PBML

The Prague Bulletin of Mathematical Linguistics
NUMBER 99 APRIL 2013 17–38

Source-Side Discontinuous Phrases for Machine Translation: A Comparative Study on Phrase Extraction and Search

Matthias Huck, Erik Scharwächter, Hermann Ney

Human Language Technology and Pattern Recognition Group, RWTH Aachen University

Abstract

Standard phrase-based statistical machine translation systems generate translations based on an inventory of continuous bilingual phrases. In this work, we extend a phrase-based decoder with the ability to make use of phrases that are *discontinuous* in the source part.

Our dynamic programming beam search algorithm supports separate pruning of coverage hypotheses per cardinality and of lexical hypotheses per coverage, as well as coverage constraints that impose restrictions on the possible reorderings. In addition to investigating these aspects, which are related to the decoding procedure, we also concentrate our attention on the question of how to obtain source-side discontinuous phrases from parallel training data. Two approaches (hierarchical and discontinuous extraction) are presented and compared.

On a large-scale Chinese→English translation task, we conduct a thorough empirical evaluation in order to study a number of system configurations with source-side discontinuous phrases, and to compare them to setups which employ continuous phrases only.

1. Introduction

In standard statistical phrase-based machine translation with continuous phrases (Koehn et al., 2003), lexical translation decisions are based on local context only. Source-side discontinuous phrases, in contrast, can explain lexical dependencies between words that appear in a wider context in the source sentence. For example, the French negation is formed with the particle "ne" followed by a verb and a subsequent negative word like "pas" for "not" or "rien" for "nothing". The lexical decision for correctly translating the negation must be based upon the wider context "ne … pas"

Corresponding author: huck@cs.rwth-aachen.de
Cite as: Matthias Huck, Erik Scharwächter, Hermann Ney. Source-Side Discontinuous Phrases for Machine Translation: A Comparative Study on Phrase Extraction and Search. The Prague Bulletin of Mathematical Linguistics No. 99, 2013, pp. 17–38. doi: 10.2478/pralin-2013-0002.

or "ne … rien". This can be achieved by using discontinuous phrases like ⟨ne ◊ pas, do not⟩ and ⟨ne ◊ rien, nothing⟩, where the ◊ symbol represents a gap.

Statistical machine translation with discontinuous phrases as we define it in this paper has been introduced by Galley and Manning (2010). We present a generalization of the source cardinality synchronous search algorithm as described by Zens and Ney (2008)—with coverage pruning per cardinality and lexical pruning per coverage—which is able to cope with source-side discontinuities. We give a formulation of a discontinuous phrase extraction algorithm and conduct an in-depth analysis of the differences to hierarchical phrase extraction (Chiang, 2005, 2007). On the NIST Chinese→English translation task, we empirically compare a broad range of setups and configuration parameters.

Note that, while Galley and Manning (2010) allow bilingual phrases with discontinuities both in the source and in the target part, we restrict our study to phrases which are allowed to be discontinuous in the source part only, but are required to be continuous on the target side.

Our implementation has been released as part of version 2.2 of Jane (Vilar et al., 2010, 2012; Wuebker et al., 2012), the RWTH Aachen University open source statistical machine translation toolkit.

2. Source cardinality synchronous search using discontinuous phrases

To translate a source sentence f_1^J of length J with the help of discontinuous phrases, known discontinuous source parts are identified in f_1^J and a target sentence e_1^I of length I is generated out of the corresponding target parts. As in the continuous case, the process yields a segmentation of the sentence.

2.1. Generalized segmentation

Let f_1^J be a source sentence, e_1^I be a target sentence. In the continuous phrase-based model, the segmentation of the sentence pair into K phrases is defined as a sequence s_1^K with $s_k = (i_k; b_k, j_k)$ for $k = 1 \ldots K$, where the source phrases $\tilde{f}_k = f_{b_k}^{j_k}$ are continuous. i_k denotes the end of the k^{th} target phrase, b_k denotes the beginning of the k^{th} source phrase and j_k its end. Now, discontinuous source phrases are allowed, so a generalized segmentation $\dot{s}_1^K$ is introduced:

$$k \rightarrow \dot{s}_k := (i_k; \tilde{C}_k), \text{ for } k = 1 \ldots K \tag{1}$$

Still, i_k denotes the end of the k^{th} target phrase, but now the k^{th} source phrase is given by a phrase coverage set $\tilde{C}_k \subseteq \{1, \ldots J\}$. The phrase coverage contains all source positions that are part of the k^{th} source phrase. If $\tilde{C}_k = \{b_k, \ldots j_k\}$ for some $b_k, j_k \in \{1, \ldots J\}$, the source phrase it represents is continuous. In that case, the set $\tilde{C}_k$ is called continuous.

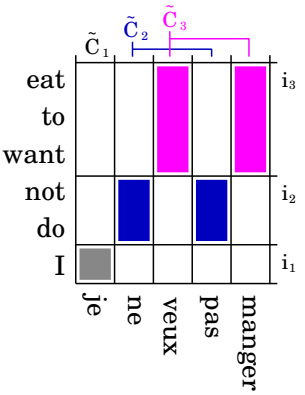

Figure 1. Discontinuous segmentation of a bilingual sentence pair.

Any $\tilde{C}_k$ can be decomposed into a union of N_k maximal continuous subsets

$$\tilde{C}_k = \{b_{k,1}, \ldots j_{k,1}\} \cup \ldots \cup \{b_{k,N_k}, \ldots j_{k,N_k}\} \tag{2}$$

such that

$$\forall\, n:\ 1 \leq n \leq N_k : b_{k,n} \leq j_{k,n} \tag{3}$$

$$\forall\, n:\ 1 \leq n < N_k : j_{k,n} + 1 < b_{k,n+1} \tag{4}$$

Each maximal continuous subset represents a maximal continuous source unit. The combination of these continuous source units gives the complete k^{th} source phrase. The k^{th} target phrase is continuous by definition, as we do not allow for gaps on the target side. This gives us the following notation:

$$\tilde{f}_{k,n} := f_{b_{k,n}}^{j_{k,n}} \qquad \forall\, n:\ 1 \leq n \leq N_k \tag{5}$$

$$\tilde{f}_k := \tilde{f}_{k,1} \Diamond \ \ldots \ \Diamond \tilde{f}_{k,N_k} \tag{6}$$

$$\tilde{e}_k := e_{i_{k-1}+1} \ldots e_{i_k} \qquad (\text{with } i_0 = 0) \tag{7}$$

The segmentation $\dot{s}_k$ describes a partition of the source and target sentence. On the target side, nothing changes from the standard model. It must hold that $i_0 = 0$, $i_K = I$ and $i_{k-1} < i_k$ for $1 \leq k \leq K$. On the source side, constraints on the phrase coverage sets must be imposed:

$$\bigcup_{k=1}^{K} \tilde{C}_k = \{1, \ldots J\} \tag{8}$$

$$\tilde{C}_k \cap \tilde{C}_{k'} = \emptyset \qquad \forall\, k \neq k' \tag{9}$$

Figure 1 shows the segmentation of a French–English sentence pair, where two discontinuous phrases are used: ⟨ne ◊ pas, do not⟩ and ⟨veux ◊ manger, want to eat⟩.

