Learning Phrase Boundaries for Hierarchical Phrase-based Translation

Zhongjun HE Yao MENG Hao YU
Fujitsu R&D Center CO., LTD.
{hezhongjun, mengyao, yu}@cn.fujitsu.com

Abstract

Hierarchical phrase-based models provide a powerful mechanism to capture non-local phrase reorderings for statistical machine translation (SMT). However, many phrase reorderings are arbitrary because the models are weak on determining phrase boundaries for pattern-matching. This paper presents a novel approach to learn phrase boundaries directly from word-aligned corpus without using any syntactical information. We use phrase boundaries, which indicate the beginning/ending of phrase reordering, as soft constraints for decoding. Experimental results and analysis show that the approach yields significant improvements over the baseline on large-scale Chinese-to-English translation.

1 Introduction

The hierarchial phrase-based (HPB) model (Chiang, 2005) outperformed previous phrase-based models (Koehn et al., 2003; Och and Ney, 2004) by utilizing hierarchical phrases consisting of both words and variables. Thus the HPB model has generalization ability: a translation rule learned from a phrase pair can be used for other phrase pairs with the same pattern, e.g. reordering information of a short span can be applied for a large span during decoding. Therefore, the model captures both short and long distance phrase reorderings.

However, one shortcoming of the HPB model is that it is difficult to determine phrase boundaries for pattern-matching. Therefore, during decoding, a rule may be applied for all possible source phrases with the same pattern. However, incorrect pattern-matching will cause wrong translation.

Consider the following rule that is used to translate the Chinese sentence in Figure 1 into English:

$$X \rightarrow \langle X_L \text{ de } X_R, X_R \text{ in } X_L \rangle \qquad (1)$$

The rule translates the Chinese word "de" into English word "in", and swaps the left sub-phrase covered by X_L and the right sub-phrase covered by X_R on the target side. However, X_L may pattern-match 5 spans on the left side of "de" and X_R may pattern-match 3 spans on the right side. Therefore, the rule produces 15 different derivations. However, 14 of them are incorrect.

The correct derivation S_c is shown in Figure 2, while one of the wrong derivations S_i is shown in Figure 3. We observe that the basic difference between S_c and S_i is the phrase boundary matched by "X_R". In S_c, X_R matches the span $[7, 9]$ and moves it as a whole unit. While in S_i, X_R matches the span $[7, 8]$ and left the last word $[9, 9]$ be translated separately. Similarly, other incorrect derivations are caused by inadequate pattern-matching of X_L and/or X_R.

Previous research showed that phrases should be constrained to some extent for improving translation quality. Most of the existing approaches utilized syntactic information to constrain phrases to respect syntactic boundaries. Chiang (2005) introduced a constituent feature to reward phrases that match a syntactic tree but did not yield significant improvement. Marton and Resnik (2008) revised this method by distinguishing different constituent syntactic types, and defined features for each type to count whether a phrase matches or crosses the syntactic boundary. This led to a substantial improvements. Gimpel and Smith (2008) presented rich contextual features on the source side including constituent syntactical features for phrase-based translation. Cherry (2008) utilized a dependency tree as a soft constraint to detect syntactic cohesion violations for a phrase-based

Coling 2010: Poster Volume, pages 383–390,
Beijing, August 2010

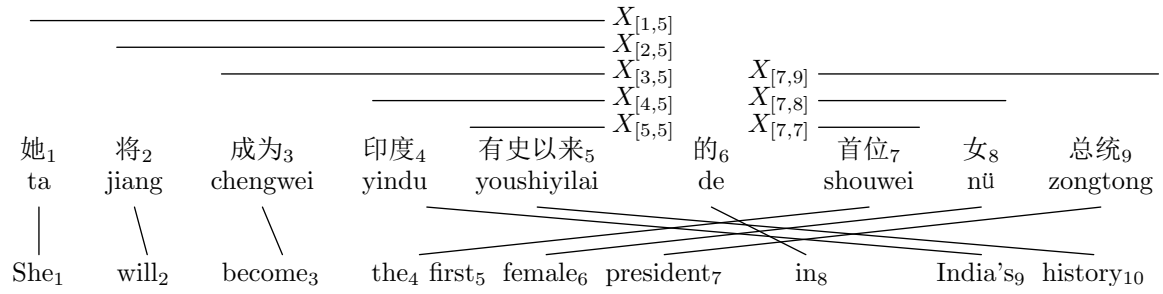

Figure 1: An example of Chinese-English translation. The rule $X \rightarrow \langle X_L \text{ de } X_R,\ X_R \text{ in } X_L \rangle$ pattern-matches 5 and 3 spans on the left and right of the Chinese word "de", respectively.

