Inner-Outer Bracket Models for Word Alignment using Hidden Blocks

Bing Zhao
School of Computer Science
Carnegie Mellon University
{bzhao}@cs.cmu.edu

Niyu Ge and **Kishore Papineni**
IBM T. J. Watson Research Center
Yorktown Heights, NY 10598, USA
{niyuge, papineni}@us.ibm.com

Abstract

Most statistical translation systems are based on phrase translation pairs, or "blocks", which are obtained mainly from word alignment. We use blocks to infer better word alignment and improved word alignment which, in turn, leads to better inference of blocks. We propose two new probabilistic models based on the inner-outer segmentations and use EM algorithms for estimating the models' parameters. The first model recovers IBM Model-1 as a special case. Both models outperform bi-directional IBM Model-4 in terms of word alignment accuracy by 10% absolute on the F-measure. Using blocks obtained from the models in actual translation systems yields statistically significant improvements in Chinese-English SMT evaluation.

1 Introduction

Today's statistical machine translation systems rely on high quality phrase translation pairs to acquire state-of-the-art performance, see (Koehn et al., 2003; Zens and Ney, 2004; Och and Ney, 2003). Here, phrase pairs, or "blocks" are obtained automatically from parallel sentence pairs via the underlying word alignments. Word alignments traditionally are based on IBM Models 1-5 (Brown et al., 1993) or on HMMs (Vogel et al., 1996). Automatic word alignment is challenging in that its accuracy is not yet close to inter-annotator agreement in some language pairs: for Chinese-English, inter-annotator agreement exceeds 90 on F-measure whereas IBM Model-4 or HMM accuracy is typically below 80s. HMMs assume that words "close-in-source" are aligned to words "close-in-target". While this locality assumption is generally sound, HMMs do have limitations: the self-transition probability of a state (word) is the only control on the duration in the state, the length of the phrase aligned to the word. Also there is no natural way to control repeated non-contiguous visits to a state. Despite these problems, HMMs remain attractive for their speed and reasonable accuracy.

We propose a new method for localizing word alignments. We use blocks to achieve locality in the following manner: a block in a sentence pair is a source phrase aligned to a target phrase. We assume that words inside the source phrase cannot align to words outside the target phrase and that words outside the source phrase cannot align to words inside the target phrase. Furthermore, a block divides the sentence pair into two smaller regions: the *inner* part of the block, which corresponds to the source and target phrase in the block, and the *outer* part of the block, which corresponds to the remaining source and target words in the parallel sentence pair. The two regions are non-overlapping; and each of them is shorter than the original parallel sentence pair. The regions are thus easier to align than the original sentence pairs (e.g., using IBM Model-1). While the model uses a single block to split the sentence pair into two independent regions, it is not clear which block we should select for this purpose. Therefore, we treat the splitting block as a hidden variable.

This proposed approach is far simpler than treating the entire sentence as a sequence of non-overlapping phrases (or chunks) and considering such sequential segmentation either explicitly or implicitly. For example, (Marcu and Wong, 2002) for a joint phrase based model, (Huang et al., 2003) for a translation memory system; and (Watanabe et al., 2003) for a complex model of insertion, deletion and head-word driven chunk reordering. Other approaches including (Watanabe et al., 2002) treat extracted phrase-pairs as new parallel data with limited success. Typically, they share a similar architecture of phrase level segmentation, reordering, translation as in (Och and Ney, 2002; Koehn and Knight, 2002; Yamada and Knight, 2001). The phrase level interaction has to be taken care of for the non-overlapping sequential segmentation in a complicated way. Our models model such interactions in a soft way. The hidden blocks are allowed to overlap with each other,

Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP), pages 177–184, Vancouver, October 2005.

while each block induced two non-overlapping regions, i.e. the model brackets the sentence pair into two independent parts which are generated synchronously. In this respect, it resembles bilingual bracketing (Wu, 1997), but our model has more lexical items in the blocks with many-to-many word alignment freedom in both inner and outer parts.

We present our localization constraints using blocks for word alignment in Section 2; we detail our two new probabilistic models and their EM training algorithms in Section 3; our baseline system, a maximum-posterior inference for word alignment, is explained in Section 4; experimental results of alignments and translations are in Section 5; and Section 6 contains discussion and conclusions.

2 Segmentation by a Block

We use the following notation in the remainder of this paper: $\mathbf{e}$ and $\mathbf{f}$ denote the English and foreign sentences with sentence lengthes of I and J, respectively. e_i is an English word at position i in $\mathbf{e}$; f_j is a foreign word at position j in $\mathbf{f}$. $\mathbf{a}$ is the alignment vector with a_j mapping the position of the English word $e_{a_j}$to which f_j connects. Therefore, we have the standard limitation that one foreign word cannot be connected to more than one English word. A *block* $\delta^{[]}$ is defined as a pair of brackets as follows:

$$\delta^{[]} = (\delta^{\mathbf{e}}, \delta^{\mathbf{f}}) = ([i_l, i_r], [j_l, j_r]), \tag{1}$$

where $\delta^{\mathbf{e}} = [i_l, i_r]$ is a bracket in English sentence defined by a pair of indices: the *left* position i_l and the *right* position i_r, corresponding to a English phrase $e_{i_l}^{i_r}$. Similar notations are for $\delta^{\mathbf{f}} = [j_l, j_r]$, which is one possible *projection* of $\delta^{\mathbf{e}}$ in $\mathbf{f}$. The subscript l and r are abbreviations of left and right, respectively.

$\delta^{\mathbf{e}}$ segments $\mathbf{e}$ into two parts: $(\delta^{\mathbf{e}}, \mathbf{e}) = (\delta^{\mathbf{e}}_{\in}, \delta^{\mathbf{e}}_{\notin})$. The inner part $\delta^{\mathbf{e}}_{\in} = \{e_i, i \in [i_l, i_r]\}$ and the outer part $\delta^{\mathbf{e}}_{\notin} = \{e_i, i \notin [i_l, i_r]\}$; $\delta^{\mathbf{f}}$ segments $\mathbf{f}$ similarly.