2.2. Discontinuous translation model

With the new segmentation, the maximization is carried out over $\dot{s}_1^K$, and the log-linear feature functions (Och and Ney, 2002) have to be adapted accordingly:

$$\hat{e}_1^{\hat{I}} = \underset{I, e_1^I}{\operatorname{argmax}} \left\{ \max_{K, \dot{s}_1^K} \sum_{m=1}^{M} \lambda_m h_m(e_1^I, \dot{s}_1^K; f_1^J) \right\} \tag{10}$$

All standard features functions for phrase-based machine translation can be utilized in the discontinuous phrase-based model with some minor notational changes. A formal redefinition of these features is omitted here. Instead, two new features are introduced which are unique for discontinuous translation.

The first one is the *gappy flag*, which counts the number of discontinuous phrases used in the segmentation:

$$h_{isGappy}(e_1^I, \dot{s}_1^K; f_1^J) = \sum_{k=1}^{K} [N_k > 1] \tag{11}$$

It can be used to reward or penalize the application of discontinuous phrases, depending on its scaling factor. As before, N_k denotes the number of maximal continuous subsets of $\tilde{C}_k$, and $[\mathcal{C}]$ evaluates to 1, if condition $\mathcal{C}$ is true, and 0 if it is false.

The second one is the *gap size* feature. It counts the number of words in between consecutive maximal continuous source units of discontinuous phrases:

$$h_{GS}(e_1^I, \dot{s}_1^K; f_1^J) = \sum_{k=1}^{K} \sum_{n=1}^{N_k - 1} (b_{k,n+1} - j_{k,n} - 1) \tag{12}$$

The gap size feature differs from the standard translation model features, as it cannot be precomputed and stored in the phrase table. Instead, it must be calculated during decoding, similar to the distortion model.

2.3. Dynamic programming beam search

When moving from continuous phrases to phrases with discontinuous source parts, the search space is only slightly altered. Still, nodes of the search graph represent triples $(C, \tilde{e}', j')$, where C is a coverage set, $\tilde{e}'$ is the language model history and j' the last translated source position. A translation decision now is a tuple $(\tilde{C}_k; \tilde{e}_k)$. It can be used if $C \cap \tilde{C}_k = \emptyset$; the successor state is given by:

$$(C \cup \tilde{C}_k, \tilde{e}' \oplus \tilde{e}_k, \max \tilde{C}_k) \tag{13}$$

```
   INPUT: source sentence f_1^J, maximum source phrase length l̃_max, sorted translation candidates E(·),
          models q_TM(·), q_LM(·), q_DM(·), rest cost estimate R(·)
 1 Q(∅, $, 0) = 0 ; all other Q(·,·,·) entries are initialized to −∞
 2 FOR cardinality c = 1 TO J DO
 3     FOR source phrase length l̃ = 1 TO l̃_max DO
 4         previous cardinality c' = c − l̃
 5         FOR ALL coverages C' ⊂ {1,...J} : |C'| = c' DO
 6             FOR ALL start positions j' ∈ {1,...J} DO
 7                 FOR ALL end positions j ∈ {j' + l̃ − 1,...J}
 8                     FOR ALL phrase coverages C̃ with |C̃| = l̃, min C̃ = j', max C̃ = j
 9                         IF C' ∩ C̃ ≠ ∅ THEN CONTINUE
10                         coverage C = C' ∪ C̃
11                         FOR ALL states ẽ', j'' ∈ Q(C', ·, ·) DO
12                             partial score q = Q(C', ẽ', j'') + q_DM(j'', j')
13                             IF q + R(C, j) + q_TM(C̃) isTooBadForCoverage C THEN
14                                 CONTINUE
15                             FOR ALL phrase translations ẽ'' ∈ E(C̃) DO
16                                 IF q + R(C, j) + q_TM(ẽ'', C̃) isTooBadForCoverage C THEN
17                                     BREAK
18                                 score = q + q_TM(ẽ'', C̃) + q_LM(ẽ''|ẽ')
19                                 IF score +R(C, j) isTooBadForCoverage C THEN
20                                     CONTINUE
21                                 language model state ẽ = ẽ' ⊕ ẽ''
22                                 IF score > Q(C, ẽ, j) THEN
23                                     Q(C, ẽ, j) = score
24                                     B(C, ẽ, j) = (C', ẽ', j'')
25                                     A(C, ẽ, j) = ẽ
26     pruneCardinality c
```

Figure 2. Dynamic programming beam search algorithm with support for source discontinuities.

The search can be carried out using dynamic programming. Let the helper function $Q(C, \tilde{e}, j)$ denote the score of the best path to node $(C, \tilde{e}, j)$ in the search graph. This node can now be reached by translating any source phrase with phrase coverage $\tilde{C} \subseteq C$ in a predecessor node $(C \backslash \tilde{C}, \tilde{e}', j'')$. It must only hold that $\max \tilde{C} = j$ and, for the translation candidate $\tilde{e}''$, $\tilde{e}' \oplus \tilde{e}'' = \tilde{e}$. The new dynamic programming recursion equation is straightforward:

$$Q(\emptyset, \$, 0) = 0 \tag{14}$$

$$Q(C, \tilde{e}, j) = \max_{\substack{\tilde{C}: \tilde{C} \subseteq C \wedge \max \tilde{C} = j \\ j'', \tilde{e}', \tilde{e}'': \tilde{e}' \oplus \tilde{e}'' = \tilde{e}}} \big\{ Q(C \backslash \tilde{C}, \tilde{e}', j'') + q_{TM}(\tilde{e}'', \tilde{C}) + q_{LM}(\tilde{e}''|\tilde{e}') + q_{DM}(j'', \min \tilde{C}) \big\} \tag{15}$$

The translation model, language model, and distortion model helper functions $q_{TM}(\cdot)$, $q_{LM}(\cdot)$, and $q_{DM}(\cdot)$ are defined as the weighted sums of the involved feature func-

tions. The score of the best translation is given by:

$$\hat{Q} = \max_{\tilde{e},j} \left\{ Q(\{1,\ldots J\}, \tilde{e}, j) + q_{LM}(\$|\tilde{e}) + q_{DM}(j, J+1) \right\} \quad (16)$$

The dynamic programming beam search algorithm can be changed to work with discontinuous source phrases. Figure 2 shows the updated algorithm. The main difference to the standard algorithm (Zens, 2008; Zens and Ney, 2008) is in lines 7 to 10. The standard algorithm loops over the source phrase length $\tilde{l}$ and the start position j', and these two parameters determine the end position $j = j' + \tilde{l} - 1$. The continuous source phrase $f_{j'}^{j}$, starts at position j' and ends at position j.

In the new algorithm, the source phrase length $\tilde{l}$ determines the cardinality of the phrase coverage $\tilde{C}$. A discontinuous phrase starting at position j' with cardinality $|\tilde{C}| = \tilde{l}$ does not necessarily end at position $j = j' + \tilde{l} - 1$, but can end at any position $j \geq j' + \tilde{l} - 1$. Therefore, a second loop over the end position is carried out in line 7. In line 8, all phrase coverages with cardinality $\tilde{l}$ starting at position j' and ending at position j are considered. A phrase matching algorithm is executed before the actual search takes place. The phrase matching algorithm finds for all $\tilde{l}$, j', and j the phrase coverages for which translation candidates are available.

