Improved Monolingual Hypothesis Alignment for Machine Translation System Combination

XIAODONG HE
Microsoft Research

MEI YANG[1]
University of Washington

JIANFENG GAO
Microsoft Research

PATRICK NGUYEN
Microsoft Research

ROBERT MOORE
Microsoft Research

This paper presents a new hypothesis alignment method for combining outputs of multiple machine translation (MT) systems. An indirect hidden Markov model (IHMM) is proposed to address the synonym matching and word ordering issues in hypothesis alignment. Unlike traditional HMMs whose parameters are trained via maximum likelihood estimation (MLE), the parameters of the IHMM are estimated indirectly from a variety of sources including word semantic similarity, word surface similarity, and a distance-based distortion penalty. The IHMM-based method significantly outperforms the state-of-the-art TER-based alignment model in our experiments on NIST benchmark datasets. Our combined SMT system using the proposed method achieved the best Chinese-to-English translation result in the constrained training track of the 2008 NIST Open MT Evaluation.

Categories and Subject Descriptors: I.2.7 [**Artificial Intelligence**]: Natural Language Processing – *Machine Translation*

General Terms: Algorithm, Languages, Human Factors, Theory

Additional Key Words and Phrases: statistical machine translation, system combination, word alignment, hidden Markov model

1. INTRODUCTION

System combination has been applied successfully to various machine translation tasks. Recently, confusion-network-based system combination algorithms have been developed to combine outputs of multiple machine translation (MT) systems to form a consensus output (Bangalore, et al. 2001, Matusov et al., 2006, Rosti et al., 2007a, Sim et al., 2007, Rosti et al., 2008). A confusion network comprises a sequence of sets of alternative words, possibly including empty words, with associated scores. The consensus output is then derived by selecting one word from each set of alternatives, to produce the sequence with the best overall score, which could be assigned in various ways such as by voting, by using posterior probability estimates, or by using a combination of these measures and other features.

Constructing a confusion network requires choosing one of the hypotheses as the backbone (also called "skeleton" in the literature), and other hypotheses are aligned to it at the word level. High quality hypothesis alignment is crucial to the performance of the resulting system combination. However, there are two challenging issues that make MT hypothesis alignment difficult. First, different hypotheses may use different synonymous words to express the same meaning, and these synonyms need to be aligned to each other. Second, correct translations may have different word orderings in different hypotheses and these words need to be properly reordered in hypothesis alignment.

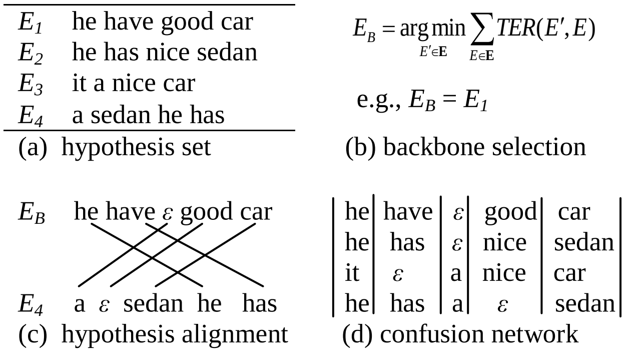
In this paper, we propose an indirect hidden Markov model (IHMM) for MT hypothesis alignment. The HMM provides a way to model both synonym matching and word ordering. Unlike traditional

[1] This research was conducted while the author was visiting Microsoft Research.

HMMs whose parameters are trained via maximum likelihood estimation (MLE), the parameters of the IHMM are estimated *indirectly* from a variety of sources including word semantic similarity, word surface similarity, and a distance-based distortion penalty, without using large amount of monolingual parallel training data. Our combined SMT system using the proposed method gave the best result on the Chinese-to-English test in the constrained training track of the 2008 NIST Open MT Evaluation (MT08).

