The CMU Statistical Machine Translation System

Stephan Vogel, Ying Zhang, Fei Huang, Alicia Tribble,
Ashish Venugopal, Bing Zhao, Alex Waibel
Language Technologies Institute, Carnegie Mellon University
vogel+,joy,fhuang,atribble,ashishv,bzhao,ahw@cs.cmu.edu

Abstract

In this paper we describe the components of our statistical machine translation system. This system combines phrase-to-phrase translations extracted from a bilingual corpus using different alignment approaches. Special methods to extract and align named entities are used. We show how a manual lexicon can be incorporated into the statistical system in an optimized way. Experiments on Chinese-to-English and Arabic-to-English translation tasks are presented.

1 Introduction

Statistical machine translation is currently the most promising approach to large vocabulary text translation. In the spirit of the Candide system developed in the early 90s at IBM (Brown et al., 1993), a number of statistical machine translation systems have been presented in the last few years (Wang and Waibel, 1998), (Och and Ney, 2000), (Yamada and Knight, 2000). These systems share the basic underlying principles of applying a translation model to capture the lexical and word reordering relationships between two languages, complemented by a target language model to drive the search process through translation model hypotheses. Their primary differences lie in the structure and source of their translation models. Whereas the original IBM system was based on purely word-based translation models, modern systems try to incorporate more complex structure.

Our system uses phrase-to-phrase translations as the primary building blocks to capture local context information, leading to better lexical choice and more reliable local reordering. The quality of the translations is largely dependent on the quality of phrase-to-phrase translation pairs extracted from bilingual corpora. We have developed and explored several methods to find and generalize bilingual phrase pairs which are described in detail in Section 2. In addition to extracted phrase translations, reliable translation of source language named entities is important as they carry information which often cannot be reconstructed from context. A short overview of our work on named entities is included in Section 3.

Section 4 outlines the architecture of the decoder that combines the translation and language model to generate complete translations, and provides details regarding our decoding procedure.

Finally, in Section 5 we present a series of experiments in Chinese-to-English and Arabic-to-English translation task. We compare the different phrase alignment methods and their combinations, give results for introducing named entities, and explore the effect of the language model and word reordering on translation quality.

2 Phrase Translations

Different methods to find phrase-to-phrase translations from a bilingual corpus have been proposed. Most of them rely on word-to-word alignment. In our system we have experimented with four different approaches to phrase pair extraction, each of which will be described below. We also describe our technique for adding generalization power by allowing for overlapping phrases.

2.1 From Viterbi Path of HMM Word Alignment

A simple approach to extract phrase translations from a bilingual corpus is to harvest the Viterbi path generated by a word alignment model. A number of

probabilistic word alignment models have been proposed (Brown et al., 1993) (Och and Ney, 2000) and shown to be effective for statistical machine translation. We use the HMM-based alignment model introduced in (Vogel et al., 1996) which estimates position alignment probabilities in addition to lexical probabilities. The HMM-based alignment model is based on relative positions: it addresses the likelihood that the word at source position $j+1$ is aligned to target position i' when source position j is aligned to target position i.

The Viterbi path can be used not only to map source words to target words, i.e. building a statistical lexicon, but also to map source phrases to target phrases. For each source phrase ranging from positions j_1 to j_2 the corresponding target phrase is given by $i_{min} = min_j\{i = a(j)\}$ and $i_{max} = max_j\{i = a(j)\}$, where $j = j_1...j_2$. This is a very simple criterion which does not test if the source phrase actually aligns to two or more non-contiguous sequences of words in the target sentence. Due to the potential for alignment errors, such a test would be unreliable. However, by preventing the length of the aligned target phrase from exceeding the length of the source phrase by a given factor, the problem of non-contiguous alignments can be reduced.

2.2 From Bilingual Bracketing

In (Wu, 1997) a word alignment model was proposed which adds additional alignment restrictions over the IBM-style alignment models. The bilingual bracketing builds an hierarchical alignment, which can be viewed as a simple top-down bilingual parse: split source and target segment into two halves $\tilde{f}_l, \tilde{f}_r$ and $\tilde{e}_l, \tilde{e}_r$. Then either align $\tilde{f}_l$ to $\tilde{e}_l$ and $\tilde{f}_r$ to $\tilde{e}_r$, which is called a straight alignment, or align $\tilde{f}_l$ to $\tilde{e}_r$ and $\tilde{f}_r$ to $\tilde{e}_l$, called a reversed alignment. Repeat this for each aligned segment pair down to the word level. At each level the optimization is over the split points and the direction, i.e. straight or reversed alignment. The resulting alignment can be viewed as an alignment of two binary trees, where the sub-trees of the target side can be swapped with respect to the sub-trees of the source side.