Thus, the block $\delta^{[]}$ splits the parallel sentence pair into two *non-overlapping* regions: the *Inner* $\delta^{[]}_{\in}$ and *Outer* $\delta^{[]}_{\notin}$ parts (see Figure 1). With this segmentation, we assume the words in the inner part are aligned to inner part only: $\delta^{[]}_{\in} = \delta^{\mathbf{e}}_{\in} \leftrightarrow \delta^{\mathbf{f}}_{\in} : \{e_i, i \in [i_l, i_r]\} \leftrightarrow \{f_j, j \in [j_l, j_r]\}$; and words in the outer part are aligned to outer part only: $\delta^{[]}_{\notin} = \delta^{\mathbf{e}}_{\notin} \leftrightarrow \delta^{\mathbf{f}}_{\notin} : \{e_i, i \notin [i_l, i_r]\} \leftrightarrow \{f_j, j \notin [j_l, j_r]\}$. We do not allow alignments to cross block boundaries. Words inside a block $\delta^{[]}$ can be aligned using a variety of models (IBM models 1-5, HMM, etc). We choose Model1 for simplicity. If the block boundaries are accurate, we can expect high quality word alignment. This is our proposed new localization method.

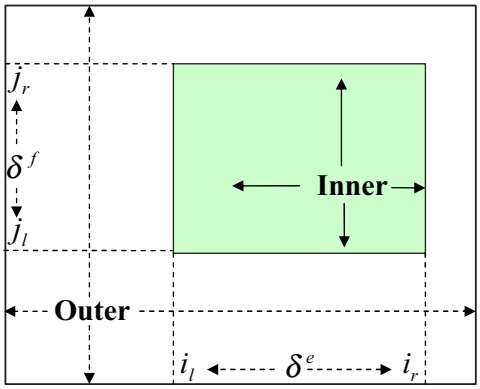

Figure 1: Segmentation by a Block

3 Inner-Outer Bracket Models

We treat the constraining block as a hidden variable in a generative model shown in Eqn. 2.

$$\begin{aligned} P(\mathbf{f}|\mathbf{e}) &= \sum_{\{\delta^{[]}\}} P(\mathbf{f}, \delta^{[]}|\mathbf{e}) \\ &= \sum_{\{\delta^{\mathbf{e}}\}} \sum_{\{\delta^{\mathbf{f}}\}} P(\mathbf{f}, \delta^{\mathbf{f}}|\delta^{\mathbf{e}}, \mathbf{e}) P(\delta^{\mathbf{e}}|\mathbf{e}), \end{aligned} \tag{2}$$

where $\delta^{[]} = (\delta^{\mathbf{e}}, \delta^{\mathbf{f}})$ is the hidden block. In the generative process, the model first generates a bracket $\delta^{\mathbf{e}}$ for $\mathbf{e}$ with a monolingual bracketing model of $P(\delta^{\mathbf{e}}|\mathbf{e})$. It then uses the segmentation of the English $(\delta^{\mathbf{e}}, \mathbf{e})$ to generate the projected bracket $\delta^{\mathbf{f}}$ of $\mathbf{f}$ using a generative translation model $P(\mathbf{f}, \delta^{\mathbf{f}}|\delta^{\mathbf{e}}, \mathbf{e}) = P(\delta^{\mathbf{f}}_{\notin}, \delta^{\mathbf{f}}_{\in}|\delta^{\mathbf{e}}_{\notin}, \delta^{\mathbf{e}}_{\in})$ — the key model to implement our proposed inner-outer constraints. With the hidden block $\delta^{[]}$ inferred, the model then generates word alignments within the inner and outer parts separately. We present two generating processes for the inner and outer parts induced by $\delta^{[]}$ and corresponding two models of $P(\mathbf{f}, \delta^{\mathbf{f}}|\delta^{\mathbf{e}}, \mathbf{e})$. These models are described in the following secions.

3.1 Inner-Outer Bracket Model-A

The first model assumes that the inner part and the outer part are generated independently. By the formal equivalence of (f, δ^f) with $(\delta^f_{\in}, \delta^f_{\notin})$, Eqn. 2 can be approximated as:

$$P(\mathbf{f}|\mathbf{e}) \approx \sum_{\{\delta^{\mathbf{e}}\}} \sum_{\{\delta^{\mathbf{f}}\}} P(\delta^{\mathbf{f}}_{\in}|\delta^{\mathbf{e}}_{\in}) P(\delta^{\mathbf{f}}_{\notin}|\delta^{\mathbf{e}}_{\notin}) P(\delta^{\mathbf{e}}|\mathbf{e}) P(\delta^{\mathbf{f}}|\delta^{\mathbf{e}}), \tag{3}$$

where $P(\delta^{\mathbf{f}}_{\in}|\delta^{\mathbf{e}}_{\in})$ and $P(\delta^{\mathbf{f}}_{\notin}|\delta^{\mathbf{e}}_{\notin})$ are two independent generative models for inner and outer parts, respec-

tively and are futher decompsed into:

$$P(\delta^{\mathbf{f}}_{\in}|\delta^{\mathbf{e}}_{\in}) = \sum_{\{a_j \in \delta^{\mathbf{e}}_{\in}\}} \prod_{f_j \in \delta^{\mathbf{f}}_{\in}} P(f_j|e_{a_j}) P(e_{a_j}|\delta^{\mathbf{e}}_{\in})$$

