAMT (Zollmann and Venugopal 2006; Mylonakis and Sima'an 2011) exploits a 'softer' notion of syntax by fitting the CCG-like syntactic labels to non-constituent phrases.

7.2 Label clustering methods

Approaches to automatically coarsen the label set used by SAMT are explored in Hanneman and Lavie (2011, 2013). In this approach, the similarity between conditional probability distributions of labels is used to merge labels. The conditional probability of a source label s_i given a target label t_j is computed with simple relative frequency estimation, counting the frequency of s_i and t_j together and dividing by the total frequency of t_j in combination with any source label $s_i \in S$. The computation of conditional probabilities for target labels given source labels is analogous. Based on these distributions $L1$ distances are computed for all pairs of labels, in both source-to-target and target-to-source directions. Finally, the pair of labels with the smallest $L1$ distance between corresponding label distributions in either direction is merged. This is further improved upon by Mino et al. (2014) who propose an alternative clustering algorithm based on the exchange algorithm (Uszkoreit and Brants 2008), which obtains comparable results, but which runs an order of magnitude faster.

7.3 Soft constraints

Soft syntactic constraints have been around for some time now (Zhou et al. 2008; Venugopal et al. 2009; Chiang 2010; Xiao and Zhu 2013). Zhou et al. (2008) reinforce Hiero with a linguistically-motivated prior. This prior is based on the level of syntactic homogeneity between pairs of non-terminals and the associated syntactic forests rooted at these nonterminals, whereby tree kernels[29] are applied to efficiently measure the amount of overlap between all pairs of sub-trees induced by the pairs of syntactic forests. Crucially, the syntactic prior encourages derivations that are more syntactically coherent but does not block derivations when they are not. Venugopal et al. (2009) associate distributions over compatible syntactic labelings with grammar rules, and combine these preference distributions during decoding, thus achieving a summation rather than competition between compatible label configurations. The latter approach requires significant changes to the decoder and comes at a considerable

[29] Informally, tree kernels are operators that efficiently compute a function $K(T, T')$ of two input tree arguments T and T', e.g. the number of common subtrees. The efficient computation of the function by tree kernels is often achieved by a form of dynamic programming.

computational cost. Soft constraints as proposed by Chiang (2010) and adopted in this paper were discussed earlier in Sect. 2.2 and will not be repeated here. Xiao and Zhu (2013) focus on unsupervised learning of sub-tree alignment based on synchronous tree substitution grammars in combination with the Expectation Maximization (EM) algorithm (Dempster et al. 1977) or a Bayesian learning approach. The translation approach in their work in contrast to ours is based on tree-to-tree translation. It uses syntax on both sides and works with rule sets that even with the labels removed still differ significantly from Hiero. However, in line with our work, this approach also requires elastic-substitution decoding (Chiang 2010) to obtain the best results.

7.4 Learning labels

Improving the quality of the extracted syntactic rules and their labels for syntactic translation with the help of the EM-algorithm is explored by Wang et al. (2010). This work uses re-structuring—binarization of syntactic trees—to make more translation patterns available. It also uses re-labeling to improve the adequacy of syntactic labels and re-aligning to improve word alignments.

Learning labels in a robust way is also explored by Mylonakis and Sima'an (2011). This work uses a special variant of the EM algorithm called Cross-Validated EM to avoid the standard problems of EM with overfitting. The algorithm is then used to learn a distribution of source-labeled hierarchical rules with labels of different levels of specificity. The labels are based on the SAMT labeling approach and also include some basic information about relative orientation with respect to parent rules.

Learning latent-variable SCFGs for hierarchical translation is explored by Saluja et al. (2014). This work uses spectral learning or the EM-algorithm to learn tensors that capture the latent variable information of rules. The tensors are used by means of tensor-vector products,[30] somewhat similar to the way label preferences are propagated in Venugopal et al. (2009). Third-order tensors as opposed to matrices (second-order tensors) are required in the case of binary rules to capture the relation whereby the rule tensor takes the vectors of its two nonterminals as inputs to produce an output vector for the left-hand-side of the rule. Learning of labels is done based on the covariances between sparse feature vectors for inside and outside trees for rules in the training corpus.[31] The work uses minimal rules to avoid the complex problem of simultaneously learning the latent variables and the segmentations of word alignments.