3. Phrase extraction

3.1. Standard phrase extraction

In the standard phrase-based approach, only continuous phrases are extracted (Och et al., 1999; Och, 2002). The set of continuous bilingual phrases $\mathcal{BP}(f_1^J, e_1^I, A)$, given a training instance consisting of a source sentence f_1^J, a target sentence e_1^I, and a word alignment $A \subseteq \{1,\ldots,I\} \times \{1,\ldots,J\}$, is defined as follows:

$$\mathcal{BP}(f_1^J, e_1^I, A) = \Big\{ \langle f_{j_1}^{j_2}, e_{i_1}^{i_2} \rangle : \exists (i,j) \in A : i_1 \leq i \leq i_2 \wedge j_1 \leq j \leq j_2 \\ \wedge \forall (i,j) \in A : i_1 \leq i \leq i_2 \leftrightarrow j_1 \leq j \leq j_2 \Big\} \quad (17)$$

Consistency for continuous phrases is based upon two constraints in this definition: (1.) At least one source and target position within the phrase must be aligned, and (2.) words from inside the source phrase may only be aligned to words from inside the target phrase and vice versa.

3.2. Discontinuous phrase extraction

A general discontinuous phrase with source and target discontinuities is expressed as a pair of coverage sets $(\tilde{C}_{src}, \tilde{C}_{tgt})$ with $\tilde{C}_{src} \subseteq \{1,\ldots,J\}$ and $\tilde{C}_{tgt} \subseteq \{1,\ldots,I\}$.

The phrase is consistent with the alignment A if and only if two conditions hold:

$$\exists(i,j) \in A : i \in \tilde{C}_{tgt} \wedge j \in \tilde{C}_{src} \tag{18}$$

$$\forall(i,j) \in A : i \in \tilde{C}_{tgt} \leftrightarrow j \in \tilde{C}_{src} \tag{19}$$

These are exactly the same constraints as for the continuous case, only with a relaxed view on what is considered a phrase.

We now define the set of discontinuous bilingual phrases $\mathcal{D}(f_1^J, e_1^I, A)$ with discontinuous source parts and continuous target parts. Let N be the total number of gaps allowed in the source part of a phrase, and let $\mathcal{D}_n(f_1^J, e_1^I, A)$ denote the sets of discontinuous phrases with exactly n gaps, $n = 0 \ldots N$. The complete set of discontinuous phrases is the union of these smaller sets:

$$\mathcal{D}(f_1^J, e_1^I, A) = \bigcup_{n=0}^{N} \mathcal{D}_n(f_1^J, e_1^I, A) \tag{20}$$

Before moving on to the definition of $\mathcal{D}_n$, the constraint from Equation 18 will be made stronger. For a phrase $\langle f_{j_{1,1}}^{j_{1,2}} \diamond f_{j_{2,1}}^{j_{2,2}}, e_{i_1}^{i_2} \rangle$, it should hold that the two maximal continuous subsequences of the source part, $f_{j_{1,1}}^{j_{1,2}}$ and $f_{j_{2,1}}^{j_{2,2}}$ are both connected to $e_{i_1}^{i_2}$ with a word alignment, i.e. there exists a pair $(i,j) \in A$ with $i_1 \leq i \leq i_2 \wedge j_{1,1} \leq j \leq j_{1,2}$ and a pair $(i,j) \in A$ with $i_1 \leq i \leq i_2 \wedge j_{2,1} \leq j \leq j_{2,2}$. This stronger constraint is also imposed by Galley and Manning (2010). Furthermore, a gap must span over at least one aligned source position, i.e. there exists a pair $(i,j) \in A$ with $j_{1,2} < j < j_{2,1}$. The constraint from Equation 19 will be kept as it is. With these additional constraints in mind, the sets $\mathcal{D}_n$ can be defined in a general way as follows:

$$\begin{aligned}
\mathcal{D}_n(f_1^J, e_1^I, A) = \Big\{ \langle f_{j_{1,1}}^{j_{1,2}} \diamond \ldots \diamond f_{j_{n+1,1}}^{j_{n+1,2}}, e_{i_1}^{i_2} \rangle \;\Big|\; & 1 \leq i_1 \leq i_2 \leq I \\
& \wedge\ \forall k : 1 \leq k \leq n+1 : \Big(1 \leq j_{k,1} \leq j_{k,2} \leq J \\
& \quad \wedge\ \exists(i,j) \in A : i_1 \leq i \leq i_2 \wedge j_{k,1} \leq j \leq j_{k,2} \Big) \\
& \wedge\ \forall k : 1 \leq k \leq n : \Big(\exists(i,j) \in A : j_{k,2} < j < j_{k+1,1} \Big) \\
& \wedge\ \forall(i,j) \in A : \Big(i_1 \leq i \leq i_2 \leftrightarrow \big(\exists k : j_{k,1} \leq j \leq j_{k,2} \big) \Big) \Big\}
\end{aligned} \tag{21}$$

3.2.1. Discontinuous extraction algorithm

The algorithm for extracting all discontinuous phrases can be found in Figure 3. The idea of the algorithm is to build up phrases with n gaps from phrases with $n-1$ gaps. It does so by using helper sets $\mathcal{W}_n$ ($n = 0 \ldots N$) which contain candidate phrases

```
   INPUT: source sentence f_1^I, target sentence e_1^I, alignment A,
          maximum number of source gaps N
1  W_0 = ∅
2  FOR j_1 = 1 TO J DO
3      IF j_1 is unaligned THEN CONTINUE
4      i_1 = ∞; i_2 = −∞
5      FOR j_2 = j_1 TO J DO
6          IF j_2 is unaligned THEN CONTINUE
7          i_1 = min{i_1, min{i|(i, j_2) ∈ A}}
8          i_2 = max{i_2, max{i|(i, j_2) ∈ A}}
9          W_0 := W_0 ∪ {⟨f_{j_1}^{j_2}, e_{i_1}^{i_2}⟩}
10 FOR n = 1 TO N DO
11     W_n = ∅
12     FOR ⟨f_{j_{1,1}}^{j_{1,2}} ◊ ... ◊ f_{j_{n,1}}^{j_{n,2}}, e_{i_1}^{i_2}⟩ IN W_{n−1} DO
13         FOR j_1 = j_{n,2} + 2 TO J DO
14             IF ∄(i, j) ∈ A : j_{n,2} < j < j_1 THEN CONTINUE
15             IF j_1 is unaligned THEN CONTINUE
16             i'_1 = i_1; i'_2 = i_2
17             FOR j_2 = j_1 TO J DO
18                 IF j_2 is unaligned THEN CONTINUE
19                 i'_1 = min{i'_1, min{i|(i, j_2) ∈ A}}
20                 i'_2 = max{i'_2, max{i|(i, j_2) ∈ A}}
21                 W_n = W_n ∪ {⟨f_{j_{1,1}}^{j_{1,2}} ◊ ... ◊ f_{j_{n,1}}^{j_{n,2}} ◊ f_{j_1}^{j_2}, e_{i'_1}^{i'_2}⟩}
22 FOR n = 0 TO N DO
23     FOR phrase r IN W_n DO
24         IF check-consistency(r) THEN D_n = D_n ∪ {r}
```

Figure 3. Discontinuous phrase extraction algorithm.