$$P(\delta^{\mathbf{f}}_{\notin}|\delta^{\mathbf{e}}_{\notin}) = \sum_{\{a_j \in \delta^{\mathbf{e}}_{\notin}\}} \prod_{f_j \in \delta^{\mathbf{f}}_{\notin}} P(f_j|e_{a_j}) P(e_{a_j}|\delta^{\mathbf{e}}_{\notin}), \quad (4)$$

where $\{a_1^J\}$ is the word alignment vector. Given the block segmentation and word alignment, the generative process first randomly selects a e_i according to either $P(e_i|\delta^{\mathbf{e}}_{\in})$ or $P(e_i|\delta^{\mathbf{e}}_{\notin})$; and then generates f_j indexed by word alignment a_j with $i = a_j$ according to a word level lexicon $P(f_j|e_{a_j})$. This generative process using the two models of $P(\delta^{\mathbf{f}}_{\in}|\delta^{\mathbf{e}}_{\in})$ and $P(\delta^{\mathbf{f}}_{\notin}|\delta^{\mathbf{e}}_{\notin})$ must satisfy the constraints of segmentations induced by the hidden block $\delta^{[]} = (\delta^{\mathbf{e}}, \delta^{\mathbf{f}})$. The English words $\delta^{\mathbf{e}}_{\in}$ inside the block can only generate the words in $\delta^{\mathbf{f}}_{\in}$ and nothing else; likewise $\delta^{\mathbf{e}}_{\notin}$ only generates $\delta^{\mathbf{f}}_{\notin}$. Overall, the combination of $P(\delta^{\mathbf{f}}_{\in}|\delta^{\mathbf{e}}_{\in})P(\delta^{\mathbf{f}}_{\notin}|\delta^{\mathbf{e}}_{\notin})$ in Eqn. 3 collaborates each other quite well in practice. For a particular observation $\delta^{\mathbf{f}}_{\in}$, if $\delta^{\mathbf{e}}_{\in}$ is too small (i.e., missing translations), $P(\delta^{\mathbf{f}}_{\in}|\delta^{\mathbf{e}}_{\in})$ will suffer; and if $\delta^{\mathbf{e}}_{\in}$ is too big (i.e., robbing useful words from $\delta^{\mathbf{e}}_{\notin}$), $P(\delta^{\mathbf{f}}_{\notin}|\delta^{\mathbf{e}}_{\notin})$ will suffer. Therefore, our proposed model in Eqn. 3 combines the two costs and requires both inner and outer parts to be explained well at the same time.

Because the model in Eqn. 3 is essentially a two-level ($\delta^{[]}$ and $\mathbf{a}$) mixture model similar to IBM Models, the EM algorithm is quite straight forward as in IBM models. Shown in the following are several key E-step computations of the posteriors. The M-step (optimization) is simply the normalization of the fractional counts collected using the posteriors through the inference results from E-step:

$$P_{\delta^{[]}_{\in}}(a_j|\delta^{\mathbf{f}}_{\in}, \delta^{\mathbf{e}}_{\in}) = \frac{P(f_j|e_{a_j})}{\sum_{e_k \in \delta^{\mathbf{e}}_{\in}} P(f_j|e_k)}$$

$$P_{\delta^{[]}_{\notin}}(a_j|\delta^{\mathbf{f}}_{\notin}, \delta^{\mathbf{e}}_{\notin}) = \frac{P(f_j|e_{a_j})}{\sum_{e_k \in \delta^{\mathbf{e}}_{\notin}} P(f_j|e_k)} \quad (5)$$

The posterior probability of $P(a_1^J|\mathbf{f}, \delta^{\mathbf{f}}, \delta^{\mathbf{e}}, \mathbf{e}) = \prod_{j=1}^{J} P(a_j|\mathbf{f}, \delta^{\mathbf{f}}, \delta^{\mathbf{e}}, \mathbf{e})$, where $P(a_j|\mathbf{f}, \delta^{\mathbf{f}}, \delta^{\mathbf{e}}, \mathbf{e})$ is either $P_{\delta^{[]}_{\in}}(a_j|\delta^{\mathbf{f}}_{\in}, \delta^{\mathbf{e}}_{\in})$ when $(f_j, e_{a_j}) \in \delta^{[]}_{\in}$, or otherwise $P_{\delta^{[]}_{\notin}}(a_j|\delta^{\mathbf{f}}_{\notin}, \delta^{\mathbf{e}}_{\notin})$ when $(f_j, e_{a_j}) \in \delta^{[]}_{\notin}$. Assuming $P(\delta^{\mathbf{e}}|\mathbf{e})$ to be a uniform distribution, the posterior of selecting a hidden block given observations: $P(\delta^{[]} = (\delta^{\mathbf{e}}, \delta^{\mathbf{f}})|\mathbf{e}, \mathbf{f})$ is proportional to block level relative frequency $P_{rel}(\delta^{\mathbf{f}}_{\in}|\delta^{\mathbf{e}}_{\in})$ updated in each iteration; and can be smoothed with $P(\delta^{\mathbf{f}}|\delta^{\mathbf{e}}, \mathbf{f}, \mathbf{e}) = P(\delta^{\mathbf{f}}_{\in}|\delta^{\mathbf{e}}_{\in})P(\delta^{\mathbf{f}}_{\notin}|\delta^{\mathbf{e}}_{\notin}) / \sum_{\{\delta'^{\mathbf{f}}\}} P(\delta'^{\mathbf{f}}_{\in}|\delta^{\mathbf{e}}_{\in})P(\delta'^{\mathbf{f}}_{\notin}|\delta^{\mathbf{e}}_{\notin})$ assuming Model-1 alignment in the inner and outer parts independently to reduce the risks of data sparseness in estimations.

In principle, $\delta^{\mathbf{e}}$ can be a bracket of any length not exceeding the sentence length. If we restrict the bracket length to that of the sentence length, we recover IBM Model-1. Figure 2 summarizes the generation process for Inner-Outer Bracket Model-A.

