CLSP Research Note No. 48

A Weighted Finite State Transducer Translation Template Model for Statistical Machine Translation

Shankar Kumar and William Byrne
Center for Language and Speech Processing,
Department of Electrical and Computer Engineering,
The Johns Hopkins University,
3400 N. Charles St., Baltimore, MD 21218, USA
{skumar,byrne}@jhu.edu

August 29, 2004

Contents

Abstract

We present a Weighted Finite State Transducer Translation Template Model for statistical machine translation. This is a source-channel model of translation inspired by the Alignment Template translation model. The model attempts to overcome the deficiencies of word-to-word translation models by considering phrases rather than words as units of translation. The approach we describe allows us to implement each constituent distribution of the model as a weighted finite state transducer or acceptor. We show that bitext word alignment and translation under the model can be performed with standard finite state machine operations involving these transducers. One of the benefits of using this framework is that it avoids the need to develop specialized search procedures, even for the generation of lattices or N-Best lists of bitext word alignments and translation hypotheses. We report and analyze bitext word alignment and translation performance on the Hansards French-English task and the NIST Chinese-English tasks under the Alignment Error Rate, BLEU, NIST and Word Error-Rate metrics. These experiments identify the contribution of each of the model components to different aspects of alignment and translation performance.

1 Introduction

Statistical machine translation originated with the pioneering work at IBM [3, 4] in modeling the movement and translation of words in bitext alignment. There has subsequently been considerable effort devoted to improving the IBM models themselves [29, 20] and developing improved translation search algorithms based on those models [30, 10, 26, 22, 9]. There also have been advancements in the understanding of the nature of these models, notably, due to the work by Knight and Al-Onaizan [10] that describes how Weighted Finite State Transducers (WFSTs) can be used to perform translation using the IBM models, albeit in slightly modified form. In addition to the efficiencies in computation that can be obtained using WFSTs, that formulation provides an accessible, intuitive description of IBM models 1 through 3. Motivated by this work, we developed WFST-based alignment bitext word alignment algorithms and used them under various alignment criteria [12] . However these applications were restricted in power by their reliance on the IBM-3 model, which is the most complex of the IBM models that can easily be treated as a WFST.

The IBM-3 model appears particularly weak in comparison to the Alignment Template Model developed by Och, Tillmann, and Ney [21], which attempts to overcome the limitations of IBM-style word-to-word translation models by considering whole phrases rather than words as the basis for translation. Under this model, a phrase in the target language (e.g. French) would be translated to a phrase in the source language (e.g. English), and the basic unit of this model is an *alignment template* that specifies the allowable word alignments within a pair of source and target phrases. In our first attempt to implement this model directly using WFSTs we developed a formulation within which each of the component models can be implemented as a weighted finite state transducer [13]. In doing so, we also generalized the model to support bitext word alignment. That implementation provided a working translation system that we used as a basis for the Chinese-to-English translation system [5] submitted in the NIST 2003 MT evaluations [18]. However, it was flawed in how it incorporated the source language model and in its treatment of phrase insertions and deletions in bitext word alignment. These shortcomings provided motivation for this current work.

We describe a source-channel model of translation inspired by the WFST implementation of

the Alignment Template Model. We have two objectives in doing so. First, by following a careful source-channel formulation we can be certain that all components of the model are correctly implemented. Secondly, each component of the overall model is constructed so that translation and bitext word alignment can be carried out using standard WFST operations.

Our current model departs from the original Alignment Template Model [21, 19] in several ways. In addition to the new formulation of the overall statistical model, the components of the model do not make use of the word alignments within the alignment templates; we model only the translation of phrases. This does not prevent using the model in bitext word alignment, however, and we describe how this can be done. We furthermore allow insertions of target language phrases in the generative translation process; this removes the restriction that the source and target language sentences contain the same number of phrases. To avoid confusion with previous work, we call this model the *Translation Template Model* (TTM), leaving out the reference to word alignment within phrases. In this paper we present the Translation Template Model and show how it can be implemented component-wise using WFSTs.

The recent developments in statistical translation have been accompanied by progress in the automatic evaluation of alignment and translation performance using metrics such as Alignment Error Rate (AER) [20], BLEU [23], NIST [8], and multi-reference Word Error Rate [19]. Like others, we have found these metrics to be extremely valuable; the development of statistical models on this scale would be impossible without fast, inexpensive evaluation metrics. We present extensive experiments analyzing the translation performance of our overall system. Our aim is to identify the contribution of each of the model components to different aspects of translation performance. In doing so, we also analyze some aspects of the performance metrics themselves; these criteria are complex enough that they have behavior of their own. We also study the influence of the bitext used to train the system. The quality of and the amount of available bitext has a strong influence on the quality of the statistical models that result, and we provide an analysis of the influence of both quality and quantity on translation performance under the TTM.

We acknowledge other recent and related work in developing phrase-based models for statistical machine translation. In particular, there are new techniques available for extracting phrase pairs from bitext, either using underlying word alignments [27, 11] or not [32, 16]. Bangalore and Ricardi [2] have also explored the use of WFSTs for machine translation. They implement a two-step translation process in which the foreign sentence is first mapped to an English word sequence, but in foreign word order; that string is then reordered into English word order. Both processing steps are implemented by WFSTs and the overall approach has been applied in a call-routing task. While related in its use of WFSTs for translation, our work (and that of Knight and Al-Onaizan [10]) differs in spirit from Bangalore et al. in that we are mainly focused on the formulation of a source-channel model of translation and its implementation via WFSTs.

The paper is organized as follows. In Section 2 we present the derivation of the overall translation model that identifies the conditional independence assumptions among the component variables. In Section 3 we describe phrase-pair inventories and their extraction from aligned bitext. The TTM has six component models, and we discuss each along with its WFST implementation in Section 4. In Section 5, we show how bitext word alignment and translation can be performed with standard FSM operations involving these transducers. In Section 6 we report and analyze bitext word alignment and translation performance on French-English and Chinese-English tasks. We discuss these experiments in Section 7, and conclude in Section 8.

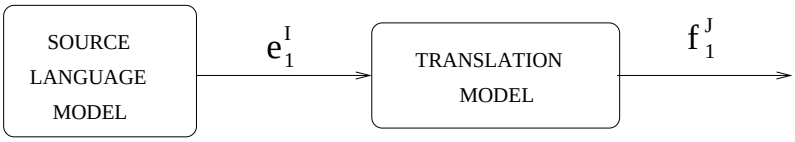

Figure 1: A Source Channel Model of Machine Translation.

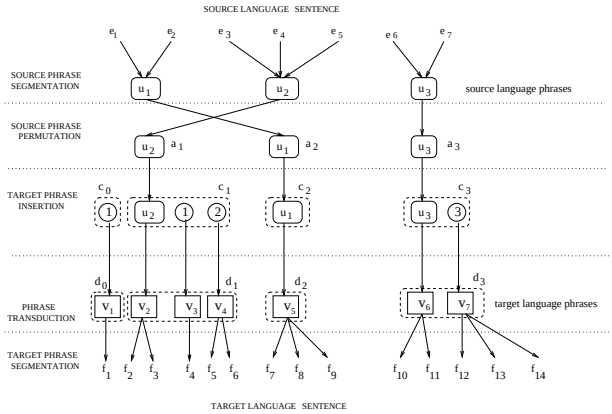

Figure 2: Translation Process underlying the TTM by which translation is modeled as a transformation of a source language sentence into a target language sentence. Conditional dependencies underlying the process are given in Equation 1.

2 The Translation Template Model

We present here a derivation of the Translation Template model, and give an implementation of the model using Weighted Finite State Transducers.

The TTM is a source-channel model of translation (Figure 1) [3] with a joint probability distribution over all possible segmentations and alignments of target language sentences and their translations in the source language. The translation process is presented in Figure 2, and the conditional dependencies underlying this process are presented in Equation 1. Each of the conditional distributions that make up the model is realized independently. In Section 4 we define each in turn and present its implementation as a weighted finite state acceptor or transducer.

$$
\begin{aligned}
P(f_1^J, v_1^R, d_0^K, c_0^K, a_1^K, u_1^K, K, e_1^I) \;=\; & \\
& P(e_1^I)\cdot && \textit{Source Language Model} \\
& P(u_1^K, K|e_1^I)\cdot && \textit{Source Phrase Segmentation} \\
& P(a_1^K|u_1^K, K, e_1^I)\cdot && \textit{Phrase Order} \\
& P(c_0^K|a_1^K, u_1^K, K, e_1^I)\cdot && \textit{Target Phrase Insertion} \\
& P(v_1^R, d_0^K|c_0^K, a_1^K, u_1^K, K, e_1^I) && \textit{Phrase Transduction} \\
& P(f_1^J|v_1^R, d_0^K, c_0^K, a_1^K, u_1^K, K, e_1^I)\cdot && \textit{Target Phrase Segmentation}
\end{aligned}
\tag{1}
$$

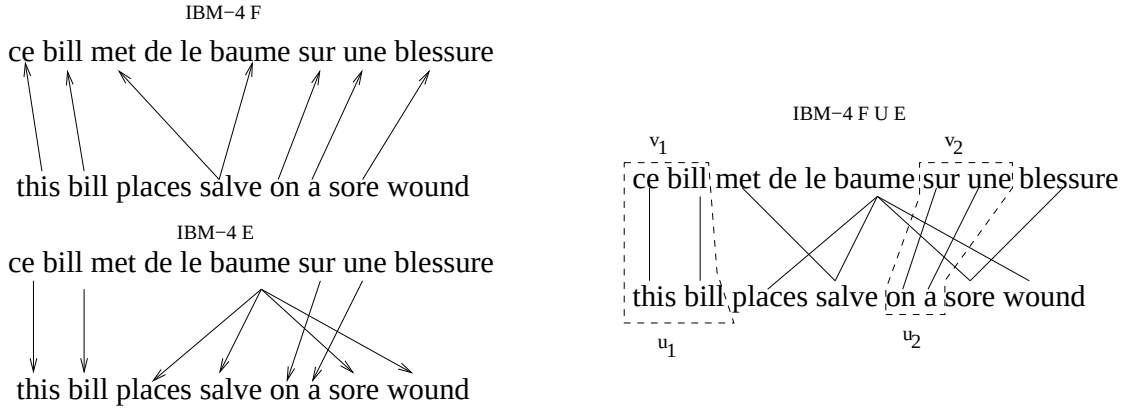

Figure 3: Phrase-Pair Collection Process from Bidirectional word alignments of an English-French sentence pair.

We begin by distinguishing words and phrases. We assume that u is a phrase in the source language sentence that consists of a variable number of words $e_1, e_2, ..., e_M$. Similarly, v is a phrase in the target language sentence of words $f_1, f_2, ..., f_N$. Throughout the model, if an I word sentence e_1^I is segmented into K phrases u_1^K, we say $u_1^K = e_1^I$ to indicate that the words in the phrase sequence are those of the original sentence.

3 The Phrase-Pair Inventory

The Translation Template Model relies on an inventory of target language phrases and their source language translations. These translations need not be unique, in that multiple translations of phrases in either language are allowed. The manner by which the inventory is created does not affect our formulation. For the experiments that will be presented in this paper, we utilize the *phrase-extract* algorithm [19] to extract a library of phrase-pairs from bitext word alignments. We first obtain word alignments of bitext using IBM-4 word level translation models [4] trained in both translation directions (IBM-4 F and IBM-4 E) , and then form the union of these alignments (IBM-4 $E \cup F$). We will refer to these initial models as the *underlying models*. We next use the algorithm to identify pairs of phrases (u, v) in the target and source language that align well according to a set of heuristics [19]. To restrict the memory requirements of the model, we extract only the phrase-pairs which have at most 5 words in the target phrase. In Figure 3, we show the extraction of phrase-pairs from bidirectional word alignments of an English-French sentence pair. For each pair of target and source phrases, we retain the matrix of word alignments that occurs most frequently in the training corpus. We augment this inventory by the most likely translations of each target (source) word from the IBM-4 translation tables [4] so as to get complete coverage of all single word phrases in either language. We note that monolingual phrase inventories can be created by projecting the phrase-pairs onto the target or the source language.

4 TTM Component Models

We now introduce the definitions of the component distributions of the Translation Template Model in Equation 1. In presenting these, we first define the component probability distribution, and then describe its implementation using a Weighted Finite State Transducer or an Acceptor.

4.1 Source Language Model

We specify this model using a standard monolingual trigram word language model

$$P(e_1^I) = \prod_{i=1}^{I} P(e_i|e_{i-1}, e_{i-2}).$$

Any n-gram or other language model that can be easily compiled as a weighted finite state acceptor could be used [1].

4.2 Source Phrase Segmentation Model

We construct a joint distribution over all phrase segmentations $u_1^K = u_1, u_2, ..., u_K$ of the source sentence e_1^I as

$$P(u_1^K, K|e_1^I) \quad = \quad P(u_1^K|K, e_1^I)P(K|I). \tag{2}$$

We choose the distribution over the number of phrases $P(K|I)$ to be uniform

$$P(K|I) = \frac{1}{I}; K \in \{1, 2, ..., I\}. \tag{3}$$

For a given number of phrases, the segmentation model is a uniform distribution over the set of K-length phrase sequences of e_1^I

$$P(u_1^K|K, e_1^I) \quad = \quad \begin{cases} C & u_1^K = e_1^I \text{ and } u_i, i \in \{1, 2, ..., K\} \text{ belongs to the source phrase inventory} \\ 0 & \text{otherwise,} \end{cases}$$

and we renormalize the above model so that $\sum_{u_1^K} P(u_1^K|K, e_1^I) = 1$. In summary, this distribution assigns a uniform likelihood to all phrase segmentations of the source sentence that can be obtained using the phrase inventory.

The WFST implementation of the Source Phrase Segmentation model involves an unweighted segmentation transducer W that maps source word sequences to source phrase sequences. The transducer performs the mapping of source word strings to phrases for every source phrase in our inventory. A portion of the segmentation transducer W is presented in Figure 4. The "_" symbol is used to indicate phrases formed by concatenation of consecutive words.

We now describe the procedure to construct a WFST for the distribution $P(u_1^K|K, e_1^I)$. In particular we must ensure that $\sum_{u_1^K} P(u_1^K|K, e_1^I) = 1$ for each source sentence e_1^I and $K \in \{1, 2, ..., I\}$.

1. We build a finite state word acceptor T for the source sentence e_1^I. We then generate a transducer of segmentations of e_1^I by composing T with W, i.e. $\mathcal{U} = T \circ W$.