[30] Tensors are multidimensional arrays that generalize vectors and matrices. A third-order tensor T can be imagined as a stack of matrices. When T is combined in a tensor-vector product with two input vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ to produce an output tensor, this corresponds to the following computation: First $\mathbf{v}_1$ is multiplied (on the right) with each of the stacked matrices, producing a single intermediate result matrix M_{int}: $T \cdot \mathbf{v}_1 = M_{int}$. Second, $\mathbf{v}_2$ is multiplied (on the right) with M_{int} to produce the final result vector $\mathbf{v}_{result}$: $M_{int} \cdot \mathbf{v}_2 = \mathbf{v}_{result}$.

[31] The concepts *inside-* and *outside-tree* are defined in terms of another concept called *skeletal tree*. The skeletal tree for an aligned sentence pair is the synchronous tree composed of the set of synchronous rules in the derivation of the aligned sentence pair. Since only minimal rules are used, there is always only one unique derivation. The inside tree for a rule in the training contains the entire sub-tree at and below the left-hand-side nonterminal, and the outside tree is everything else in the synchronous skeletal-tree except the inside-tree.

7.5 Improvement and evaluation of reordering with permutation trees

Stanojević and Sima'an (2015) propose a method for inducing reordering grammars based on permutation trees (PETs) for preordering. Their work uses a modified form of PETs (Gildea et al. 2006) in combination with variable splitting for the permutation labels of PET nodes (Matsuzaki et al. 2005; Prescher 2005; Petrov et al. 2006). The reported results show significant improvements over no preordering, a rule-based preordering baseline (Isozaki et al. 2010) and an ITG-based preordering baseline (Neubig et al. 2012). Usage of all PETs yields better results than working with a single PET in the reported experiments. This work is relevant in the context of ours because it also shows that working with PETs gives significant improvement over using only ITG reordering operators. There are large differences with our work. Our work uses all hierarchical alignment trees (HATs) in combination with bucketing to form labels for elastic-substitution decoding, improving hierarchical translation within the decoder. Stanojević and Sima'an (2015) instead restrict the set of used HATs to only PETs (bijective mappings), and learn the labels. Nevertheless, both contribute evidence to the thesis that word order can be significantly improved without using syntax.

Stanojević and Sima'an (2014) propose a new and highly successful machine translation evaluation method called BEER. This metric uses a multitude of weighted features, with weights that are directly trained to maximize correlation with human ranking. As such, the metric shows very high correlation with human evaluation of translation performance. Training is done for pairwise rankings using learning-to-rank techniques in a way that is similar to PRO MT system tuning (Hopkins and May 2011). Some of the successful new features that are proposed are character n-grams and features based on PETs. The latter features are concerned with reordering and turn out to be an important component in the success of the metric. In the context of this work, the effectiveness of PETs in characterizing the correctness of translation word order as part of the complete evaluation gives yet another reason to believe that the information present in PETs (and more generally, HATs) may be particularly suitable for improving word order in SMT.

8 Conclusion

We presented a novel method to enrich hierarchical statistical machine translation with bilingual labels that help to improve the translation quality. Considerable and significant improvements in the BLEU, METEOR and the Kendall Reordering Score (KRS) are achieved simultaneously for Chinese–English translation while tuning on BLEU, where the KRS is specifically designed to measure improvement of reordering in isolation. Significant improvements in the BLEU score are achieved for German–English. Our work differs from related approaches that use syntactic or part-of-speech information in the formation of reordering constraints in that it needs no such additional information. It also differs from related work on reordering constraints based on lexicalization in that it uses no such lexicalization but instead strives to achieve more globally coherent translations, afforded by global, holistic constraints that take the local reordering history of the derivation directly into account. Our experiments also

once again reinforce the established wisdom that soft, rather than strict constraints, are a necessity when aiming to include new information to an already strong system without the risk of effectively worsening performance through constraints that have not been directly tailored to the data through a proper learning approach. While lexicalized constraints on reordering have proven to have great potential, unlexicalized soft bilingual constraints, which are more general and transcend the rule level, have their own place in providing another agenda of improving translation which focusses more on the global coherence direction by directly putting soft alignment-informed constraints on the combination of rules. Finally, while more research is necessary in this direction, there are strong reasons to believe that in the right setup these different approaches can be made to further reinforce each other.