Again, this leads to a word alignment between source and target sentence which can be used to extract phrase translation pairs. In this case we extract phrases corresponding to aligned sub-trees in the bilingual bracketing.

Instead of estimating the lexical probabilities for the bilingual bracketing alignment using the Inside-Outside algorithm as in (Wu, 1997), we use the IBM1 alignment model to estimate the lexical probability $p(f|e)$ and calculate a forced alignment using the restrictions of the bilingual bracketing alignment model (Zhao, 2003).

2.3 Robust Alignment Based Phrase Extraction

The third phrase alignment method starts from a high recall sentence level word alignment for generating phrase translation pairs and uses occurrence statistics collected over the entire corpus to achieve higher precision.

We begin by training a high order IBM translation model in both directions, i.e. from source language to target language and vice versa. The resulting alignments are unioned at the sentence level to achieve high recall when evaluated against manually aligned sentences. For a given sentence pair, we consider each possible sequence of source and target words and evaluate them using a series of metrics that estimate the quality of the phrase translation. We consider metrics that measure within sentence consistency (ratio of hypothesized alignment points within this phrase region to the alignment points inconsistent with this region), across sentence consistency (evaluating the number of similar phrases extracted across the corpus), and language pair specific measures to ensure that phrases have appropriate lengths. These metrics are combined using experimentally determined weights and the candidate phrase list is pruned to reduce the computational burden when introduced into the decoding process. This method has been described in detail in (Venugopal, 2003).

2.4 Integrated Segmentation and Phrase Alignment (ISA)

In moving from word level lexicons to phrase based extractions, most techniques rely on an initial word level alignment as the foundation for phrase level extraction. As (Marcu, 2002) argues by example, the word level estimates are liable to provide non-intuitive translation probabilities, and lexical cor-

respondence can in fact be estimated at the phrase level by moving toward joint probability models. Our fourth phrase translation method extends this work by proposing a generative phrase correspondence model that attempts to segment sentences across phrase boundaries.

A bilingual sentence pair (f, e) can be represented by a two-dimensional matrix $R_{m \times n}$, where m in the number of words in f, n in e respectively. The value for cell $[i, j]$ is the point-wise mutual information (MI) between word pairs (f_i, e_j), denoted as $I(f_i, e_j)$. If, for example, the translation for phrase $e_1 e_2$ is f_1, then $I(e_1, f_1)$ and $I(e_2, f_1)$ should be similar. Based on this observation, a phrase pair $(\tilde{f}, \tilde{e})$ should correspond to a contiguous rectangle region in R, where MI values for cells in this region are similar to each other. We use a greedy search algorithm to find all possible phrase pairs for a sentence pair. These phrase pairs represent the segmentations over f and e as well as the alignment between f and e at the phrase level (Zhang, 2003).

2.5 Phrase Translation Probabilities

One general problem with using phrase translations in a statistical machine translation system is that most phrase pairs are seen only a few times, even in very large corpora. This is especially true for longer phrases. As our translation system is based on Bayes' decision rule, we are looking for phrase translation probabilities $p(\tilde{f}|\tilde{e})$, where $\tilde{f}$ denotes the source phrase and $\tilde{e}$ denotes the target phrase. If a phrase $\tilde{f}$ is seen three times in the training corpus, but each time it is aligned to a different translation, then the probabilities of all three phrase pairs is equal, $1/3$ in this example. Therefore, probabilities based on occurrence counts have little discriminative power. Selecting one translation over the others is left to the language model within the decoder.

To get more discriminative probabilities in the phrase translation models we calculate phrase translation probabilities based on a statistical lexicon for the constituent words in the phrase. As the IBM1 alignment model gives the global optimum for the lexical probabilities, it is the natural choice. This leads to the phrase translation probability

$$p(\tilde{f}|\tilde{e}) = \prod_j \sum_i p(f_j|e_i)$$

Table 1: Example of an OP merge.

	Src.		Tgt.
	a b c	#	w x y
	c d e	#	x y z
merge result:	a b c d e	#	w x y z

where the word probabilities $p(f_j|e_i)$ are estimated using the IBM1 word alignment model.

The phrase translations still show some advantage over word-for-word translation due to the summation over all aligned target words. However, if there is no appropriate translation for one of the source words, this will lead to a small word alignment factor making the overall phrase translation probability small. Probabilities are calculated in this fashion for phrases generated by the HMM and bilingual bracketing Viterbi alignments.