$$
\begin{aligned}
S_c \ &\Rightarrow\ \langle \text{她 将 成为 } X,\ \text{She will become } X \rangle \\
&\Rightarrow\ \langle \text{她 将 成为 } X_{[4,5]} \text{ 的 } X_{[7,9]},\ \text{She will become } X_{[7,9]} \text{ in } X_{[4,5]} \rangle \\
&\Rightarrow\ \langle \text{她 将 成为 } \| \text{ 印度 有史以来 } \| \text{ 的 } \| \text{ 首位 女 总统,} \\
&\qquad \text{She will become the first female president in India's history} \rangle
\end{aligned}
$$

Figure 2: The correct derivation with adequate pattern-matching of X_R.

$$
\begin{aligned}
S_i \ &\Rightarrow\ \langle \text{她 将 成为 } X \text{ 总统,}\ \text{She will become } X \text{ president} \rangle \\
&\Rightarrow\ \langle \text{她 将 成为 } X_{[4,5]} \text{ 的 } X_{[7,8]} \text{ 总统,}\ \text{She will become } X_{[7,8]} \text{ in } X_{[4,5]} \text{ president} \rangle \\
&\Rightarrow\ \langle \text{她 将 成为 } \| \text{ 印度 有史以来 } \| \text{ 的 } \| \text{ 首位 女 } \| \text{ 总统,} \\
&\qquad \text{She will become the first female in India's history president} \rangle
\end{aligned}
$$

Figure 3: A wrong derivation with inadequate pattern-matching of X_R.

system. Xiong et al. (2009) presented a syntax-driven bracketing model to predict whether two phrases are translated together or not, using syntactic features learned from training corpus. Although these approaches differ from each other, the main basic idea is the utilization of syntactic information.

In this paper, we present a novel approach to learn phrase boundaries for hierarchical phrase-based translation. A phrase boundary indicates the beginning or ending of a phrase reordering. Motivated by Ng and Low (2004) that built a classifier to predict word boundaries for word segmentation, we build a classifier to predict phrase boundaries. We classify each source word into one of the 4 boundary tags: "b" indicates the beginning of a phrase, "m" indicates a word appears in the middle of a phrase, "e" indicates the end of a phrase, "s" indicates a single-word phrase.

We use phrase boundaries as soft constraints for decoding. To do this, we incorporate our classifier as a feature into the HPB model and propose an efficient decoding algorithm.

Compared to the previous work, out approach has the following advantages:

- Our approach maintains the strength of the phrase-based models since it does not require any syntactical information. Therefore, phrases do not need to respect syntactic boundaries.
- The training instances are directly learned from a word-aligned bilingual corpus, rather than from manually annotated corpus.

- The decoder outputs phrase segmentation information as a byproduct, in addition to translation result.

We evaluate our approach on large-scale Chinese-to-English translation. Experimental results and analysis show that using phrase boundaries as soft constraints achieves significant improvements over the baseline system.

2 Previous Work

2.1 Learning Word Boundaries

In some languages, such as Chinese, words are not demarcated. Therefore, it is a preliminary task to determine word boundaries for a sentence, which is the so-called word segmentation.

Ng and Low (2004) regarded word segmentation as a classification problem. They labelled each Chinese character with one of 4 possible boundary tags: "b", "m", "e" respectively indicates the begin, the middle and the end of a word, and "s" indicates a single-character word. Their segmenter was built within a maximum entropy framework and trained on manually segmented sentences.

