Natural Language Engineering **12** (1): 35–75. © 2005 Cambridge University Press
doi:10.1017/S1351324905003815 First published online 6 December 2005 Printed in the United Kingdom

A weighted finite state transducer translation template model for statistical machine translation

SHANKAR KUMAR, YONGGANG DENG 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
email: {skumar,dengyg,byrne}@jhu.edu

(*Received 24 March 2004; revised 8 September 2004*)

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 FBIS Chinese-English task 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. We finally discuss translation performance with large bitext training sets on the NIST 2004 Chinese-English and Arabic-English MT tasks.

1 Introduction

The premise underlying the statistical approach to automatic machine translation is that statistics describing the translation of words and word sequences (phrases and sentences) can be reliably and consistently extracted from large collections of example translations (bitext) and used to create systems capable of translation. In general terms, the methodology is to formulate a model of translation, refine the model using information derived from the available bitext, and embed the model in a system to generate translations of sentences not seen in the original bitext.

The original and influential work in this area is a generative source-channel model of translation (Brown *et al.* 1990; Brown *et al.* 1993). Bitext is modeled as observations of a stochastic process that first generates source (English) sentences and then transforms them into target (French) sentences by reordering the English words and then translating them into French. As usual with generative probabilistic models, translation is performed by inverting the channel. Given a French sentence, the system searches for the English sentence that, under the model, is most likely to have generated the French sentence. This English sentence is chosen as the translation hypothesis. The original framework was a series of models of increasing complexity, known as IBM Models 1 through 5, along with estimation procedures by which the models can be refined from bitext. To date, the IBM models have been found to be very effective for modeling word alignment in bitext but less effective in actual translation, despite enhancements to the IBM models themselves (Vogel, Ney and Tillmann 1996; Och and Ney 2000) and improved translation search algorithms (Wang and Waibel 1997; Knight and Al-Onaizan 1998; Tillmann and Ney 2003; Och, Ueffing and Ney 2001; Germann *et al.* 2001). In translation, the IBM models are particularly weak relative to the Alignment Template Model developed by Och, Tillmann and Ney (1999) which overcomes the limitations of word-to-word translation models by using phrases rather than words as the basis for translation.

One of the challenges in the source-channel approach to translation is that the transformation of a generative model into an effective translation system, or decoder, is rarely straightforward. Searching for the best English translation of a given French sentence becomes more difficult as the underlying model becomes more powerful. If a model formulation cannot be transformed into an efficient and exact decoder, translation can be achieved only through simplifying approximations that may yield sub-optimal translations and thus undo even powerful models of translation. The best strategy in formulating a model may be to forgo some descriptive power to ensure that translation can be performed exactly.

In this paper we formulate a generative, source-channel model for phrase-based translation in such a way that its transformation into a translation system is direct and clear. The model is named the *Translation Template Model* (TTM) and it defines a series of stochastic transformations by which a French sentence is generated from an English sentence via the translation, reordering, insertion, and deletion of phrases. Each of these stochastic transformations will be formulated so that it can be implemented as a Weighted Finite State Transducer (WFST). This approach simultaneously achieves two results. The stochastic transformations come together to define a complete probability distribution over French-English sentence pairs. Furthermore, translation and bitext word alignment can be realized almost immediately by standard algorithms to merge the component models into an overall processing system (Mohri, Pereira and Riley 1997). This is the great value of WFST-based modeling. Stochastic model formulation and realization are indistinguishable and are achieved as one.

Consequently our approach focuses almost entirely on modeling. There is almost no effort spent on decoder design. Instead, the models are designed so that the overall

system can be exactly realized by a composition of the component transducers. It may happen that one of the component models must be simplified for the overall system to be realized efficiently. An example is in the modeling of phrase movement. The overall model permits English phrases to be reordered arbitrarily, and we do explore the role of phrase movement in bitext word alignment. However for reasons that we will discuss, in translation we disable the phrase movement component model to disallow phrase reordering. This is an instance of sacrificing modeling power to ensure that the overall system is realized exactly. We may have a slightly weaker translation model in that it does not allow phrase movement, but this weakness is clearly identified and we know that the translation hypotheses produced will be exact with respect to the model.

Our approach is very much influenced by the description of the IBM models as WFSTs by Knight and Al-Onaizan (1998). In addition to showing how the IBM Models 1 through 3 can be implemented as WFSTs, they give a tutorial explanation of the role played by each model component. Motivated by their work, we developed WFST-based bitext word alignment algorithms and explored their use under various alignment criteria (Kumar and Byrne 2002). We found however that the performance of that approach was limited by its reliance on the IBM-3 model, which is the most complex of the IBM models that can easily be formulated as a WFST. This experience led us to investigate WFST implementations of phrase-based translation models. We developed an alignment template formulation that can be implemented using weighted finite state transducers (Kumar and Byrne 2003) and in doing so 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 submitted in the NIST 2003 MT evaluations (Byrne, Khudanpur, Kim, Kumar, Pecina, Virga, Xu and Yarowsky 2003).

The TTM as described here departs from the original Alignment Template Model (Och, Tillmann and Ney 1999; Och 2002) in several ways. In addition to the complete formulation of the source-channel generative model of phrase-based translation and its implementation via WFSTs, the TTM is wholly phrasal in that it makes no use of word-to-word alignments at any level; but as we will show this does not prohibit using the TTM for bitext word alignment. We furthermore allow the insertion of target language phrases within the generative translation process and this removes the restriction that the source and target language sentences contain the same number of phrases.

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) (Och and Ney 2000), BLEU (Papineni, Roukos, Ward and Zhu 2001), NIST score (Doddington 2002), and multi-reference Word Error Rate (Och 2002). 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 model component to overall translation

performance. In doing so we also analyze some aspects of the performance metrics themselves; these criteria are complex enough that they can be said to have behavior of their own.

There has been 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 (Tillmann 2003; Koehn, Och and Marcu 2003) or not (Zhang, Vogel and Waibel 2003; Marcu and Wong 2002). Bangalore and Riccardi (2001) have also explored the use of WFSTs for machine translation and we briefly summarize their work to contrast it to our own. They first construct word-level alignment of sentence pairs using a tree-based alignment procedure incorporating a synchronous dependency grammar and from these alignments they extract word-level and phrase-level lexicons that are represented using WFSTs. They also extract lexical reordering schemas from the bitext alignment and encode these in a second WFST. These WFSTs are used in a two step translation process in which the first WFST is used to map the target sentence into a sequence of source language words that remain in their original target language word order. Reordering into source language word order is then done by the second WFST. These transducers are constructed independently from word aligned bitext. In contrast our model supports both bitext word alignment and translation. It is based on a joint translation and reordering process that works by reordering both phrases and their constituent words. Moreover our work, as well as that of Knight and Al-Onaizan (1998), is different in spirit in that we focus on a complete source-channel model of translation which can be realized using WFSTs.

The article is organized as follows. In section 2 we present the TTM formulation. 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 describe exploratory alignment and translation experiments on the Hansards French-English and the FBIS Chinese-English tasks. We report Arabic-English and Chinese-English translation performance with large bitext training sets in section 7 and we discuss these experiments in section 8 and conclude in section 9.

2 The translation template model

The Translation Template Model (TTM) is a source-channel model of translation (Brown, Cocke, Della Pietra, Della Pietra, Jelinek, Lafferty, Mercer and Roossin 1990). It defines a joint probability distribution over all possible phrase segmentations and alignments of target language sentences and their translations in the source language. The steps in the translation process are presented with the aid of an example in figure 1, and the conditional dependencies underlying this process are defined in equation 1.

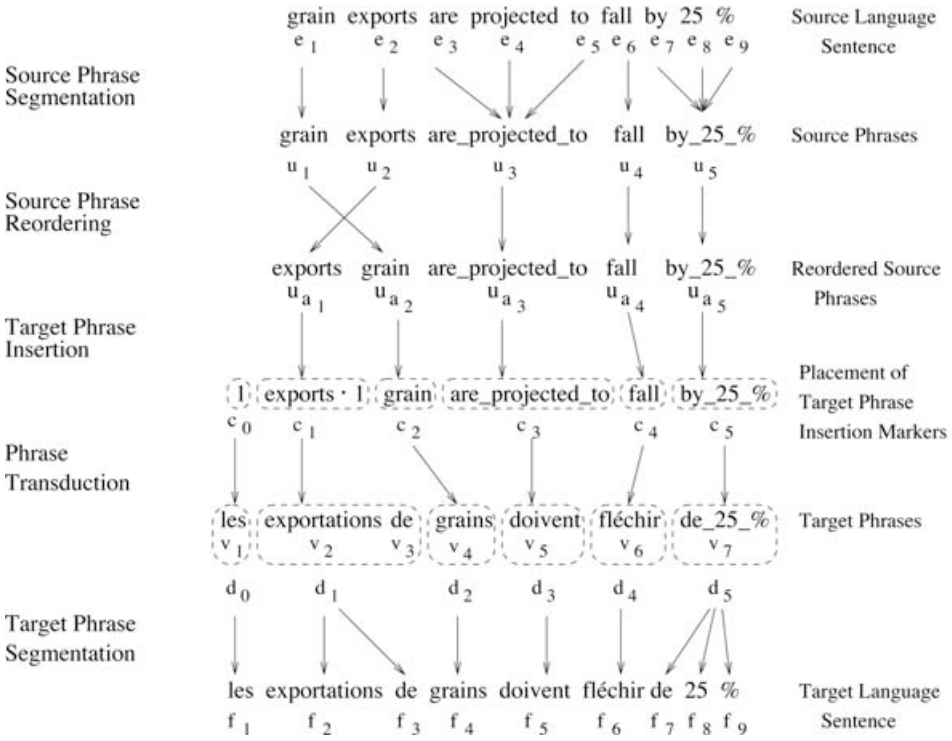

Fig. 1. An example showing the generative translation process through which the TTM transforms a source language sentence into its translation in the target language. We show the inputs and outputs for each TTM constituent model as well as the TTM variables from Equation 1. In this example, $I = 9, K = 5, R = 7, J = 9$.