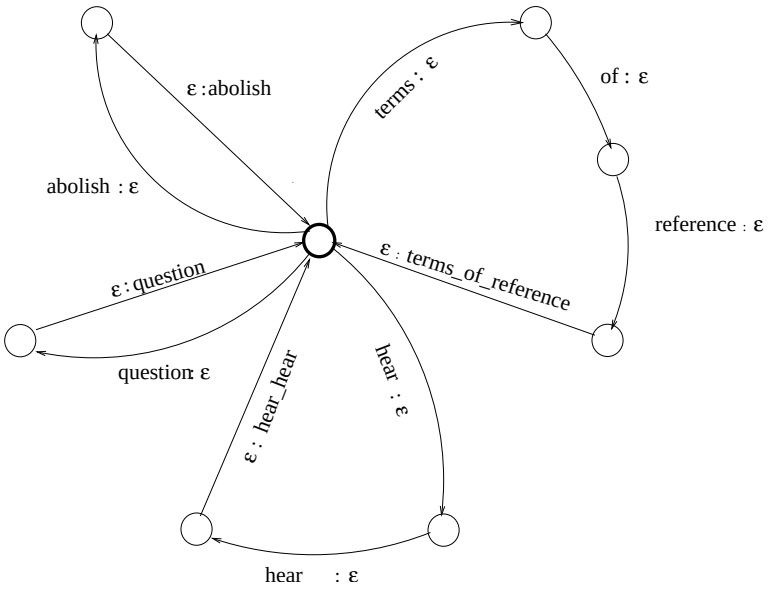

Figure 4: A portion of the Source Phrase Segmentation Transducer W that maps word sequences to phrases. There is a distinct loop for each phrase in the source language phrase inventory. Suppose an example input for this transducer is the source language sentence: *What are its terms of reference*, then a possible output of WFST would be the source language phrase sequence: *what_are its terms_of_reference*.

2. The transducer $\mathcal{U}$ can be partitioned into I disjoint transducers $\mathcal{U}_K$ so that $\cup_{K=1}^{I}\mathcal{U}_K = \mathcal{U}$; each $\mathcal{U}_K$ consists of those segmentations of the source sentence with exactly K phrases. To construct $\mathcal{U}_K$, we create an unweighted acceptor P_K that accepts any phrase sequence of length K; for efficiency, the phrase vocabulary is restricted to the phrases in $\mathcal{U}$. $\mathcal{U}_K$ is then obtained by the finite state composition: $\mathcal{U}_K = \mathcal{U} \circ P_K$.

3. For $K = 1, 2, ..., J$
   Obtain the total number of distinct paths C_K in $\mathcal{U}_K$. This can be computed efficiently using lattice forward probabilities [31]. Set the probability of each path to $\frac{1}{C_K}\frac{1}{I}$ to obtain a new transducer $\mathcal{U'}_K$.

4. Construct a new segmentation lattice $\mathcal{U'} = \cup_{K=1}^{I}\mathcal{U'}_K$.

The segmentation lattice $\mathcal{U'}$ obtained through the above procedure will be normalized so that probabilities of all segmentations of a given length would sum up to one, i.e. $\sum_{u_1^K} P(u_1^K|K, e_1^I) = 1; K \in \{1, 2, ..., I\}$.

We emphasize that these forms of the segmentation distribution are exceedingly simple and were chosen for ease of presentation. More complex phrase segmentation models can easily be implemented in this framework.

4.3 Phrase Order Model

We now define a model for the reordering of the source phrase sequences that make up the source sentence. The phrase alignment sequence a_1^K specifies a reordering of source phrases into *target language phrase order*; note that the words within the phrases remain in the original order. In this way the phrase sequence u_1^K is reordered into $u_{a_1}, u_{a_2}, ..., u_{a_K}$ under the model $P(a_1^K|u_1^K, K, e_1^I)$. We now discuss several phrase order models.

4.3.1 Markov Phrase Order Model

The phrase alignment sequence is modeled as a first order Markov process

$$\begin{aligned} P(a_1^K|u_1^K, K, e_1^I) &= P(a_1^K|u_1^K) \\ &= P(a_1)\prod_{k=2}^{K} P(a_k|a_{k-1}, u_1^K). \end{aligned} \tag{4}$$

with $a_k \in \{1, 2, ..., K\}$. The alignment sequence distribution is constructed to assign lower likelihood to phrase re-orderings that diverge from the original word order. Suppose $u_{a_k} = e_l^{l'}$ and $u_{a_{k-1}} = e_m^{m'}$, we set the Markov chain probabilities as follows [21]

$$\begin{aligned} P(a_k|a_{k-1}, u_1^K) &\propto p_0^{|l-m'-1|} \\ P(a_1 = k) &= \frac{1}{K}; k \in \{1, 2, ..., K\}. \end{aligned} \tag{5}$$

In the above equations, p_0 is a tuning factor and we normalize the probabilities $P(a_k|a_{k-1})$ so that $\sum_{j=1, j\neq a_{k-1}}^{K} P(a_k = j|a_{k-1}) = 1$.

The finite state implementation of the phrase order model involves two acceptors. We first build a unweighted permutation acceptor Π_U that contains all reorderings of the source language phrase sequence u_1^K [10]. We note that a path through Π_U corresponds to an alignment sequence a_1^K. Figure 5 shows the acceptor Π_U for the source phrase sequence *we have run_away_inflation*.

A source phrase sequence U of length K words requires a permutation acceptor Π_U of 2^K states. For long phrase sequences we compute a score $\max_j P(a_k = i|a_{k-1} = j)$ for each arc and then prune the arcs by this score, i.e. phrase alignments containing $a_k = i$ are included only if this score is above a threshold. Pruning can therefore be applied while Π_U is constructed.

The second acceptor H in the implementation of the Phrase Order Model assigns alignment probabilities (Equation 5) to a given reordering a_1^K of the source phrase sequence u_1^K (Figure 6). In this example, the phrases in the source phrase sequence are specified as follows: $v_1 = f_1$ (*we*), $v_2 = f_2$ (*have*) and $v_3 = f_3^5$ (*run_away_inflation*). We now show the computation of some of the alignment probabilities (Equation 5) in this example ($p_0 = 0.9$)

$$\begin{aligned} P(a_3 = 1|a_2 = 3) &\propto p_0^{|1-5-1|} = 0.59 \\ P(a_3 = 2|a_2 = 3) &\propto p_0^{|2-5-1|} = 0.66. \end{aligned}$$

Normalizing these terms gives $P(a_3 = 1|a_2 = 3) = 0.47$ and $P(a_3 = 2|a_2 = 3) = 0.53$.

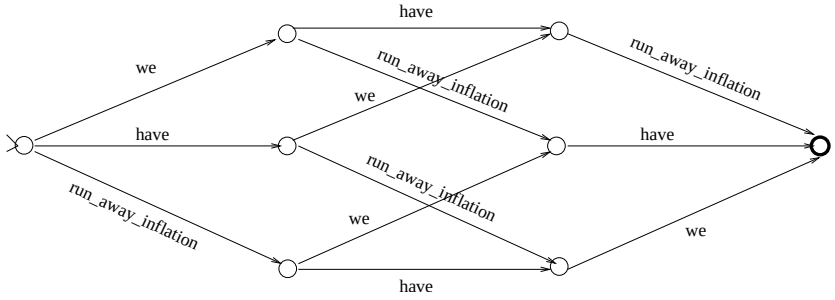

Figure 5: The permutation acceptor Π_U for the source-language phrase sequence *we have run_away_inflation*. For this phrase sequence, an example of a reordering allowed by this acceptor is *run_away_inflation we have*, so that the alignment sequence is given by: $a_1 = 3, a_2 = 1, a_3 = 2$.

4.3.2 Practical Phrase Order Models

The permutation acceptor described above must be constructed for each segmentation u_1^K of the source sentence e_1^I. As a source sentence typically has several segmentations, it is infeasible to construct a separate permutation acceptor for every segmentation. Moreover, during decoding, this process has be carried out for every source sentence that is allowable by the source language model. As a practical approximation, we therefore consider a degenerate model that does not allow any reordering of the source phrase sequence u_1^K. Therefore the model would be specified as

$$P(a_1^K|u_1^K, K, e_1^I) = \begin{cases} 1 & \{a_1 = 1, a_2 = 2, a_3 = 3, ..., a_K = K\} \\ 0 & \text{otherwise.} \end{cases} \quad (6)$$

We will refer to this model as the *Fixed Phrase Order Model*.

4.4 Target Phrase Insertion Model

The processes described thus far allow a mapping of a source language sentence into a reordered sequence of source language phrases, whose order is the phrase order of the target language. The constraint that the target language phrase sequence have the same number of phrases as this source language phrase sequence is overly restrictive. Our goal is to construct a model to allow insertion of target language phrases anywhere in the reordered source language phrase sequence. This process will be governed by a probability distribution over insertion of target language phrases such that the likelihood of inserting a phrase is inversely proportional to the number of words in the phrase. Therefore there will be a greater penalty for the insertion of longer phrases.

This model transforms the reordered source language phrase sequence $u_{a_1}, u_{a_2}, ..., u_{a_k}$ into a new sequence called c_0^K. The process replaces each source language phrase by a structure that retains the phrase itself and additionally specifies how many target language phrases should be appended to that phrase. Given $u_{a_1}, u_{a_2}, ..., u_{a_k}$, an element in the transformed sequence has the following form

$$c_k = u_{a_k} \cdot p_k \; ; \; p_k \in \{1, 2, ..., M\}^*$$

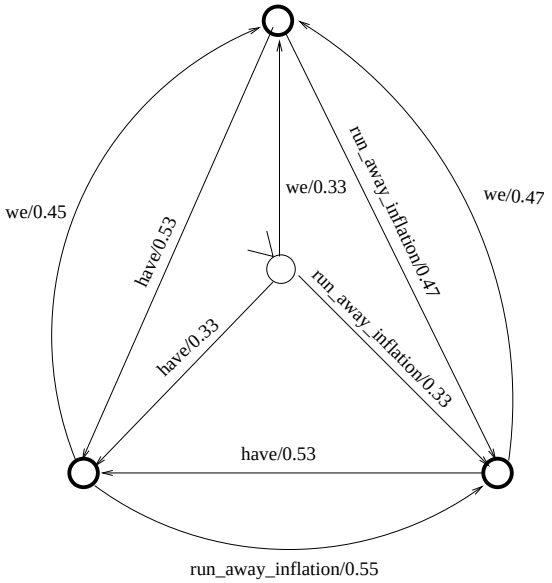

Figure 6: Acceptor H that assigns probabilities to reorderings of the source language phrase sequence *we have run_away_inflation* ($p_0 = 0.9$). Given the reordering *run_away_inflation we have* with alignment sequence $a_1 = 3, a_2 = 1, a_3 = 2$, H would assign it a probability: $P(a_1 = 3)P(a_2 = 1|a_1 = 3)P(a_3 = 2|a_2 = 1) = 0.33 \times 0.47 \times 0.53 = 0.08$.

The term p_k specifies the number and length of the target language phrases that can be spontaneously generated to follow the translation of u_{a_k}. The term has the following form: $p_k = p_k[1] \cdot p_k[2] \cdot ...$ and $p_k[i] \in \{1, 2, ..., M\}$. For example, if $u_{a_k} = terms\_of\_reference$, c_k might equal $terms\_of\_reference \cdot 1 \cdot 3 \cdot 4$, which specifies that the translations of *terms_of_reference* must be followed by three target language phrases of length one word, three words, and four words respectively. We note that these target language phrases must be drawn from the phrase-pair inventory, and therefore are of known maximum word length M. The probability of the element c_k is specified as

$$P(c_k|u_{a_k}) = \begin{cases} \alpha_0 & c_k = u_{a_k} \cdot \epsilon \\ \alpha^{\sum_i p_k[i]} & c_k = u_{a_k} \cdot p_k \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

We will refer to α as the *Phrase Exclusion Probability* (PEP). We note that $c_0, c_1, ..., c_k$ contains one additional term relative to the original sequence $u_{a_1}, u_{a_2}, ..., u_{a_k}$. This term c_0, has the form $c_0 = \epsilon \cdot p_0$, and its probability is given by

$$P(c_0) = \begin{cases} \alpha_0 & c_0 = \epsilon \\ \alpha^{\sum_i p_0[i]} & c_k = p_0 \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

The total probability of the sequence c_0^K is obtained as

$$P(c_0^K|u_{a_1}, u_{a_2}, ..., u_{a_k}) = P(c_0) \prod_{k=1}^{K} P(c_k|u_{a_k}). \quad (9)$$

In the above equations, the value of α_0 is set to ensure that the probability distribution (given in Equation 7) is normalized.

$$\begin{aligned}
\sum_{c_k} P(c_k|u_{a_k}) &= P(c_k = u_{a_k} \cdot \epsilon) + \sum_{p_k \neq \epsilon} P(c_k = u_{a_k} \cdot p_k) \\
&= \alpha_0 + \sum_{l=1}^{\infty} \sum_{p_k:|p_k|=l} P(c_k = u_{a_k} \cdot p_k) \\
&= \alpha_0 + \sum_{l=1}^{\infty} \sum_{p_k[1]p_k[2]...p_k[l]} \alpha^{\sum_{i=1}^{l} p_k[i]} \\
&= \alpha_0 + \sum_{l=1}^{\infty} \prod_{i=1}^{l} \sum_{j=1}^{M} \alpha^j \\
&= \alpha_0 + \sum_{l=1}^{\infty} (\sum_{j=1}^{M} \alpha^j)^l.
\end{aligned}$$

We can set α so that $\sum_{j=1}^{M} \alpha^j < 1$. This imposes a permissible range on α values: $0 \leq \alpha < \alpha_{\max}$, so that $(\sum_{j=1}^{M} \alpha^j)^l$ forms an infinite geometric series in l with sum of its terms given by

$$S = \frac{(\sum_{j=1}^{M} \alpha^j)}{1 - (\sum_{j=1}^{M} \alpha^j)}.$$

Therefore $\sum_{c_k} P(c_k) = \alpha_0 + S$, so that α_0 is fixed by α as $\alpha_0 = 1 - S$.

The WFST Implementation of the Target Phrase Insertion Model involves a transducer Φ shown in Figure 7. When a source phrase sequence is composed with Φ, it spontaneously inserts target phrases to generate an output sequence c_0^K according to Equation 9.

4.5 Phrase Transduction Model

We have described the segmentation and reordering processes that transform a source language sentence into source language phrases in target language phrase order. The Target Phrase Insertion Model decides the number and length of target phrases that are to be spontaneously inserted within this reordered source phrase sequence. The next step is to map this sequence into a sequence of target phrases.