Learning phrase boundaries is analogous to word boundaries. The basic difference is that the unit for learning word boundaries is character while the unit for learning phrase boundaries is word. In this paper, we adopt the boundary tags presented by Ng and Low (2004) and build a classifier to predict phrase boundaries within maximum entropy framework. We train it directly on a word-aligned bilingual corpus, without any manually annotation and syntactical information.

2.2 The Hierarchical Phrase-based Model

We built a hierarchical phrase-based MT system (Chiang, 2007) based on weighted SCFG. The translation knowledge is represented by rewriting rules:

$$\mathrm{X} \rightarrow \langle \alpha, \gamma, \sim \rangle \tag{2}$$

where X is a non-terminal, α and γ are source and target strings, respectively. Both of them contain words and possibly co-indexed non-terminals. $\sim$ describes a one-to-one correspondence between non-terminals in α and γ.

Chiang (2007) used the standard log-linear framework (Och and Ney, 2002) to combine various features:

$$Pr(e|f) \propto \sum_i \lambda_i h_i(\alpha, \gamma) \tag{3}$$

where $h_i(\alpha, \gamma)$ is a feature function and λ_i is the weight of h_i. Analogous to the previous phrase-based model, Chiang defined the following features: translation probabilities $p(\gamma|\alpha)$ and $p(\alpha|\gamma)$, lexical weights $p_w(\gamma|\alpha)$ and $p_w(\alpha|\gamma)$, word penalty, rule penalty, and a target n-gram language model.

In this paper, we integrate a phrase boundary classifier as an additional feature into the log-linear model to provide soft constraint for pattern-matching during decoding. The feature weights are optimized by MERT algorithm (Och, 2003).

3 Learning Phrase Boundaries

We build a phrase boundary classifier (PBC) within a maximum entropy framework. The PBC predicts a boundary tag for each source word, considering contextual features:

$$P_{tag}(t|f_j, F_1^J) = \frac{exp(\sum_i \lambda_i h_i(t, f_j, F_1^J))}{\sum_t exp(\sum_i \lambda_i h_i(t, f_j, F_1^J)} \tag{4}$$

where, $t \in \{b, m, e, s\}$, f_j is the jth word in source sentence F_1^J, h_i is a feature function and λ_i is the weight of h_i.

To build PBC, we first present a method to recognize phrase boundaries and extract training examples from word-aligned bilingual corpus, then we define contextual feature functions.

3.1 Phrase Boundary

During decoding, intuitively, words within a phrase should be translated or moved together. Therefore, a phrase boundary should indicate reordering information. We assign one of the boundary tags (b, m, e, s) to each word in source sentences. Thus the word with tag *b*, *e* or *s* is a phrase boundary. One question is that how to assign boundary tag to a word? In this paper, we recognize the largest source span which has the monotone translation. Then we assign boundary

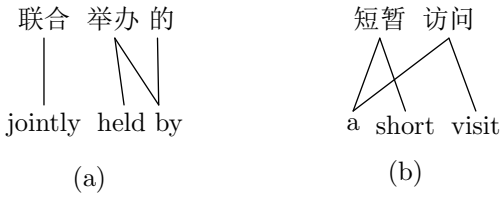

Figure 4: Illustration for monotone span (a) and PM span (b).

tags to each word in the source span, according to their position.

To do this, we first introduce some notations. Given a bilingual sentence (F_1^J, E_1^I) together with word alignment matrix A, we use $L(A_j)$ and $H(A_j)$ to represent the lowest and highest target word position which links to the source word f_j, respectively. Since the word alignment for f_j maybe "one-to-many", all the corresponding target words will appear in the span $[L(A_j), H(A_j)]$.

we define a source span $[j_1, j_2]$ $(1 \leq j_1 \leq j_2 \leq J)$ a *monotone* span, iff:

1. $\forall (j,i) \in A, j_1 \leq j \leq j_2 \leftrightarrow L(A_{j_1}) \leq i \leq H(A_{j_2})$
2. $\forall k_1, k_2 \in [j_1, j_2], k_1 \leq k_2 \rightarrow H(A_{k_1}) \leq L(A_{k_2})$

The first condition indicates that $(F_{j_1}^{j_2}, E_{L(A_{j_1})}^{H(A_{j_2})})$ is a phrase pair as described previously in phrase-based SMT models. While the second condition indicates that the lower target bound linked to a source word cannot be lower than any target word position linked to the previous source word. Therefore, a monotone span does not contain crossed links or internal reorderings.