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

We start with an example (figure 1) showing the generative process through which the TTM transforms a source language sentence into its translation in the target language. In this example, the Source Language Model generates the Source Language Sentence *grain exports are projected to fall by 25%*. This sentence is segmented into a source phrase sequence: *grain exports are_projected_to fall by_25_%* under the Source Phrase Segmentation Model. This source phrase sequence is reordered into the target language phrase order: *exports grain are_projected_to fall by_25_%* under the Phrase Order Model. The reordered source phrase sequence is

then transformed into a sequence: *1 exports*·1 *grain are_projected_to fall by_25_%*, where the integers indicate the length of target phrases to be spontaneously inserted; this process is governed by the Target Phrase Insertion Model. The above sequence is next converted into a target language phrase sequence *les exportations de grains doivent fléchir de_25_%* under the Phrase Transduction Model. We note that the words *les* and *de* are spontaneously inserted. Finally the target language phrase sequence is transformed into the target language sentence: *les exportations de grains doivent fléchir de 25%* under the Target Phrase Segmentation Model. It should be understood that all of the above steps are stochastic, and the example shown is only one possible realization.

We now define some notation. e_1^I refers to a sequence of I elements, and e_i^j refers to the subsequence that begins with the i^{th} element and ends with the j^{th}, *e.g.* if $e_1^I = A\ B\ C\ D$, then $e_2^3 = B\ C$, where $I = 4$. We next distinguish words and phrases. We assume that u is a phrase in the source language sentence that consists of a variable number of words $e_1, e_2, \ldots, e_M$. Similarly, v is a phrase in the target language sentence of words $f_1, f_2, \ldots, f_N$. 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. This inventory will be used in the creation of both the Phrase Transduction Model and the Source Phrase Segmentation Model. 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.

We utilize the *phrase-extract* algorithm (Och 2002) to extract a library of phrase-pairs from bitext word alignments for the experiments that will be presented in this paper. We first obtain word alignments of bitext using IBM-4 word level translation models (Brown, Della Pietra, Della Pietra and Mercer 1993) 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 phrase-extract algorithm to identify pairs of phrases (u, v) in the target and source language that align well according to a set of heuristics (Och 2002). The phrase pairs are gathered so that the source and target words within a phrase pair are aligned only to each other and not to any words outside the phrase-pair (Och 2002; Koehn, Och and Marcu 2003).

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 2 we show the extraction of phrase-pairs from bidirectional word alignments of an English-French sentence pair. We augment the phrase-pair inventory by the most likely translations of each target (source) word from the IBM-4 translation tables (Brown *et al.* 1993) so as to get complete coverage of all single word phrases in either language. We note

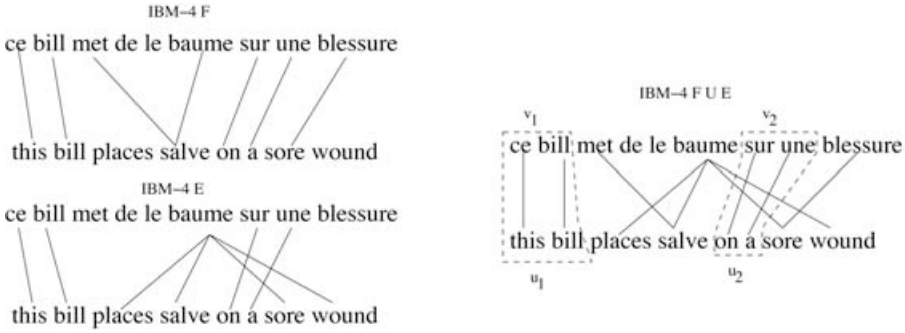

Fig. 2. Phrase-Pair collection process from bidirectional word alignments of an English-French sentence pair.

that monolingual phrase inventories can be created by projecting the phrase-pairs onto either the target or the source language.

We retain the matrix of word alignments that occurs most frequently for each pair of source and target phrases. The best phrase-to-phrase alignment of two sentences is easily obtained under the TTM (section 5). Once this alignment is found, the best word-to-word alignments of the aligned phrases are found using the matrices extracted from the training bitext.

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

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

Any n-gram or other language model that can be easily compiled as a weighted finite state acceptor could be used (Allauzen, Mohri and Roark 2003). We will use G to denote the language model WFSA.

4.2 Source phrase segmentation model

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

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

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

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

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

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

where C is chosen to ensure that the above model is normalized, which means that $\sum_{u_1^K} P(u_1^K|K, e_1^I) = 1$. This distribution assigns a uniform likelihood to all phrase segmentations of the source sentence that can be obtained using the inventory of phrases.

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 3. The '_' symbol is used to indicate phrases formed by concatenation of consecutive words. Using W, we construct a WFST for the distribution $P(u_1^K|K, e_1^I)$ to ensure that $\sum_{u_1^K} P(u_1^K|K, e_1^I) = 1$ for each source sentence e_1^I and $K \in \{1, 2, \ldots, I\}$. We have described this WFST construction procedure in earlier work (Kumar and Byrne 2003).

4.3 Phrase order model

We now define a model for the reordering of the source phrase sequence that makes 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 $u_{a_1}, u_{a_2}, \ldots, u_{a_K}$ 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

We begin by defining a first order Markov process over phrase alignment sequences

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

with $a_k \in \{1, 2, \ldots, K\}$. The phrase alignment sequence is further constrained to be a valid reordering of u_1^K, i.e. the phrase alignment sequence is constrained to be a permutation of the set $\{1, 2, \ldots, K\}$. 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

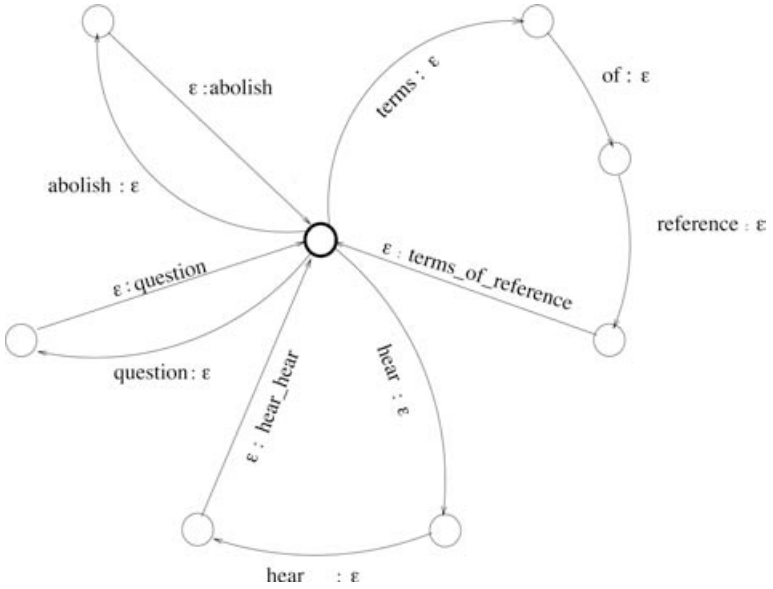

Fig. 3. A portion of the Source Phrase Segmentation Transducer W that maps word sequences to phrases. There is a distinct loop for each phrase in the source language phrase inventory. Suppose an example input for this transducer is the source language sentence: *What are its terms of reference*, then a possible output of this WFST would be the source language phrase sequence: *what_are its terms_of_reference*.

probabilities as follows (Och, Tillmann and Ney 1999)

$$P\left(a_k|a_{k-1}, u_1^K\right) \propto p_0^{|l-m'-1|} \tag{5}$$

$$P(a_1 = k) = \frac{1}{K}; k \in \{1, 2, \ldots, K\}.$$

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 (Knight and Al-Onaizan 1998; Kumar and Byrne 2003). Suppose this acceptor allows reorderings of the source language phrase sequence *we have run_away_inflation*. An example of a reordering would be *run_away_inflation we have*, so that the alignment sequence is given by: $a_1 = 3, a_2 = 1, a_3 = 2$.

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 (Kumar and Byrne 2003). Suppose 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*). Given a reordering of this phrase sequence *run_away_inflation we have* with alignment sequence $a_1 = 3$, $a_2 = 1$, $a_3 = 2$, H would assign it a probability: $P(a_1 = 3)P(a_2 = 1|a_1 = 3)P(a_3 = 2|a_2 = 1) = 0.33 \times 0.47 \times 0.53 = 0.08$.

4.3.2 Monotone 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

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

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

4.4 Target phrase insertion model

The steps introduced so far segment the source language sentence into phrases and then reorder the phrases. But it would be overly restrictive to insist that the source language phrase sequence have the same number of phrases as the target language phrase sequence. In particular, we wish to allow for the spontaneous insertion and deletion of phrases. It may happen that the bitext translations are not literal, so that not all target language phrases can be generated by phrases in the source language sentence. Even for literal translations, we may wish to allow for flaws in the phrase-pair inventory. We note that these phenomena could be captured to a large extent by allowing null phrase translations in the phrase-pair inventory, although the process would then be conflated with phrase translation.