2.6 Overlapping Phrases

Each of the phrase alignment methods described so far helps the system generate more fluent translations by essentially memorizing useful examples from the training data. In order to take better advantage of these examples and add some generalization power, we also combine phrase alignments to generate translations for unseen phrases. Specifically, we combine phrase alignments that overlap on both source and target side as described in (Tribble, 2003). The Overlapping Phrases (OP) that result can be used as an additional source of phrase alignments during translation.

In the OP approach, a set of phrase alignments is read in and stored according to its prefixes and suffixes of length 1-4 tokens. For each source-side prefix string s, rules beginning with s are paired with rules ending in the same string. The target sides of the candidate pair are checked for an overlapping substring t, where the length of t must be 1-4 tokens but may differ from the length of s. If substrings s and t are found for a particular phrase pair, then the alignments are merged to form a new, usually longer, phrase alignment. An example merge between two overlapping rules is given in Table 1.

Phrase-level alignment probabilities are assigned to the new rules using to the IBM1 lexical probabilities as described above.

3 Named Entities

Translating named entities (NE), which include named persons, locations and organizations, is both semantically important and technically challenging. NE translation involves both semantic translation and phonetic transliteration, and is made more difficult by the frequent occurrence of OOV words in NEs.

An integrated two-step strategy, Offline and Online NE translation, is proposed and implemented in the current SMT system. Offline NE translation automatically extracts NE translingual equivalence from a parallel corpus, where NEs have been manually or automatically annotated. Starting from a bilingual corpus where NEs are automatically tagged for each language, NE pairs are aligned in order to minimize a multi-feature alignment cost including the transliteration cost, the NE tagging cost, and word-based translation cost. These features are designed to capture the semantic or phonetic similarities between NE pairs as well as NE tagging confidence, and are derived from several information sources using unsupervised and partly supervised methods. A greedy search algorithm is applied to minimize the alignment cost (Huang et al., 2003).

Online NE translation is specially designed for translating NEs which appear in the given test document, but are not covered by the Offline translation. The missing source NEs and target NE translations are "retrieved" cross-lingually from topic-relevant documents (w.r.t. the test document). Relevant documents are retrieved from a monolingual corpus using a 1st-pass translation of the test document as the query. NEs in the retrieved documents are extracted and aligned with source NEs according to their transliteration cost. The NE pairs with minimum transliteration cost are considered as translingual equivalence, and added for the 2nd pass translation. This approach works well for translating foreign person/location names, which is an important issue in word-based translation systems.

4 Decoding

The decoding process works in two stages: First, the word-to-word translations and the phrase-to-phrase translations and, if available, other specific information like NE translation tables are used to generate a translation lattice. Second, a standard n-gram language model is applied to find the best path in this lattice. Both steps will now be described in more detail.

4.1 Building the Translation Lattice

We define a *transducer* as a set of translation pairs generated by the methods described above as well as by alternative knowledge sources such as manual dictionaries. Each translation pair has the form

Label # Source # Target # Probability,

where the label can be used to build hirarchical transducers (Vogel et al., 2000), but in most cases functions just as a name for the transducer. The first step in the decoding process is to build a translation lattice by applying all the transducers, resulting in a lattice over the source words similair to the lattice employed in speech recognition. The transducers are organized as prefix trees over the source side, with translations and translation probabilities attached to the final nodes. This allows for efficient search, as a node in the transducer represents all source phrases consisting of the words along the path to this node and all possible paths to final nodes in the sub-tree under this node.

As we build a translation graph over the source sentence, we construct an initial graph from this sentence, which has nodes $0..J$, where J is the sentence length, and each edge $e_j = (n_j, n_{j+1})$ is labeled with word f_j. An hypothesis $h = (j_1, j_2, \sigma)$ describes a partial translation for the sentence, covering the words between the nodes j_1 and j_2 and matching the path from the inital state σ_0 in the transducer to the state σ. Matching a path through a transducer with part of a sentence can start at each position in the sentence. Therefore, an initial hypothesis $(j, j, \sigma = \sigma_0)$, where σ_0 denotes the root node or initial state of the transducer, is set for each position $j = 0, ..., J-1$.