We assume that the target phrases are conditionally independent of each other and depend only on the source language phrase which generated each of them. Each term c_k is mapped to a

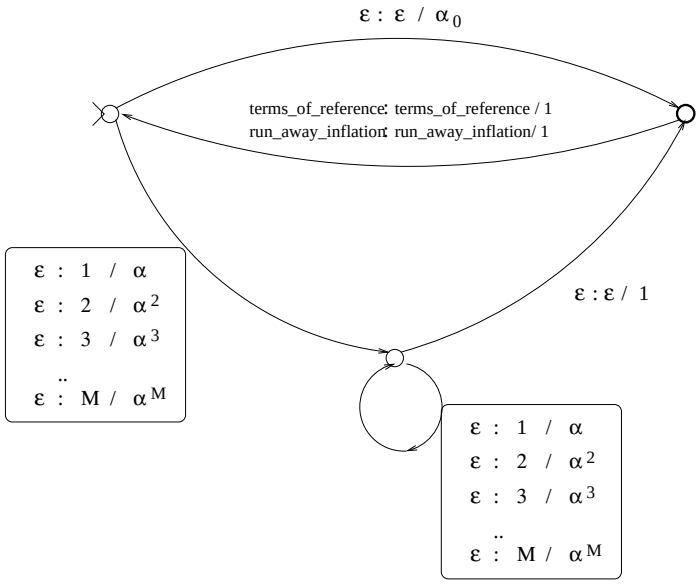

Figure 7: A portion of the Weighted Finite State Transducer Φ used to implement the Target Phrase Insertion Model. Suppose an example input for this transducer is the reordered source language phrase sequence *exports grain are projected_to_fall*, then a possible output of the WFST is the sequence $1\,exports \cdot 1\,grain\,are\,projected\_to\_fall$, which means that two target phrases are spontaneously inserted in the translation of source phrase sequence. The first target phrase is of length one word and inserted at the start of the sentence, and the second target phrase, also of length one, follows the translation of the source phrase *exports*.

sequence of target phrases d_k which are concatenated to obtain the final target phrase sequence $v_1^R = d_0^K$.

$$\begin{aligned} P(v_1^R, d_0^K | c_0^K, a_1^K, u_1^K, K, e_1^I) &= P(d_0^K | c_0^K) 1\{d_0^K = v_1^R\} \qquad (10) \\ P(d_0^K | c_0^K) &= \prod_{k=0}^{K} P(d_k | c_k) \\ &= \prod_{l=1}^{|p_0|} P(d_0 | c_{0l}) \prod_{k=1}^{K} \prod_{l=1}^{1+|p_k|} P(d_{kl} | c_{kl}), \end{aligned}$$

where $1\{d_0^K = v_1^R\}$ ensures that the target phrase sequence v_1^R agrees with the sequence d_0^K produced by the model. We note that this is the main component model of the TTM. We estimate the phrase translation probabilities by the relative frequency of phrase translations found in bitext alignments. We will implement this model using a transducer Y that maps any reordering of the target language phrase sequence into a source language phrase sequence v_1^R as in Equation 10. For every phrase u, this transducer allows only the target phrases v which are present in our library of phrase-pairs. In addition, for each $m \in \{1, 2, ..., M\}$, the transducer allows a mapping from

the target-phrase symbol m to all the m-length target phrases from our phrase-pair inventory V_T^m with probability given by

$$P(v|m) = \frac{1}{|V_T^m|}; v \in V_T^m. \tag{11}$$

A small portion of the phrase-pair inventory used to build the tranducer Y is shown in Table 1.

Source Phrase	Target Phrase	Phrase Transduction Probability
run_away_inflation	une_inflation_galopante	0.5
run_away_inflation	une_inflation_galopante	0.5
hear_hear	bravo	0.8
hear_hear	bravo_bravo	0.15
hear_hear	ordre	0.05
terms_of_reference	mandat	0.8
terms_of_reference	de_son_mandat	0.2

Table 1: A portion of the phrase-pair inventory used to build the Phrase Transducer Y. Y is a trivial single state transducer with number of arcs equal to the size of the inventory.

4.6 Target Phrase Segmentation Model

The composition of the previous transducers overgenerates the set of target language sequences. We build a model to constrain this set to agree with the target sentence. We specify this model as

$$P(f_1^J|v_1^R, d_0^K, c_0^K, a_1^K, u_1^K, K, e_1^I) = 1\{f_1^J = v_1^R\},$$

where $1\{f_1^I = v_1^R\}$ enforces the requirement that words in the target sentence agree with those in the phrase sequence. The WFST implementation of this model involves an unweighted segmentation transducer that enforces the above requirement, and maps target phrase sequences to target sentences. We build a weighted finite state transducer Ω for each target language sentence f_1^J to be translated. The transducer segments the sentence into all possible phrase sequences v_1^R permissible given the inventory of phrases.

A portion of the segmentation transducer Ω for the French sentence *nous avons une inflation galopante* is presented in Figure 8. When composed with the acceptor for the target sentence, Ω generates the following two phrase segmentations: *nous avons une_inflation_galopante* and *nous_avons_une inflation_galopante*.

We now present a example showing the translation process through which the TTM transforms a source language sentence into its translation in the target language (Figure 9).

5 Bitext Word Alignment and Translation Under the TTM

We will now describe how the Translation Template Model can be used to perform word-level alignment of bitexts and translation of target language sentences.

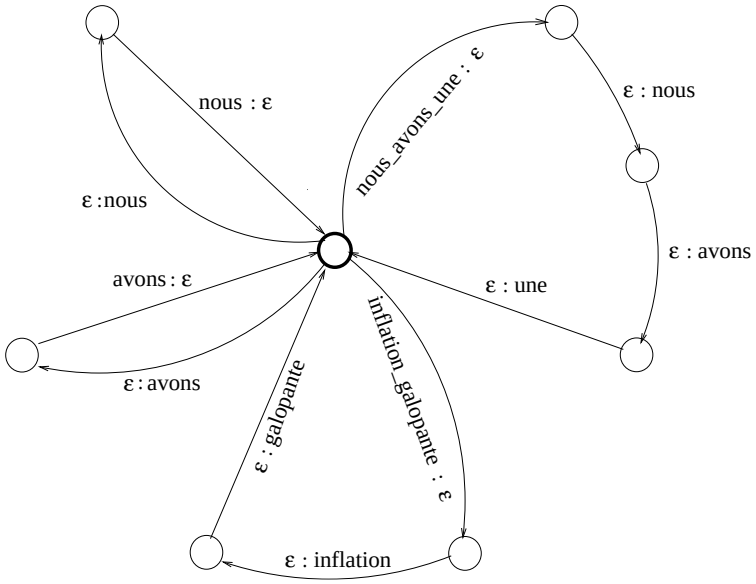

Figure 8: A portion of the target phrase segmentation transducer Ω for the target language phrase sequence: *nous avons une_inflation_galopante*. Given this sentence, the output of this transducer is the target language sentence *nous avons une inflation galopante*.

5.1 Bitext Word Alignment

Given a target language sentence f_1^J and a source sentence e_1^I, the word-to-word alignment between the sentences can be found using *Maximum A Posteriori* (MAP) decoding

$$\{\hat{K}, \hat{u}_1^{\hat{K}}, \hat{a}_1^{\hat{K}}, \hat{c}_0^{\hat{K}}, \hat{d}_0^{\hat{K}}, \hat{v}_1^{\hat{R}}\} = \underset{K, u_1^K, a_1^K, c_0^K, d_0^K, v_1^R}{\operatorname{argmax}} P(K, u_1^K, a_1^K, c_0^K, d_0^K, v_1^R | e_1^I, f_1^J). \tag{12}$$

$\hat{u}_1^{\hat{K}}$ and $\hat{d}_0^{\hat{K}} = \hat{v}_1^{\hat{R}}$ specify the MAP source phrase sequence and target phrase sequence respectively. $\hat{c}_0^{\hat{K}}$ specifies the position and length of the spontaneously generated target phrases within the reordered source phrase sequence. $\hat{a}_1^{\hat{K}}$ describes the MAP phrase-to-phrase alignment between the phrase sequences so that $\hat{c}_i$ is aligned to the target phrase $\hat{d}_i$. The MAP hypotheses are generated at the phrasal level, however using the knowledge that $\hat{c}_i$ is aligned to $\hat{d}_i$, we can obtain the word level alignments within the phrases directly from the phrase pair inventory. In this way we can generate the single MAP alignment.

We first describe how MAP word alignment under the TTM can be obtained when all phrase segmentations of the source sentence are considered and no reorderings of the source phrase sequence are considered. In this case a lattice of possible word alignments between e_1^I and f_1^J can be obtained by the finite state composition

$$\mathcal{B} = T \circ W \circ \Phi \circ Y \circ \Omega \circ S,$$

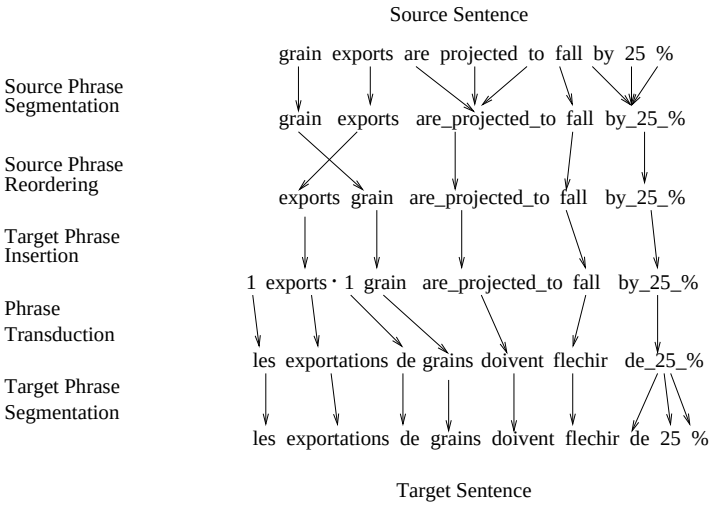

Figure 9: An example showing the translation process through which the TTM transforms a source language sentence into its translation in the target language. Inputs and Outputs for each TTM constituent model are shown.

where T is an acceptor for the source sentence e_1^I, and S is an acceptor for the target sentence f_1^J. An alignment lattice can be generated by pruning $\mathcal{B}$ based on likelihoods or number of states. The MAP alignment $\hat{B}$ (Equation 12) is found as the path with the highest probability in $\mathcal{B}$.

If only one phrase segmentation of the source sentence is to be considered during alignment, we follow a two-step procedure proposed earlier [13] in place of Equation 12. The first step is MAP phrase segmentation of the source sentence, followed by the MAP alignment of the fixed segmentation.

$$\begin{aligned} \{\tilde{u}_1^{\tilde{K}}, \tilde{K}\} &= \underset{u_1^K, K}{\operatorname{argmax}} P(u_1^K, K | e_1^I) \\ \{\tilde{a}_1^{\tilde{K}}, \tilde{c}_0^{\tilde{K}}, \tilde{d}_0^{\tilde{K}}, \tilde{v}_1^{\tilde{R}}\} &= \underset{a_1^{\tilde{K}}, c_0^{\tilde{K}}, d_0^{\tilde{K}}, v_1^R}{\operatorname{argmax}} P(a_1^{\tilde{K}}, c_0^{\tilde{K}}, d_0^{\tilde{K}}, v_1^R | \tilde{u}_1^{\tilde{K}}, \tilde{K}, e_1^I, f_1^J). \end{aligned} \tag{13}$$

This is implemented via WFSTs as follows. We first obtain a segmentation lattice of the source sentence: $\mathcal{U} = T \circ W$. The MAP source phrase seqmentation $\tilde{U}$ is obtained as the path with the highest probability in $\mathcal{U}$. Given the MAP segmentation $\tilde{U}$, the alignment lattice can be obtained by the WFST composition: $\mathcal{B} = \tilde{U} \circ \Phi \circ Y \circ \Omega \circ S$.

The above presentation assumes that the source phrase sequence is not reordered while performing alignment. If reorderings of the MAP source phrase segmentation are to be considered when obtaining MAP word alignment, we perform the following procedure. We first obtain the MAP phrase segmentation of the source language sentence as described above. We next build a permutation acceptor $\Pi_{\tilde{U}}$ that generates reorderings of the source phrase sequence $\tilde{U}$. The N-best reorderings of $\tilde{U}$ are obtained by considering the N most likely paths in the permutation acceptor under the Markov Phrase Order Model (Equation 5). Given this set of reorderings of the source

phrase sequence, the alignment lattice is found by a WFST composition. These two steps are given by

$$\begin{aligned} \Pi_{\tilde{U}}^{N} &= \text{N-Best Paths}(\Pi_{\tilde{U}} \circ H) \\ \mathcal{B} &= \Pi_{\tilde{U}}^{N} \circ \Phi \circ Y \circ \Omega \circ S. \end{aligned} \tag{14}$$

5.2 Translation

Given a target language sentence f_1^J, its translation in the source language can be found via MAP decoding as:

$$\{\hat{e}_1^{\hat{I}}, \hat{K}, \hat{u}_1^{\hat{K}}, \hat{a}_1^{\hat{K}}, \hat{c}_0^{\hat{K}}, \hat{d}_0^{\hat{K}}, \hat{v}_1^{\hat{R}}\} = \underset{e_1^I, K, u_1^K, a_1^K, c_0^K, d_0^K, v_1^R}{\operatorname{argmax}} P(K, u_1^K, a_1^K, c_0^k, d_0^K, v_1^R | f_1^J). \tag{15}$$

where $\hat{e}_1^I$ is the translation of f_1^J. $\hat{u}_1^{\hat{K}}$, $\hat{a}_1^{\hat{K}}$, $\hat{d}_0^{\hat{K}} = \hat{v}_1^{\hat{R}}$ and $\hat{c}_0^{\hat{K}}$ are the corresponding source phrase sequence, alignment sequence, target phrase sequence, and the sequence that specifies the position and length of spontaneously inserted target phrases within the reordered source phrase sequence; all these variables are hypothesized in the decoding process.

In translation we do not consider reorderings of the source phrase sequence due to limitations in the current WFST translation framework. In this case the set of possible translations of f_1^J is obtained using the weighted finite state composition:

$$\mathcal{T} = G \circ U \circ \Phi \circ Y \circ \Omega \circ S.$$

A translation lattice [28] can be generated by pruning $\mathcal{T}$ based on likelihoods or number of states. The translation with the highest probability (Equation 15) can be computed by obtaining the path with the highest score in $\mathcal{T}$.