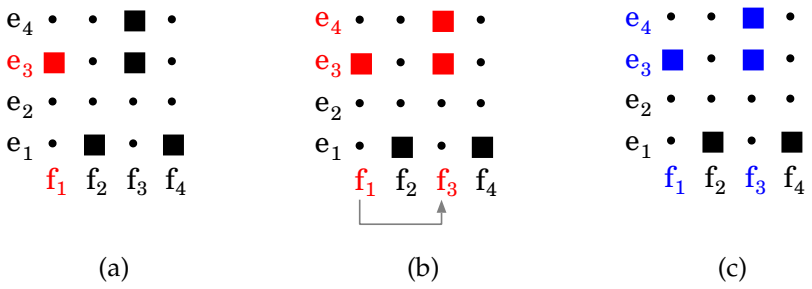

Figure 4. Visualization of discontinuous phrase extraction. Starting from the inconsistent phrase $\langle f_1, e_3 \rangle$ from $\mathcal{W}_0$ (red in (a)), the algorithm skips one aligned position and reads another continuous source sequence. The result is first stored in $\mathcal{W}_1$ (red in (b)) and then, since it is a consistent discontinuous phrase, in $\mathcal{D}_1$ (blue in (c)).

that do not necessarily fulfill all consistency constraints. In the end, from these helper sets only the phrases that fulfill all constraints are extracted to $\mathcal{D}_n$.

The first part of the algorithm, lines 2–9, is identical to the standard phrase extraction algorithm. All continuous source sequences from the source sentence are enumerated and the aligned continuous target parts collected, but without checking the consistency constraints. All these phrases are stored in $\mathcal{W}_0$. In lines 10–21, each candidate phrase with $n-1$ gaps is extended to a phrase with n gaps by skipping at least one aligned position (lines 13 and 15), finding a new continuous source sequence (line 17), and collecting the newly covered target positions (lines 19 and 20). Finally, for all phrase candidates the consistency constraints are checked and the valid phrases are added to the discontinuous phrase set (lines 22–24). Figure 4 visualizes this process.

This algorithm is inspired by the one presented by Lopez (2007) for hierarchical phrase extraction using suffix arrays, which itself is based upon the pattern matching algorithm for variable length gaps by Rahman et al. (2006).

3.3. Hierarchical phrase extraction

Hierarchical phrases (Chiang, 2005, 2007; Vilar, 2011) are essentially special discontinuous phrases where gaps are denoted by the non-terminals. The crucial difference is that non-terminals on the source side and on the target side of hierarchical rules are linked with a one-to-one relation. Typically, a single generic non-terminal symbol X is used as a placeholder for the gaps within the right-hand side of hierarchical translation rules as well as on all left-hand sides of the translation rules that are extracted from the training corpus.

3.3.1. Hierarchical rules for the discontinuous search algorithm

Interpreting hierarchical rules as discontinuous phrases is straight-forward. From a given hierarchical rule

$$X \rightarrow \langle \alpha X^{\sim 1} \beta X^{\sim 2} \gamma, \delta X^{\sim 1} \epsilon X^{\sim 2} \zeta \rangle \tag{22}$$

with $\alpha, \beta, \gamma \in \mathcal{F}^+$ and $\delta, \epsilon, \zeta \in \mathcal{E}^+$, where $\mathcal{F}$ denotes the source vocabulary and $\mathcal{E}$ the target vocabulary, the left-hand side is discarded and all non-terminals are replaced by gap symbols:

$$\langle \alpha \diamondsuit \beta \diamondsuit \gamma, \delta \diamondsuit \epsilon \diamondsuit \zeta \rangle \tag{23}$$

Hierarchical rules naturally have gaps in their target parts. When using hierarchical extraction to obtain a phrase inventory for application in our discontinuous search procedure, discontinuous target parts must be discarded, and non-terminals at the phrase boundary be removed. This can either be done as a post-processing step, enforcing a renormalization of the phrase probabilities, or it can directly be integrated into the extraction algorithm.

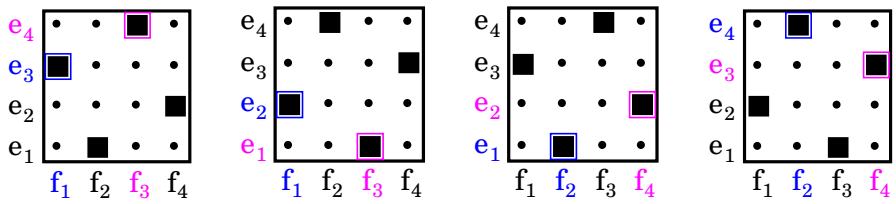

Figure 5. Tricky phrase reorderings. The hierarchical extractor can only generate the phrases $\langle f_2 \Diamond f_4, e_1 e_2 \rangle$ and $\langle f_2 \Diamond f_4, e_3 e_4 \rangle$ from the first two examples by considering the larger phrase $\langle f_1 f_2 f_3 f_4, e_1 e_2 e_3 e_4 \rangle$ and first cutting out f_1, then f_3. The same holds for the last two examples and the phrases $\langle f_1 \Diamond f_3, e_3 e_4 \rangle$ and $\langle f_1 \Diamond f_3, e_1 e_2 \rangle$ with words f_2 and f_4 respectively.

Some properties of hierarchical rules should be considered before using them with the discontinuous search algorithm. For a hierarchical translation system, the two rules $X \rightarrow \langle \alpha, \beta \rangle$ and $X \rightarrow \langle \alpha X^{\sim 1}, \beta X^{\sim 1} \rangle$ are different. For a discontinuous system, both represent the same phrase pair $\langle \alpha, \beta \rangle$. It is tempting to automatically discard all hierarchical rules with non-terminals at the boundaries of the source part to avoid counting the same phrase pair multiple times. In fact, when preparing the hierarchical rule set for discontinuous translation, most of these rules must be discarded. However, there are cases where these phrases are needed because they enable extracting rules which otherwise could not be extracted.

When extracting with at most two non-terminal symbols, there are two alignment configurations that enforce keeping a source part with a non-terminal at its boundary, because there is no other way to extract the resulting phrase pair. Figure 5 shows these two configurations. Wu (1997) characterized them as inside-out reorderings, because they involve a phrase moving from inside the source part to the boundary of the target part and vice versa. Table 1 shows all hierarchical source and target parts that can be used in our source cardinality synchronous discontinuous search algorithm, when extracting with at most two non-terminals (or gaps) per phrase. To avoid confusion, we use the term *hierarchical phrase* only for those phrases, and the term *discontinuous phrase* only for phrases extracted with discontinuous phrase extraction.

The set of hierarchical phrases with up to N non-terminal symbols per rule is a subset of the set of discontinuous phrases with up to N gaps. The difference in the phrase tables from the discontinuous and the hierarchical extraction is analyzed in Section 4.1.

4. Empirical evaluation

We present an empirical evaluation on the NIST Chinese→English translation task.[1] We work with a parallel training corpus of 3.0 M Chinese–English sentences pairs (77.5 M Chinese / 81.0 M English running words). Word alignments are calculated with GIZA++[2] in both directions with four IBM model 1 iterations, five HMM iterations and four IBM model 4 iterations (Brown et al., 1993; Vogel et al., 1996; Och and Ney, 2003). The two directions are combined using the refined heuristic by Och and Ney (2003) to obtain a symmetrized alignment.