Acknowledgments This work is supported by Stichting voor de Technische Wetenschappen (STW) grant nr. 12271, and the Netherlands Organization for Scientific Research (NWO) VICI Grant nr. 277-89-002. The authors would like to thank Dr. Wilker Aziz for very helpful feedback on earlier versions of the manuscript. We also thank the anonymous reviewers for their helpful comments.

References

Almaghout H, Jiang J, Way A (2010) CCG augmented hierarchical phrase-based machine translation. In: Federico M, Lane I, Paul M, Yvon F (eds) Proceedings of the seventh international workshop on spoken language translation (IWSLT). France, Paris, pp 211–218

Almaghout H, Jiang J, Way A (2012) Extending CCG-based syntactic constraints in hierarchical phrase-based SMT. In: Proceedings of the annual conference of the European Association for Machine Translation (EAMT)

Birch A, Osborne M, Blunsom P (2010) Metrics for MT evaluation: evaluating reordering. Mach Trans 24(1):15–26

Birch A, Osborne M (2010) LRscore for evaluating lexical and reordering quality in MT. In: Proceedings of the joint fifth workshop on statistical machine translation and metricsMATR. Uppsala, pp 327–332

Chang PC, Galley M, Manning CD (2008) Optimizing chinese word segmentation for machine translation performance. In: Proceedings of the 3rd workshop on statistical machine translation. Columbus, pp 224–232

Chen S, Goodman J (1999) An empirical study of smoothing techniques for language modeling. Comput Speech Lang 4(13):359–393

Cherry C, Foster G (2012) Batch tuning strategies for statistical machine translation. In: NAACL HLT 2012, The 2012 conference of the North American chapter of the association for computational linguistics: human language technologies, proceedings of the conference, Montréal, pp 427–436

Chiang D (2005) A hierarchical phrase-based model for statistical machine translation. In: Proceedings of the 43rd annual meeting of the association for computational linguistics (ACL'05). Michigan, pp 263–270

Chiang D (2006) An introduction to synchronous grammars. Tutorials at the annual meeting of the association for compuational linguistics (ACL), Sydney. http://www3.nd.edu/~dchiang/papers/synchtut.pdf

Chiang D (2010) Learning to translate with source and target syntax. In: ACL 2010, The 48th annual meeting of the association for computational linguistics, conference proceedings, Uppsala, pp 1443–1452

Chiang D (2007) Hierarchical phrase-based translation. Comput Linguist 33(2):201–228

Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the EM algorithm. Je R Stat Soc 39:1–38

Denkowski M, Lavie A, (2011) Meteor 1.3: automatic metric for reliable optimization and evaluation of machine translation systems. In: WMT, (2011) 6th workshop on statistical machine translation. Proceedings of the workshop, Edinburgh pp 85–91

Dixon WJ, Mood AM (1946) The statistical sign test. J Am Stat Assoc 41(236):557–566

Eisele A, Chen Y (2010) MultiUN: a multilingual corpus from United Nation documents. In: Proceedings of the 7th conference on international language resources and evaluation. Valletta, pp 2868–2872

Galley M, Manning CD (2008) A simple and effective hierarchical phrase reordering model. In: Proceedings of the 2008 conference on empirical methods in natural language processing. Honolulu, pp 847–855

Ganitkevitch J, Cao Y, Weese J, Post M, Callison-Burch C (2012) Joshua 4.0: packing, pro, and paraphrases. In: Proceedings of the 7th workshop on statistical machine translation, Montréal, pp 283–291

Gildea D, Satta G, Zhang H (2006) Factoring synchronous grammars by sorting. In: COLING-ACL 2006, 21st international conference on computational linguistics and the 44th annual meeting of the association for computational linguistics, Proceedings of the main conference poster sessions, Sydney, pp 279–286

Hanneman G, Lavie A (2011) Automatic category label coarsening for syntax-based machine translation. In: Proceedings of the 5th workshop on syntax, semantics and structure in statistical translation, Portland, pp 98–106

Hanneman G, Lavie A (2013) Improving syntax-augmented machine translation by coarsening the label set. In: NAACL HLT 2013, The 2013 conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, Atlanta, pp 288–297

He Y, Way A (2009) Metric and reference factors in minimum error rate training. Mach Transl 24(1):27–38

Hopkins M, May J (2011) Tuning as ranking. In: EMNLP 2011, conference on empirical methods in natural language processing, proceedings of the conference, Edinburgh, pp 1352–1362