5.3 Issues in Bitext Word Alignment

We now describe some issues that arise in the implementation of bitext word alignment using the TTM. Given a target language sentence and its translation in the source language, bitext word alignment under the TTM is performed by considering all segmentations of each sentence and finding the best possible alignment between the phrases under the constraint that all phrases are aligned. However, our inventory of phrase-pairs is not rich enough to cover all possible sentences, and as a result the sentence-pair contains phrase-pairs not in the inventory. When a sentence pair cannot be covered by the inventory, the pair is assigned a probability of zero under the model. In practice, we observe that even in the bitext collection from which the phrase inventory was gathered, most sentence pairs have a probability of zero. We see such an example in Figure 3 where the phrase-pairs extracted from the bitext do not completely cover the words in either the target or the source sentence. To overcome this limitation, we allow deletion of source phrases during the alignment process. This is done in addition to the insertion of target phrases under the Target Phrase Insertion Model (Equation 9). This will make it possible to align sentences containing phrases not found in the phrase pair inventory. The phrase transducer Y is modified by adding extra transitions to allow deletions of source phrases. Therefore each source phrase u can be mapped to an empty string in addition to its regular transductions to target phrases v.

The parameters $P(\epsilon|u)$ for deletions of source phrases u are not estimated; they are tied to the *Phrase Exclusion Probability* (α) introduced in the Target Phrase Insertion Model so that $P(\epsilon|u) = \alpha$ for all source phrases u in our inventory. The parameter α will be tuned to optimize the alignment performance on a development set. We modify the original estimates of phrase transduction probabilities $P(v|u)$ to ensure that the Phrase Transduction Model is correctly normalized while allowing deletions. For each source phrase u in the source phrase inventory, this is done as follows

$$P'(v|u) = \begin{cases} P(v|u)(1-\alpha) & v \neq \epsilon \\ \alpha & v = \epsilon, \end{cases}$$

so that $\sum_v P'(v|u) = 1$.

6 Translation and Alignment Experiments

We now report alignment and translation performance of the Translation Template Model. The finite state modeling is performed using the AT&T FSM Toolkit [17]. We present experiments on two tasks that involve both word alignment and translation - the Hansards French-to-English task [20] and the NIST Chinese-to-English task [18].

6.1 Source Language Texts, Bitexts, and Phrase-Pair Inventories

6.1.1 French-to-English

The goal of this task is the translation of the Canadian Hansards which are the official records of the Canadian parliament [24] maintained in both English and French. The translation model training data consists of 48,739 French-English sentence pairs from the Hansards [20]. The French side of the bitext contains 816,545 words (24,096 unique tokens). The English side has a total of 743,633 words (18,430 unique tokens) and is used to train the source language model. The test set consists of 500 unseen French sentences from Hansards for which both reference translations and word alignments are available [20].

On this task our phrase-pair inventory is found as described in Section 3 and consists of 772,691 entries, with 473,741 unique target phrases and 434,014 unique source phrases. We restrict the phrase-pairs to the target phrases which have at most 5 words. The distribution of the number of words in the source and target phrases over the inventory is shown in Table 2.

Target Phrase Length (in French words)	Source Phrase Length (in English words)							
	1	2	3	4	5	6-7	8-10	≥ 11
1	**414,347**	53,074	10,731	2,168	523	140	33	2
2	102,760	**190,072**	44,352	12,146	3,206	1,102	144	10
3	27,817	89,866	**119,501**	35,012	10,778	4,699	505	45
4	6,789	30,097	73,564	**79,147**	27,650	13,568	1,852	127
5	1,738	9,925	29,368	**57,207**	55,537	31,834	5,703	391

Table 2: Distribution of the number of words in the target and source phrases over the Phrase-Pair Inventory on the French-English Task. The bold entries denote the maximum count in each row.

6.1.2 Chinese-to-English

The goal of this task [18] is the translation of news stories from Chinese to English. The translation model training data consists of the FBIS Chinese-English parallel corpus [18] that consists of 9.76M words (49,108 unique tokens) in English and 7.82M words (55,767 unique tokens) in Chinese. The Chinese side of the corpus is segmented into words using the LDC segmenter [14]. The original bitext is aligned at the document level; documents are aligned automatically into chunk-pairs using a statistical chunk model [7] to generate 440,000 chunk pairs; on an average there are 38 chunk pairs per document pair, 1.72 chunks per sentence in each document, and 22 sentences per document pair. Our language model training data comes from English news text derived from two sources: online archives (Sept 1998 to Feb 2002) of *The People's Daily* (16.9M words) [6] and the English side of the Xinhua Chinese-English parallel corpus [18] (4.3M words). The total language model corpus size is 21M words.

Our translation test set is the NIST 2002 MT evaluation set [18] consisting of 878 sentences. Each Chinese sentence in this set has four reference translations. Our alignment test set consists of 124 sentences from the NIST 2001 dry-run MT-eval set [18] that are word aligned manually.

On this task our phrase-pair inventory is found as described in Section 3 and consisted of 8.05M entries, with 3.12M unique target phrases and 4.98M unique source phrases. We restrict the phrase-pairs to the target phrases which have at most 5 words. The distribution of the number of words in the source and target phrases over the inventory is shown in Table 3.

Target Phrase Length (in Chinese words)	Source Phrase Length (in English words)							
	1	2	3	4	5	6	7-8	≥ 9
1	**3,142,325**	1,720,267	775,320	266,181	80,127	24,565	12,219	3,973
2	705,287	**1,461,448**	1,134,843	635,510	295,413	123,223	69,124	18,583
3	149,409	479,069	**781,015**	696,161	461,924	262,585	201,714	64,835
4	34,096	130,745	300,534	**451,314**	441,086	340,730	359,742	162,452
5	9,134	34,186	95,821	196,055	283,960	**300,150**	449,388	314,298

Table 3: Distribution of the number of words in the target and source phrases over the Phrase-Pair Inventory on the Chinese-English Task. The bold entries denote the maximum count in each row.

6.2 Bitext Word Alignment

Given a pair of translations, the goal of bitext word alignment is to find word-to-word correspondences between these sentences. Performance is measured with respect to a reference word alignment created by a competent human translator, and we measure the alignment performance against the reference alignment using Alignment Precision, Alignment Recall, and Alignment Error Rate metrics [20].

We first present definitions of alignment metrics. Let $e = e_0^l$ and $f = f_1^m$ denote a pair of translated sentences in the source and the target language. A source word token is defined as an ordered pair $e = (j, w) : w \in V_E, j \in \{0, 1, 2, ..., l\}$, where the index j refers to the position of the word in the source sentence; V_E is the vocabulary of the source language; and the word at position 0 is the NULL word to which "spurious" target words may be aligned. Similarly, a target word token is written as $f = (i, w) : w \in V_F, i \in \{1, 2, 3, ..., m\}$, where the index i refers to the position of the word in the target sentence; V_F is the vocabulary of the target language; and the word at position 0 is the NULL word to which "spurious" source words may be aligned.

An *alignment* between e and f is defined to be a link set $B = \{b_1, b_2, ..., b_m\}$ whose elements are given by the alignment links b_k. An alignment link $b = (i, j)$ specifies that the source word e_i is connected to the target word f_j under the alignment. Alignment metrics allow us to measure the quality of an automatic word alignment B' relative to a reference alignment B. Alignment Precision is defined as the fraction of links in the automatic alignment which are also in the reference alignment. Alignment Recall is the fraction of links in the reference alignment that are also in the automatic alignment. Alignment Error Rate (AER) is the fraction of links by which the automatic alignment differs from the reference alignment. In all these measurements, links to the NULL word are ignored. This is done by defining modified link sets for the reference alignment: $\bar{B} = B - \{(i, j) : i = 0 \text{or} j = 0\}$ and the automatic alignment: $\bar{B}' = B' - \{(i', j') : i' = 0 \text{or} j' = 0\}$.

The reference annotation procedure allowed the human transcribers to identify which links in $\bar{B}$ they judged to be unambiguous. In addition to the reference alignment, this gives a set of *sure links* (S) which is a subset of $\bar{B}$. The alignment metrics are defined as follows [20]

$$\text{Alignment Precision } (S, B; B') = \frac{|\bar{B}' \cap \bar{B}|}{|\bar{B}'|} \tag{16}$$

$$\text{Alignment Recall } (S, B; B') = \frac{|\bar{B}' \cap S|}{|S|} \tag{17}$$

$$\text{AER } (S, B; B') = 1 - \frac{|\bar{B}' \cap S| + |\bar{B}' \cap \bar{B}|}{|\bar{B}'| + |S|}. \tag{18}$$

We present word alignment performance of the WFST translation model on the two alignment tasks in Table 4. For comparison, we also align the bitext using IBM-4 word translation models [4][20] trained in both translation directions (IBM-4 E and IBM-4 F), and their union (IBM-4 $E \cup F$). For all experiments presented here, we will use the Fixed Phrase Order Model (Equation 6). We will justify the choice of this model through the experiments in Section 6.2.6.

Model	Alignment Metrics (%)					
	French-English			Chinese-English		
	Precision	Recall	AER	Precision	Recall	AER
IBM-4 F	89.4	90.5	10.1	82.8	48.0	39.2
IBM-4 E	89.6	90.0	10.2	73.9	58.3	34.9
IBM-4 $F \cup E$	84.5	94.5	11.7	66.0	63.1	35.5
TTM	94.5	84.6	9.9	89.0	37.7	47.0

Table 4: TTM Alignment Performance on the French-English and the Chinese-English Alignment Tasks.

We note that the alignment error rate of the TTM is comparable to the baseline IBM-4 models on the French-English task, but worse than IBM-4 models on the Chinese-English task. On both tasks the model obtains a very high Alignment Precision but a relatively poor Alignment Recall. The high Alignment Precision suggests that the word alignments within the phrase-pairs are very accurate. However, the poor performance under the recall measure suggests that the phrase-pair inventory has relatively poor coverage of the phrases in the alignment test set. Alignment Recall is influenced by the words in the target and the source language sentence which are either spontaneously inserted or deleted during word alignment. In analyzing the French-English word

alignments, we found that on average, 32.5% of the target-phrases are inserted and 34.1% of the source phrases are deleted. On the Chinese-English task, 51.6% of the target-phrases are inserted and 52.1% of the source phrases are inserted. Clearly the Alignment Recall in the Chinese-English will therefore be much lower than in French-English, whereas the Alignment Precision degrades only slightly. An additional factor that affects the Alignment Recall is the presence of words in the test set that are unseen in training. These are treated as single word phrases and are left out of the alignment, thus reducing the Alignment Recall.

6.2.1 Phrase Exclusion Probability

MAP word alignment under the TTM is affected by the number of target and source phrases that are excluded during bitext word alignment; this behavior is governed by the Phrase Exclusion Probability (PEP) as described in Section 5.3. We will now measure word alignment quality as a function of PEP (α) (Figure 10). In Figure 10 we observe that Alignment Precision increases monotonically with PEP over most of its permissible range, however there is a critical value above which Alignment Precision decreases. Alignment Recall at first improves slightly with PEP but then decreases. and then decreases slighly. AER closely follows the Alignment Recall.

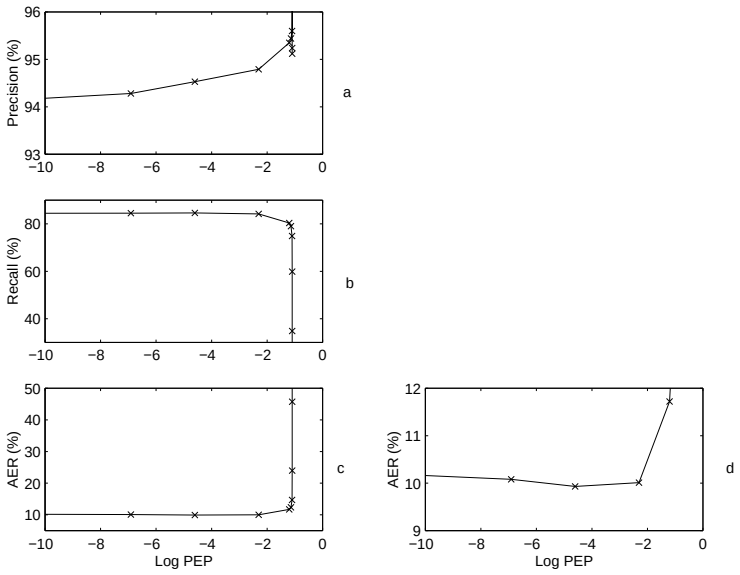

Figure 10: Alignment Performance of TTM as a function of Phrase Exclusion Probability (PEP). For each value of PEP, we measure Precision (Panel a), Recall (Panel b), and AER (Panel c). Results are shown on the French-English task. The plot in Panel d focuses in on the values of PEP where AER attains the minimum.

We now study this behavior more closely. The TTM is constructed so that as PEP (α) increases, the likelihood of excluding phrases increases. To assess this, we measure the percentage of Excluded Phrase Counts (EPC) which is the ratio of the number of source and target phrases ex-

cluded under the MAP alignment to the total number of transductions (phrase-pair transductions, spontaneous insertions of target phrases, and deletions of source phrases) in the MAP alignment. In Figure 11, we see that EPC is in fact increasing in PEP. We see furthermore that there is a critical value above which EPC increases rapidly; at this point the model simply finds it more likely to exclude phrases rather than align them. This has a direct influence on Alignment Recall (Equation 17), which is proportional to the number of correctly aligned words. This quantity is necessarily dominated by the number of aligned phrases, so that Alignment Recall falls off sharply with a sharp rise in PEP.

The influence of PEP on Alignment Precision is more complex. As PEP increases, the model is able to avoid aligned phrase pairs whose transduction probability is low. As a result, the phrase pairs that remain in the alignment are those with higher phrase transduction likelihoods. For each aligned phrase pair, this quantity is based simply on the relative frequencies of their occurrences in the bitext word alignments (see Section 4.5). As PEP increases, the alignment favors source language phrases that are uniquely aligned to one target phrase phrase. It is plausible that the word alignments within these phrase pairs are of higher quality than found in general. This would explain the increase in Alignment Precision at intermediate values of PEP.

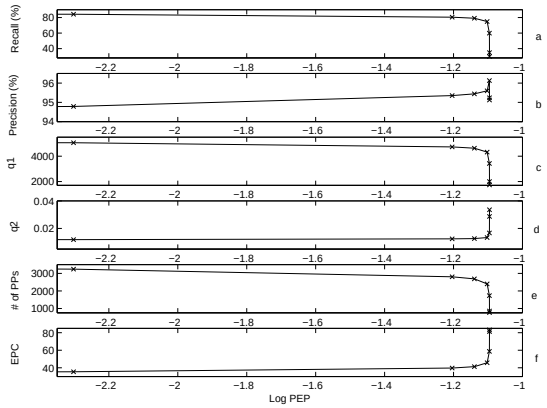

Figure 11: Variation of Alignment Precision (Panel b) and Recall (Panel a) for values of Phrase Exclusion Probability (PEP) near the critical value. We also plot four additional quantities derived from the MAP alignment. These include the number of wrongly hypothesized links q_1 (Panel c), penalty per incorrectly hypothesized alignment link q_2 (Panel d), the number of phrase-pair transductions (Panel e), and the percentage of Excluded Phrase Counts (Panel f). Results are shown on the French-English task.