We 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 so 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 $u_{a_1}, u_{a_2}, \ldots, u_{a_k}$ 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 $u_{a_1}, u_{a_2}, \ldots, u_{a_k}$, an element in the transformed sequence has the following form

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

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: $p_k = p_k[1] \cdot p_k[2] \cdot \ldots$ and $p_k[i] \in \{1, 2, \ldots, M\}$. 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

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

This distribution over c_k is specified through the Phrase Exclusion Probability (PEP), denoted by α. α^n is the probability of inserting a target language phrase of length n. In Section 6.2.1 we will show that the PEP can be used to tune alignment and translation by governing the tendency of the systems to insert target language phrases. The parameter α_0 is the probability that no target language phrase is inserted and it is dependent on α so that equation 7 sums to one (see below).

We note that $c_0, c_1, \ldots, c_k$ contains one additional term relative to the original sequence $u_{a_1}, u_{a_2}, \ldots, u_{a_k}$. 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

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

We now set the value of α_0 to ensure that the probability distribution (given in Equation 7) is normalized.

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

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

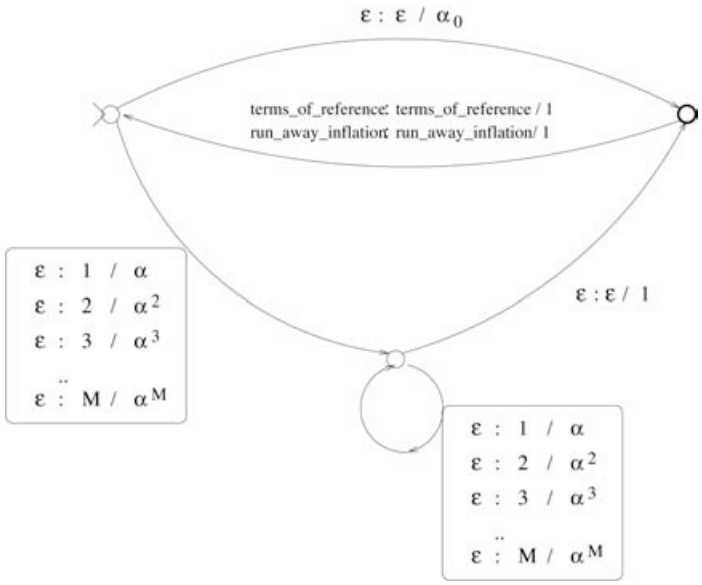

Fig. 4. A portion of the Weighted Finite State Transducer Φ used to implement the Target Phrase Insertion Model. Suppose an example input for this transducer is the reordered source language phrase sequence *exports grain are projected_to_fall*, then a possible output of the WFST is the sequence 1 *exports* · 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*.

its terms given by

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

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 4. 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 and we have described the process by which target phrases are spontaneously inserted within this reordered source phrase sequence. The next step is to map the sources phrases in this sequence to 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 sequence of target phrases d_k which are concatenated to

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

Source phrase	Target phrase	Phrase transduction probability
run_away_inflation	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

obtain the final target phrase sequence $v_1^R = d_0^K$.

$$P\left(v_1^R, d_0^K | c_0^K, a_1^K, u_1^K, K, e_1^I\right) = P\left(d_0^K | c_0^K\right) 1\left\{d_0^K = v_1^R\right\} \tag{10}$$

$$P\left(d_0^K | c_0^K\right) = \prod_{k=0}^{K} P(d_k | c_k)$$

$$= \prod_{l=1}^{|p_0|} P(d_0 | c_{0l}) \prod_{k=1}^{K} \prod_{l=1}^{1+|p_k|} P(d_{kl} | c_{kl}),$$

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 $m \in \{1, 2, \ldots, M\}$, 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

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

A small portion of the phrase-pair inventory used in constructing the transducer Y is shown in Table 1.

4.6 Target phrase segmentation model

The operations described so far allow a mapping of a source language sentence into a sequence of target language phrases. We now specify a model to enforce the constraint that words in the target sentence f_1^J agree with those in the target phrase sequence v_1^R.

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

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 a simple unweighted segmentation transducer Ω that maps target phrase sequences to target sentences. We build a transducer Ω for each target language sentence f_1^J to be translated. Suppose we have built the segmentation transducer Ω for the target language sentence: *nous avons une inflation galopante*. When Ω is composed with a valid phrase segmentation, *e.g. nous avons une_inflation_galopante*, it generates the target sentence: *nous avons une inflation galopante*.

5 Bitext word alignment and translation under the TTM

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

5.1 Bitext word alignment

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

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

$\hat{u}_1^{\hat{K}}$ 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 word 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

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

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 $\mathscr{B}$ based on likelihoods or number of states. The MAP alignment $\hat{B}$ (equation 12) is found as the path with the highest probability in $\mathscr{B}$.

If only one phrase segmentation of the source sentence is to be considered during alignment, we follow a two-step procedure proposed earlier (Kumar and Byrne 2003) 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.

$$\{\tilde{u}_1^{\tilde{K}}, \tilde{K}\} = \underset{u_1^K, K}{\operatorname{argmax}} P\left(u_1^K, K \middle| e_1^I\right) \tag{13}$$

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

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

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

$$\Pi_{\tilde{U}}^N = \text{N-Best Paths}(\Pi_{\tilde{U}} \circ H) \tag{14}$$

$$\mathscr{B} = \Pi_{\tilde{U}}^N \circ \Phi \circ Y \circ \Omega \circ S.$$

5.2 Translation

The translation of a target language sentence f_1^J into the source language can be found via MAP decoding as:

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

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:

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

A translation lattice (Ueffing, Och and Ney 2002) can be generated by pruning $\mathscr{T}$ 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 probability in $\mathscr{T}$.

5.3 Source phrase deletion in bitext word alignment

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. We see such an example in figure 2 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. 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.

6 Exploratory translation and alignment experiments

We now report alignment and translation performance of the Translation Template Model. We present experiments on two tasks that involve both word alignment and translation - the Hansards French-to-English task (Och and Ney 2000) and the FBIS Chinese-to-English task.

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 (Hansards 2003) maintained in both English and French. The translation model training data consists of 48,739 French-English sentence pairs from the Hansards (Och and Ney 2000). 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 (Och and Ney 2000).

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.

Table 2. *Distribution of the number of words in the target and source phrases over the Phrase-Pair Inventory on the French-English Hansards Task. The entries are phrase-pair counts (in 1000s), and the bold entries denote the maximum count in each row*

Target phrase length (in French words)	Source phrase length (in English words)							
	1	2	3	4	5	6–7	8–10	≥ 11
1	**414.3**	53.1	10.7	2.1	0.5	0.1	0.0	0.0
2	102.8	**190.0**	44.3	12.1	3.2	1.1	0.1	0.0
3	27.8	89.9	**119.5**	35.0	10.8	4.7	0.5	0.0
4	6.8	30.1	73.6	**79.1**	27.7	13.6	1.9	0.1
5	1.7	9.9	29.4	**57.2**	55.5	31.8	5.7	0.4

6.1.2 Chinese-to-English

The goal of this task (NIST 2004) is the translation of news stories from Chinese to English. The translation model training data consists of the Foreign Broadcast Information Service (FBIS) Chinese-English parallel corpus (LDC2003E14) 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 Linguistic Data Consortium (LDC) segmenter (LDC 2002). The provided bitext is aligned at the document level. Documents are aligned automatically at sentence and sub-sentence level into chunk-pairs using a statistical chunk model (Deng and Byrne 2004) 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) (People's Daily 2002) and the English side of the Xinhua Chinese-English parallel corpus (LDC2002E18) (4.3M words). The total language model corpus size is 21M words.

Our translation test set is the NIST 2002 MT evaluation set (LDC2003T17) 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 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.

6.2 Bitext word alignment

The goal of bitext word alignment is to find word-to-word correspondences between a pair of translated sentences. We measure performance with respect to a reference

Table 3. *Distribution of the number of words in the target and source phrases over the Phrase-Pair Inventory on the Chinese-English FBIS Task. The entries are phrase-pair counts (in 1000s), and the bold entries denote the maximum count in each row*

Target phrase length (in Chinese words)	Source phrase length (in English words)							
	1	2	3	4	5	6	7–8	≥ 9
1	**3,142.3**	1,720.3	775.3	266.2	80.1	24.6	12.2	4.0
2	705.3	**1,461.5**	1,134.8	635.5	295.4	123.2	69.1	18.6
3	149.4	479.1	**781.0**	696.2	461.9	262.6	201.7	64.9
4	34.1	130.7	300.5	**451.3**	441.1	340.7	359.7	162.5
5	9.1	34.2	95.8	196.1	284.0	**300.2**	449.4	314.3

word alignment created by a competent human translator. We evaluate alignment performance against the reference alignment using Alignment Precision, Alignment Recall and Alignment Error Rate (AER) metrics (Och and Ney 2000).

An *alignment* between a pair of source and target sentences e and f is defined to be a link set $B = \{b_1, b_2, \ldots, b_m\}$ 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. In these measurements, links to the NULL word are ignored. This is done by defining modified link sets $\bar{B}$ for the reference alignment and $\bar{B}'$ for the automatic alignment.

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 (Och and Ney 2000):

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

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

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

We present in Table 4 word alignment performance of the TTM on the two alignment tasks. We will use the Fixed Phrase Order Model (equation 6) in all the TTM experiments presented here. We will justify the choice of this model through the experiments in section 6.2.3.