Expansion of an hypothesis means moving over an edge in the translation lattice and simultaneously over an edge in the transducer tree. If this expansion of hypothesis $h = (j_1, j, \sigma)$ is possible then a new hypothesis $h = (j_1, j+1, \sigma')$ is generated. If the expansion of an hypothesis leads into a final state of the transducer, a new edge is created and in-

serted into the translation lattice for each translation attached to this final state. All relevant information (translation and translation probability) is attached to the new edges.

4.2 Searching for the Best Path

Once the complete translation lattice has been built, a first-best search through this lattice is performed. In addition to the translation probabilities, or rather translation costs, as we use the negative logarithms of the probabilities for numerical stability, the language model costs are added and the path which minimizes the combined cost is returned.

Starting with a special begin-of-sentence hypothesis attached to the first node in the translation lattice, hypotheses are expanded over all outgoing edges from the current node. To allow for local reordering, the search algorithm can be extended by leaving a gap and jumping to a distant node in the translation lattice. This requires that we also keep track of positions already covered in the source sentence. To restrict reordering we use position alignment probabilities; specifically, the jump probabilities as estimated in the HMM alignment.

The decoder allows for recombination of hypotheses in a flexible way. It is important to keep hypotheses apart if the partial translations end in different words, as this will result in different scores from the language model during the next expansion step. In addition, we can distinguish hypotheses if they cover different positions in the source sentence, and also if the length of the translation generated so far is different. The latter comes into effect when a sentence length model is applied at the sentence end.

The search space, especially when allowing for reordering, becomes very large. Pruning is applied to keep decoding times reasonable. Our decoder realizes a standard beam search, where a best hypothesis is stored based on some of the features used for hypothesis recombination, and all hypotheses which are worse by some margin are deleted.

The new hypothesis stores information about the hypothesis which was just expanded and the edge over which it was expanded. This allows us to trace back and reconstruct the translation along the best path.

5 Experiments

5.1 The Corpora

We report a number of experiments carried out on Chinese-to-English and Arabic-to-English translation tasks. As defined for the TIDES machine translation evaluation, the small Chinese-to-English data track allows system training on limited bilingual data but. In addition to a 100K bilingual corpus, a 10k subset of the LDC Chinese-English dictionary can be used. For the large data track, the bilingual corpora consist of the full LDC dictionary with appr. 54,000 Chinese entries, and a number of corpora adding up to about 150 million words. To train the Arabic-to-English system, no small data track in this case, we use the 80 million word UN corpus and the small Ummah corpus. No restrictions apply as to the monolingual English data used for building language models. All the data were made available by LDC.

We tested our system on the 878 test sentences used in the June 2002 TIDES MT evaluation. The Arabic system was tested on the devtest data consisting of 203 sentences. 4 reference translations are available for automatic evaluation of these test sentences.

5.2 Preprocessing

A number of preprocessing steps were performed on these corpora. For the Chinese data these were: 2 byte character to 1 byte character conversion, esp. to convert names written in two-byte encoded Latin characters into their one-byte equivalent; word segmentation using the LDC segmenter with a 43k word list; number and date conversion using a small number of regular expressions.

For Arabic the preprocessing included conversion from UTF-8 encoding into a romanized form and correction of errors in numbers.

5.3 The Evaluation Method

We report results using the NIST mteval metric that was applied in the TIDES evaluation (MTeval, 2002). It compares the system output with several human translations (in our case 4) and uses n-gram matches to calculate a translation quality score. More specifically, information gains for all matching n-grams are calculated and summed up.

This score functions as a weighted precision: how many of the generated n-grams are correct. To balance high precision a length penalty is applied to translations which are too short compared to the reference translations.

For the experiments on overlapping phrases and word reordering we use an additional metric called the Bleu score, described in (Papineni, 2001). While both the Bleu and NIST metrics correlate well with human judgements of translation quality, the Bleu score gives more scoring weight to high order (2, 3, 4) n-grams and as a result is better for highlighting the longer fluent phrases generated by the OP technique.

5.4 The LDC Chinese-English Dictionary

A manually created dictionary can be a valuable addition to the statistical translation system, as the translations are generally very reliable.

To make the dictionary even more useful we augment it with some morphological variations and other additions, on the English side only. Typical augmentation steps include adding plural or past tense forms and prepending nouns with the articles 'a' and 'the'. This technique tends to over-generate and we rely somewhat on the language model to select the reasonable translations. We can also recalculate the probabilities of the entries in the dictionary using a trained statistical lexicon just as we do for phrase translations.

Table 2 shows translation results using the 10K and full 56K LDC dictionaries under several different conditions. A 20 million word language model was used in this experiment.

Table 2: Translation results for the June-2002 test data when using only the LDC dictionary.