For PEP above the critical point, we observe a decrease in Alignment Precision (Figure 11 e). To analyze this behavior, we write Alignment Precision as

$$\text{Alignment Precision}(S, B; B') \quad = \quad \frac{|\bar{B}' \cap \bar{B}|}{|\bar{B}'|} \\ 1 - q_1 q_2,$$

where $q_1 = |\bar{B}'| - |\bar{B}' \cap \bar{B}|$ and $q_2 = \frac{1}{|\bar{B}'|}$. Considered in this way, q_1 is the number of incorrectly

hypothesized alignment links, and q_2 is the penalty associated with each wrong alignment link; this penalty decreases inversely with the number of hypothesized links. The interaction between q_1 and q_2 as PEP varies will determine the Alignment Precision. In Figure 11, we see that as EPC increases (Figure 11f) the absolute number of phrase-pairs in the alignment decreases (Figure 11e). The quantity q_2 (Figure 11d) can be expected to vary inversely with the number of aligned phrase pairs, and we in fact observe this behavior. We separately measure q_1 , the number of incorrectly hypothesized alignment links, and find that this number does decrease for PEP above the critical value (Figure 11c), suggesting that the relatively few phrase pairs that remain in the alignments are of high quality. However we see that the Alignment Precision (Figure 11b) is dominated by q_2 so that performance falls for PEP above the critical value.

6.2.2 Richness of the Phrase-Pair Inventory

It has been established [20] that alignment performance of IBM-4 models improves as the size of the bitext training set grows. In contrast, the alignment performance of the TTM is more complex. The phrase-pair inventory is created from a set of word alignments generated by underlying IBM-4 models so that the TTM alignment performance depends, in part, on the quality of the underlying word alignments. In addition, the TTM alignment performance also depends on the richness of the phrase-pair inventory which determines coverage of the test set. We now perform experiments to tease apart these two factors.

In this section we study the effect of richness of phrase-pair inventory on word alignment quality. For this purpose, we train IBM-4 translation models on the 48K French-English Hansards bitext collection (Section 6.1) and obtain word alignments over this set. We then construct four subsets of the bitext word alignments consisting of 5K, 12K, 24K, and 48K sentence-pairs respectively. From each subset, we extract a phrase-pair inventory (using the procedure described in Section 3). Statistics over the four phrase-pair inventories are shown in Table 5. We also measure coverage by each inventory of the test set in the following way. Multiple segmentations of each target (source) sentence are first obtained under the inventory. For each word in the target (source) sentence, we obtained the number of phrases that covered the word in alternative segmentations of the sentence. The number of phrases per word is then averaged over the test set. The *Average Number of Phrases per Word* is measured for both the target and the source language. A higher value of these quantities indicates a better coverage of the test set under the particular inventory.

Subset ID	# of Sentence Pairs	Phrase-Pair Inventory Statistics			Av. phrases/word	
		# of Target Phrases	# of Source Phrases	# of Phrase-Pairs	Target	Source
PPI-1	5K	81,640	73,283	122,621	3.1	3.2
PPI-2	12K	167,752	151,534	259,010	5.5	6.2
PPI-3	24K	288,395	261,685	456,946	7.3	8.2
PPI-4	48K	473,741	434,014	772,691	10.6	11.8

Table 5: Statistics over Phrase-Pair Inventories extracted from four subsets of the French-English Hansards. IBM-4 models are trained on 48K sentence-pairs from Hansards and word alignments are obtained on the training set. Four phrase-pair inventories are then constructed from four nested subsets of these word alignments. The coverage by each inventory of the test set (Average Number of Phrases per Word) is reported.

Using the PPIs as described in Table 5, we construct four different TTMs and use each to obtain MAP word alignments of the test set (Equation 12). In Figure 12 we show the Alignment Precision, Alignment Recall, and AER as a function of Phrase Exclusion Probability (α), for values

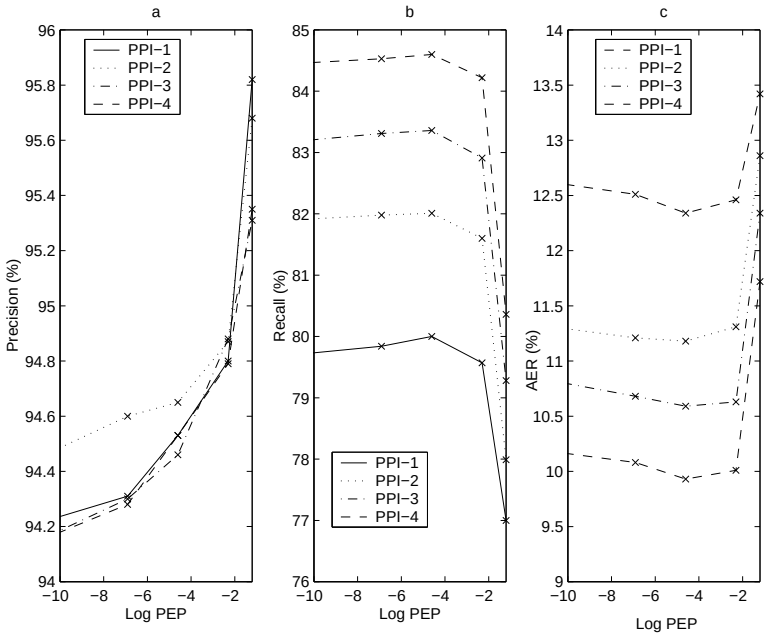

Figure 12: Effect of phrase-pair inventory size on word alignment quality. IBM-4 models are trained on 48K sentence-pairs from French-English Hansards and word alignments are obtained over the collection. Four subsets are constructed from this set of word alignments and phrase-pair inventories were collected over each subset. For each inventory, MAP word alignments under the TTM are obtained, and Alignment Precision (Panel a), Alignment Recall (Panel b), and AER (Panel c) are measured as functions of Phrase Exclusion Probability (PEP). Inventories are shown in Table 5.

below the critical value. Examining these results shows that Alignment Precision changes only slightly with an increase in the size of the phrase-pair inventory (Figure 12a). However Alignment Recall decreases dramatically as the size of the phrase-inventory is reduced (Figure 12b). AER is dominated by the decrease in Alignment Recall and decreases with a reduction in the size of the inventory (Figure 12c). The variation in all three alignment metrics with respect to Phrase Exclusion Probability (PEP) is identical for all the four subsets.

We first explain the variation in Alignment Precision as the size of the phrase-pair inventory is reduced. We note that the four phrase-pair inventories are extracted from word alignments generated by the same IBM-4 models. Therefore the word alignments within the phrase-pair inventories are of uniform quality; this in turn suggests that the word alignments generated by the TTM will yield nearly identical Alignment Precision regardless of the size of the inventory employed. We explain the variation in Alignment Recall across the four inventories by measuring the coverage of the inventories on the test set. As the size of the underlying phrase-pair inventory is reduced, the coverage of test set drops as seen in Table 5. Alignment Recall (Equation 17) is proportional to the number of correctly aligned words on the test set and is therefore dependent

on the coverage by the inventory of the test set. This suggests that as the size of the phrase-pair inventory is reduced, Alignment Recall will decrease due to a decrease in test set coverage. We conclude from this experiment that if the underlying word alignment quality does not change, the main influence of increasing bitext size is to increase phrase-pair coverage and consequently improve Alignment Recall.

6.2.3 Word Alignment Quality of Underlying IBM-4 Models

In the previous experiment, the quality of the underlying word alignments is held constant while we vary the size of the bitext from which the phrase-pair inventories are extracted. As an alternative, we would like to fix the size of the phrase-pair inventory and allow the underlying word alignments to vary in quality. However this is not possible, since the phrase-pair inventories themselves are extracted from word alignments. We take the following approach. We constructed systems over varying amounts of bitext and adjust the PEP (α) so that two different sized systems have the same Alignment Recall; this implies that they have comparable coverage. At these points we will measure Alignment Precision and AER.

For this experiment, we construct four nested subsets of the Hansards bitext collection containing 5K, 12K, 22K, and 48K sentence pairs respectively; these are the same four subsets used in the previous experiment. On each subset, we trained IBM-4 translation models and used these models to obtain word alignments over the smallest (5K) subset. From each set of word alignments over the 5K subset, we construct a phrase-pair inventory using the procedure described in Section 3. Statistics over these four phrase-pair inventories are shown in Table 6.

Subset ID	# of Sent. Pairs	Phrase-Pair Inventory Statistics			AER (%) of IBM-4 models	Av. phrase/word	
		# of Target Phrases	# of Source Phrases	# of Phrase-Pairs		Target	Source
PPI-1	5K	58,266	51,318	80,256	20.6	4.6	5.2
PPI-2	12K	67,242	59,138	95,953	15.9	5.0	5.6
PPI-3	24K	74,526	65,856	108,952	13.9	5.3	6.0
PPI-4	48K	81,800	73,442	123,314	12.1	5.6	6.2

Table 6: Statistics over four different Phrase-Pair Inventories collected from a 5K subset of the French-English Hansards. IBM-4 models are trained on four nested subsets of the French-English Hansards bitext and word alignments are obtained over the smallest (5K) subset. A phrase-pair inventory is collected over each word alignment. The alignment quality (in AER) of each underlying IBM-4 model and the coverage by each inventory of the test set (Average number of Phrases per Word) are reported.

Using the four phrase-pair inventories as described in Table 6, we construct four different TTMs and use each to obtain a MAP word alignment of the test set. In Figure 13, we study Alignment Precision, Alignment Recall, and AER as a function of PEP for values below the critical value. Contrary to the previous experiment in which we alignment quality was held constant, we observe that as the size of bitext increases, the Alignment Precision improves (Figure 13a). We see that Alignment Recall also improves with the size of the bitext (Figure 13b). AER reflects the combined Alignment Precision and Recall, and improves consistently as the bitext size is increased (Figure 13c). The variation in alignment performance (precision, recall and AER) with respect to Phrase Exclusion Probability is seen to be identical for all the four subsets.

To understand this behavior, we note that as the size of bitext is increased, the alignment performance of the IBM-4 models improves (Table 6). We therefore attribute the increase in Alignment Precision to the improvement of the underlying word alignments. We attempt to

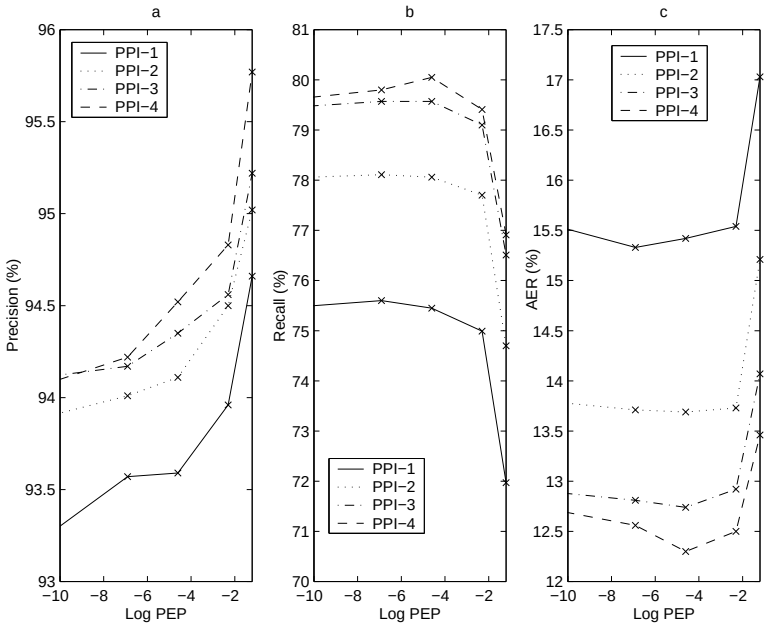

Figure 13: Effect of word alignment quality of underlying IBM-4 models on alignment performance of TTM. IBM-4 models are trained on four nested subsets of the French-English Hansards bitext and word alignments are obtained over the smallest subset (5K sentence pairs). A phrase-pair inventory are constructed over each word alignment. For each inventory, MAP word alignments under the TTM are obtained, and Alignment Precision (Panel a), Alignment Recall (Panel b), and AER (Panel c) are measured as functions of Phrase Exclusion Probability. Inventories are shown in Table 6.

measure Alignment Precision for constant values of Alignment Recall. Table 7 presents PPI-1 at $log(PEP) = -2.30$ and PPI-2 at $log(PEP) = -1.25$. We observe that although the Alignment Recall values for the two systems are equal, the PPI-1 system has a lower Alignment Precision than PPI-2. Since the underlying models for PPI-1 are trained on approximately half the number of sentence pairs as the underlying models for PPI-2, we conclude that, if Alignment Recall can be held constant, the effect of increasing bitext is to improve Alignment Precision of the TTM. As the size of bitext increases, test-set coverage improves ; this results in an increase in the Alignment Recall.

6.2.4 Multiple Source Phrase Segmentations

Ideally the word alignment of sentence pairs under the TTM is obtained after considering all possible phrase segmentations of the source sentence (Equation 12). An alternative, approximate approach could be done following the two-step procedure (Equation 13) that consists of MAP phrase segmentation of the source sentence, then followed by the MAP alignment of the fixed source sentence phrase segmentation. Figure 14 compares the performance of the two approaches

Subset ID	Bitext Size	$log(PEP)$	Precision (%)	Recall (%)	AER (%)
PPI-1	5K	-2.30	94.0	75.0	15.5
PPI-2	12K	-1.25	94.8	75.0	15.1

Table 7: Analysis of the effect of Word Alignment Quality on TTM Alignment Performance. We select two systems from Figure 13 with constant Alignment Recall, and measure Alignment Precision and AER for these systems.

as a function of the Phrase Exclusion Probability for values above the critical value. We find that the two-step approach (Equation 13) is markedly inferior relative to the exact MAP word alignment (Equation 12).

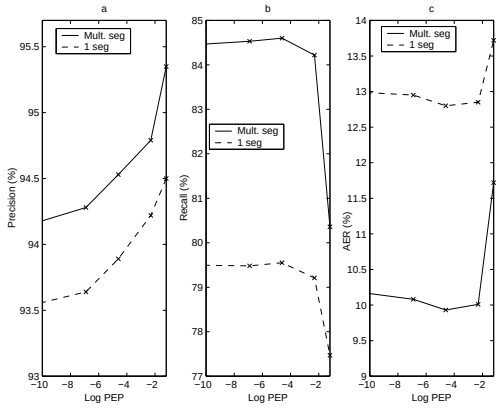

Figure 14: Effect of multiple phrase segmentations of the source sentence on word alignment quality. MAP Word Alignments Under the TTM are obtained using the two-step alignment process (Equation 13) that considers only a single phrase segmentation of the source sentence These are compared to MAP word alignments obtained using all segmentations of the source sentence (Equation 12). In both cases, Alignment Precision (Panel a), Alignment Recall (Panel b), and AER (Panel c) are measured are functions of Phrase Exclusion Probability.