As a basis for comparison we measure the Alignment Error Rate (AER) without any word reordering. The alignment between words in a sentence-pair is set proportionally, i.e. a target word at position j in a J-length target language sentence is aligned to the source word at position $i = \text{ceil}(j * I/J)$ in the I-length source

Table 4. *TTM alignment performance on the French-English and the Chinese-English alignment tasks*

	Alignment metrics (percent)					
	French-English			Chinese-English		
Model	Precision	Recall	AER	Precision	Recall	AER
Linear Order	43.6	35.3	59.2	16.5	12.9	85.5
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

language sentence. We note that the 'ceil' function is used to round a real number upward to the nearest integer value. AER results are given in Table 4 for this simple alignment. Comparison to other methods shows that linear order alignment is clearly inferior and that word reordering is crucial.

We also align the bitext using IBM-4 word translation models (Brown *et al.* 1993; Och and Ney 2000) trained in both translation directions (IBM-4 E and IBM-4 F), and their union (IBM-4 $E \cup F$).

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 TTM obtains a very high Alignment Precision but a relatively poor Alignment Recall unlike IBM-4, which is more balanced in Alignment Precision and Alignment Recall.

The TTM is the more conservative of the methods. It hypothesizes word alignments only within phrase-pairs that were encountered in training, and the hypothesized word alignments are in fact those assigned by IBM-4 to the training bitext. In this way it achieves high Alignment Precision. In contrast, the word alignments under IBM-4 need not respect the phrase pair inventory and may also cross phrase boundaries. In this way the IBM-4 model is able to achieve better 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 5). In figure 5 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. AER closely follows the Alignment Recall.

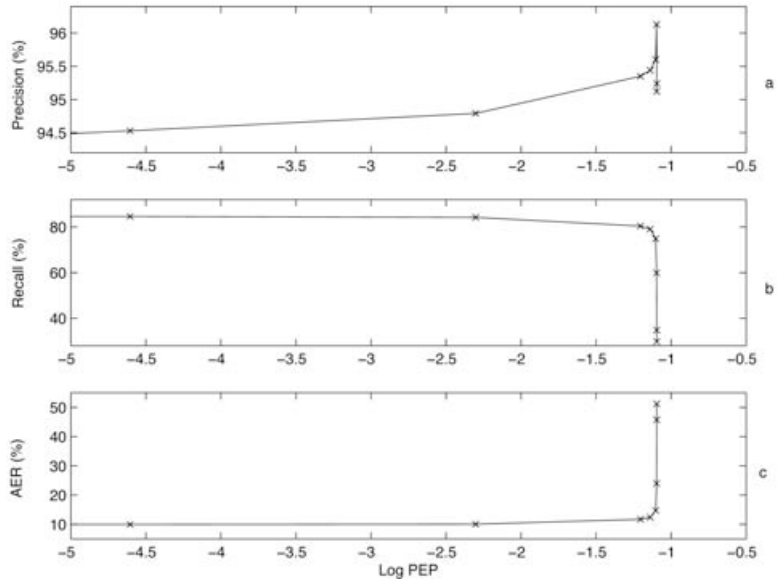

Fig. 5. Alignment performance of TTM as a function of Phrase Exclusion Probability (PEP). For each value of PEP, we measure Alignment Precision (Panel a), Alignment Recall (Panel b), and AER (Panel c). Results are shown on the French-English task. At a log PEP value of −4.6 the AER achieves a minimum value of 9.9%.

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 excluded 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 6, 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. This quantity for each phrase pair is based simply on the relative frequency of its occurrence 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. 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.

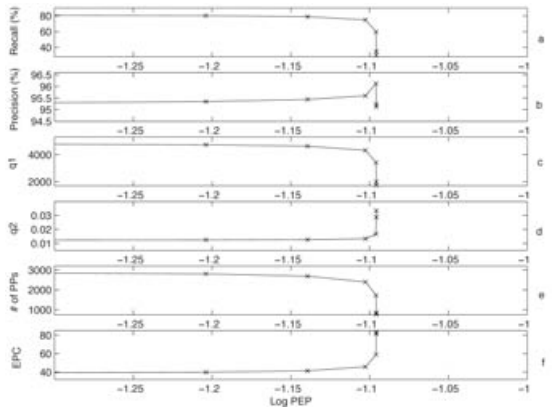

Fig. 6. Variation of alignment precision (Panel b) and recall (Panel a) for values of Phrase Exclusion Probability (PEP) near the critical value. We also plot four additional quantities derived from the MAP alignment. These include the number of wrongly hypothesized links q_1 (Panel c), penalty per incorrectly 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 6e). To analyze this behavior, we write Alignment Precision as

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

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 6, we see that as EPC increases (figure 6f) the absolute number of phrase-pairs in the alignment decreases (figure 6e). The quantity q_2 (figure 6d) 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 6c), suggesting that the relatively few phrase pairs that remain in the alignments are of high quality. However we see that the Alignment Precision (figure 6b) is dominated by q_2 so that performance falls for PEP above the critical value.

6.2.2 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

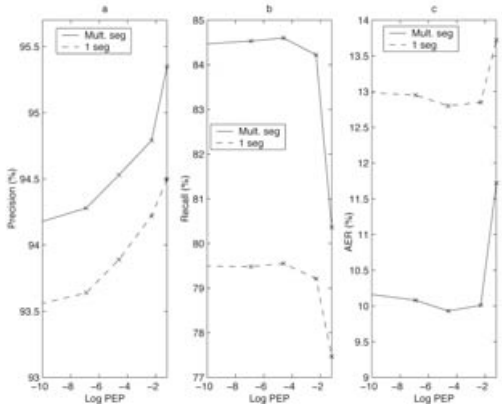

Fig. 7. Effect of multiple phrase segmentations of the source sentence on word alignment quality. MAP word alignments Under the TTM are obtained using the two-step alignment process (Equation 13) that considers only a single phrase segmentation of the source sentence These are compared to MAP word alignments 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.

(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 7 compares the performance of the two approaches 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).

In experiments not reported here, we have observed that excluding the segmentation model has almost no impact on the alignment quality. We can therefore avoid the expensive step which ensures that for a given sentence the probabilities over all segmentations of a fixed length are correctly normalized.

6.2.3 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) (Och 2002), measures the degree of non-monotonicity in the MAP word alignment (equation 12). Suppose any two consecutive phrases in the

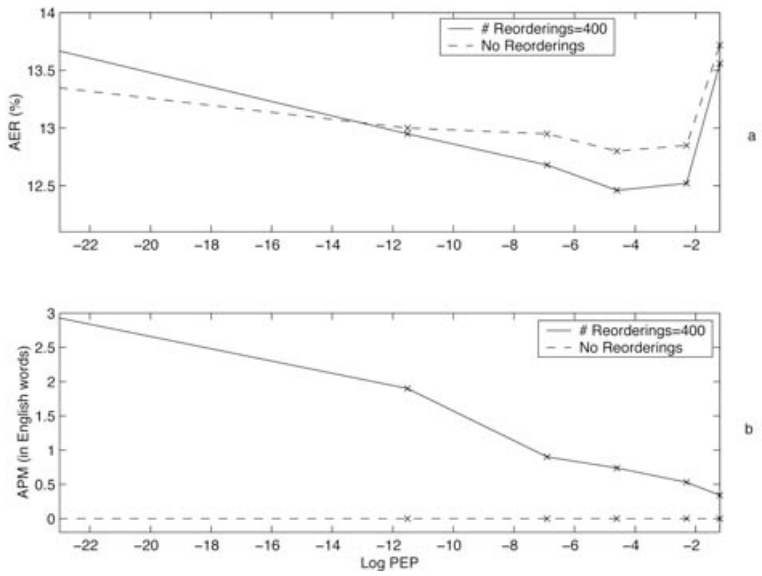

Fig. 8. Effect of reorderings of the source phrase sequence on alignment quality. MAP Word Alignments under the TTM are obtained using a fixed number of reorderings ($N = 400$) of the single phrase segmentation of the source sentence. Performance is compared with MAP word alignments obtained without 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.

reordered source phrase sequence $\hat{u}_{\hat{a}_1}, \ldots, \hat{u}_{\hat{a}_k}$ 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.

In this 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 8). For each value of 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.

6.2.4 Discussion

We have investigated the role of phrase movement in word alignment. We have described the role the PEP plays in the overall model and how smaller values

encourage phrase reordering in MAP alignment. We observe that encouraging phrase movement need not lead to better AER. However allowing some movement does lead to gains relative to the Fixed Phrase Order Model, although these gains are small ($< 0.5\%$ AER). Furthermore, the Average Phrase Movement is less than one word at the best achieved AER. We do not claim that these results hold generally for other language pairs or other translation models, although we note that Chinese-English alignment behavior is similar to what we have reported in French-English alignment. Based on the limited AER gains that come from phrase reordering, we will use only the Fixed Phrase Order Model for translation. These experiments show that this is not an entirely unreasonable choice. Given the small amount of phrase movement observed in the best alignments (less than one word), we might hope to achieve similar performance using Fixed Phrase Order models with longer phrase pairs. This is a constraint which can be varied. The maximum length of the source language phrases is a design parameter in the construction of the phrase pair inventory. It determines the size of the phrase pair inventory, and therefore balances the coverage of the test set against memory usage.

6.3 Translation

We now measure the translation performance of the TTM described in section 5.2. In implementing translation under the TTM we use the same components analyzed in our word alignment experiments (section 6.2). We use the unweighted Source Phrase Segmentation Model (section 4.2). Allowing phrase movement in FSM-based implementations such as this is expensive in memory usage (Kumar and Byrne 2003; Knight and Al-Onaizan 1998). We use the Fixed Phrase Order Model (section 4.3.2). Translation is performed in monotone phrase order, as has been done by others (Zens and Ney 2004).