Considering that word alignments could be very noisy and complex in real-world data, we define *pseudo-monotone* (PM) span by loosening the second condition:

$$\forall k_1, k_2 \in [j_1, j_2], k_1 \leq k_2 \rightarrow L(A_{k_1}) \leq L(A_{k_2}) \quad (5)$$

This condition allows crossed links to some extent by loosening the bound of A_{k_1} from upper to lower. Figure 4 (a) shows an example of monotone span, in which the translation is monotone. While Figure 4 (b) is not a monotone span because there is a cross link between the upper bound of "短暂" and the lower bound of "访问" on the target side. However, it is a PM span according to the definition. Note that in some cases, a source word may not be contained in any phrase pair, therefore we consider a single word span as a PM span, specificly.

An interesting feature of PM span is that if two PM spans are consecutive on both source side and their corresponding target side, the two PM spans can be combined as a larger PM span. Formally,

$$(F_{j_1}^{j}, E_{i_1}^{i}) \bigoplus (F_{j+1}^{j_2}, E_{i+1}^{i_2}) = (F_{j_1}^{j_2}, E_{i_1}^{i_2}) \quad (6)$$

where $[j_1, j]$ and $[j+1, j_2]$ are PM spans, $[i_1, i]$ and $[i+1, i_2]$ are the target spans corresponding to $[j_1, j]$ and $[j+1, j_2]$, respectively. For example, Figure 4 (a) shows a PM phrase pair that consists of two small PM pairs "联合, jointly" and "举办的, held by".

In this paper, we are interested in phrase reordering boundaries for a source sentence. We define *translation span* (TS) the largest possible PM span. A TS may consist of one or more PM spans. According to our definition, cross links may appear within PM spans but do not appear between PM spans within a TS. Therefore, TS is the largest possible span that will be translated as a unit and phrase reorderings may occur between TSs during decoding.

To obtain phrase boundary examples from word-aligned bilingual sentences, we first find all possible TSs and then assign boundary tags to each word. For a TS $[j_1, j_2]$ $(j_1 < j_2)$ that contain more than two words, we assign "b" to the first word f_{j_1} and "e" to the last word f_{j_2}, and "m" to the middle words f_j $(j_1 < j < j_2)$. For a single word span TS $[j, j]$, we assign "s" to the word f_j.

Figure 5 shows an example of labelling source words with boundary tags. The source sentence is segmented into 4 TSs. Using the phrase boundary information to guide decoding, the decoder will produce the correct derivation and translation as shown in Figure 2.

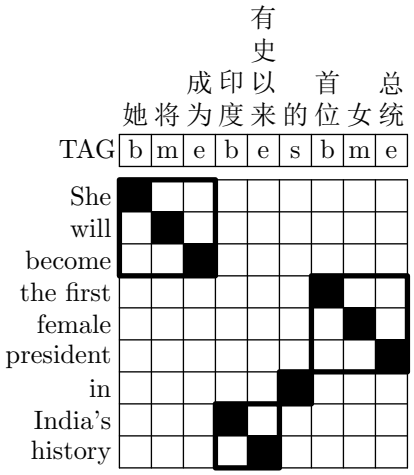

Figure 5: Illustration for labelling the source words with boundary tags. The solid boxes present word alignments. The bordered boxes are TSs.

3.2 Feature Definition

The features we used for the PS model are analogous to (Ng and Low, 2004). For a word W_0, we define the following contextual features with a window of "n":

- The word feature W_n, which denotes the left (right) n words of the current word W_0;
- The part-of-speech (POS) feature P_n, which denotes the POS tag of the word W_n.