Huang L, Chiang D (2007) Forest rescoring: faster decoding with integrated language models. In: Proceedings of the 45th annual meeting of the association of computational linguistics. Czech Republic, pp 144–151

Huang L, Knight K, Joshi A (2006) 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. New York City, pp 1–8

Huck M, Wuebker J, Rietig F, Ney H (2013) A phrase orientation model for hierarchical machine translation. In: ACL 2013 8th workshop on statistical machine translation. Sofia, pp 452–463

Isozaki H, Sudoh K, Tsukada H, Duh K (2010) Head finalization: a simple reordering rule for sov languages. In: Proceedings of the joint 5th workshop on statistical machine translation and metricsMATR. Uppsala, pp 244–251

Koehn P (2005) Europarl: a parallel corpus for statistical machine translation. In: Proceedings of MT summit X. Phuket, pp 79–86

Koehn P (2010) Stat Mach Transl, 1st edn. Cambridge University Press, New York

Koehn P, Hoang H, Birch A, Callison-Burch C, Federico M, Bertoldi N, Cowan B, Shen W, Moran C, Zens R, Dyer C, Bojar O, Constantin A, Herbst E (2007) Moses: open source toolkit for statistical machine translation. In: Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions. Czech Republic, pp 177–180

Li J, Tu Z, Zhou G, van Genabith J (2012) Using syntactic head information in hierarchical phrase-based translation. In: Proceedings of the 7th workshop on statistical machine translation. Montréal, pp 232–242

Liu Y, Liu Q, Lin S (2006) Tree-to-string alignment template for statistical machine translation. In: Proceedings of the 21st international conference on computational linguistics and the 44th annual meeting of the association for computational linguistics. Sydney, pp 609–616

Maillette de Buy Wenniger G, Sima'an K (2014a) Bilingual markov reordering labels for hierarchical SMT. In: Proceedings of the 8th workshop on syntax, semantics and structure in statistical translation, Denver, pp 11–21

Maillette de B, Wenniger G, Sima'an K (2014b) Visualization, search and analysis of hierarchical translation equivalence in machine translation data. Prague Bull Math Linguist 101:43–54

Marton Y, Chiang D, Resnik P (2012) Soft syntactic constraints for Arabic–English hierarchical phrase-based translation. Mach Transl 26(1–2):137–157

Matsuzaki T, Miyao Y, Tsujii J (2005) Probabilistic CFG with latent annotations. In: ACL-05, 43rd annual meeting on association for computational linguistics, proceedings of the conference, Ann Arbor, pp 75–82

Mi H, Huang L (2008) Forest-based translation rule extraction. In: Proceedings of the 2008 conference on empirical methods in natural language processing. Honolulu, pp 206–214

Mi H, Huang L, Liu Q (2008) Forest-based translation. In: ACL-08: HLT, 46th annual meeting of the association for computational linguistics: human language technologies, proceedings of the conference, Columbus, pp 192–199

Mino H, Watanabe T, Sumita E (2014) Syntax-augmented machine translation using syntax-label clustering. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). Doha, pp 165–171

Müller G (2002) Free word order, morphological case, and sympathy theory. In: Fanselow G, Fery C (eds) Resolving conflicts in grammars: optimality theory in syntax, morphology, and phonology. BuskeVerlag, pp 265–397

Mylonakis M (2012) Learning the latent structure of translation. PhD thesis, Institute for Logic, Language and Computation, University of Amsterdam, Amsterdam

Mylonakis M, Sima'an K (2011) Learning hierarchical translation structure with linguistic annotations. In: Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies. Portland, pp 642–652

Neubig G, Watanabe T, Mori S (2012) Inducing a discriminative parser to optimize machine translation reordering. In: EMNLP-CoNLL 2012, 2012 joint conference on empirical methods in natural language processing and computational natural language learning, proceedings of the conference, Jeju Island, pp 843–853

Nguyen T, Vogel S (2013) Integrating phrase-based reordering features into a chart-based decoder for machine translation. In: ACL 2013, 51st annual meeting of the association for computational linguistics, proceedings of the conference, vol 1: Long Papers. Sofia, pp 1587–1596

Och F, Ney H (2002) Discriminative training and maximum entropy models for statistical machine translation. In: Proceedings of the 40th annual meeting of the association for computational linguistics, pp 160–167

Och FJ, Ney H (2004) The alignment template approach to statistical machine translation. Comput Linguist 30(4):417–449