4.1. Phrase coverage

We first compare the phrase tables that result from the hierarchical and from the discontinuous approach to phrase extraction.

A single-word extraction heuristic, forced single-word extraction heuristic, and an extraction heuristic for unaligned words as described by Stein et al. (2011) are active in all approaches. Phrases are restricted to a maximum source and target length of 10 words (including gap symbols) with at most two gaps in the source part. For the discontinuous phrase extractor, a maximum gap size of 10 words is used, meaning that the extractor may skip at most 10 words to introduce a new gap.

Let the *span* of a source phrase $\tilde{f}$ in a training sentence f_1^J be the distance between its first word and its last word. For a continuous source phrase $\tilde{f} = f_{j_1}^{j_2}$ the span is given by $j_2 - j_1 + 1$. For a discontinuous phrase with n gaps $\tilde{f} = f_{j_{1,1}}^{j_{1,2}} \Diamond \ldots \Diamond f_{j_{n+1,1}}^{j_{n+1,2}}$ the span is given by $j_{n+1,2} - j_{1,1} + 1$. For hierarchical phrase extraction, a restriction of the maximum source phrase length is also a restriction for the span of the source phrases. A hierarchical phrase is generated by taking a standard phrase and cutting out another standard phrase that is contained in the first one. When the initial standard phrase has a maximum source length of 10, there is no way to generate a hierarchical phrase with a span larger than 10. To extract more discontinuous phrases, this constraint was not imposed in the discontinuous phrase extractor. A discontinuous phrase may span over more than 10 words as long as it consists of at most 10 of them and as long as in each gap at most 10 words are skipped.

After extraction, the phrase tables are filtered towards a larger collection of test sets, i.e. phrases that are not applicable for the translation of any input sentence from one of the test sets are removed from the phrase table. Table 2 shows the number of phrase pairs in the filtered tables extracted with the standard, hierarchical and discontinuous approach. By definition, the hierarchical and discontinuous approach extract all standard phrases from the standard approach.

[1] http://www.itl.nist.gov/iad/mig/tests/mt/

[2] http://code.google.com/p/giza-pp/

source part	allowed target parts	resulting phrase
α	β	$\langle \alpha, \beta \rangle$
$\alpha X^{\sim 1} \beta$	$X^{\sim 1} \gamma$ $\gamma X^{\sim 1}$	$\langle \alpha \lozenge \beta, \gamma \rangle$
$\alpha X^{\sim 1} \beta X^{\sim 2} \gamma$	$X^{\sim 1} \delta X^{\sim 2}$ $X^{\sim 2} \delta X^{\sim 1}$ $X^{\sim 1} X^{\sim 2} \delta$ $X^{\sim 2} X^{\sim 1} \delta$ $\delta X^{\sim 1} X^{\sim 2}$ $\delta X^{\sim 2} X^{\sim 1}$	$\langle \alpha \lozenge \beta \lozenge \gamma, \delta \rangle$
$X^{\sim 1} \alpha X^{\sim 2} \beta$	$\gamma X^{\sim 1} X^{\sim 2}$ $X^{\sim 2} X^{\sim 1} \gamma$	$\langle \alpha \lozenge \beta, \gamma \rangle$
$\alpha X^{\sim 1} \beta X^{\sim 2}$	$X^{\sim 1} X^{\sim 2} \gamma$ $\gamma X^{\sim 2} X^{\sim 1}$	$\langle \alpha \lozenge \beta, \gamma \rangle$

Table 1. Hierarchical phrases for the discontinuous model. The overview is complete if no more than two non-terminals are allowed. The last four rows represent the special cases from Figure 5.

approach	total	standard	gappy	% gappy
standard	34.0 M	34.0 M	0	0
hierarchical	48.0 M	34.0 M	14.0 M	29
discontinuous	85.4 M	34.0 M	51.4 M	60
discontinuous*	53.0 M	34.0 M	19.0 M	36

Table 2. Chinese–English phrase table statistics. In the row marked with an asterisk (*), the maximum span of the discontinuous phrases is limited to 10 for hierarchical compatibility.

	absolute	relative
different length constraints	32.4 M	87 %
gaps over non-standard phrases	2.7 M	7 %
obstacle alignment dots	2.3 M	6 %
total additional	37.4 M	100 %

Table 3. Reasons for additional discontinuous phrases.

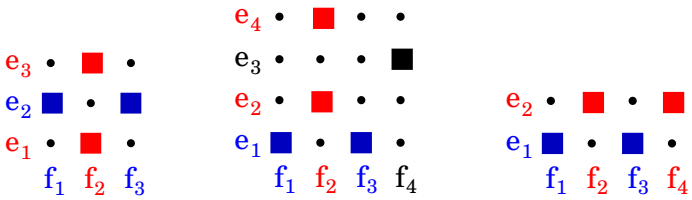

Figure 6. Gaps over non-standard phrases. Out of these three training examples, the hierarchical phrase extractor cannot extract the blue colored phrases, because the red colored initial phrases are non-standard.

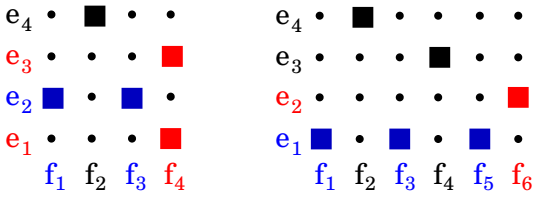

Figure 7. Obstacle alignment dots. In both examples, the gaps span over standard phrases, but the hierarchical extractor cannot extract the blue colored phrases due to the red colored alignment dots.

An analysis of the 37.4 M additional discontinuous phrases has revealed three classes of phrases that are extracted with the discontinuous approach, but not with the hierarchical approach in this setting. These classes can be characterized by *different length constraints*, *gaps over non-standard phrases* and *obstacle alignment dots*. The different length constraints were already mentioned above. A description of the other classes follows in the remainder of this section. Table 3 shows how the additional discontinuous phrases are distributed over the three classes. In fact, 32.4 M of the 37.4 M additional phrases result from the different length constraints. From the remaining 5.0 M additional discontinuous phrases that obey the hierarchical length constraints, 2.7 M have gaps over non-standard phrases and 2.3 M have obstacle alignment dots.

4.1.1. Gaps spanning over non-standard phrases

Some discontinuous phrases cannot be extracted with hierarchical phrase extraction because they include gaps over non-standard phrases. These discontinuous phrases cannot result from cutting out a standard phrase from another standard phrase, regardless of the chosen extraction parameters. Figure 6 depicts three alignments

with gaps over non-standard phrases, where the hierarchical extractor cannot extract a phrase with the source part $f_1 \Diamond f_3$. In the first two examples, the gap comprises a phrase with a discontinuous target part. In the third example, there are two discontinuous phrases in a cross-serial configuration (Søgaard and Kuhn, 2009).

4.1.2. Obstacle alignment dots

The third class consists of all discontinuous phrases that do not belong to the first two classes, i.e. they span over at most 10 words and have no gaps over non-standard phrases. Some of these phrases can be extracted with hierarchical phrase extraction if different extraction parameters are chosen. Some are discontinuous phrases with gaps over standard phrases which are not hierarchical for other reasons. All phrases of the third class are extracted from alignments with obstacle alignment dots at some position. These obstacle dots either prevent the phrase from being extracted with the hierarchical approach due to the chosen extraction parameters or prevent them from being hierarchical at all. See Figure 7 for two examples.