Unlike word alignment, translation requires a source language model (section 4.1). Here we use a trigram word language model estimated using modified Kneser-Ney smoothing as implemented in the SRILM tookit (Stolcke 2002). 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 (Doddington 2002) (Och 2002), 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 (Papineni *et al.* 2001) 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

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

Table 5. *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*

	French-English		Chinese-English	
Model	BLEUr1n4 (percent)	NISTr1n4	BLEUr4n4 (percent)	NISTr4n4
IBM-4	17.09	5.02	9.67	3.57
TTM	22.29	5.52	22.45	7.73

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

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

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

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

where $\mathscr{V}^n$ 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 (Marcu and Germann 2002) with the French-English and Chinese-English IBM-4 translation models used in creating the phrase-pair inventories. We see in Table 5 that the performance of the TTM compares favorably to that of the ReWrite decoder on both the Chinese-English and French-English tasks.

Our results are consistent with previously published comparisons between phrase-based translations and IBM-4 based translations (Och 2002; Marcu and Wong 2002). We know from our experiments in bitext word alignment that IBM-4 models yield better overall alignments than the TTM, although the TTM achieves better alignment precision and as such is the more conservative of the two. The same holds in translation. The TTM creates translations by assembling phrase pairs extracted from the bitext. IBM-4 based translation has more freedom to generate novel translations but runs a greater risk of producing invalid translations. The

TTM may err on the side of caution, but this appears to be the better strategy given the current quality of the models.

In the TTM translation experiments on the FBIS corpus, the entire translation process ran without pruning on an IBM X335 with dual Xeon 2.4 GHz processors and 4GB of RAM. Processing time is 10.1 second per sentence, with 26.1 words per sentence on average.

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 allows the model the flexibility of deleting phrases in sentence to be translated and it is achieved within the source-channel model 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 the role 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 9 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 slightly with PEP, unlike BLEU.

Since BLEU is influenced by both BPrecision (Equation 20) and Brevity Penalty (equation 21), we plot these components separately in figure 10. We note first that as PEP (α) increases, the translations grow shorter as measured by the Source-to-Target Length Ratio (STLRatio) (figure 10d) 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 10c) is governed by the number of words in the translation hypothesis, and therefore closely tracks the STLRatio. Somewhat surprisingly, BLEU score (figure 10a) closely tracks the Brevity Penalty and does not improve despite gains in BPrecision. Analogous to AER in bitext word alignment, increasing PEP allows the model to produce higher quality translation when BPrecision (figure 10b) 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. WER does not have an explicit length penalty although it does have an implicit length penalty in that the number of deletions increases as the hypothesized translations grow

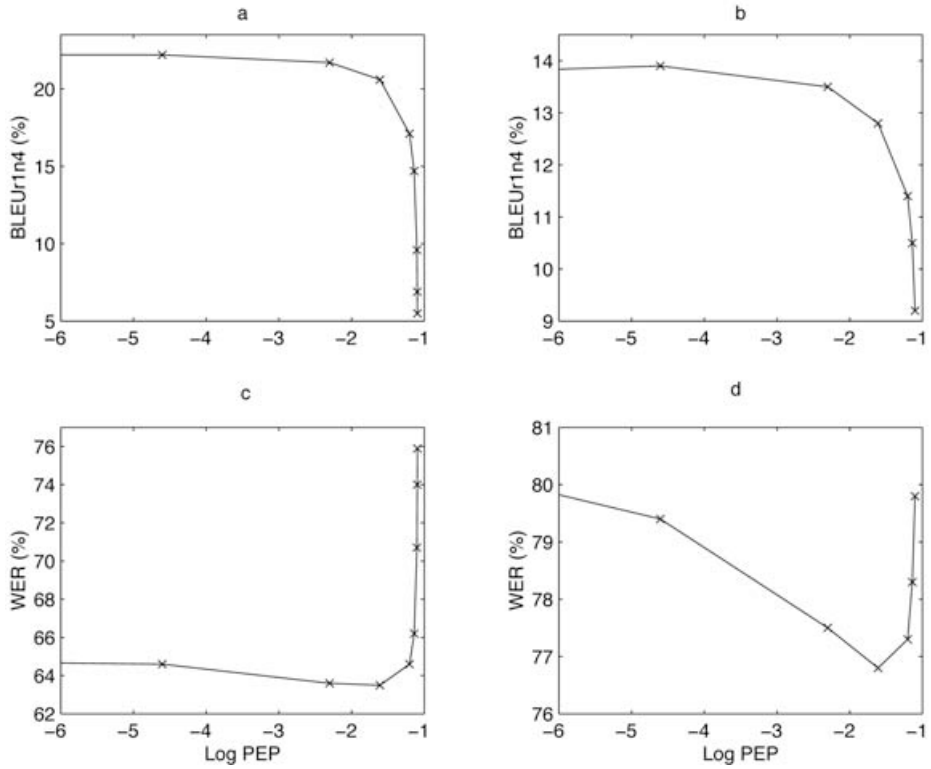

Fig. 9. Translation performance of the TTM as a function of the Phrase Exclusion Probability (PEP) when one reference translation is considered. We measure BLEU (Panel a,b) and WER (Panel c,d) on the French-English and the Chinese-English Tasks.

shorter and deletions are penalized. However, the added cost is linear in the number of deletions, unlike BLEU where it is exponential (equation 21). As PEP is increased we first observe an improvement in WER that corresponds with the increase in the BLEU precision. Beyond a certain value of PEP the translations grow shorter to an extent that WER also degrades.

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 BPrecision. However we note in figure 10b 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 11). In the TTM the dominant mechanism by which shorter translations are produced is simply to delete French phrases (this corresponds to insertions of target phrases in the generative model). As a result English phrases in the translation arise from French phrases which are likely to be separated by phrases that are deleted and not translated. It is unlikely that English phrases generated would follow each other in a fluent translation, i.e. the hypothesis translation contains phrases that are unlikely to be found next to each other in the reference

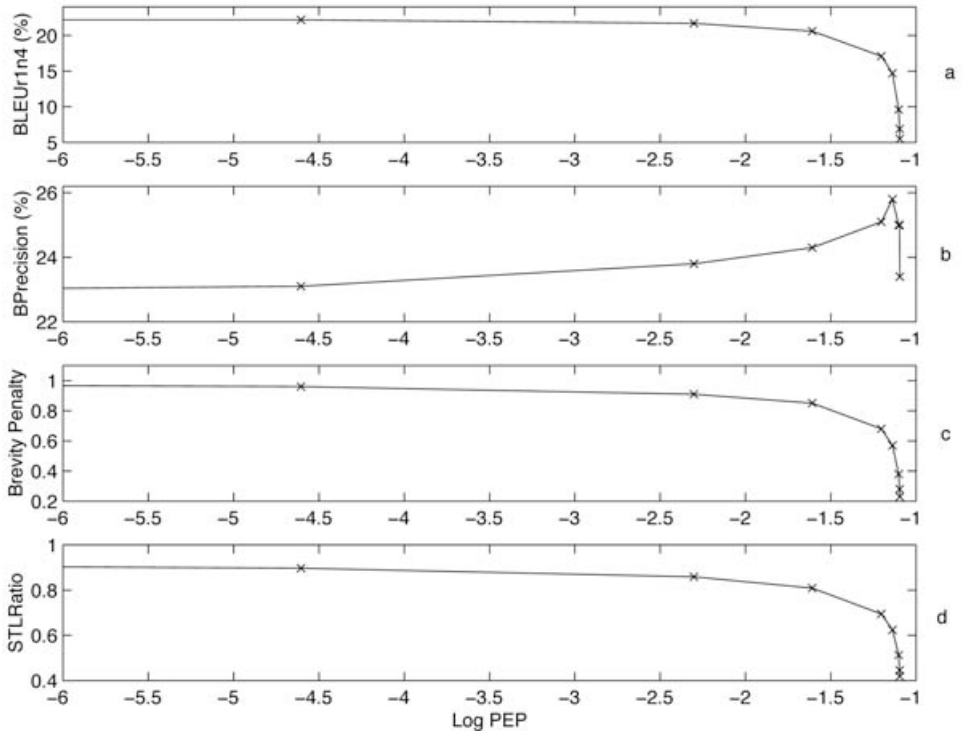

Fig. 10. Translation performance of the TTM as a function of the Phrase Exclusion Probability (PEP) on the French-English task. We measure BLEU (Panel a), BPrecision (Panel b), Brevity Penalty (Panel c), and STLRatio (Panel d) as functions of PEP.

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 11 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.

We now discuss translation performance when multiple reference translations are available (figure 12). The most notable difference in overall score is that BLEUr4n4 peaks at a log PEP of -1.6. This is in contrast to BLEUr1n4 which is largely insensitive to PEP below its critical value. BPrecision behaves nearly identically (except for absolute value) as a function of PEP, but the Brevity Penalty under BLEUr4n4 has a higher critical value.