For example, the tag of the word "成为 (become)" in Figure 5 is "e", indicating that it is the end of a phrase. If we set the context window $n = 2$, the features of the word "成为 (become)" are:

- W_{-2}=她 W_{-1}=将 W_0=成 为 W_1=印 度 W_2=有史以来
- P_{-2}=r P_{-1}=d P_0=v P_1=ns P_2=l

We collect TSs from bilingual sentences together with the contextual features and used a MaxEnt toolkit (Zhang, 2004) to train a PBC.

	她	将	成为
b	**0.78**	0.10	1.2e-5
m	6.4e-8	**0.75**	5.4e-5
e	2.1e-8	0.11	**0.87**
s	0.22	0.04	0.13

Table 1: The TPM for a source sentence. The highest probability of each word is in bold.

4 Phrase Boundary Constrained Decoding

Give a source sentence, we can assign boundary tags to each word by running the PBC. During decoding, a rule is prohibited to pattern-match across phrase boundaries. By doing this, the PBC is integrated as a hard constraint. However, this method will invalidate a large number of rules and the decoder suffers from a risk that there are not enough rules to cover the source sentence.

Alternatively, inspired by previous approaches, we integrate the phrase boundary classifier as a soft constraint by incorporating it as a feature into the HPB model:

$$h_{pbc}(F_1^J) = log(\prod_{j=1}^{J} P_{tag}(t|f_j, F_1^J)) \quad (7)$$

To perform translation, for each word f_j in a source sentence F_1^J, we first compute all tag probabilities $P_{tag}(t|f_j)$, where $t \in (b, m, e, s)$, $j \in [1, J]$, according to Equation 4. Therefore, we build a $4 \times J$ tag-word probability matrix (TPM). $TPM[i, j]$ indicates the probability of the word f_j labelled with the tag t_i. Table 1 shows the TPM for a source text "她 将 成为".

Then we select rule options from the rule table that can be used for translating the source text. Since each rule option $(\widetilde{f}, \widetilde{e}, a)$ [1] can be regarded as a bilingual sentence with word alignments, thus we find all TS in $\widetilde{f}$ and assign an *initial tag* (IT) for each source word. This procedure is analogous to label phrase boundary tags for a word-aligned bilingual sentence. For example, the following rules are used for translating the Chinese sentence in Table 1:

[1] We keep word alignments of a rule when it is extracted from bilingual sentence.

$$X \rightarrow \langle \text{她}^{b} X_1^{*},\ \text{She } X_1 \rangle \quad (8)$$

$$X_1 \rightarrow \langle \text{将}^{b}\ \text{成为}^{e},\ \text{will become} \rangle \quad (9)$$

Since both the source sides of these two rules are PM spans according to the word alignments, the IT sequences for rule (8) and (9) are "b *"[2] and "b e", respectively. According to Table 1, the initial h_{pbc} score for these two rules can be computed as follows:

$$h_{pbc}^{(7)} = log(P_{tag}(b|\text{她})) = log(TPM[1,1]) \quad (10)$$

$$\begin{aligned} h_{pbc}^{(8)} &= log(P_{tag}(b|\text{将})) + log(P_{tag}(e|\text{成为})) \\ &= log(TPM[1,2]) + log(TPM[3,3]) \end{aligned} \quad (11)$$

Note that to keep the tag sequence valid, e.g. "m" follows "b" rather than "s", the ITs maybe updated during decoding. The tag-updating should be consistent with the definition of TS as described in Section 3.1. Specifically, when the non-terminal symbol X is derived from its covered span $f(X)$, the boundary tags should be updated.

When a tag of word f_j is updated from t_{k_1} to t_{k_2}, the PBC score should also be updated according to TPM:

$$\Delta PBC = log(TPM[k_2, j]) - log(TPM[k_1, j]) \quad (12)$$

The following is a derivation of the source sentence in Table 1:

$$\begin{aligned} S &\Rightarrow \langle \text{她}^{b} X_1^{*},\ \text{She } X_1 \rangle \\ &\Rightarrow \langle \text{她}^{b}\text{将}^{b \rightarrow m}\ \text{成为}^{e},\ \text{She will become} \rangle \end{aligned}$$

When X_1 is derived, the tag of its left boundary word "将" is updated from "b" to "m". The reason is that after derivation, the combined span forms a larger PM span and the left boundary of $f(X_1)$ should be updated.