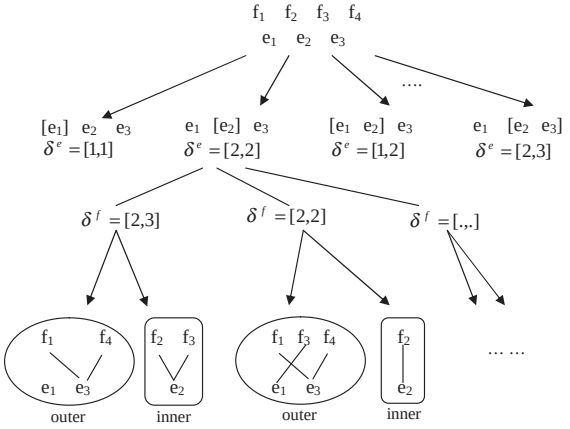

Figure 2: Illustration of generative Bracket Model-A

3.2 Inner-Outer Bracket Model-B

A block $\delta^{[]}$ invokes both the inner and outer generations simultaneously in Bracket Model A (BM-A). However, the generative process is usually more effective in the inner part as $\delta^{[]}$ is generally small and accurate. We can build a model focusing on generating only the inner part with careful inferences to avoid errors from noisy blocks. To ensure that all f_1^J are generated, we need to propose enough blocks to cover each observation f_j. This constraint can be met by treating the whole sentence pair as one block.

The generative process is as follows: First the model generates an English bracket $\delta^{\mathbf{e}}$ as before. The model then generates a projection $\delta^{\mathbf{f}}$ in $\mathbf{f}$ to localize all a_j's for the given $\delta^{\mathbf{e}}$ according to $P(\delta^{\mathbf{f}}|\delta^{\mathbf{e}}, \mathbf{e})$. $\delta^{\mathbf{e}}$ and $\delta^{\mathbf{f}}$ forms a hidden block $\delta^{[]}$. Given $\delta^{[]}$, the model then generates only the inner part $f_j \in \delta^{\mathbf{f}}_{\in}$ via $P(\mathbf{f}|\delta^{\mathbf{f}}, \delta^{\mathbf{e}}, \mathbf{e}) \simeq P(\delta^{\mathbf{f}}_{\in}|\delta^{\mathbf{f}}, \delta^{\mathbf{e}}, \mathbf{e})$. Eqn. 6 summarizes this by rewriting $P(\mathbf{f}, \delta^{\mathbf{f}}|\delta^{\mathbf{e}}, \mathbf{e})$:

$$\begin{aligned} P(\mathbf{f}, \delta^{\mathbf{f}}|\delta^{\mathbf{e}}, \mathbf{e}) &= P(\mathbf{f}|\delta^{\mathbf{f}}, \delta^{\mathbf{e}}, \mathbf{e}) P(\delta^{\mathbf{f}}|\delta^{\mathbf{e}}, \mathbf{e}) \quad (6) \\ &= P(\mathbf{f}|\delta^{\mathbf{f}}, \delta^{\mathbf{e}}, \mathbf{e}) P([j_l, j_r]|\delta^{\mathbf{e}}, \mathbf{e}) \\ &\simeq P(\delta^{\mathbf{f}}_{\in}|\delta^{\mathbf{f}}, \delta^{\mathbf{e}}, \mathbf{e}) P([j_l, j_r]|\delta^{\mathbf{e}}, \mathbf{e}). \end{aligned}$$

$P(\delta^{\mathbf{f}}_{\in}|\delta^{\mathbf{f}}, \delta^{\mathbf{e}}, \mathbf{e})$ is a bracket level *emission* probabilistic model which generates a bag of contiguous words $f_j \in \delta^{\mathbf{f}}_{\in}$ under the constraints from the given hidden block $\delta^{[]} = (\delta^{\mathbf{f}}, \delta^{\mathbf{e}})$. The model is simplified in Eqn. 7 with the assumption of bag-of-words' independence within the bracket $\delta^{\mathbf{f}}$:

$$P(\delta^{\mathbf{f}}_{\in}|\delta^{\mathbf{f}}, \delta^{\mathbf{e}}, \mathbf{e}) = \sum_{a_1^J} \prod_{j \in \delta^{\mathbf{f}}_{\in}} P(f_j|e_{a_j}) P(e_{a_j}|\delta^{\mathbf{f}}, \delta^{\mathbf{e}}, \mathbf{e}). \quad (7)$$

The $P([j_l, j_r]|\delta^{\mathbf{e}}, \mathbf{e})$ in Eqn. 6 is a *localization* probabilistic model, which has resemblances to an HMM's transition probability, $P(a_j|a_{j-1})$, implementing the assumption "close-in-source" is aligned to "close-in-target". However, instead of using the simple position variable a_j, $P([j_l, j_r]|\delta^{\mathbf{e}}, \mathbf{e})$ is more expressive with word identities to localize words $\{f_j\}$ aligned to $\delta^{\mathbf{e}}_{\in}$. To model $P([j_l, j_r]|\delta^{\mathbf{e}}, \mathbf{e})$ reliably, $\delta^{\mathbf{f}} = [j_l, j_r]$ is equivalently defined as the *center* and *width* of the bracket $\delta^{\mathbf{f}}$: $(\odot_{\delta^{\mathbf{f}}}, w_{\delta^{\mathbf{f}}})$. To simplify it further, we assume that $w_{\delta^{\mathbf{f}}}$ and $\odot_{\delta^{\mathbf{f}}}$ can be predicted independently.