The explanation is simply that as the number of references grows, it is more likely that a translation will find a close match in length. This weakening of the Brevity Penalty makes it easier to take advantage of the increase in BPrecision that comes with shorter translations. It is interesting to observe this direct relationship between

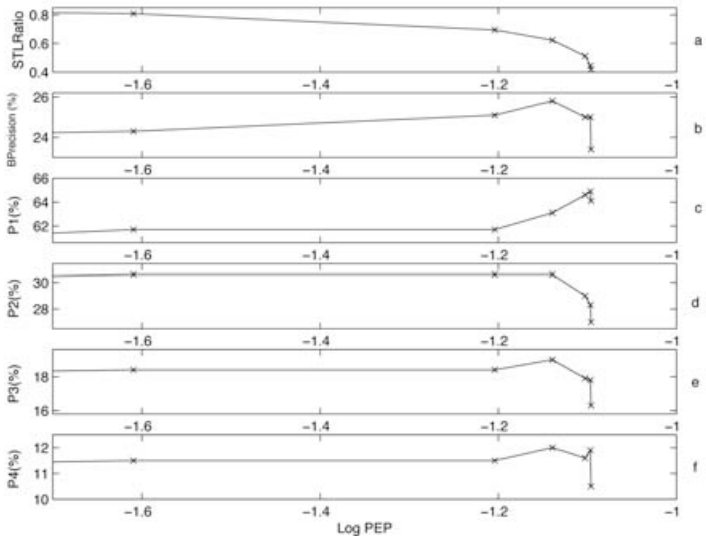

Fig. 11. Analysis of BLEU Precision for values of Phrase Exclusion Probability (PEP) close to its maximum permissible value. We measure the following as functions of PEP : STLRatio (Panel a), BPrecision (Panel b) and each of the n-gram precisions, $n = 1, 2, 3, 4$ (Panels c-f). Results are shown on the French-English task.

TTM component distributions and behavior of the BLEU score. A system tuned under BLEUr4n4 may not be optimized with respect to BLEUr1n4.

6.3.2 Translation lattice quality

The goal of this experiment is to study the usefulness of translation lattices for rescoring purposes. 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 13). The oracle-best BLEU score is obtained in the following way. We obtain the oracle hypothesis for each sentence in the test set by selecting from its N-best list the translation 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. This is only an underbound on the actual oracle BLEU score but it is still useful.

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.

6.3.3 Translation examples

We now present and analyze examples of translations under the TTM. The examples are selected from the NIST 2002 Chinese-English evaluation test set.

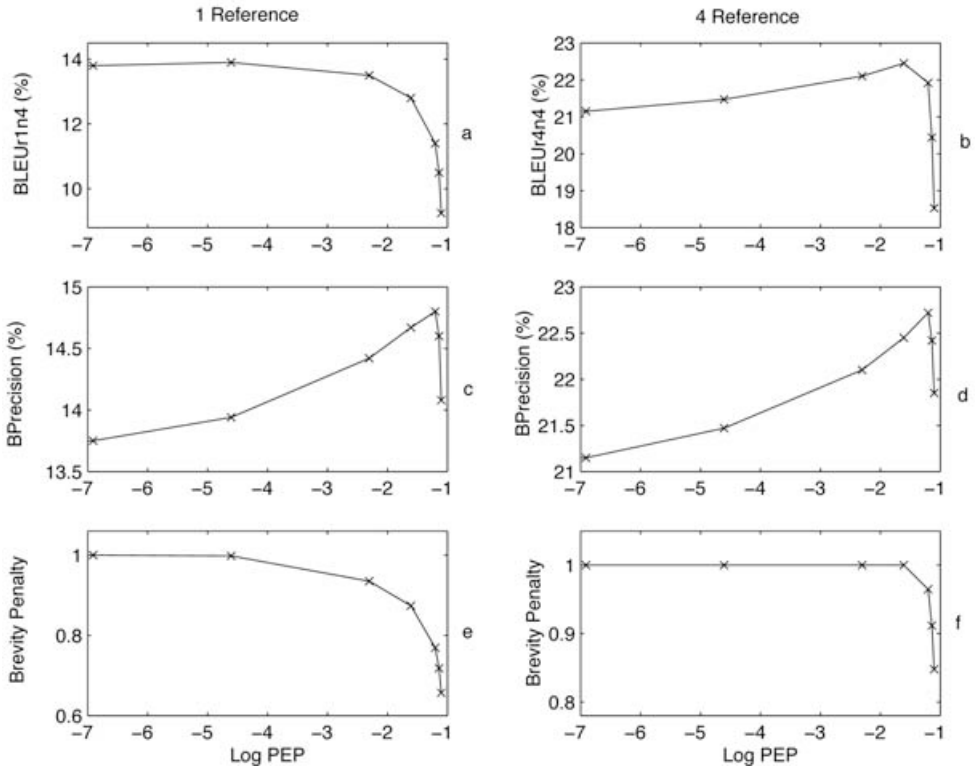

Fig. 12. Translation performance of TTM as a function of the Phrase Exclusion Probability when multiple reference translations are considered for scoring. We obtain BLEU, BPrecision, and Brevity Penalty as functions of PEP in two situations: when 1 reference is considered (Panels a,c,e), and when 4 references are considered (Panels b,d,f).

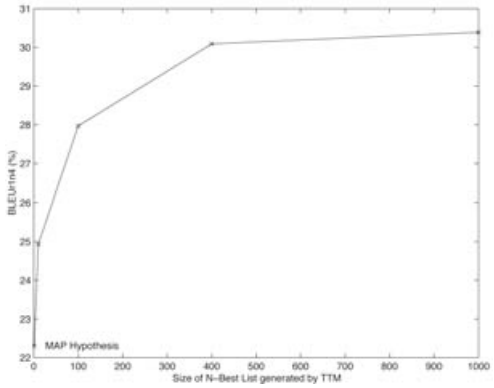

Fig. 13. Variation of oracle-best BLEU scores with the size of the N-best list on the French-English Task. For each N-best list on the test set, the oracle BLEU hypothesis is computed under the sentence-level BLEU metric. The oracle hypotheses are concatenated over the test set, and the test-set BLEU score is measured.

Fig. 14. An example of a good translation under the TTM. The top panel shows the word-segmented Chinese sentence, its English gloss, and target phrase segmentation and phrase transduction under the TTM. The lower panel shows the four reference translations.

We first show an example sentence for which the TTM produces a good translation with a sentence-level BLEU score of 85%. In figure 14 we present the Chinese sentence segmented into words with their English glosses. We also show the Target Phrase Segmentation and Phrase Transduction of this sentence under the TTM. The lower panel of this figure shows the four reference translations for this sentence.

We first note that there is an error in word segmentation of the Chinese sentence: the two word sequence with English gloss "internet" should have been left as a single word. However, as can be seen in Figure 14, this segmentation error does not affect the translation of this word sequence because the two words are contained within the same Chinese phrase. Clearly, word segmentation errors made consistently in training and test data need not be harmful. A more serious error is the translation of the Chinese word (with English gloss "whole country") into the English word *the*. In the phrase-pair inventory, this Chinese word occurred with 152 distinct English phrases. In fact, *the* was one of least likely alternatives as measured by relative frequencies of phrase translations in bitext alignments (section 4.5). It clearly is an error that crept in either due to word alignment errors in the bitext or weaknesses in the phrase-pair extraction heuristics. However, *the* was preferred by the language model.

Interestingly, the next most likely translation produced by the TTM was:

By 2005, the national internet users will reach 200 million .

which is correct, but would not be scored any higher under the BLEU score given these reference translations. This example also shows the need for the model to allow for the insertion of the phrase *the number of* in bitext alignment. This phrase was included by human translators in two of the four reference translations without strict supporting evidence in the Chinese text.

We next present an example sentence (figure 15) on which the TTM produces a poor translation with a sentence-level BLEU score of 0%. Though this translation is

Fig. 15. An example of a poor translation under the TTM. The top panel shows the word-segmented Chinese sentence, its English gloss, and target phrase segmentation and phrase transduction under the TTM. The lower panel shows the four reference translations.

far from perfect and does not match any of the references, the most significant error is the incorrect translation of the last Chinese word in the sentence into the English word *tender*. This error again points to inaccurate translations in the phrase-pair inventory underlying the TTM.

7 Translation performance with large bitext training sets

We report the performance of the Translation Template Model on the Chinese-to-English and Arabic-to-English translation tasks in the NIST 2004 MT evaluation (NIST 2004). We will describe the training and test data, model training procedures, and the experiments performed in the development of our evaluation systems.

7.1 Monolingual texts and bitexts

The goal of the NIST 2004 MT task (NIST 2004) is the translation of news stories, editorials, and speeches from Chinese to English (C-E) and Arabic to English (A-E). The large data track in this task restricts the allowable bitext to that provided by the LDC but places no restrictions on the monolingual English text used by the systems.

Bitexts and source language texts Our translation model training data is derived from various bitext sources. For the Chinese-English task, these sources include the FBIS, Hong Kong News, Xinhua News, Hong Kong Hansards, Translations from the Chinese Treebank, Sinorama Magazine, and the United Nations (LDC2003E14, LDC2003E25, LDC2002E18, LDC2004E09, LDC2003E07, LDC2002E58, and LDC2004E12, respectively). For the Arabic-English task we obtain the bitext

training text from the A-E UN corpus and Arabic news corpora released by the LDC (LDC2004E13, LDC2004E08, LDC2003E05, LDC2003E09, LDC2004E07, and LDC2004E11, respectively).

Our language model training data consists of English text derived from the following English text sources: Xinhua and Agency France Presse (AFP) sections of the English Gigaword corpus (LDC2003T05), the English side of FBIS, the UN and A-E news texts, and the online archives from September 1998 to February 2002 of The People's Daily (PD) (People's Daily 2002).