6.2.5 Unweighted Source Phrase Segmentation Model

In Section 4, we described an exact procedure to ensure that the probabilities over all segmentations of a given length sentence are correctly normalized. As this procedure is expensive in practice, we consider excluding the source phrase segmentation model in the following way. We obtain word alignments under the TTM using an unweighted source phrase segmentation model, i.e. a source phrase segmentation transducer W is constructed as in Section 4.2 but no weights are assigned to the transitions. The MAP word alignments without the source phrase segmentation model are compared to the exact MAP word alignments in Table 8.

We observe that excluding the segmentation model has almost no impact on the alignment quality. We can therefore avoid this expensive step in practice. On observing that the phrase

segmentation model likelihoods have no practical benefit for word alignment, we conclude that this particular instance of the segmentation model is so weak that the overall alignment performance is dominated by the phrase transduction probabilities.

Source Phrase Segmentation Model	Alignment Metrics (%)		
	Precision	Recall	AER
Weighted	94.5	84.6	9.9
Unweighted	94.4	84.6	10.0

Table 8: Effect of an Unweighted Source Segmentation Model on Alignment Quality. Results are shown on the French-English Hansards Task.

6.2.6 Source Phrase Reorderings

In the experiments described thus far we have used the Fixed Phrase Order Model (Equation 6) that does not reorder the source phrase sequence while performing word alignment (Equation 12). We now measure the effect of reorderings of the MAP source phrase segmentation on alignment performance of the TTM.

We follow the procedure described earlier (Section 5) and obtain an N-best list of reorderings under the Markov Phrase Order model (Equation 5). Word alignment of each sentence-pair under the TTM (Equation 12) is then performed given the N-best reorderings of the source phrase sequence.

We first derive a quantity that characterizes the tendency of the model to relocate phrases in order to achieve the MAP word alignment. This quantity, called Average Phrase Movement (APM) [19], measures the degree of non-monotonicity in the MAP word alignment (Equation 12). Suppose any two consecutive phrases in the reordered source phrase sequence $\hat{u}_{\hat{a}_1}, ..., \hat{u}_{\hat{a}_k}$ are given by $\hat{u}_{\hat{a}_k} = e_l^{l'}$ and $\hat{u}_{\hat{a}_{k-1}} = e_m^{m'}$, the movement between these phrases is measured as $d_k = |l - m' - 1|$. The total phrase movement over the sentence pair is taken as the sum of the individual movements: $d = \sum_{k=1}^{K} d_k$. The Average Phrase Movement is obtained by averaging the total movement over the sentences in the test set. We emphasize that the target phrase order is unchanged during the alignment process, so the Average Phrase Movement measures variation in the source phrase order relative to both the original source phrase order and the target phrase order.

We perform two experiments to study the effect of reorderings on TTM word alignments. In the first experiment, we fix the number of reordered source phrase sequences (an N-best list of size 400) and obtained MAP word alignments under the TTM as a function of PEP (α) (Figure 15). For each PEP we also measure the percentage of Excluded Phrase Counts (EPC). We observe that there is only a slight improvement of AER by allowing reorderings relative to the no reordering case. When reorderings are allowed, the Average Phrase Movement drops monotonically as PEP is increased. We also note the AER peaks at the same value of PEP whether or not reordering of the source phrase sequence is allowed.

Our conclusion is that to induce phrase reorderings in the MAP word alignment, PEP must be set to a value that leads to a degradation in AER. In contrast at the optimal value of AER, we observe that the Average Phrase Movement of the MAP word alignment is less than one word; this suggests that we could obtain similar gains in AER by increasing the maximum word length of the source phrases in the phrase-pair inventory instead of allowing source phrase reorderings

during alignment.

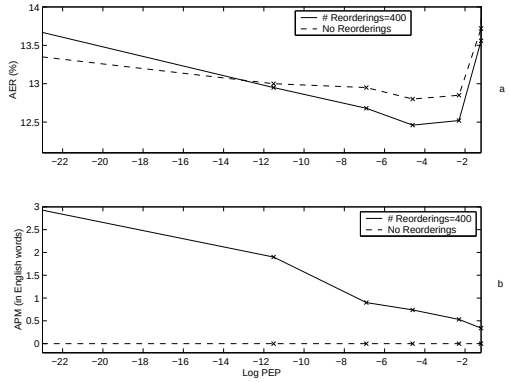

Figure 15: Effect of Reorderings of the Source Phrase Sequence on alignment quality. MAP Word Alignments under the TTM are obtained using a fixed number of reorderings ($N = 400$) of the single phrase segmentation of the source sentence. Performance is compared with MAP word alignments obtained without reordering the source phrase sequence. We measure AER (Panel a) and Average Phrase Movement (Panel b) as functions of the Phrase Exclusion Probability (PEP). Results are shown on the French-English Task.

In the second experiment we fixed the Phrase Exclusion Probability at its optimal value from the the first experiment (PEP = 0.005), and then obtained MAP word alignments under the TTM as the number of reordered source phrase sequences is varied (Table 9). For comparison we also show the performance when the source phrase is not reordered in computing the MAP word alignment. As the number of reordered source phrase sequences is increased from 1 to $1,000$, we note that the Average Phrase Movement increases slightly. When reorderings are allowed, there is a slight reduction in PEP relative to the no-reordering case. AER decreases only slightly by allowing more reorderings of the source phrase sequence during alignment.

We conclude from this experiment that allowing more reorderings leads to a greater Average Phrase Movement in the MAP alignment. In addition this also allows more phrase pairs to be aligned as seen by the reduction in PEP. However, the AER does not improve much by allowing reorderings of the source phrase sequence; we observe that most of the AER gains can be obtained by using a 100-best list of reorderings. This experiment provides evidence that we can avoid reordering the source phrase sequence without much degradation in alignment performance.

In summary we have investigated several factors that affect the alignment performance of the TTM. We have observed in all these experiments that the variation in alignment performance with respect to the Phrase Exclusion Probability is invariant to the number of segmentations, richness of inventory or the reorderings of the source phrase sequence allowed during alignment. The best performance among these configurations is achieved when multiple segmentations of the source phrase sequence are considered while computing the MAP alignment (Table 4).

# of reordered source phrase sequences	Alignment Metrics (%) Precision	Recall	AER	Average Phrase Movement	EPC (%)
No Reordering	93.9	79.5	12.8	0.0	34.8
1	93.9	79.5	12.8	0.2	34.8
100	94.0	80.1	12.5	0.7	34.2
200	94.0	80.1	12.5	0.7	34.2
400	94.0	80.1	12.5	0.7	34.2
600	94.0	80.1	12.5	0.8	34.2
800	94.0	80.1	12.5	0.8	34.2
1000	94.0	80.1	12.5	0.8	34.2

Table 9: Effect of number of reorderings of the source phrase sequence on alignment quality. MAP word alignments under the TTM is obtained as a function of the number of reorderings of the source phrase sequence in the French-English Task. In each case, we measure Alignment Quality (Precision, Recall and AER), Average Phrase Movement and the percentage of Excluded Phrase Counts (EPC).

6.3 Translation

We now measure the translation performance of the TTM as described in Section 5.2. In implementing translation under the TTM we employ the unweighted Source Phrase Segmentation Model (Section 4.2) and the Fixed Phrase Order Model (Section 4.3.2). The other components of the TTM remain as described in the bitext word alignment experiments (Section 6.2). Unlike word alignment, translation requires a source language model (Section 4.1). For this we use a trigram word language model estimated using modified Kneser-Ney smoothing as implemented in the SRILM tookit [25]. As described in Section 6.1, separate source (English) language models are trained for the French-English and Chinese-English tasks.

Translation performance is measured using the BLEU and NIST MT-eval metrics, and Multi-Reference Word Error Rate (mWER). The NIST and mWER metrics are described at length elsewhere [8] [19], and we will not review them. However we wish to provide a detailed analysis of translation performance under BLEU, so we will review its formulation.

The BLEU score [23] measures the agreement between a hypothesis translations E' and its reference translation E by computing the geometric mean of the precision of their common n-grams. The score also includes a 'Brevity Penalty' $\gamma(E, E')$ that is applied if the hypothesis is shorter than the reference. The functional form is

$$\text{BLEU}(E, E') = \gamma(E, E') \times \text{BPrecision}(E, E') \tag{19}$$

$$\text{BPrecision}(E, E') = exp\left(\frac{1}{N}\sum_{n=1}^{N} log\, p_n(E, E')\right) \tag{20}$$

$$\gamma(E, E') = \begin{cases} 1 & |E'| \geq |E| \\ e^{(1-|E|/|E'|)} & |E'| < |E| \end{cases} \tag{21}$$

In the above equations, $p_n(E, E')$ is a modified precision of n-gram matches in the hypothesis E',

and is specified as

$$p_n(E, E') = \frac{\sum_{g \in \mathcal{V}^n} \min(\#_E(g), \#_{E'}(g))}{\sum_{g \in \mathcal{V}^n} \#_{E'}(g)}, \tag{22}$$

where $\mathcal{V}^n$ denoted all n-grams (order n), $\#_E(g)$ and $\#_{E'}(g)$ are the number of occurrences of the n-gram g in the reference E and in the hypothesis E' respectively. We will use the notation BLEUrXnY to refer to BLEU score measured with respect to X reference translations and a maximum n-gram length $N = Y$ in Equation 20. The BLEU score (Equations 19-22) is defined over all sentences in the test set, i.e. E' and E are concatenations of hypothesis (reference) translations over sentences in a test set. We can also define a sentence-level BLEU score between the hypothesis and reference translations of each individual sentence using Equations 19-22.

To serve as a baseline translation system, we use the ReWrite decoder [15] with the French-English and Chinese-English IBM-4 translation models used in creating the phrase-pair inventories. We see in Tables 10 that in both the Chinese-English and French-English tasks, the performance of the TTM compares favorably to that of the ReWrite decoder.

Model	French-English		Chinese-English	
	BLEUr1n4 (%)	NISTr1n4	BLEUr4n4 (%)	NISTr4n4
IBM-4	17.09	5.02	9.67	3.57
TTM	22.29	5.52	20.83	7.54

Table 10: Translation Performance of the TTM on the French-English and Chinese-English Translation Tasks. For comparison, we also report performance of ReWrite Decoder with the French-English and Chinese-English IBM-4 translation models used to create the Phrase-Pair inventories.

6.3.1 Phrase Exclusion Probability

In Section 6.2, we have seen that the Phrase Exclusion Probability (PEP) strongly influences bitext alignment quality. We now evaluate the effect of this parameter on translation. The role of PEP in translation is to control spontaneous insertions of target phrases. This is in contrast to word alignment where PEP affects both the spontaneous insertions of target phrases and the deletions of source phrases. We would like to allow the model the flexibility of deleting phrases in sentence to be translated. Within the source-channel model, this is achieved through the insertion of target language phrases. We could also allow the generative model to delete source language phrases, but this would correspond to the insertion of English phrases in translation independent of any evidence in the Chinese or French sentence; in other words, they would be hypothesized entirely by the source language model. We do not consider this scenario.

We now discuss several aspects of Phrase Exclusion Probability in translation. We first observe that there is sensitivity in the BLEU score to the number of reference translations. In the French-English task, we have only one reference per sentence to be translated, while in the Chinese-English task we have four references. In Figure 16 we measure BLEU and WER metrics as functions of PEP when one reference is considered in measuring performance. We see that BLEU decreases as the PEP increases to allow target (French/Chinese) phrases to be deleted in translation. As in bitext word alignment, there is a critical value of PEP above which BLEU and WER quickly degrade. We note that WER does decrease slighly with PEP, unlike BLEU.

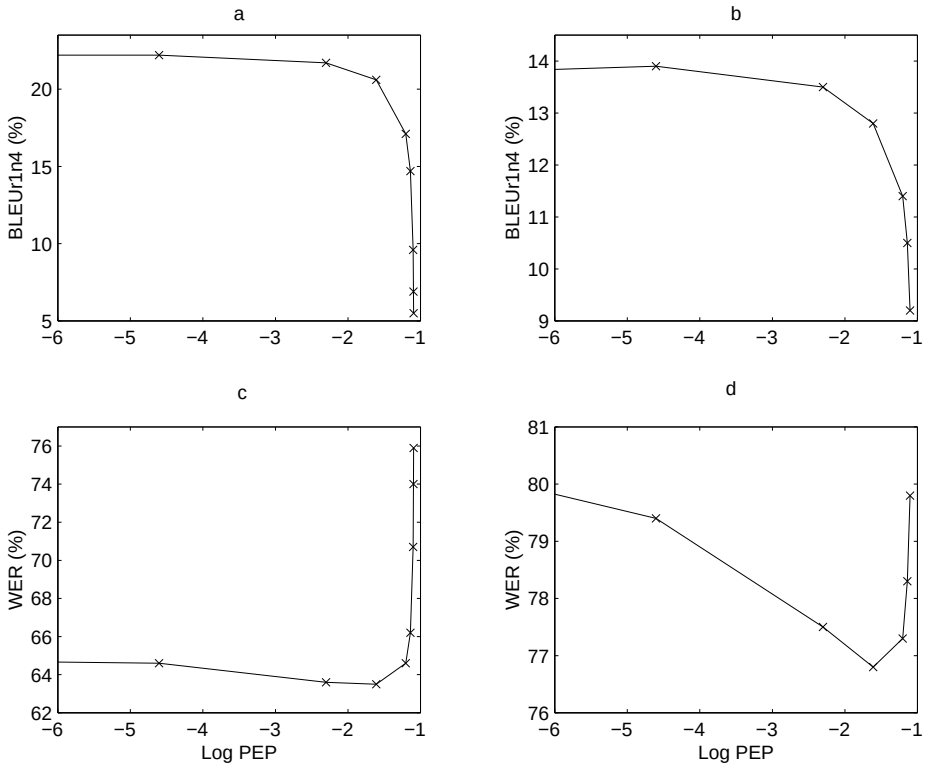

Figure 16: Translation Performance of the TTM as a function of the Phrase Exclusion Probability (PEP) when one reference translation is considered. We measure BLEU (Panel a,b) and WER (Panel c,d) on the French-English and the Chinese-English Tasks.