The *width* model, $P(w_{\delta^{\mathbf{f}}}|\delta^{\mathbf{e}}, \mathbf{e})$, depends on the length of the English bracket and the fertilities of English words in it. To simplify M-step computations, we can compute the expected width as in Eqn. 8.

$$E\{w_{\delta^{\mathbf{f}}}|\delta^{\mathbf{e}}, \mathbf{e}\} \simeq \gamma \cdot |i_r - i_l + 1|, \tag{8}$$

where γ is the expected bracket length ratio and is approximated by the average sentence length ratio computed using the whole parallel corpus. For Chinese-English, $\gamma = 1/1.3 = 0.77$. In practice, this estimation is quite reliable.

The *center* model $P(\odot_{\delta^{\mathbf{f}}}|\delta^{\mathbf{e}}, \mathbf{e})$ is harder. It is conditioned on the translational equivalence between the English bracket and its projection. We compute the expected $\odot_{\delta^{\mathbf{f}}}$ by averaging the weighted expected centers from all the English words in $\delta^{\mathbf{e}}$ as in Eqn. 9.

$$\begin{aligned} E\{\odot_{\delta^{\mathbf{f}}}|\delta^{\mathbf{e}}, \mathbf{e}\} &= \textstyle\sum_{j=0}^{J} j \cdot P(j|\delta^{\mathbf{e}}, \mathbf{e}) \\ &\simeq \textstyle\sum_{j=0}^{J} j \cdot \frac{\sum_{i \in \delta^{\mathbf{e}}} P(f_j|e_i)}{\sum_{j'=0}^{J} \sum_{i \in \delta^{\mathbf{e}}} P(f_{j'}|e_i)}. \end{aligned} \tag{9}$$

The expectations of $(\odot_{\delta^{\mathbf{f}}}, w_{\delta^{\mathbf{f}}})$ from Eqn. 8 and Eqn. 9 give a reliable starting point for a local search for the optimal estimation of $(\hat{\odot}_{\delta^{\mathbf{f}}}, \hat{w}_{\delta^{\mathbf{f}}})$ as in Eqn 10:

$$(\hat{\odot}_{\delta^{\mathbf{f}}}, \hat{w}_{\delta^{\mathbf{f}}}) = \underset{\{(\odot_{\delta^{\mathbf{f}}}, w_{\delta^{\mathbf{f}}})\}}{\arg\max} P(\delta^{\mathbf{f}}_{\in}|\delta^{\mathbf{e}}_{\in}) P(\delta^{\mathbf{f}}_{\notin}|\delta^{\mathbf{e}}_{\notin}), \tag{10}$$

where the score functions of $P(\delta^{\mathbf{f}}_{\in}|\delta^{\mathbf{e}}_{\in})P(\delta^{\mathbf{f}}_{\notin}|\delta^{\mathbf{e}}_{\notin})$ are in Eqn. 4 with the word alignment explicitly given from the previous iteration. For the very first iteration, *full alignment* is assumed; this means that every word pair is connected in the parallel sentences. During the local search in Eqn. 10, one can choose the top-1 (Viterbi) $(\hat{\odot}_{\delta^{\mathbf{f}}}, \hat{w}_{\delta^{\mathbf{f}}})$ or top-N candidates and normalize over these candidates to obtain the posteriors. Except for the local search of $(\hat{\odot}_{\delta^{\mathbf{f}}}, \hat{w}_{\delta^{\mathbf{f}}})$, the remainder EM steps are similar to Bracket Model-A, though with different interpretations.

By performing local search in Eqn. 10, Model-B localizes hidden blocks more accurately than the scheme of the smoothed relative frequency in Model-A's EM iterations. The model is also more focused on the predictions in the inner part. Figure 3 summarizes the generative process of Model-B (BM-B).

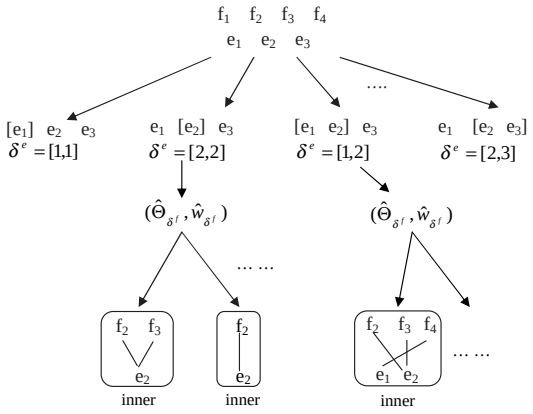

Figure 3: Generative Bracket Model-B

3.3 A Null Word Model

The null word model allows words to be aligned to nothing. In the traditional IBM models, there is a universal null word which is attached to every sentence pair to compete with word generators. In our inner-outer bracket models, we use two context-specific null word models which use both the left and right context as competitors in the generative process for each observation f_j: $P(f_j|f_{j-1}, \mathbf{e})$ and $P(f_j|f_{j+1}, \mathbf{e})$. This is similar to the approach in (Toutanova et al., 2002), in which the null word model is part of an extended HMM using left context only. With two null word models, we can associate f_j with its left or right context (i.e., a null link) when the null word models are very strong, or when the word's alignment is too far from the expected center $\hat{\odot}_{\delta^{\mathbf{f}}}$ in Eqn. 9.

4 A Max-Posterior for Word Alignment

In the HMM framework, (Ge, 2004) proposed a *maximum-posterior* method which worked much better than *Viterbi* for Arabic to English translations. The difference between maximum-posterior and Viterbi, in a nutshell, is that while Viterbi computes the best state *sequence* given the observation, the maximum-posterior computes the best state *one at a time*.