Papineni K, Roukos S, Ward T, Zhu WJ (2002) BLEU: A method for automatic evaluation of machine translation. In: Proceedings of the 40th annual meeting on association for computational linguistics. Pennsylvania, pp 311–318

Petrov S, Barrett L, Thibaux R, Klein D (2006) 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. Sydney, pp 433–440

Prescher D (2005) Inducing head-driven PCFGs with latent heads: Refining a tree-bank grammar for parsing. In: Proceedings of the 16th European conference on machine learning, Porto, ECML'05, pp 292–304

Rayner J, Best DJ (1999) Modeling ties in the sign test. Biometrics 2(55):663–665

Saluja A, Dyer C, Cohen SB (2014) Latent-variable synchronous cfgs for hierarchical translation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). Doha, pp 1953–1964

Sima'an K, Maillette de Buy Wenniger G (2013) Hierarchical alignment trees: a recursive factorization of reordering in word alignments with empirical results. Internal Report. http://staff.science.uva.nl/~simaan/D-Papers/HATsReport2013.pdf

Snover M, Dorr B, Schwartz R, Micciulla L, Makhoul J (2006) A study of translation edit rate with targeted human annotation. In: AMTA 2006: Proceedings of the 7th conference of the association for machine translation in the Americas, visions for the future of machine translation, Cambridge, pp 223–231

Stanojević M, Sima'an K (2014) BEER: BEtter evaluation as ranking. In: Proceedings of the 9th workshop on statistical machine translation. Baltimore, pp 414–419

Stanojević M, Sima'an K (2015) Reordering grammar induction. In: Proceedings of the 2015 conference on empirical methods in natural language processing, Lisbon, pp 44–54

Steedman M (2000) The syntactic process. MIT Press, Cambridge

Tiedemann J (2012) Parallel data, tools and interfaces in OPUS. In: Proceedings of the 8th international conference on language resources and evaluation (LREC-2012). Istanbul, pp 2214–2218

Tillmann C (2004) A unigram orientation model for statistical machine translation. In: HLT-NAACL 2004, human language technology conference of the North American Chapter of the association for computational linguistics companion volume: short papers, student research workshop, demonstrations, tutorials abstracts, Boston, pp 101–104

Uszkoreit J, Brants T (2008) Distributed word clustering for large scale class-based language modeling in machine translation. In: ACL-08: HLT, 46th annual meeting of the association for computational linguistics: human language technologies, proceedings of the conference, Columbus, pp 755–762

Venugopal A, Zollmann A, Smith NA, Vogel S (2009) Preference grammars: softening syntactic constraints to improve statistical machine translation. In: NAACL HLT 2009, human language technologies: the 2009 annual conference of the North American Chapter of the association for computational linguistics, proceedings of the conference, Boulder, pp 236–244

Wang W, May J, Knight K, Marcu D (2010) Re-structuring, re-labeling and re-aligning for syntax-based machine translation. Comput Linguist 36:247–277

Wu D (1997) Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. Comput Linguist 23:377–404

Xiao X, Su J, Liu Y, Liu Q, Lin S (2011) An orientation model for hierarchical phrase-based translation. IALP 2011, proceedings of the 2011 international conference on Asian language processing. Penang, pp 165–168

Xiao T, Zhu J (2013) Unsupervised sub-tree alignment for tree-to-tree translation. J Artif Intell Res 48(1):733–782

Zhang H, Gildea D, Chiang D (2008) Extracting synchronous grammar rules from word-level alignments in linear time. In: Coling 2008, 22nd international conference on computational linguistics, proceedings of the conference, Manchester, pp 1081–1088

Zhou B, Xiang B, Zhu X, Gao Y (2008) Prior derivation models for formally syntax-based translation using linguistically syntactic parsing and tree kernels. In: Proceedings of the ACL-08: HLT second workshop on syntax and structure in statistical translation (SSST-2). Columbus, pp 19–27

Zollmann A (2011) Learning multiple-nonterminal synchronous grammars for statistical machine translation. PhD thesis, Carnegie Mellon University, Pittsburgh. http://www.cs.cmu.edu/~zollmann/publications/thesis.pdf

Zollmann A, Venugopal A (2006) Syntax augmented machine translation via chart parsing. In: HLT-NAACL 06, statistical machine translation, proceedings of the workshop, New York City, pp 138–141