We next discuss how PEP influences BLEU. Since BLEU is influenced by both BPrecision (Equation 20) and Brevity Penalty (Equation 21), we plot these components separately in Figure 17. We note first that as PEP (α) increases, the translations grow shorter. This is measured by the Source-to-Target Length Ratio (STLRatio) (Figure 17d) which is the ratio of the number of words in the translation to number of words in the French sentence. This behavior is consistent with the role of PEP; it allows target phrases to delete in translation. The Brevity Penalty (Figure 17c) is governed by the number of words in the translation hypothesis, and therefore closely tracks the STLRatio. Somewhat surprisingly, BLEU score (Figure 17a) closely tracks the Brevity Penalty and does not improve despite improvements in BPrecision. Analogous to the case of bitext word alignment, increasing PEP allows the model to produce higher quality translation when BPrecision (Figure 17b) is taken alone. However, the interaction between BPrecision and Brevity Penalty is such that the shorter sentences, although of higher precision, incur a very high Brevity Penalty so that the increase in precision does not improve BLEU overall.

The behavior of BPrecision is interesting in itself. Intuitively, it should be possible to increase the PEP so that only the most likely phrase translations are retained and thus improve the

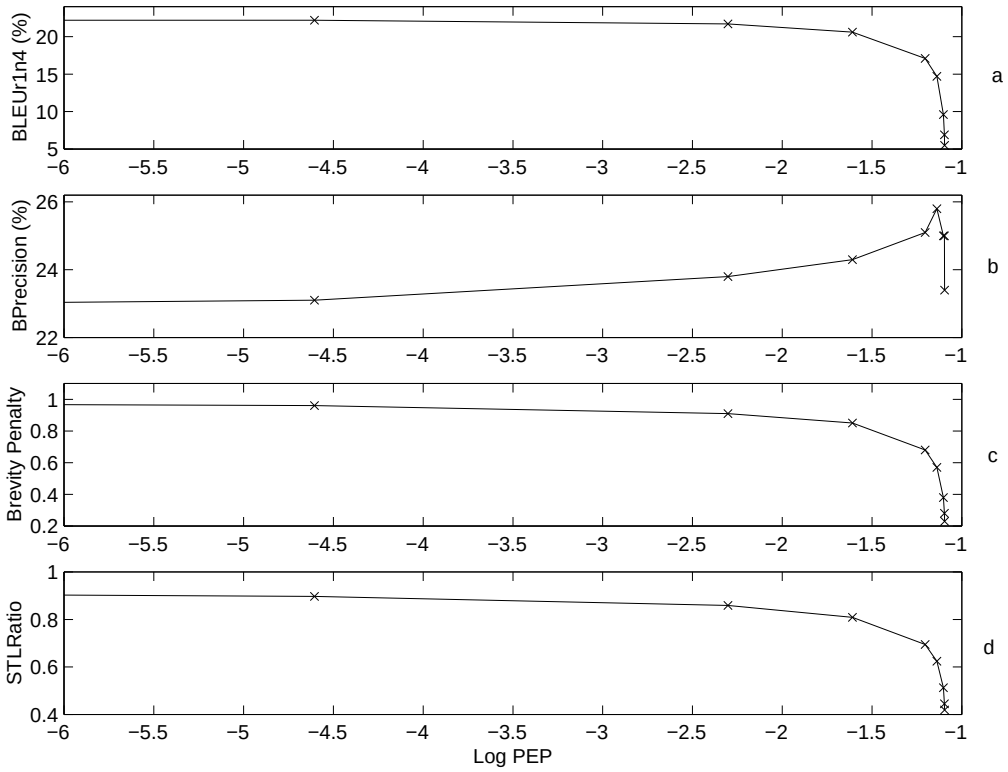

Figure 17: Translation Performance of the TTM as a function of the Phrase Exclusion Probability (PEP) on the French-English task. We measure BLEU (Panel a), BPrecision (Panel b), Brevity Penalty (Panel c), and STLRatio (Panel d) as functions of PEP.

BPrecision. However we note in Figure 17b that BPrecision itself falls off above a critical value of PEP.

To explain this behavior of BPrecision, we study the contribution to the BLEU precision of the four n-gram precision measures (Equation 22) in the French-English task (Figure 18). In the TTM, the dominant mechanism by which shorter translations are produced is to insert French phrases so that fewer French phrases are translated. As a result, English phrases in the translation arise from French phrases which are likely to be separated in the French sentence. It is correspondingly unlikely that English phrases in the translation (generated by separated French phrases) would follow each other in a fluent translation. Therefore the hypothesis translation contains phrases that are unlikely to be found next to each other in the reference translation. Consequently when precision statistics (Equation 22) are gathered over the translation, the hypothesized n-grams spanning these phrase boundaries are unlikely to be present in the reference translation, thus reducing precision. Figure 18 shows this behavior, the precision of higher n-grams ($n > 1$) falls off as the translations get shorter. Because of the need to account for n-grams spanning phrase boundaries, it is not possible to 'game' precision by merely producing shorter translations.

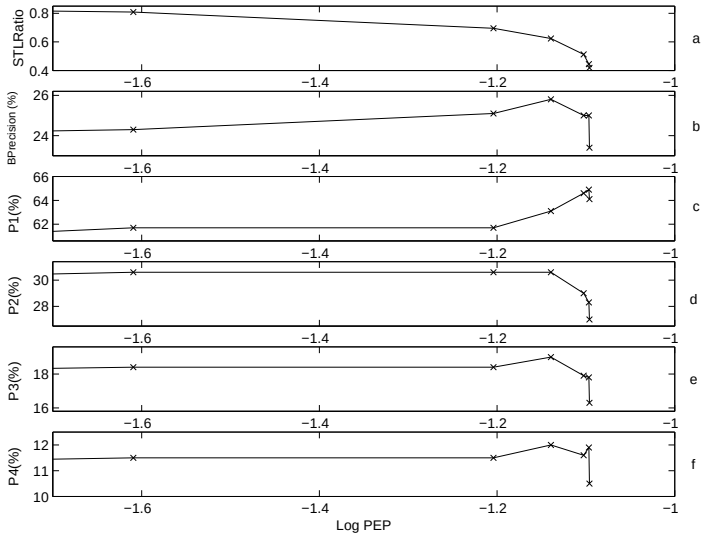

Figure 18: Analysis of BLEU Precision for values of Phrase Exclusion Probability (PEP) close to its maximum permissible value. We measure the following as functions of PEP : STLRatio (Panel a), BPrecision (Panel b) and each of the n-gram precisions, $n = 1, 2, 3, 4$ (Panels c-f). Results are shown on the French-English task.

We now discuss the translation performance when multiple reference translations are considered for measuring translation performance (Figure 19). The most notable difference between the four-reference and one-reference scenarios is that BLEU score actually shows a substantial increase as PEP varies when four references are available. Relative to the single reference case, over the range of PEP, BPrecision also shows a greater increase and the Brevity Penalty is less severe.

We can explain this behavior by noting that Brevity Penalty is less severe when multiple reference translations are available in scoring (Figure 19e,f). In the single reference and multiple reference cases, the BPrecision increases with PEP, although the absolute values of BPrecision in the multiple reference case is higher due to the greater diversity of n-grams in the references. However in the multi-reference case, there is a greater range of PEP values over which the Brevity Penalty has little influence on the overall BLEU score. Within this range, the BPrecision can be improved substantially by varying PEP so that BLEU shows a strong maximum.

6.3.2 Richness of the Phrase-Pair Inventory

We have described how the richness of the phrase-pair inventory can influence word alignment under the TTM (Section 6.2.2). We now investigate whether translation performance of the TTM might vary similarly. We used the four inventories described in Table 5 that are constructed using different amounts of bitext and we measure translation performance (under BLEU, NIST, and WER metrics) (Figure 20). In this experiment we measure performance at the optimal value of the PEP (α) that was determined previously (Section 6.3.1). As the bitext size employed to construct the inventory increases, we observe an improvement in performance as measured with

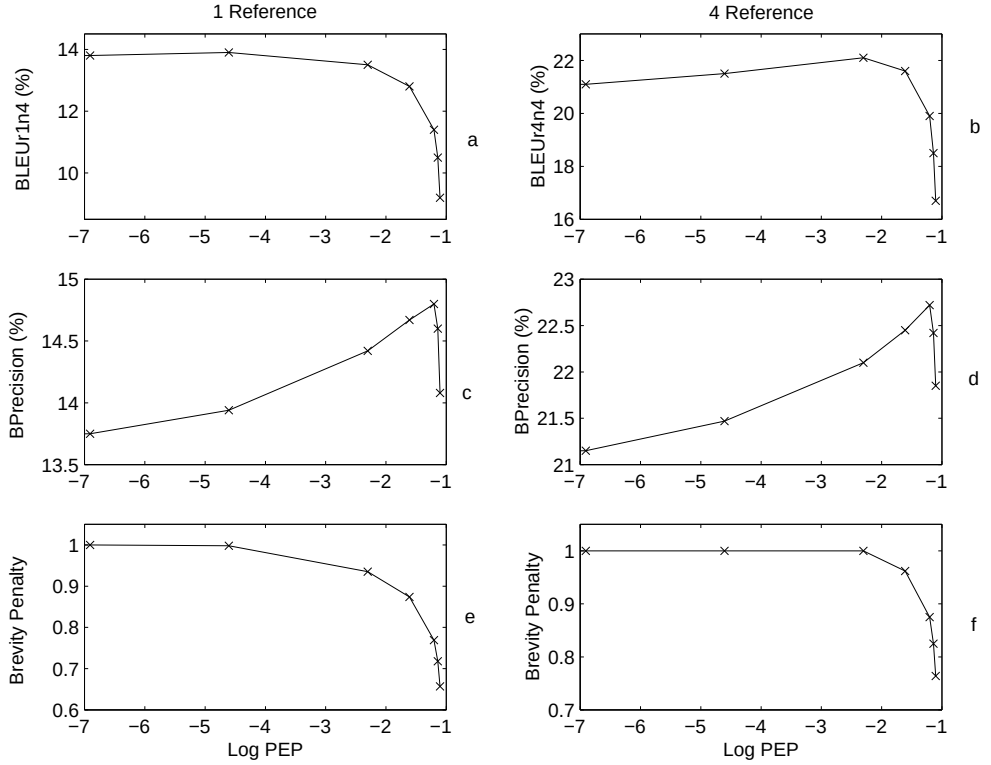

Figure 19: Translation Performance of TTM as a function of the Phrase Exclusion Probability when multiple reference Translations are considered for scoring. We obtain BLEU, BPrecision, and Brevity Penalty as functions of PEP in two situations: when 1 reference is considered (Panels a,c,e), and when 4 references are considered (Panels b,d,f).

respect to all three translation metrics. This shows that the additional data helps to improve coverage of the test set by the inventory, and therefore improves the translation performance.

6.3.3 Word Alignment Quality of Underlying IBM-4 Models

We have studied how word alignment performance of the TTM varies with the quality of its underlying IBM-4 models (Section 6.2.3). We now study translation performance of the TTM in a similar way. We use the four inventories as described in Table 6 which are built with varying amounts of bitext. We measure translation performance (under BLEU, NIST, and WER metrics) as a function of the bitext training set size in Figure 21, where we fix the PEP (α) to its optimal value found in Section 6.3.1. We observe that as the training set is increased, the AER of the IBM-4 models decreases, and the translation performance improves under all three translation metrics. We conclude that more bitext improves word alignment quality and in turn improves translation quality.

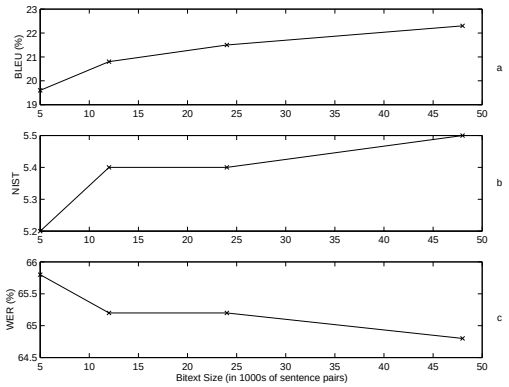

Figure 20: Effect of phrase-pair inventory size on translation performance. IBM-4 models are trained on 48K sentence-pairs from French-English Hansards, and word alignments are obtained over the collection. Four subsets are constructed from this set of word alignments and phrase-pair inventories were collected over each subset. For each inventory, translations under the TTM are obtained, and BLEU (Panel a), NIST (Panel b), and WER (Panel c) are plotted as functions of the bitext size employed to construct the inventory. Inventories are shown in Table 5.

6.3.4 Lattice Quality

The goal of this experiment is to study the usefulness of translation lattices for rescoring purposes. For this purpose we generate N-best lists of translation hypotheses from each translation lattice, and show the variation of their oracle-best BLEU scores with the size of the N-best list (Figure 22). The oracle-best BLEU score is obtained in the following way. For each sentence in the test set, we obtain the oracle hypothesis by selecting the hypothesis from N-best list with the highest sentence-level BLEU score. We concatenate these oracle hypotheses over all sentences in the test set and then measure the test-set BLEU score of the resulting hypothesis.

We observe that the oracle-best BLEU score sharply increases with the size of the N-Best List. We can therefore expect to rescore the lattices and N-best lists generated by TTM with more sophisticated models and achieve improvements in translation quality.

7 Discussion

We present the Translation Template Model (TTM) for statistical machine translation. We have developed this model with two intentions in mind. First the model should be formulated in a way that the conditional dependencies underlying the model are clearly stated. Second we intend to formulate the model in a way that allows bitext word alignment and translation under the model to be implemented using Weighted Finite State Transducer (WFST) operations.

The TTM is a source-channel model of the translation process. It defined a joint distribution over phrase segmentations, reorderings, and phrase-pair translations needed to describe how the source language sentence is translated into the target language.

The model relies on an underlying inventory of target language phrases and their source lan-

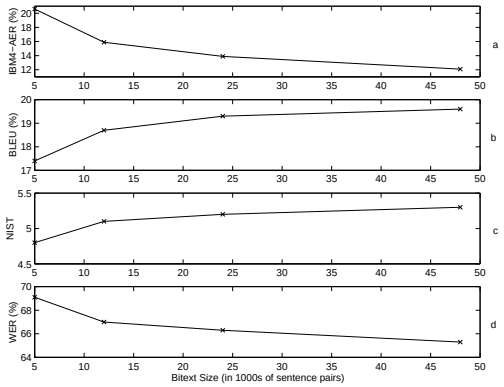

Figure 21: Translation performance of TTM as a function of the bitext size employed in training the underlying IBM-4 models. IBM-4 models are trained on four nested subsets of the French-English Hansards bitext, and word alignments are obtained over the smallest subset (5K sentence pairs). A phrase-pair inventory is constructed over each word alignment. For each inventory, translations under the TTM are obtained, and BLEU (Panel b), NIST (Panel c), and WER (Panel d) are plotted as functions of the bitext size employed in training the underlying IBM-4 models. We also measure AER of the underlying IBM-4 models (Panel a). Inventories are shown in Table 6.

guage translations. The manner by which the inventory is created does not affect the model formulation. In this paper we have employed IBM-4 word translation models to generate an initial bitext word alignment, and then collected the phrase-pair inventory over this alignment using a set of heuristics [19]. However any word alignment or methodology of collecting phrase pairs could be used.