In the terminology of HMM, let the states be the words in the foreign sentence f_1^J and observations be the words in the English sentence e_1^T. We use the subscript t to note the fact that e_t is observed (or emitted) at time step t. The posterior probabilities $P(f_j|e_t)$ (state given observation) are obtained after the forward-backward training. The maximum-posterior word alignments are obtained by first com-

puting a pair $(j,t)^*$:

$$(j,t)^* = \arg\max_{(j,t)} P(f_j|e_t), \quad (11)$$

that is, the point at which the posterior is maximum. The pair (j,t) defines a word pair (f_j, e_t) which is then aligned. The procedure continues to find the next maximum in the posterior matrix. Contrast this with Viterbi alignment where one computes

$$\hat{f}_1^T = \arg\max_{\{f_1^T\}} P(f_1, f_2, \cdots, f_T|e_1^T), \quad (12)$$

We observe, in parallel corpora, that when one word translates into multiple words in another language, it usually translates into a contiguous sequence of words. Therefore, we impose a contiguity constraint on word alignments. When one word f_j aligns to multiple English words, the English words must be contiguous in **e** and vice versa. The algorithm to find word alignments using max-posterior with contiguity constraint is illustrated in Algorithm 1.

Algorithm 1 A maximum-posterior algorithm with contiguity constraint

```
while (j,t) = (j,t)* (as computed in Eqn. 11) do
    if (f_j, e_t) is not yet aligned then
        align(f_j, e_t);
    else if (e_t is contiguous to what f_j is aligned)
        or (f_j is contiguous to what e_t is aligned) then
        align(f_j, e_t);
    end if
end while
```

The algorithm terminates when there isn't any 'next' posterior maximum to be found. By definition, there are at most JxT 'next' maximums in the posterior matrix. And because of the contiguity constraint, not all (f_j, e_t) pairs are valid alignments. The algorithm is sure to terminate. The algorithm is, in a sense, *directionless*, for one f_j can align to multiple e_t's and vise versa as long as the multiple connections are contiguous. Viterbi, however, is *directional* in which one state can emit multiple observations but one observation can only come from one state.

5 Experiments

We evaluate the performances of our proposed models in terms of word alignment accuracy and translation quality. For word alignment, we have 260 hand-aligned sentence pairs with a total of 4676 word pair links. The 260 sentence pairs are randomly selected from the CTTP[1] corpus. They were then word aligned by eight bilingual speakers. In this set, we have one-to-one, one-to-many and many-to-many alignment links. If a link has one target *functional* word, it is considered to be a *functional* link (Examples of funbctional words are prepositions, determiners, etc. There are in total 87 such functional words in our experiments). We report the overall F-measures as well as F-measures for both content and functional word links. Our significance test shows an overall interval of ±1.56% F-measure at a 95% confidence level.

For training data, the small training set has 5000 sentence pairs selected from XinHua news stories with a total of 131K English words and 125K Chinese words. The large training set has 181K sentence pairs (5k+176K); and the additional 176K sentence pairs are from FBIS and Sinorama, which has in total 6.7 million English words and 5.8 million Chinese words.

5.1 Baseline Systems

The baseline is our implementation of HMM with the maximum-posterior algorithm introduced in section 4. The HMMs are trained unidirectionally. IBM Model-4 is trained with GIZA++ using the best reported settings in (Och and Ney, 2003). A few parameters, especially the maximum fertility, are tuned for GIZA++'s optimal performance. We collect *bi-directional* (**bi**) refined word alignment by growing the intersection of *Chinese-to-English* (**CE**) alignments and *English-to-Chinese* (**EC**) alignments with the neighboring unaligned word pairs which appear in the union similar to the "final-and" approaches (Koehn, 2003; Och and Ney, 2003; Tillmann, 2003). Table 1 summarizes our baseline with different settings. Table 1 shows that **HMM EC-P** gives the

F-measure(%)		**Func**	**Cont**	**Both**
Small	**HMM EC-P**	**54.69**	**69.99**	**64.78**
	HMM EC-V	31.38	53.56	55.59
	HMM CE-P	51.44	69.35	62.69
	HMM CE-V	31.43	63.84	55.45
Large	**HMM EC-P**	**60.08**	78.01	**71.92**
	HMM EC-V	32.80	74.10	64.26
	HMM CE-P	58.45	**79.44**	71.84
	HMM CE-V	35.41	79.12	68.33
Small	GIZA MH-bi	45.63	69.48	60.08
	GIZA M4-bi	48.80	73.68	63.75
Large	GIZA MH-bi	49.13	76.51	65.67
	GIZA M4-bi	52.88	81.76	70.24
-	Fully-Align [2]	5.10	15.84	9.28

Table 1: Baseline: **V**: **V**iterbi; **P**: Max-**P**osterior

[1] LDC2002E17

best baseline, better than bidirectional refined word alignments from GIZA M4 and the HMM Viterbi aligners.

5.2 Inner-Outer Bracket Models

We trained HMM lexicon $P(f|e)$ to initialize the inner-outer Bracket models. Afterwards, up to 15–20 EM iterations are carried out. Iteration starts from the fully aligned[2] sentence pairs, which give an F-measure of 9.28% at iteration one.

5.2.1 Small Data Track

Figure 4 shows the performance of Model-A (BM-A) trained on the small data set. For each English bracket, *Top-1* means only the fractional counts from the Top-1 projection are collected, *Top-all* means counts from all possible projections are collected. *Inside* means the fractional counts are collected from the inner part of the block only; and *outside* means they are collected from the outer parts only. Using the Top-1 projection from the inner parts of the block (*top-1-inside*) gives the best performance: an F-measure of 72.29%, or a 7.5% absolute improvement over the best baseline at iteration 5. Figure 5 shows