Test sets For the Chinese-English task, we report performance on the NIST 2001 (LDC2002T01), 2002 (LDC2003T17), 2003, and 2004 evaluation sets (NIST 2004). The test sets consist of 993, 878, 919, and 1788 sentences respectively. For the Arabic-English task, we report performance on the NIST 2002 (LDC2003T18), 2003, and 2004 evaluation sets consisting of 1043, 663, and 1353 sentences respectively. In both tasks, the NIST 2004 is our blind test set while the other corpora form our development sets; the performance reported here on the 2004 tasks are our evaluation results. There are four reference translations for each Chinese (Arabic) sentence in all test sets.

Automatic Text Processing Our Chinese text processing consists of word segmentation (using the LDC word segmenter (LDC 2002)) followed by grouping of numbers. For Arabic our text processing consisted of a modified Buckwalter analysis (LDC2002L49) followed by post processing to separate conjunctions, prepostions and pronouns, and Al-/w- deletion. The English text is processed using a simple tokenizer based on the text processing utility available in the the NIST MT evaluation toolkit (NIST 2004).

Bitext processing Some of these sources in the C-E task (FBIS, Xinhua, HKNews and HKHansards) are provided in document-aligned form by LDC. These collections are refined further by a chunk alignment model (Deng and Byrne 2004) which produces aligned segments within a document pair, possibly allowing for alignment of subsentence segments.

For the other bitext sources (UN, Sinorama and Chinese Treebank), the original document pairs are not available; we therefore use the sentence alignments provided by the LDC. From the LDC sentence alignments, we retain sentence pairs for which: (1) both English and Chinese sentences are shorter than 60 words and (2) the ratio of the number of words in the English sentence to the number of words in the Chinese sentence is less than 6. If a sentence pair violates either condition, it is realigned at the sub-sentence level to obtain shorter chunk-pairs.

In the A-E task, the original bitext from all sources is aligned at the sentence level by LDC. As in C-E task, we retain sentence pairs that satisfy conditions 1 and 2 above. The remaining sentence-pairs are realigned at the sub-sentence level to obtain shorter chunk-pairs.

Statistics computed over the bitext chunk pairs in the C-E and A-E tasks are shown in Table 6.

Table 6. *Chunk pairs extracted from the Chinese-English and Arabic-English large-data track training bitext for the NIST 2004 MT evaluation*

	Chinese-English	Arabic-English
# of chunk-pairs (M)	7.6	5.1
# of words		
English (M)	207.4	132.6
Foreign (M)	175.7	123.0
Vocabulary sizes		
English	169,561	302,282
Foreign	233,183	334,796

7.2 Translation model training

We now describe the procedures involved in training the translation model and the language model.

IBM-4 translation model training and phrase-pair extraction We partition the bitext to form manageable training sets for GIZA++ (Och and Ney 2000) to generate IBM-4 word alignments in each translation direction (IBM4-F, IBM4-E). In the C-E task we create three partitions that contain 53.5M, 95.5M and 95.6M English words, respectively, while in the A-E task, we form 2 partitions with 68.5M and 67.5M English words. Following IBM-4 model training, the IBM-4 word alignments from the training set partitions are merged and phrase-pairs are extracted from the resulting word alignments (using the procedure described in section 3). For reducing storage requirements of the phrase-pair inventory, we extract only those phrase-pairs whose Chinese (or Arabic) side is seen in the test set.

English language models We build language models (LMs) from the English texts using modified Kneser-Ney smoothing as implemented in the SRILM toolkit (Stolcke 2002). For the C-E task, we train a small trigram LM over 20.5M words from PD and Xinhua; and we train large trigram and four-gram LMs over 373.3M words from FBIS, PD, Xinhua, and AFP. For the A-E task, we train a small trigram LM over 266.0M words from Xinhua, AFP, and A-E news, and we train large trigram and four-gram LMs over 428.0M words from Xinhua, AFP, Arabic news, and UN.

7.3 Performance of TTM Evaluation Systems

We here report the performance of the C-E and A-E TTM systems. Translation under the TTM is performed as described in Section 5 except that the translation lattice generated in an initial pass is rescored using various trigram and four-gram language models as described below.

Table 7. *Performance of the Chinese-to-English and Arabic-to-English TTM systems with trigram and four-gram language models. TTM systems are trained on the large-data track training bitexts for the NIST 2004 MT evaluation*

Language Model	BLEUr4n4 (%)						
	Chinese-English				Arabic-English		
	eval01	eval02	eval03	eval04	eval02	eval03	eval04
Small 3g	29.5	27.0	25.7	–	36.7	38.3	–
Large 3g	30.2	28.0	27.2	26.5	38.1	40.1	33.1
Large 4g	31.2	28.8	27.5	27.6	39.4	42.1	36.1

Translation performance is measured using the case insensitive BLEU score on the development sets, and using the case sensitive BLEU score on the NIST 2004 test set. For case-sensitive evaluation, we need to restore case information in our translations; this is done using a capitalizer (built in the JHU Summer Workshop WS'03 (JHU WS 2003)) that uses a trigram language model trained on case-preserved English texts from FBIS, PD, AFP, and Xinhua.

In Table 7 we report the performance of the TTM system under the small trigram LM (Small 3g), large trigram LM (Large 3g), and the large four-gram LM (Large 4g). For performing translation under either of the two trigram language models, we first generate a translation lattice using a pruned version of the language model and then rescore the lattice using the unpruned language model. For performing translation under the four gram language model, we first generate a translate lattice under a pruned version of the large trigram LM (Large 3g), and then rescore this lattice with the four-gram language model. Language model pruning is performed using an entropy criterion (Stolcke 1998) as implemented in the SRILM toolkit (Stolcke 2002).

In the C-E task we observe that the large trigram LM outperforms the smaller trigram LM by about 1.0% absolute in BLEU. The four-gram LM yields a further improvement of about 1.0% BLEU over the large trigram LM. In the A-E task the large trigram LM yields an improvement of about 2.0% BLEU over the small trigram LM. The four-gram LM gives an additional improvement of $1.3 - 3.0\%$ BLEU over the large trigram LM.

7.4 Pruning in translation, translation speed, and memory usage

All steps in the translation process ran on an IBM X335 with dual Xeon 2.4 GHz processors running RedHat Linux 9.1 with Kernel version 2.4.26 where the process size is restricted to be less than 2.7GB. The processing time of the Large 3g system on the NIST 2002 evaluation set is 1.44 minute per sentence, with 26.1 words per sentence on average.

We now describe how pruning can be applied to the component WFSTs of the TTM during translation. We first generate a lattice containing multiple phrase

segmentations of the target language sentence to be translated and obtain the shortest phrase segmentation in this lattice. If this segmentation contains more than 45 phrases, we retain only this segmentation; otherwise, we retain the entire segmentation lattice. To determine whether to allow deletion of target phrases, we compute the average number of phrases per target word from the segmentation lattice. We obtain the number of phrases covering each target language word in the segmentation lattice. We average this quantity over the words in the target sentence to obtain the average number of phrases per word. If either the average number of phrases per word is greater than 2.3 or the number of phrases in the shortest segmentation is greater than 23, we do not allow any deletion of target phrases in translation. Finally we apply pruning when generating a translation lattice under the language model. We prune the lattice so that all paths in the resulting lattice have a total path likelihood that is within 5.0 of the log-likelihood of the best path.

7.5 Summary of evaluation systems

We have described the use of TTM in building Chinese-English and Arabic-English MT systems from large bitexts. The respectable performance of the TTM on the NIST 2004 task shows that our approach is competitive relative to contemporary research MT systems. Our MT system has benefitted considerably from the experiments in sections 6.2 and 6.3. The WFST-TTM architecture supports the generation and rescoring of translation lattices and N-best lists, and we have found this to be valuable in performing rescoring under various language models and decoding criteria.

8 Discussion

The TTM is a source-channel model of the translation process. It defines a joint distribution over the 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 language translations. formulation. In this work we have employed IBM-4 word translation models to generate an initial bitext word alignment and we extracted the phrase-pair inventory from these alignments (Och 2002). The quality of the underlying word alignments and the richness of the phrase-pair inventory play a crucial role in the alignment and translation performance of the TTM (Kumar and Byrne 2004a; Och 2002; Koehn, Och and Marcu 2003). Although we have not discussed these issues, we note that any word alignment or methodology of collecting phrase pairs could be used with the TTM and improving the phrase-pair inventory would undoubtedly improve translation and alignment performance.

We have presented the first use of phrase-based models for bitext word alignment. The technical difficulty that had to be overcome was caused by the inability of the phrase-pair inventory to cover the bitext. We developed source and target phrase deletion models that make aligning arbitrary sentence pairs possible even with an

impoverished phrase-pair inventory. The ability to do this is crucial to implement iterative parameter estimation procedures such as Expectation Maximization (EM) for this model. EM re-estimation of a model requires assigning non-zero probability to the training data. If a finite inventory of phrase-pairs is not rich enough to cover all possible sentence-pairs in a training set, then there will be sentence-pairs with probability zero under the model. The ability to delete phrases within a consistent statistical model was missing from our earlier work (Kumar and Byrne 2003), and addressing that shortcoming was one of the motivations of the TTM.

The deletion of phrases is governed by the Phrase Exclusion Probability (PEP) and this parameter can be tuned in both word alignment and translation. In each case it balances precision against either recall or the Brevity Penalty. Intuitively, one might think that arbitrary high precision is attainable if these other measures are ignored. But in both translation and alignment there are mechanisms that limit precision with resulting practical consequences. In word alignment, even if a user is willing to sacrifice coverage to gain precision, for instance in picking words aligned with high confidence in order to obtain a high quality translation lexicon, arbitrarily high performance cannot be obtained. In translation, choosing only highly likely phrases is not a successful strategy, even if length penalties are ignored.