The distinction between phrases with gaps over non-standard phrases and phrases with obstacle alignment dots is not very strict. In the first example from Figure 7 the gap consists of part of the discontinuous phrase $\langle f_2 \Diamond f_4, e_1 \Diamond e_3 e_4 \rangle$, while in the second example the second gap could result from cutting out the discontinuous phrase $\langle f_4 \Diamond f_6, e_2 e_3 \rangle$. However, the *smallest* phrases both gaps comprise are standard phrases.

4.2. Translation quality

4.2.1. Experimental setup

In our translation setups, we use the following features (apart from the ones that have been or will be explicitly mentioned): phrase translation probabilities, lexical translation probabilities from IBM model 1 (Brown et al., 1993) and discriminative word lexicon models (Mauser et al., 2009) in the manner of Huck et al. (2011), each for both translation directions, length penalties on word and phrase level, source-to-target and target-to-source phrase length ratios, insertion models (Huck and Ney, 2012), four binary features marking phrases that have been seen more than one, two, three or five times, respectively, a distance-based distortion penalty, and an n-gram language model. The language model is a 4-gram with modified Kneser-Ney smoothing (Kneser and Ney, 1995) which was trained with the SRILM toolkit (Stolcke, 2002) on a large collection of English data including the target side of the parallel corpus. Phrase tables have not been prepruned to contain a maximum number of translation candidates per source side, but the decoder is configured to load at most the 200 best candidates with respect to the weighted phrase-level model scores. We do not impose any hard restriction on the jump width, but the distance-based distortion cost is linear up to a certain limit and quadratic beyond that. The soft jump distance limit is

set to 10 in the experiments. Our decoder computes a rest score estimate for the language model, translation model, and distortion model (Zens and Ney, 2008; Moore and Quirk, 2007). Model weights are optimized against BLEU (Papineni et al., 2002) with Minimum Error Rate Training (MERT) (Och, 2003), performance is measured in truecase with BLEU and TER (Snover et al., 2006). We employ MT06 as development set, MT08 is used as held-out test set.

In the experiments with source-side discontinuous phrases, we depart from the description as given in Section 2 in one aspect: After the application of a discontinuous phrase with coverage vector $\tilde{C} = \{b_1, \ldots, j_1\} \cup \ldots \cup \{b_N, \ldots, j_N\}$ we set the last translated position to j_1, i.e. the maximum of the leftmost maximal continuous subset of the coverage vector, not to $\max \tilde{C}$. The modifications to the notation of Section 2 as well as to the beam search algorithm from Figure 2 are straightforward, and we will omit them here. We extensively compared both variants and found that they are performing at the same level in terms of translation quality. The reason why we decided in favor of this variant is that it may encourage the usage of discontinuous phrases to a larger extent due to a reduced distortion cost (and likewise a reduced distortion rest cost estimate for partial hypotheses).

4.2.2. Experimental results

In a first series of experiments, different reordering constraints are evaluated. Results can be found in Table 4. The last column indicates the *gappy usage* (GU), i.e. the amount of sentence translations in the respective hypothesis for the test set which use at least one discontinuous phrase. Starting with monotonic decoding[3] in the first row, more and more non-monotonicity is allowed in the following rows. First, a soft jump distance limit of 10 is used with phrase-level IBM reordering constraints (Zens et al., 2004) set to 2 (thus allowing only one gap at a time in each coverage set). Then, the number of allowed uncovered blocks according to the reordering constraints is increased step by step. The histogram size for reordering pruning (RH) is set to 64, for lexical pruning (LH) it is also set to 64 for these experiments.

In a second series of experiments, the pruning settings are analyzed. Using the phrase-level IBM reordering constraint with a maximum of 4 runs (thus allowing up to three gaps at a time in each coverage set), different combinations of reordering histogram size and lexical histogram size are tested. We keep the same scaling factors in the log-linear model combinations for all pruning settings. These optimized model weights have been obtained by running MERT on configurations (with and without discontinuous phrases, respectively) with a relatively large search space (RH=64, LH=64).

[3] In pseudo-monotonic decoding with discontinuous phrases, a new phrase must always start at the leftmost uncovered position in the coverage set. Any explicit jumps are not permitted, and we do not compute distance-based distortion costs. Discontinuous phrases may introduce gaps, though.

IBM reordering constraints	gaps	MT06 (dev) BLEU [%]	MT06 (dev) TER [%]	MT08 (test) BLEU [%]	MT08 (test) TER [%]	GU
monotonic	no	30.3	62.1	24.7	66.1	–
	yes	31.6	61.3	25.4	65.7	38.8 %
2	no	32.7	61.0	25.8	66.2	–
	yes	32.9	60.9	25.8	66.1	26.2 %
3	no	32.7	61.1	26.1	65.8	–
	yes	32.8	61.1	25.9	66.0	25.4 %
4	no	32.6	61.1	26.1	65.8	–
	yes	32.8	61.0	26.1	65.7	26.2 %
5	no	32.5	61.0	26.1	65.4	–
	yes	32.7	60.9	25.8	65.7	25.7 %

Table 4. Effect of reordering constraints (with LH=64, RH=64). The experiments have been carried out with a soft jump distance limit of 10. Results are reported in truecase.

Table 5 shows the results. As the gappy usage indicates, a considerable amount of discontinuous phrases is applied for the generation of the single-best hypotheses for all combinations of pruning parameters. However, clear advantages over the setups without discontinuous phrases become evident with very restrictive pruning settings only (RH=4 or LH=4).

We next examine the impact of the gappy flag and the gap size feature (with RH=64 and LH=64). Both features can be used by the decoder to either penalize or reward the use of discontinuous phrases. Table 6 shows that the decoder uses discontinuous phrases most if these two features are not present (last row). In this case, there is no way to distinguish discontinuous from continuous phrases, and the translation quality drops by 1.1 %BLEU on the development set and 1.6 %BLEU on the test set. The features seem to be required to penalize the application of discontinuous phrases.

Finally, the effect of the different phrase inventories is analyzed. The results are presented in Table 7. With the hierarchical phrase table, the number of sentence translations that use phrases with gaps is quite low compared to the discontinuous phrase table.[4] With the discontinuous phrase table, no improvement is achieved by adding a binary feature which enables the system to distinguish those gappy entries which are also extracted with the hierarchical approach.