	10K	Full
original no-LM	3.79	3.72
original with-LM	5.40	5.52
augmented with-LM	5.78	6.15
probs renorm with-LM	5.91	6.28
probs no-ren with-LM	4.77	6.59

Using no language model results in always picking the first translation alternative. Augmenting the dictionaries provides some useful new translations but they are only selected appropriately when the LM is also added, helping the system discriminate between good and bad augmentations. An improvement of 1.99 and 2.43 in NIST MTeval scores is achieved for 10K and full dictionary when using both.

Adding probabilities allows the translation model to be more discriminative and gives an additional improvement. For the small data track (10K) these probabilities must be normalized since the training data is so small that most word pairs from the dictionary are not seen in the corpus and are assigned a small default probability only. For the large data track (Full) the probabilities from the corpus are reliable enough to be used without renormalization.

5.5 Different Phrase Translation Approaches

In Section 2 we presented four different approaches to phrase alignment: from HMM Viterbi alignment (HMM), from bilingual bracketing Viterbi alignment (BiBr), robust alignment based phrase extraction (RPE), and integrated phrase segmentation and alignment (ISA). Table 3 gives translation results using each of these methods alone and in combination. As the bilingual bracketing has high time complexity this alignment could not be trained on enough data to make a comparison meaningful. All of these translations were done using a 20 million word 3-gram language model and the augmented LDC dictionary with probabilities assigned as described section 5.4

We see that each phrase alignment approach gives different results when used alone, with high scores coming from the ISA phrases. We also observe that the different methods complement each other: ISA generates short, reliable phrases, while the other methods find longer phrases but tend to overgenerate. Combining methods always leads to improvement.

5.6 Overlapping Phrases

In addition to translating with several combinations of phrase alignment models, we applied the Overlapping Phrases approach and translated with these new rules in addition to the original ones. We tested on Chinese-to-English and Arabic-to-English translation. Table 4 gives some experimental results. The baseline used the ISA and HMM phrase transducers.

Table 3: Translation results for the June-2002 test data for the different phrases translation approaches.

	Small	Large
LDC	5.91	6.59
+ HMM	6.70	7.76
+ BiBr	6.50	-
+ RPE	6.52	7.68
+ ISA	6.86	7.88
+ ISA,HMM	6.97	7.97
+ HMM,ISA,RPE	7.00	8.11
All	7.03	-

Table 4: Translation results using overlapping phrases for large Chinese-to-English and Arabic-to-English translation tasks.

	NIST	Bleu
Baseline Chinese	7.97	0.201
With Overlapping Phrases	8.09	0.210
Baseline Arabic	8.59	0.385
With Overlapping Phrases	8.78	0.425

The overall improvement for Arabic-to-English is 2.2% in NIST score and 10.4% in Bleu score. The effect of adding overlapping phrases is to increase the number of long phrases that are correctly translated by the SMT system. As the Bleu score takes longer n-grams more into account, the effect is more visible with this metric.

In the Chinese-to-English translation tasks we see only a smaller effect when adding overlapping phrases. This is in line with our general observation that word order places a more difficult problem in Chinese-to-English translation when compared to Arabic-to-English translation.

5.7 Named Entities

The baseline for the NE experiments was the LDC dictionary (augmented and with probabilities) with the ISA phrase translations. Table 5 shows that the Offline NE approach gave only a small improvement over the baseline system, as many of the NEs were already covered by the LDC dictionary or by the phrase transducer. The Online NE approach gave a nice additional improvement by finding NEs that did not appear in the bilingual training data.

Table 5: Translation results for the June-2002 test data when using named entities.

	Small	Large
Baseline	6.57	7.82
+ Offline NEs	6.61	7.87
+ Online NEs	6.81	7.96

5.8 Effect of the LM

In this experiment we investigated the effect of LM training corpus size. Small data track experiments use the 10K LDC dictionary and all 4 phrase translation methods, for the large data track we used the full LDC dictionary, HMM and ISA phrases. The results are shown in Table 6.

Table 6: Test set perplexity and translation results for LMs of different sizes small and large data tracks.

LM	PP	Small	Large
M001	243.91	6.62	7.46
M010	184.59	6.96	7.88
M020	172.51	7.03	7.97
M050	161.17	7.07	8.05
M160	147.81	7.08	8.15

On the small data track, the translation candidates provided by the translation model are restricted due to the limited bilingual training data. This restricts the effect of the LM and we see little improvement beyond the 20 million word LM. For the large data track, the translation model generates a larger translation lattice with more paths to choose from and the larger LMs give greater improvement.