The TTM consists of six component models each of which can be implemented independently as a weighted finite state acceptor or transducer. The TTM component models are the Source Language Model, Source Phrase Segmentation Model, Phrase Order Model, Target Phrase Insertion Model, Phrase Transduction Model, and the Target Phrase Segmentation Model.

The Source Language Model is a standard monolingual trigram word language model that assigns probabilities to source language sentences. The source sentences are then segmented into phrases under the Source Phrase Segmentation Model; the phrases remain in the phrase order of the source language. These source phrase sequences are then reordered into the phrase order of the target language under the Source Phrase Order Model. The Target Phrase Insertion Model then specifies the number and length of target phrases which will be spontaneously inserted within these reordered source phrase sequences. The Phrase Transduction model next transforms these sequences into sequences of target phrases. The composition of the previous models overgenerates the set of target language sentences; the Target Phrase Segmentation Model constrains these to agree with the target sentence.

The Translation Template Model can be used to perform both MAP word-level alignment of bitexts and translation of target language sentences. Once the component models of the TTM are implemented as weighted finite state transducers, we have shown how MAP word alignment and

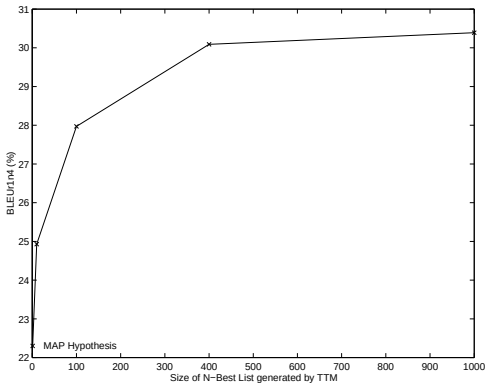

Figure 22: Variation of oracle-best BLEU scores with the size of the N-best list on the French-English Task. For each N-best list on the test set, the oracle BLEU hypothesis is computed under the sentence-level BLEU metric. The oracle hypotheses are concatenated over the test set, and the test-set BLEU score is measured.

translation can be obtained immediately using standard weighted finite state operations involving these transducers. In addition, these WFST operations facilitate generation of alignment and translation lattices without any extra effort in implementation.

This is first time that phrase-based models of this variety have been employed for bitext word alignment. The ability to do this is crucial in order to implement iterative parameter estimation procedures such as Expectation Maximization (EM) for this model. In general we note that a finite inventory of phrase-pairs is not rich enough to cover all possible sentences in any given bitext collection. As a result sentence-pairs from the collection can contain phrase-pairs not in the inventory; unless addressed, these sentence-pairs are therefore assigned a probability of zero under the model. We have described how modeling the deletion of source phrases during the alignment process can overcome this limitation, and thus make it possible for the TTM to be used to align any bitext.

We present a detailed experimental analysis of the TTM in early stages of its development, and analyzed several factors that influence alignment and translation performance. Our experiments are aimed at throwing light on the strengths and weaknesses of the model. We will now highlight some of the key results and conclusions that we draw from these experiments.

We first review the word alignment experiments. We observe that the Alignment Error Rate (AER) of the TTM is comparable to the baseline IBM-4 models on the French-English task, but worse than that of the IBM-4 models on the Chinese-English task. On both tasks the model obtains a very high Alignment Precision but a relatively poor Alignment Recall. The lower recall on the Chinese-English task can be attributed to the poorer coverage by the inventory of the test set.

Source and target phrases excluded during word alignment affect alignment performance of the TTM. This behavior is governed by varying the Phrase Exclusion Probability (PEP). Alignment Recall at first improves slightly with PEP but then decreases. Alignment Precision increases monotonically with PEP over most of its permissible range, however there is a critical value above

which Alignment Precision decreases. The initial increase in Alignment Precision suggests that as PEP increases the model favors phrase-pairs that yield higher quality word alignments than found in general. However as PEP is increased above the critical value, the percentage of excluded phrases increases sharply. As a result, the Alignment Precision drops even though the relatively few phrase-pairs that remain in the alignments are of high quality. We conclude from this behavior that we cannot 'game' Alignment Precision by arbitrarily decreasing the number of hypothesized alignment links.

The quality of underlying word alignments and the richness of the phrase-pair inventory both influence alignment performance of the TTM. If the underlying word alignment quality is held constant, the main influence of increasing bitext size is to increase phrase-pair coverage and consequently improve Alignment Recall. By contrast, if Alignment Recall can be held constant, the effect of increasing bitext size is to improve Alignment Precision of the TTM.

All phrase segmentations of the source sentence are generally considered when obtaining the MAP word alignment of sentence pairs under the TTM. An alternate approach is a two step procedure that consists of MAP phrase segmentation of the source sentence, followed by the MAP alignment of the fixed source phrase segmentation. We find that this two-step approach is markedly inferior relative to the exact MAP word alignment.

Excluding the source segmentation model has almost no impact on the alignment quality of the TTM. We conclude that this particular instance of the segmentation model is so weak that the overall alignment process is dominated by the phrase translation probabilities.

Reorderings of the source phrase sequence can be allowed during TTM word alignment. However there is only a slight improvement in AER by allowing any reordering. We observe that to induce phrase reorderings in the MAP word alignment, PEP must be set to a value that leads to a degradation in AER. In contrast, at the optimal value of AER, the Average Phrase Movement of the MAP word alignment is less than one word; this suggests that we can obtain the benefits of phrase reordering by increasing the maximum word length of the source phrases in the phrase-pair inventory. Furthermore, if we fix PEP and consider more reorderings, we observe a greater phrase movement in the MAP alignment. However, we find no gains in AER beyond a 100-best listing of reorderings.

The translation performance of the TTM compares favorably to that of the ReWrite decoder that employs the same set of IBM-4 translation models. We find that Phrase Exclusion Probability (PEP) influences translation performance of TTM. As PEP is increased, we observe that BLEU degrades but WER improves slightly at first before degrading. Examining this behavior shows that as PEP is increased, the translations become shorter and the Brevity Penalty increases while the BPrecision increases. However, the interaction between BPrecision and Brevity Penalty is such that the shorter sentences, although of higher precision, incur a very high Brevity Penalty so that the increase in precision does not improve BLEU overall. Furthermore, BPrecision itself falls off above a critical value of PEP; therefore it is not possible to 'game' BLEU Precision by merely producing shorter translations.

Interestingly, we find that the variation in BLEU score is sensitive to the number of reference translations used for scoring. The most notable difference between the four-reference and one-reference scenarios is that BLEU score actually shows a substantial increase as PEP varies when four references are available. We can attribute this difference to the greater diversity in the reference translations with respect to n-grams and length; therefore multiple reference translations are more permissive of variations in the hypothesized translation length.

The quality of underlying word alignments and richness of the phrase-pair inventory influence

translation performance of the TTM. When the alignment quality of underlying IBM-4 models is fixed, additional data helps to improve coverage of the test set by the inventory, and therefore improves the translation performance. On the other hand, we can fix the bitext collection from which the phrase-pair inventory is gathered and vary the amount of bitext for training the underlying IBM-4 models; we find that this improves word alignment quality of the underlying models and in turn improves translation quality of the TTM.

Finally we study the variation of oracle-best BLEU scores on N-best lists generated by the TTM. We observe that the oracle-best BLEU score sharply increases with the size of the N-Best List; this shows that we can expect to rescore the translation lattices with more sophisticated models and achieve improvements in translation quality. This concludes the overview of the TTM alignment and translation experiments.

8 Conclusion

The main motivation for our investigation into this WFST modeling framework for statistical machine translation lies in the simplicity of the alignment and translation processes relative to other dynamic programming or A^* decoders [19]. Once the components of the Translation Template Model are implemented as WFSTs, both word alignment and translation can be performed using standard FSM operations that have already been implemented and optimized. It is not necessary to develop specialized search procedures, even for the generation of lattices and N-best lists of alignment and translation alternatives.

Our derivation of the TTM was presented with the intent of clearly identifying the conditional independence assumptions that underly the WFST implementation. This approach leads to modular implementations of the component distributions of the translation model. These components can be refined and improved by changing the corresponding transducers without requiring changes to the overall search procedure.

It is a strength of the TTM that it can be directly constructed from bitext word alignments. However this construction should only be considered an initialization of the TTM model parameters. We expect word alignment and translation performance of the model to improve with further refinement in the models and through iterative parameter estimation schemes.

We have presented a novel approach to generate alignments and alignment lattices under the TTM. These lattices will likely be very helpful in developing TTM parameter estimation procedures, in that they can be used to provide conditional distributions over the latent model variables.

The Translation Template Model is a very promising modeling framework for statistical machine translation. The model offers a simple and unified framework for bitext word alignment and translation. The simplicity of the model has allowed us to perform a detailed investigation of the several factors that influence the alignment and translation performance of the model; we believe that this analysis has improved our understanding of the strengths and weaknesses of the model. It is our goal in future to improve the model by increasing complexity as well as through novel parameter estimation schemes.

References

[1] C. Allauzen, M. Mohri, and B. Roark. Generalized algorithms for constructing statistical language models. In *Proc. of ACL*, pages 40–47, Sapporo, Japan, 2003.

[2] S. Bangalore and G. Ricardi. A finite-state approach to machine translation. In *Proc. of the North American Chapter of the Association for Computational Linguistics*, Pittsburgh, PA, USA, 2001.

[3] P. F. Brown, J. Cocke, S. A. Della Pietra, V. J . Della Pietra, F. Jelinek, J. D. Lafferty, R. L. Mercer, and P. S. Roossin. A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85, 1990.

[4] P. F. Brown, S. A. Della Pietra, V. J. Della Pietra, and R. L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, 1993.

[5] W. Byrne, S. Khudanpur, W. Kim, S. Kumar, P. Pecina, P. Virga, P. Xu, and D. Yarowsky. The Johns Hopkins University 2003 Chinese-English machine translation system. In *MT Summit IX*, New Orleans, LA, USA, 2003.

[6] The People's Daily, 2002. http://www.english.people.com.cn.

[7] Y. Deng and W. Byrne. Statistical chunk alignment models for machine translation. CLSP Tech Report, Jan 2003.

[8] G. Doddington. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proc. of HLT 2002*, San Diego, CA. USA, 2002.

[9] U. Germann, M. Jahr, K. Knight, D. Marcu, and K. Yamada. Fast decoding and optimal decoding for machine translation. In *Proc. of COLING*, New Brunswick, NJ, USA, 2001.

[10] K. Knight and Y. Al-Onaizan. Translation with finite-state devices. In *Proc. of the AMTA Conference*, pages 421–437, Langhorne, PA, USA, 1998.

[11] P. Koehn, F. Och, and D. Marcu. Statistical phrase-based translation. In *Proc. of the Conference on Human Language Technology*, Edmonton, Canada, 2003.

[12] S. Kumar and W. Byrne. Minimum Bayes-risk alignment of bilingual texts. In *Proc. of the Conference on Empirical Methods in Natural Language Processing*, Philadelphia, PA, USA, 2002.

[13] S. Kumar and W. Byrne. A weighted finite state transducer implementation of the alignment template m odel for statistical machine translation. In *Proc. of the Conference on Human Language Technology*, Edmonton, Canada, 2003.

[14] LDC. *Chinese Segmenter*, 2002. http://www.ldc.upenn.edu/Projects/Chinese.

[15] D. Marcu and U. Germann. *The ISI ReWrite Decoder Release 0.7.0b*, 2002. http://www.isi.edu/licensed-sw/rewrite-decoder/.

[16] D. Marcu and W. Wong. A phrase-based, joint probability model for statistical machine translation. In *Proc. of the Conference on Empirical Methods in Natural Language Processing*, Philadelphia, PA, USA, 2002.

[17] M. Mohri, F. Pereira, and M. Riley. *ATT General-purpose finite-state machine software tools*, 1997. http://www.research.att.com/sw/tools/fsm/.

[18] NIST. The NIST machine translation evaluations, 2003. http://www.nist.gov/speech/tests/mt/.

[19] F. Och. *Statistical Machine Translation: From Single Word Models to Alignment Templates*. PhD thesis, RWTH Aachen, Germany, 2002.

[20] F. Och and H. Ney. Improved statistical alignment models. In *Proc. of ACL-2000*, pages 440–447, Hong Kong, China, 2000.

[21] F. Och, C. Tillmann, and H. Ney. Improved alignment models for statistical machine translation. In *Proc. of the Joint Conf. of Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 20–28, College Park, MD, USA, 1999.

[22] F. Och, N. Ueffing, and H. Ney. An efficient A* search algorithm for statistical machine translation. In *Proceedings of ACL*, pages 55–62, Toulouse, France, July 2001.

[23] K. Papineni, S. Roukos, T. Ward, and W. Zhu. Bleu: a method for automatic evaluation of machine translation. Technical Report RC22176 (W0109-022), IBM Research Division, 2001.

[24] Canadian Parliament. Canadian hansards, 2003. http://www.parl.gc.ca/.

[25] A. Stolcke. SRILM – an extensible language modeling toolkit. In *Proc. of the International Conference on Spoken Language Processing*, pages 901–904, Denver, CO, USA, 2002. http://www.speech.sri.com/projects/srilm/.

[26] C. Tillmann. *Word Reordering and Dynamic Programming based Search Algorithm for Statistical Machine Translation*. PhD thesis, RWTH Aachen, Germany, 2001.

[27] C. Tillmann. A projection extension algorithm for statistical machine translation. In *Proc. of the Conference on Empirical Methods in Natural Language Processing*, Sapporo, Japan, 2003.

[28] N. Ueffing, F. Och, and H. Ney. Generation of word graphs in statistical machine translation. In *Proc. of the Conference on Empirical Methods in Natural Language Processing*, pages 156–163, Philadelphia, PA, USA, 2002.

[29] S. Vogel, H. Ney, and C. Tillmann. Hmm based word alignment in statistical translation. In *Proc. of the COLING*, pages 836–841, Copenhagen, Denmark, 1996.

[30] Y. Wang and A. Weibel. Decoding algorithm in statistical machine translation. In *Proc. of ACL*, Madrid, Spain, 1997.

[31] F. Wessel, K. Macherey, and R. Schlueter. Using word probabilities as confidence measures. In *Proc. of ICASSP-98*, pages 225–228, Seattle, WA, USA, 1998.

[32] Y. Zhang, S. Vogel, and A. Weibel. Integrated phrase segmentation and alignment model for statistical machine translation. In *Proc. of the NLPKE*, Beijing, China, 2003.