[4]We would like to emphasize that *hierarchical* in Table 7 denotes the utilization of a phrase inventory with source-side gaps that has been produced with the hierarchical extractor. The search is conducted with the source cardinality synchronous search algorithm from Figure 2. No synchronous context free grammar (SCFG) formalism is pursued. See (Huck et al., 2012) for results on the same data with the SCFG hierarchical pipeline and a parsing-based cube pruning decoder.

pruning RH	pruning LH	gaps	MT06 (dev) Bleu [%]	MT06 (dev) Ter [%]	MT08 (test) Bleu [%]	MT08 (test) Ter [%]	GU
4	4	no	31.2	61.8	25.2	66.1	–
		yes	31.7	61.2	25.7	65.7	25.9 %
4	16	no	31.7	61.5	25.5	66.0	–
		yes	32.1	60.8	25.8	65.8	24.8 %
4	64	no	31.8	61.4	25.5	66.1	–
		yes	32.2	60.9	25.7	65.8	24.9 %
4	128	no	31.9	61.4	25.4	66.1	–
		yes	32.2	61.0	25.8	65.9	25.3 %
16	4	no	31.7	61.5	25.3	66.0	–
		yes	32.0	61.1	25.9	65.8	28.4 %
16	16	no	32.4	61.1	25.9	65.9	–
		yes	32.5	60.9	26.0	65.5	26.3 %
16	64	no	32.7	61.1	26.1	65.8	–
		yes	32.6	60.9	26.0	65.7	25.4 %
16	128	no	32.6	61.1	26.0	65.9	–
		yes	32.6	61.0	26.0	65.7	25.2 %
64	4	no	31.8	61.5	25.4	66.0	–
		yes	32.1	61.1	25.7	65.9	28.2 %
64	16	no	32.4	61.2	25.9	65.8	–
		yes	32.6	61.0	26.0	65.7	26.4 %
64	64	no	32.6	61.1	26.1	65.8	–
		yes	32.8	61.0	26.1	65.7	26.2 %
64	128	no	32.5	61.1	26.1	65.8	–
		yes	32.7	61.0	26.1	65.7	26.3 %
128	4	no	31.8	61.5	25.4	66.0	–
		yes	32.0	61.2	25.7	65.8	28.4 %
128	16	no	32.4	61.2	25.9	65.8	–
		yes	32.6	61.0	26.1	65.7	26.5 %
128	64	no	32.6	61.2	26.0	65.9	–
		yes	32.7	61.0	26.1	65.7	25.9 %

Table 5. Effect of pruning parameters. Results are reported in truecase.

4.2.3. Discussion

The discontinuous model does not yield significant improvements over the continuous baseline model. Indeed, both models perform on a similar level in almost all directly comparable system configurations. Galley and Manning (2010), in contrast,

features			MT06 (dev)		MT08 (test)		
isGappy	gapSize	gaps	BLEU [%]	TER [%]	BLEU [%]	TER [%]	GU
–	–	no	32.6	61.1	26.1	65.8	–
yes	yes	yes	32.8	61.0	26.1	65.7	26.2 %
yes	no	yes	32.7	60.9	25.9	65.6	23.9 %
no	yes	yes	32.6	61.2	25.5	65.8	25.9 %
no	no	yes	31.7	62.1	24.5	66.9	69.2 %

Table 6. Effect of gappy features (with LH=64, RH=64). Results are reported in truecase.

		MT06 (dev)		MT08 (test)		
phrase table	gaps	BLEU [%]	TER [%]	BLEU [%]	TER [%]	GU
standard	no	32.6	61.1	26.1	65.8	–
hierarchical	yes	32.6	61.1	25.8	65.9	6.1 %
discontinuous	yes	32.8	61.0	26.1	65.7	26.2 %
discontinuous^{+HF}	yes	32.8	60.9	25.9	65.6	23.4 %

Table 7. Effect of the phrase inventory (with LH=64, RH=64). In the experiment marked with ($^{+HF}$), a binary feature has been added which enables the system to distinguish those gappy entries of the discontinuous phrase table which are also extracted with the hierarchical approach. Results are reported in truecase.

have reported uncased gains of +0.6 %BLEU on MT06 and +0.4 %BLEU on MT08 with source-side gaps (and without lexicalized reordering) in their system.[5]

First, we need to discuss whether the differences of the search organization in our decoder as compared to the system by Galley and Manning (2010) may be harmful. Galley and Manning (2010) do not prune reordering hypotheses and lexical hypotheses separately, and their decoder does not impose any reordering constraints in the manner of our phrase-level IBM reordering constraints. Apart from that, their pruning and maximum jump distance settings are rather more restrictive than those we utilized in our setups. Zens and Ney (2003) found that IBM constraints are quite limiting, e.g. as compared to ITG constraints. Regardless of that, reordering constraints and separate pruning of reordering and lexical hypotheses typically guide towards promising translations in an early stage of the search process. At least we would have

[5]The uncased BLEU scores of the system with *standard* phrase table from Table 7 are 34.7 on MT06 and 27.6 on MT08, the uncased BLEU scores of the system with *discontinuous* phrase table are 34.6 on MT06 and 27.6 on MT08. We furthermore ran these systems on MT02, MT04, and MT05, but did not observe any larger gains on any of these alternative test sets.

expected to see an advantage with discontinuous phrases over setups with standard phrases only as we increase the permissible amount of reordering. This is however not the case.

Another aspect we should consider is the quality of the word alignment and its suitability for the discontinuous translation model. We trained our word alignment with four IBM model 1, five HMM and four IBM model 4 iterations, Galley and Manning (2010) theirs with two IBM model 1 and two HMM iterations. We symmetrized our word alignment with the refined heuristic by Och and Ney (2003), which is comparable to the widely-used `grow-diag-final-and` heuristic (Koehn et al., 2003). It usually performs very well for standard phrase-based systems in our experience. Galley and Manning (2010) employ `grow-diag-final` for their hierarchical setup and `grow-diag` for the standard baseline and the discontinuous setup. It is possible that our standard heuristic works well for standard phrase-based translation, while discontinuous phrase-based translation might come up to its best performance based on different properties of the word alignment. We will try to empirically verify this supposition in future work.

5. Conclusion

In this work, a dynamic programming beam search algorithm for phrase-based statistical machine translation with coverage pruning per cardinality and lexical pruning per coverage (Zens and Ney, 2008) has been extended to support phrases with a discontinuous source part similar to (Galley and Manning, 2010). Two approaches to extract phrases with source parts that are allowed to contain gaps have been presented: the hierarchical approach and the discontinuous approach. The hierarchical phrase table is in fact a subset of the discontinuous phrase table. The differences have been discussed and analyzed empirically.

The experimental evaluation on the NIST Chinese→English translation task has been conducted with a focus on reordering constraints, pruning settings, and feature functions, as well as on the different phrase inventories. We found that the setups which employ source-side discontinuous phrases unfortunately barely outperform comparable setups which employ continuous phrases only. The translation quality as measured in BLEU remains at the same level. In future work, we intend to examine a possible impact of word alignment symmetrization heuristics.

Our implementations of the algorithms which we described in this paper have been released as part of Jane, the RWTH Aachen University statistical machine translation toolkit. The Jane toolkit is publicly available under an open source non-commercial license and can be downloaded from `http://www.hltpr.rwth-aachen.de/jane/`.

Acknowledgments

This work was partly achieved as part of the Quaero Programme, funded by OSEO, French State agency for innovation.

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n^o 287658.

Bibliography

Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, 19(2):263–311, June 1993.

Chiang, David. A Hierarchical Phrase-Based Model for Statistical Machine Translation. In *Proc. of the Annual Meeting of the Assoc. for Computational Linguistics (ACL)*, pages 263–270, Ann Arbor, MI, USA, June 2005.

Chiang, David. Hierarchical Phrase-Based Translation. *Computational Linguistics*, 33(2): 201–228, June 2007.

Galley, Michel and Christopher D. Manning. Accurate Non-Hierarchical Phrase-Based Translation. In *Proc. of the Human Language Technology Conf. / North American Chapter of the Assoc. for Computational Linguistics (HLT-NAACL)*, pages 966–974, Los Angeles, CA, USA, June 2010.