5.9 Reordering

The results reported so far were all obtained by essentially monotone decoding. Word reordering was restricted to the local reordering captured within phrase translation pairs. In the final experiment we investigated the effect of allowing for additional word reordering during the decoding process. The effect of increasing this window for the Arabic-to-English translation task can be seen in Table 7.

The best improvement was obtained for a reordering window of size 4. Similar to the case of overlapping phrases, the improvement in Bleu score is more pronounced. So far, we have observed a lesser effect for Chinese-to-English translation, only about 5% for the Bleu score.

Table 7: Effect of reordering on translation quality for Arabic-to-English translation task.

	NIST	BLEU
1	8.59	0.385
2	8.87	0.424
3	8.94	0.432
4	9.02	0.441
5	8.99	0.433

6 Summary

In summary, this paper presents the central components of the CMU statistical machine translation system including approaches to extract phrase translation from bilingual corpora, LDC dictionary augmentation, and named entity translation, along with the decoding framework for the translation engine itself. Experimental results demonstrate that each of these components contributes positively towards translation performance. Translation experiments for Chinese-to-English and Arabic-to-English translations tasks on the TIDES June 2002 test data were presented, giving results which are comparable to the best results reported so far on these test sets. Using overlapping phrases and word reordering gave less improvement in the Chinese-to-English translation experiments, compared to Arabic-to-English translation. This indicates that word order poses a more difficult problem when translation from Chinese to English. We plan to study this problem in more detail, especially adding class-based and syntax-based language models to our decoder.

References

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, vol. 19, no. 2, pp. 263–311, 1993.

Fei Huang, Stephan Vogel and A. Waibel. Automatic Extraction of Named Entity Translingual Equivalence Based on Multi-feature Cost Minimization. *Proceedings of the ACL-03, Workshop on Multilingual and Mixed-language Named Entity Recognition*, pp. 9-16, July, 2003. Sapporo, Japan.

Daniel Marcu and William Wong. A Phrase-Based, Joint Probability Model for Statistical Machine Translation. *Proceedings of EMNLP-2002*, Philadelphia, PA, July 6-7, 2002.

NIST MT evaluation kit version 9. Available at: http://www.nist.gov/speech/tests/mt/.

Franz Josef Och and Hermann Ney. Improved Statistical Alignment Models. *Proceedings of ACL-00*, pp. 440-447, Hongkong, China.

Kishore Papineni, Salim Roukos, Todd Ward and Wei-Jing Zhu. Bleu: a Method for Automatic Evaluation of Machine Translation. *Technical Report RC22176 (W0109-022)*, IBM Research Division, T. J. Watson Research Center.

Alicia Tribble, Stephan Vogel, and Alex Waibel. Overlapping Phrase-Level Translation Rules in an SMT Engine. submitted to *Proc. of International Confrerence on Natural Language Processing and Knowledge Engineering (NLP-KE)*, 2003, Beijing, China.

Ashish Venugopal, Stephan Vogel and Alex Waibel. Effective Phrase Translation Extraction from Alignment Models. in *Proc. of 41st Annual Meeting of ACL*, pp. 319-326, Sopporo, Japan, July 2003.

Stephan Vogel, Hermann Ney, and Christoph Tillmann. HMM-based Word Alignment in Statistical Translation. in *COLING '96: The 16th Int. Conf. on Computational Linguistics*, pp. 836–841, Copenhagen, August 1996.

Stephan Vogel and Hermann Ney. Translation with Cascaded Finite State Transducers. *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL 2000)*, pp. 23-30. Hongkong, China, October 2000.

Yeyi Wang and Alex Waibel. Fast Decoding for Statistical Machine Translation. *Proc. ICSLP 98*, Vol. 6, pp. 2775-2778, Sidney, Australia, 1998.

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

Kenji Yamada and Kevin Knight. A Syntax-based Statistical Translation Model. in *Proc. of the 39th Annual Meeting of ACL*, Nancy, France, 2000.

Ying Zhang, Stephan Vogel and Alex Waibel. Integrated Phrase Segmentation and Alignment Model for Statistical Machine Translation. submitted to *Proc. of International Confrerence on Natural Language Processing and Knowledge Engineering (NLP-KE)*, 2003, Beijing, China.

Bing Zhao and Stephan Vogel. Word Alignment Based on Bilingual Bracketing. *HLT-NAACL 2003 Workshop: Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, pp. 15–18, Edmonton, Alberta, Canada, May 2003.