As a result, the h_{pbc} score is recomputed:

$$h_{pbc}(F_1^3) = h_{pbc}^{(7)} + h_{pbc}^{(8)} + \Delta PBC \quad (13)$$

where,

$$\Delta PBC = log(TPM[2,2]) - log(TPM[1,2]) \quad (14)$$

[2]We use "*" as a tag of the non-terminal symbol "X_1" since it has not been derived.

The decoding algorithm is efficient since the computing of the PBC score is a procedure of table-lookup.

5 Experiments

5.1 Experimental Setup

Our experiments were on Chinese-to-English translation. The training corpus (77M+81M) we used are from LDC [3]. The evaluation metric is BLEU (Papineni et al., 2002), as calculated by mteval-v11b.pl with case-insensitive matching of n-grams, where $n = 4$.

To obtain word alignments, we first ran GIZA++ (Och and Ney, 2002) in both translation directions and then refined it by "grow-diag-final" method (Koehn et al., 2003).

For the language model, we used the SRI Language Modeling Toolkit (Stolcke, 2002) to train two 4-gram models on xinhua portion of GigaWord corpus and the English side of the training corpus.

The NIST MT03 test set is used to tune the feature weights of the log-linear model by MERT (Och, 2003). We tested our system on the NIST MT06 and MT08 test sets.

5.2 Results

The results are shown in Table 2. We tested various settings of the context window. It is observed that the small values of n ($n = 1, 2$) drop the BLEU score, suggesting that perhaps there are not enough contextual information. With more contextual information is used, the BLEU scores are improved over all test sets. When $n = 3$, the most significant improvements are obtained on MT06G and MT08. The improvements over the baseline are statistically significant at $p < 0.01$ by using the significant test method described in (Koehn, 2004). While for MT06N, the optimized context window size is $n = 4$ but the improvement is not statistically significant. In most cases, with n larger than 3, we do not obtain further improvements because of the data sparseness for training

[3]LDC2002E18, LDC2002L27, LDC2002T01, LDC2003E07, LDC2003E14, LDC2004T07, LDC2005E83, LDC2005T06, LDC2005T10, LDC2005T34, LDC2006E24, LDC2006E26, LDC2006E34, LDC2006E86, LDC2006E92, LDC2006E93, LDC2004T08(HK_News, HK_Hansards).

System	MT06G	MT06N	MT08
baseline	14.66	34.42	26.29
+PBC (n=1)	13.78	33.20	24.58
+PBC (n=2)	14.34	34.21	25.87
+PBC (n=3)	**15.19***	34.63	**27.25***
+PBC (n=4)	14.76	**34.73**	26.70

Table 2: Results on the test sets with different context window (n) of the phrase boundary classifier. The largest BLEU score on each test set is in bold. MT06G: MT06 GALE set. MT06N: MT06 NIST set. *: significantly better than the baseline at $p < 0.01$.

the classifier.

6 Discussion

The experimental results show that the phrase boundary constrained method improves the BLEU score over the baseline system. Furthermore, we are interested in how the PBC affects the translation results? We compared the outputs generated by the baseline and "+PBC ($n = 3$)" system and found some interesting translations. For example, the translations of a source sentence of NIST08 are as follows [4]:

- Src: 美$_1^b$ 财长$_2^m$ 抵$_3^m$ 中国$_4^m$ 访问$_5^e$ ‖ 环保$_6^b$ 与$_7^m$ 汇率$_8^e$ ‖ 是$_9^b$ 关切$_{10}^m$ 重点$_{11}^e$
- Ref: US$_1$ Treasury-Secretary$_2$ Arrives-in$_3$ China$_4$ for-a-Visit-with$_5$ Environment$_6$ and$_7$ Exchange-Rate$_8$ as$_9$ Focus$_{10,11}$
- HPB: US$_1$ Treasury$_2$ in-environmental-protection$_6$ and$_7$ visit$_5$ China$_4$ is$_9$ key$_{11}$ to-the-concern-of$_{10}$ the-exchange-rate$_8$
- +PBC: US$_1$ Treasury$_2$ arrived-in$_3$ China$_4$ for-a-visit$_5$ environmental-protection$_6$ and$_7$ exchange-rate$_8$ is$_9$ concerned-about$_{10}$ the-key$_{11}$