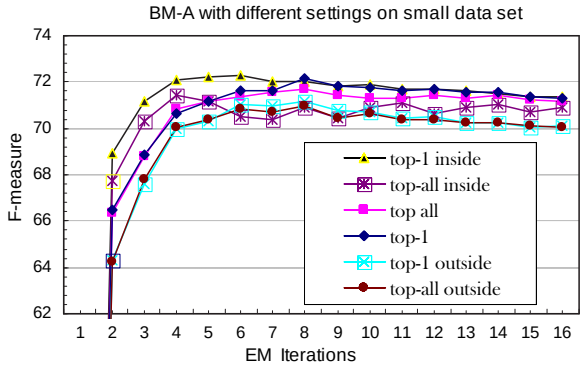

Figure 4: BM-A with different settings on small data

the performance of Inner-Outer Bracket Model-B (BM-B) over EM iterations. *smoothing* means when collecting the fractional counts, we reweigh the updated fractional count by 0.95 and give the remaining 0.05 weight to original fractional count from the links, which were aligned in the previous iteration. *w/null* means we applied the proposed Null word model in section 3.3 to infer null links. We also predefined a list of 15 English function words, for which there might be no corresponding Chinese words as translations. These 15 English words are "*a, an, the, of, to, for, by, up, be, been, being, does, do, did, -*". In the *drop-null* experiments, the links containing these predefined function words are simply dropped in the final word alignment (this means they are left unaligned).

[2]Every possible word pair is aligned

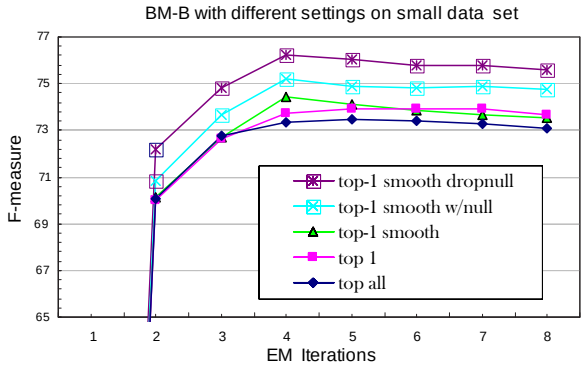

Figure 5: BM-B with different settings on small data

Empirically we found that doing more than 5 iterations lead to overfitting. The peak performance in our model is usually achieved around iteration 4∼5. At iteration 5, setting "BM-B Top-1" gives an F-measure of 73.93% which is better than BM-A's best performance (72.29%). This is because Model B leverages a local search for less noisy blocks and hence the inner part is more accurately generated (which in turn means the outer part is also more accurate). From this point on, all of our experiments are using Model B. With smoothing, BM-B improves to 74.46%. After applying the null word model, we get 75.20%. By simply dropping links containing the 15 English functional words, we get 76.24%, which is significantly better than our best baseline obtained from even the large training set (HMM EC-P: 71.92%).

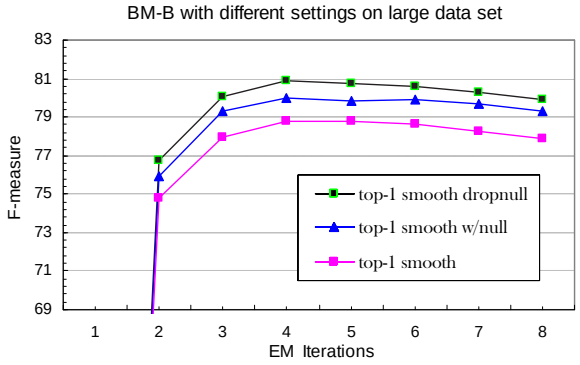

Figure 6: BM-B with different settings on large data

5.2.2 Large Data Track

Figure 6 shows performance pictures of model BM-B on the large training set. Without dropping English functional words, the best performance is

80.38% at iteration 4 using the Top-1 projection together with the null word models. By additionally dropping the links containing the 15 functional English words, we get 81.47%. These results are all significantly better than our strongest baseline system: 71.92% F-measure using HMM EC-P (70.24% using bidirectional Model-4 for comparisons).

On this data set, we experimented with different maximum bracket length limits, from one word (unigram) to nine-gram. Results show that a maximum bracket length of four is already optimal (79.3% with top-1 projection), increased from 62.4% when maximum length is limited to one. No improvements are observed using longer than five-gram.

5.3 Evaluate Blocks in the EM Iterations

Our intuition was that good blocks can improve word alignment and, in turn, good word alignment can lead to better block selection. The experimental results above support the first claim. Now we consider the second claim that good word alignment leads to better block selection.

Given reference human word alignment, we extract reference blocks up to five-gram phrases on Chinese. The block extraction procedure is based on the procedures in (Tillmann, 2003).

During EM, we output all the hidden blocks actually inferred at each iteration, then we evaluate the precision, recall and F-measure of the hidden blocks according to the extracted reference blocks. The results are shown in Figure 7. Because we extract all

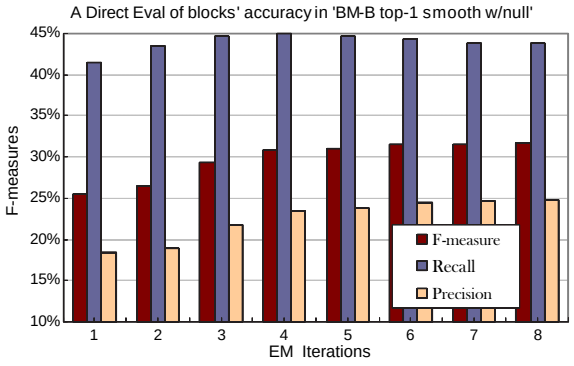

Figure 7: A Direct Eval. of Blocks in BM-B

possible n-grams at each position in **e**, the precision is low and the recall is relatively high as shown by Figure 7. It also shows that blocks do improve, presumably benefiting from better word alignments.