Huck, Matthias and Hermann Ney. Insertion and Deletion Models for Statistical Machine Translation. In *Proc. of the Human Language Technology Conf. / North American Chapter of the Assoc. for Computational Linguistics (HLT-NAACL)*, pages 347–351, Montréal, Canada, June 2012.

Huck, Matthias, Saab Mansour, Simon Wiesler, and Hermann Ney. Lexicon Models for Hierarchical Phrase-Based Machine Translation. In *Proc. of the Int. Workshop on Spoken Language Translation (IWSLT)*, pages 191–198, San Francisco, CA, USA, Dec. 2011.

Huck, Matthias, Jan-Thorsten Peter, Markus Freitag, Stephan Peitz, and Hermann Ney. Hierarchical Phrase-Based Translation with Jane 2. *The Prague Bulletin of Mathematical Linguistics*, (98):37–50, Oct. 2012.

Kneser, Reinhard and Hermann Ney. Improved Backing-Off for M-gram Language Modelling. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 181–184, Detroit, MI, USA, May 1995.

Koehn, Philipp, Franz Joseph Och, and Daniel Marcu. Statistical Phrase-Based Translation. In *Proc. of the Human Language Technology Conf. / North American Chapter of the Assoc. for Computational Linguistics (HLT-NAACL)*, pages 127–133, Edmonton, Canada, May/June 2003.

Lopez, Adam. Hierarchical Phrase-Based Translation with Suffix Arrays. In *Proc. of the Joint Conf. on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 976–985, Prague, Czech Republic, June 2007.

Mauser, Arne, Saša Hasan, and Hermann Ney. Extending Statistical Machine Translation with Discriminative and Trigger-Based Lexicon Models. In *Proc. of the Conf. on Empirical Methods for Natural Language Processing (EMNLP)*, pages 210–217, Singapore, Aug. 2009.

Moore, Robert C. and Chris Quirk. Faster Beam-Search Decoding for Phrasal Statistical Machine Translation. In *Proc. of MT Summit XI*, Copenhagen, Denmark, Sept. 2007.

Och, Franz Josef. *Statistical Machine Translation: From Single-Word Models to Alignment Templates*. PhD thesis, RWTH Aachen University, Aachen, Germany, Oct. 2002.

Och, Franz Josef. Minimum Error Rate Training for Statistical Machine Translation. In *Proc. of the Annual Meeting of the Assoc. for Computational Linguistics (ACL)*, pages 160–167, Sapporo, Japan, July 2003.

Och, Franz Josef and Hermann Ney. Discriminative Training and Maximum Entropy Models for Statistical Machine Translation. In *Proc. of the Annual Meeting of the Assoc. for Computational Linguistics (ACL)*, pages 295–302, Philadelphia, PA, USA, July 2002.

Och, Franz Josef and Hermann Ney. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–51, Mar. 2003.

Och, Franz Josef, Christoph Tillmann, and Hermann Ney. Improved Alignment Models for Statistical Machine Translation. In *Proc. of the Joint SIGDAT Conf. on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 20–28, University of Maryland, College Park, MD, USA, June 1999.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proc. of the Annual Meeting of the Assoc. for Computational Linguistics (ACL)*, pages 311–318, Philadelphia, PA, USA, July 2002.

Rahman, Mohammad Sohel, Costas S. Iliopoulos, Inbok Lee, Manal Mohamed, and William F. Smyth. Finding Patterns with Variable Length gaps or Don't Cares. In *Proc. of the International Computing and Combinatorics Conf. (COCOON)*, pages 146–155, Aug. 2006.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. A Study of Translation Edit Rate with Targeted Human Annotation. In *Proc. of the Conf. of the Assoc. for Machine Translation in the Americas (AMTA)*, pages 223–231, Cambridge, MA, USA, Aug. 2006.

Søgaard, Anders and Jonas Kuhn. Empirical Lower Bounds on Alignment Error Rates in Syntax-Based Machine Translation. In *Proc. of the Third Workshop on Syntax and Structure in Statistical Translation (SSST)*, pages 19–27, Boulder, CO, USA, June 2009.

Stein, Daniel, David Vilar, Stephan Peitz, Markus Freitag, Matthias Huck, and Hermann Ney. A Guide to Jane, an Open Source Hierarchical Translation Toolkit. *The Prague Bulletin of Mathematical Linguistics*, (95):5–18, Apr. 2011.

Stolcke, Andreas. SRILM – an Extensible Language Modeling Toolkit. In *Proc. of the Int. Conf. on Spoken Language Processing (ICSLP)*, Denver, CO, USA, Sept. 2002.

Vilar, David. *Investigations on Hierarchical Phrase-Based Machine Translation*. PhD thesis, RWTH Aachen University, Aachen, Germany, Nov. 2011.

Vilar, David, Daniel Stein, Matthias Huck, and Hermann Ney. Jane: Open Source Hierarchical Translation, Extended with Reordering and Lexicon Models. In *Proc. of the Workshop on Statistical Machine Translation (WMT)*, pages 262–270, Uppsala, Sweden, July 2010.

Vilar, David, Daniel Stein, Matthias Huck, and Hermann Ney. Jane: an Advanced Freely Available Hierarchical Machine Translation Toolkit. *Machine Translation*, 26(3):197–216, Sept. 2012.

Vogel, Stephan., Hermann Ney, and Christoph Tillmann. HMM-Based Word Alignment in Statistical Translation. In *Proc. of the Int. Conf. on Computational Linguistics (COLING)*, pages 836–841, Copenhagen, Denmark, Aug. 1996.

Wu, Dekai. Stochastic Inversion Transduction Grammars and Bilingual Parsing of Parallel Corpora. *Computational Linguistics*, 23(3):377–404, Sept. 1997.

Wuebker, Joern, Matthias Huck, Stephan Peitz, Malte Nuhn, Markus Freitag, Jan-Thorsten Peter, Saab Mansour, and Hermann Ney. Jane 2: Open Source Phrase-based and Hierarchical Statistical Machine Translation. In *Proc. of the Int. Conf. on Computational Linguistics (COLING)*, pages 483–491, Mumbai, India, Dec. 2012.

Zens, Richard. *Phrase-Based Statistical Machine Translation: Models, Search, Training*. PhD thesis, RWTH Aachen University, Aachen, Germany, Feb. 2008.

Zens, Richard and Hermann Ney. A Comparative Study on Reordering Constraints in Statistical Machine Translation. In *Proc. of the Annual Meeting of the Assoc. for Computational Linguistics (ACL)*, pages 144–151, Sapporo, Japan, July 2003.

Zens, Richard and Hermann Ney. Improvements in Dynamic Programming Beam Search for Phrase-based Statistical Machine Translation. In *Proc. of the Int. Workshop on Spoken Language Translation (IWSLT)*, pages 195–205, Honolulu, HI, USA, Oct. 2008.

Zens, Richard, Hermann Ney, Taro Watanabe, and Eiichiro Sumita. Reordering Constraints for Phrase-Based Statistical Machine Translation. In *Proc. of the Int. Conf. on Computational Linguistics (COLING)*, pages 205–211, Geneva, Switzerland, Aug. 2004.

Address for correspondence:
Matthias Huck
huck@cs.rwth-aachen.de
Human Language Technology and Pattern Recognition Group
Computer Science Department
RWTH Aachen University
D-52056 Aachen, Germany