In the example, both "环保" and "汇率" in the source sentence are the concern of the "visit". Therefore, the source span $[6, 8]$ indicates a cohesive phrase, which should be translated as a whole unit. However, the baseline translates the spans $[6, 7]$ and $[8, 8]$ separately. It moves $[6, 7]$ before "visit China" and $[8, 8]$ after "concern". This makes an mistake on phrase reordering. We observe that the "+PBC" system produces a better translation. After incorporating the PBC as a soft constraint, the system assigns a boundary tag to each source word and segments the source sentence into three TSs. According to our definition, TSs are encouraged as pseudo-monotone translation unit during decoding. As a result, the "+PBC" system discourages some arbitrary reordering rules and produces more fluent translation.

[4]The co-indexes of the words on the source and target sentence indicate word alignments.

7 Conclusion and Future Work

This paper presented a phrase boundary constrained method for hierarchical phrase-based translation. A phrase boundary indicates begin or end of a phrase reordering. We built a phrase boundary classifier within a maximum entropy framework and learned phrase boundary examples directly from word-aligned bilingual corpus. We proposed an efficient decoding method to integrate the PBC into the decoder as a soft constraint. Experiments and analysis show that the phrase boundary constrained method achieves significant improvements over the baseline system.

The most advantage of the PBC is that it handles both syntactic and non-syntactic phrases. In the future, We would like to try different methods to determine more informative phrase boundaries, e.g. Xiong et al. (2010) proposed a method to learn translation boundaries from a hierarchical tree that decomposed from word alignments using a shift-reduce algorithm. In addition, we will try more features as described in (Chiang et al., 2008; Chiang et al., 2009), e.g. the length of the phrases that covered by non-terminals.

References

Cherry, Colin. 2008. Cohesive phrase-based decoding for statistical machine translation. In *Proceedings of the 46rd Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, page 72 – 80.

Chiang, David, Yuval Marton, and Philip Resnik.

2008. Online large-margin training of syntactic and structural translation features. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, page 224 – 233.

Chiang, David, Wei Wang, and Kevin Knight. 2009. 11,001 new features for statistical machine translation. In *Proceedings of Human Language Technologies: the 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, page 218 – 226.

Chiang, David. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 263–270.

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

Gimpel, Kevin and Noah A. Smith. 2008. Rich source-side context for statistical machine translation. In *In Proceedings of the ACL-2008 Workshop on Statistical Machine Translation (WMT-2008)*, pages 9–17.

Koehn, Philipp, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of HLT-NAACL 2003*, pages 127–133.

Koehn, Philipp. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 388–395.

Marton, Yuval and Philip Resnik. 2008. Soft syntactic constraints for hierarchical phrased-based translation. In *Proceedings of the 46rd Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1003–1011.

Ng, Hweetou and Jinkiat Low. 2004. Chinese part-of-speech tagging: One-at-a-time or all-at-once? word-based or character-based? In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, pages 277–284.

Och, Franz Josef and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 295–302.

Och, Franz Josef and Hermann Ney. 2004. The alignment template approach to statistical machine translation. 30:417–449.

Och, Franz Josef. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167.

Papineni, K., S. Roukos, T. Ward, and W.-J. Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Stolcke, Andreas. 2002. SRILM – An extensible language modeling toolkit. In *Proceedings of the International Conference on Spoken language Processing*, volume 2, pages 901–904.

Xiong, Deyi, Min Zhang, Aiti Aw, and Haizhou Li. 2009. A syntax-driven bracketing model for phrase-based translation. In *ACL-IJCNLP 2009*, page 315 – 323.

Xiong, Deyi, Min Zhang, and Haizhou Li. 2010. Learning translation boundaries for phrase-based decoding. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL*, page 136 – 144.

Zhang, Le. 2004. Maximum entropy modeling toolkit for python and c++. available at http://homepages.inf.ed.ac.uk/s0450736/maxent_toolkit.html.