Table 2 summarizes word alignment performances of Inner-Outer BM-B in different settings. Overall, without the handcrafted function word list, BM-B gives about 8% absolute improvement in F-measure on the large training set and 9% for the small set

F-measure(%)		**Func**	**Cont**	**Both**
Small	Baseline	54.69	69.99	64.78
	BM-B-drop	**62.76**	**82.99**	**76.24**
	BM-B w/null	61.24	82.54	75.19
	BM-B smooth	59.61	82.99	74.46
Large	Baseline	60.08	78.01	71.92
	BM-B-drop	**63.95**	**90.09**	**81.47**
	BM-B w/null	62.24	89.99	80.38
	BM-B smooth	60.49	90.09	79.31

Table 2: BM-B with different settings

with a confidence interval of ±1.56%.

5.4 Translation Quality Evaluations

We also carried out the translation experiments using the best settings for Inner-Outer BM-B (i.e. *BM-B-drop*) on the TIDES Chinese-English 2003 test set. We trained our models on 354,252 test-specific sentence pairs drawn from LDC-supplied parallel corpora. On this training data, we ran 5 iterations of EM using BM-B to infer word alignments. A monotone decoder similar to (Tillmann and Ney, 2003) with a trigram language model[3] is set up for translations. We report *case sensitive Bleu* (Papineni et al., 2002) score **BleuC** for all experiments. The baseline system (*HMM*) used phrase pairs built from the HMM-EC-P maximum posterior word alignment and the corresponding lexicons. The baseline BleuC score is 0.2276 ± 0.015. If we use the phrase pairs built from the bracket model instead (but keep the HMM trained lexicons), we get case sensitive BleuC score 0.2526. The improvement is statistically significant. If on the other hand, we use baseline phrase pairs with bracket model lexicons, we get a BleuC score 0.2325, which is only a marginal improvement. If we use *both* phrase pairs and lexicons from the bracket model, we get a case sensitive BleuC score 0.2750, which is a statistically significant improvement. The results are summarized in Table 3.

Settings	**BleuC**
Baseline (HMM phrases and lexicon)	0.2276
Bracket phrases and HMM lexicon	0.2526
Bracket lexicon and HMM phrases	0.2325
Bracket (phrases and lexicon)	0.2750

Table 3: Improved *case sensitive* BleuC using BM-B

Overall, using Model-B, we improve translation quality from 0.2276 to 0.2750 in case sensitive BleuC score.

[3]Trained on 1-billion-word ViaVoice English data; the same data is used to build our True Caser.

6 Conclusion

Our main contributions are two novel Inner-Outer Bracket models based on segmentations induced by hidden blocks. Modeling the Inner-Outer hidden segmentations, we get significantly improved word alignments for both the small training set and the large training set over the widely-practiced bidirectional IBM Model-4 alignment. We also show significant improvements in translation quality using our proposed bracket models. Robustness to noisy blocks merits further investigation.

7 Acknowledgement

This work is supported by DARPA under contract number N66001-99-28916.

References

P.F. Brown, Stephen A. Della Pietra, Vincent. J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. In *Computational Linguistics*, volume 19(2), pages 263–331.

Niyu Ge. 2004. A maximum posterior method for word alignment. In *Presentation given at DARPA/TIDES MT workshop.*

J.X. Huang, W. Wang, and M. Zhou. 2003. A unified statistical model for generalized translation memory system. In *Machine Translation Summit IX*, pages 173–180, New Orleans, USA, September 23-27.

Philipp Koehn and Kevin Knight. 2002. Chunkmt: Statistical machine translation with richer linguistic knowledge. Draft, Unpublished.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based machine translation. In *Proc. of HLT-NAACL 2003*, pages 48–54, Edmonton, Canada, May-June.

Philipp Koehn. 2003. Noun phrase translation. In *Ph.D. Thesis*, University of Southern California, ISI.

Daniel Marcu and William Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In *Proc. of the Conference on Empirical Methods in Natural Language Processing*, pages 133–139, Philadelphia, PA, July 6-7.

Franz J. Och and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of the 40th Annual Meeting of ACL*, pages 440–447.

Franz J. Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. In *Computational Linguistics*, volume 29, pages 19–51.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proc. of the 40th Annual Conf. of the ACL (ACL 02)*, pages 311–318, Philadelphia, PA, July.

Christoph Tillmann and Hermann Ney. 2003. Word reordering and a dp beam search algorithm for statistical machine translation. In *Computational Linguistics*, volume 29(1), pages 97–133.

Christoph Tillmann. 2003. A projection extension algorithm for statistical machine translation. In *Proc. of the Conference on Empirical Methods in Natural Language Processing.*

Kristina Toutanova, H. Tolga Ilhan, and Christopher D. Manning. 2002. Extensions to hmm-based statistical word alignment models. In *Proc. of the Conference on Empirical Methods in Natural Language Processing*, Philadelphia, PA, July 6-7.

S. Vogel, Hermann Ney, and C. Tillmann. 1996. Hmm based word alignment in statistical machine translation. In *Proc. The 16th Int. Conf. on Computational Lingustics, (Coling'96)*, pages 836–841, Copenhagen, Denmark.

Taro Watanabe, Kenji Imamura, and Eiichiro Sumita. 2002. Statistical machine translation based on hierarchical phrases. In *9th International Conference on Theoretical and Methodological Issues*, pages 188–198, Keihanna, Japan, March.

Taro Watanabe, Eiichiro Sumita, and Hiroshi G. Okuno. 2003. Chunk-based statistical translation. In *In 41st Annual Meeting of the ACL (ACL 2003)*, pages 303–310, Sapporo, Japan.

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

K. Yamada and Kevin. Knight. 2001. Syntax-based statistical translation model. In *Proceedings of the Conference of the Association for Computational Linguistics (ACL-2001).*

R. Zens and H. Ney. 2004. Improvements in phrase-based statistical machine translation. In *Proceedings of the Human Language Technology Conference (HLT-NAACL)s*, pages 257–264, Boston, MA, May.