In the alignment experiments we investigate what gains in AER can be obtained from considering multiple phrase segmentations of the source language sentence. We also study alignment under reordering of the best source phrase segmentation. Both can improve alignments relative to a single source phrase segmentation in monotone phrase order, but allowing multiple source phrase segmentations was far more powerful than the reordering a single segmentation. These results are not conclusive, but for French-English and Chinese-English tasks we studied, multiple phrase segmentations within the TTM is the more valuable of the two approaches. Even though we decided to focus on monotone phrase order models in translation, it is certainly the case that some language pairs do produce long-distance word and phrase movement. From a practical point of view, it is not clear that our model suffers much by ignoring word movement outside phrases or phrase movement itself. Given the current quality of MT, we chose to focus on improving word movement within phrases before addressing what we consider the more challenging problem of moving phrases themselves. Even with our current implementation, Arabic-to-English translation is very near to state-of-the-art for the tasks we report, despite the widely made observation that canonical Arabic word order is Verb-Subject-Object and its translation into English Subject-Verb-Object order requires modeling long distance word and phrase movement (Schafer and Yarowsky 2003).

The TTM formulation does in fact allow phrase movement; the difficulty arises from our insistence on a generative model that can be implemented by WFTs. We have shown previously that how WFSTs can be used for phrase reordering in translation (Kumar and Byrne 2003). However that approach was a direct model of translation, i.e. a direct implementation of $P(e_1^I|f_1^J)$, and as such does not correctly incorporate the source language model. The language model plays an important role in translation, as is especially evident in section 7.3, and one of the advantages

of the generative, source-channel approach is that the language model appears in the model naturally (see equation 1).

There are several potential approaches to model phrase movement based on the present formulation. As mentioned in section 6.2.3, simply increasing the length of phrases can account for a great deal of phrase movement. Improvements in constructing the phrase-pair inventory should certainly address this issue. Another possibility is simply to rescore the lattices and N-Best lists generated by the current implementation under a model that allows long distance word and phrase movement. We have shown that the lattices and N-Best lists are quite rich, and while such a rescoring approach is not optimal, it may be more effective than incorporating a complex phrase movement model in the initial translation pass.

We have demonstrated that the TTM can be successfully used to build Chinese-English and Arabic-English MT systems from large training bitexts. The TTM approach has shown very competitive performance relative to contemporary research MT systems on the NIST 2004 tasks. Although not reported here, Minimum Bayes Risk rescoring under the BLEU criterion can be used to further improve translation performance (Kumar and Byrne 2004b) providing further evidence that N-Best and lattice rescoring can be effective.

The results presented here were obtained using the FSM tools publicly available from AT&T research (Mohri *et al.* 1997). Other than scripts and other programs for building the phrase-pair inventory from aligned bitext, all operations were carried out using standard FSM operations and no special purpose algorithms were employed. All operations ran with command-line tools that can be downloaded.

9 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. The approach requires a careful construction of the underlying random processes in the translation model and care must be taken that they can be realized as WFSTs. Once this is done, 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.

The Translation Template Model is a promising modeling framework for statistical machine translation. The model offers a simple and unified framework for bitext word alignment and translation and this simplicity has allowed us to perform a detailed investigation of the alignment and translation performance of the model.

The model has both strengths and weaknesses for translation and addressing these will form the basis of our future work.

Acknowledgements

This work was supported by an ONR MURI grant N00014-01-1-0685.

We would like to thank F. J. Och of Google, Inc. for providing us the GIZA++ SMT toolkit, the *mkcls* toolkit to train word classes, the Hansards 50K training and test data, and the reference word alignments and *AER* metric software. We thank Woosung Kim and Paola Virga for assistance in building language models for the NIST Chinese-English and Arabic-English tasks, and David Smith for assistance with Arabic morphological analysis and post-processing. We are grateful to AT&T Labs – Research for use of the FSM Toolkit and Andreas Stolcke for use of the SRILM Toolkit.

References

Allauzen, C., Mohri, M. and Roark, B. (2003) Generalized algorithms for constructing statistical language models. *Proceedings 41st Annual Meeting of the Association of Computational Linguistics*, pp. 40–47. Sapporo, Japan.

Bangalore, S. and Riccardi, G. (2001) A finite-state approach to machine translation. *Proceedings 2nd meeting of the North American Chapter of the Association for Computational Linguistics*. Pittsburgh, PA.

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

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

Byrne, W., Khudanpur, S., Kim, W., Kumar, S., Pecina, P., Virga, P., Xu, P. and Yarowsky, D. (2003) The Johns Hopkins University 2003 Chinese-English Machine Translation System. *Proceedings of MT Summit IX*, pp. 447–450. New Orleans, LA.

Deng, Y. and Byrne, W. (2004) Bitext Chunk Alignment for Statistical Machine Translation. *Research Note*, Center for Language and Speech Processing, Johns Hopkins University.

Doddington, G. (2002) Automatic evaluation of machine translation quality using N-gram co-occurrence statistics. *Proceedings of the Conference on Human Language Technology*, pp. 138–145, San Diego, CA.

Germann, U., Jahr, M., Knight, K., Marcu, D. and Yamada, K. (2001) Fast decoding and optimal decoding for machine translation. *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*, pp. 228–235. Toulouse, France.

Canadian Parliament (2003) Canadian Hansards. http://www.parl.gc.ca/.

Knight, K. and Al-Onaizan, Y. (1998) Translation with finite-state devices, *Proceedings of the AMTA Conference*, pp. 421–437. Langhorne, PA.

Koehn, P., Och, F. and Marcu, D. (2003) Statistical phrase-based translation. *Proceedings of the Conference on Human Language Technology*, pp. 127–133. Edmonton, Canada.

Kumar, S. and Byrne, W. (2002) Minimum Bayes-risk alignment of bilingual texts. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 140–147. Philadelphia, PA.

Kumar, S. and Byrne, W. (2003) A weighted finite state transducer implementation of the alignment template model for statistical machine translation. *Proceedings of the Conference on Human Language Technology*, pp. 142–149. Edmonton, Canada.

Kumar, S. and Byrne, W. (2004) A weighted finite state transducer translation template model for statistical machine translation. *Research Note No. 48*, Center for Language and Speech Processing, Johns Hopkins University.

Kumar, S. and Byrne, W. (2004) Minimum Bayes-risk decoding for statistical machine translation *Proceedings of the Conference on Human Language Technology*, pp. 169–176. Boston, MA.

LDC (2002), Chinese Segmenter. http://www.ldc.upenn.edu/Projects/Chinese.

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

Marcu, D. and Wong, W. (2002) A phrase-based, joint probability model for statistical machine translation. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 133–139. Philadelphia, PA.

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

NIST (2004) The NIST Machine Translation Evaluations. http://www.nist.gov/speech/tests/mt/.

Och, F. (2002) Statistical Machine Translation: From Single Word Models to Alignment Templates. *PhD Thesis*, RWTH Aachen, Germany.

Och, F. and Ney, H. (2000) Improved statistical alignment models. *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics* pp. 440–447. Hong Kong, China.

Och, F., Tillmann, C. and Ney, H. (1999) Improved alignment models for statistical machine translation. *Proceedings of the Joint Conference of Empirical Methods in Natural Language Processing and Very Large Corpora*, pp. 20–28. College Park, MD.

Och, F., Ueffing, N. and Ney, H. (2001) An efficient A* search algorithm for statistical machine translation. *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*, pp. 55–62. Toulouse, France.

Papineni, K., Roukos, S., Ward, T. and Zhu, W. (2001) Bleu: a Method for Automatic Evaluation of Machine Translation. *Technical Report RC22176 (W0109-022),* IBM Research Division.

The People's Daily (2002). http://www.english.people.com.cn.

Schafer, C. and Yarowsky, D. (2003) Statistical machine translation using coercive two-level syntactic transduction. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Sapporo, Japan.

Stolcke, A (1998) Entropy-based pruning of backoff language models. *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, pp. 270–274, Lansdowne, VA.

Stolcke, A. (2002) SRILM – An Extensible Language Modeling Toolkit. http://www.speech.sri.com/projects/srilm/. *Proceedings of the International Conference on Spoken Language Processing*, pp. 901–904. Denver, CO.

Tillmann, C. and Ney, H. (2003) Word reordering and a dynamic programming beam search algorithm for statistical machine translation. *Computational Linguistics* **29**(1): 97–133.

Tillmann, C. (2003) A projection extension algorithm for statistical machine translation. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Sapporo, Japan.

Ueffing, N., Och, F. and Ney, H. (2002) Generation of word graphs in statistical machine translation. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 156–163. Philadelphia, PA.

Vogel, S., Ney, H. and Tillmann, C. (1996) HMM based word alignment in statistical translation. *Proceedings of the 16th International Conference on Computational Linguistics* pp. 836–841. Copenhagen, Denmark.

Wang, Y. and Waibel, A. (1997) Decoding algorithm in statistical machine translation. *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pp. 366–372. Madrid, Spain.

JHU (2003), Syntax for Statistical Machine Translation, Final Report, JHU Summer Workshop. http://www.clsp.jhu.edu/ws2003/groups/translate/.

Zens, R. and Ney, H. (2004) Improvements in phrase-based statistical machine translation. *Proceedings of the Conference on Human Language Technology*, pp. 257–264. Boston, MA.

Zhang, Y. Vogel, S. and Waibel A. (2003), Integrated phrase segmentation and alignment model for statistical machine translation. *Proceedings of the Conference on Natural Language Processing and Knowledge Engineering*. Beijing, China.