2. CONFUSION-NETWORK-BASED MT SYSTEM COMBINATION

The current state-of-the-art is confusion-network-based MT system combination as described by Rosti and colleagues (Rosti et al., 2007a, Rosti et al., 2007b, Rosti et al., 2008). As described in (Rosti et al., 2007a), the major steps are illustrated in Figure 1. In Fig. 1 (a), hypotheses from different MT systems are first collected. Then in Fig. 1 (b), one of the hypotheses is selected as the backbone for hypothesis alignment. This can be done by a sentence-level minimum Bayes risk (MBR) method which selects a hypothesis that has the minimum average distance compared to all hypotheses. The backbone determines the word order of the combined output. Then as illustrated in Fig. 1 (c), all other hypotheses are aligned to the backbone. Note that in Fig. 1 (c) the symbol ε denotes an empty word *null*, which is inserted by the alignment normalization algorithm described in section 3.4. Fig. 1 (c) also illustrates the handling of synonym alignment (e.g., aligning “car” to “sedan”), and word re-ordering of the hypothesis. Then in Fig. 1 (d), a confusion network is constructed based on the aligned hypotheses, which consists of a sequence of sets in which each word is aligned to a list of alternative words (including *null*) in the same set. Then, a set of global and local features are used to decode the confusion network.



E_1	he have good car
E_2	he has nice sedan
E_3	it a nice car
E_4	a sedan he has

(a) hypothesis set

$$E_B = \arg\min_{E' \in \mathbf{E}} \sum_{E \in \mathbf{E}} TER(E', E)$$

e.g., $E_B = E_1$

(b) backbone selection

(c) hypothesis alignment

he	have	ε	good	car
he	has	ε	nice	sedan
it	ε	a	nice	car
he	has	a	ε	sedan

(d) confusion network

Figure 1: Confusion-network-based MT system combination.

3. INDIRECT-HMM-BASED HYPOTHESIS ALIGNMENT

In confusion-network-based system combination for SMT, a major difficulty is aligning hypotheses to the backbone (Matusov et al., 2006, Rosti et al., 2007a, Sim et al., 2007). One possible statistical model for word alignment is the HMM, which has been widely used for bilingual word alignment (Vogel et al., 1996, Och and Ney, 2003). In this paper, we propose an indirect-HMM method for monolingual hypothesis alignment.

3.1 IHMM for hypothesis alignment

Let $e_1^I = (e_1, ..., e_I)$ denote the backbone, $e_1'^J = (e_1', ..., e_J')$ a hypothesis to be aligned to e_1^I, and $a_1^J = (a_1, ..., a_J)$ the alignment that specifies the position of the backbone word aligned to each hypothesis word. We treat each word in the backbone as an HMM state and the words in the hypothesis as the observation sequence. We use a first-order HMM, assuming that the emission probability $p(e_j' | e_{a_j})$ depends only on the backbone word, and the transition probability $p(a_j | a_{j-1}, I)$ depends only on the position of the last state and the length of the backbone. Treating the alignment as hidden variable, the conditional probability that the hypothesis is generated by the backbone is given by

$$p(e_1'^J \mid e_1^I) = \sum_{a_1^J} \prod_{j=1}^{J} \left[p(a_j \mid a_{j-1}, I) p(e_j' \mid e_{a_j}) \right] \quad (1)$$

As in HMM-based bilingual word alignment (Och and Ney, 2003), we also associate a *null* with each backbone word to allow generating hypothesis words that do not align to any backbone word.
In HMM-based hypothesis alignment, emission probabilities model the similarity between a backbone word and a hypothesis word, and will be referred to as the similarity model. The transition probabilities model word reordering, and will be called the distortion model.

3.2 Estimation of the similarity model

The similarity model, which specifies the emission probabilities of the HMM, models the similarity between a backbone word and a hypothesis word. Since both words are in the same language, the similarity model can be derived based on both semantic similarity and surface similarity, and the overall similarity model is a linear interpolation of the two:

$$p(e_j' \mid e_i) = \alpha \cdot p_{sem}(e_j' \mid e_i) + (1-\alpha) \cdot p_{sur}(e_j' \mid e_i) \quad (2)$$

where $p_{sem}(e_j' \mid e_i)$ and $p_{sur}(e_j' \mid e_i)$ reflect the semantic and surface similarity between e_j' and e_i, respectively, and a is the interpolation factor.

Since the semantic similarity between two target words is source-dependent, the semantic similarity model is derived by using the source word sequence as a hidden layer:

$$\begin{aligned} & p_{sem}(e_j' \mid e_i) \\ & = \sum_{k=0}^{K} p(f_k \mid e_i) p(e_j' \mid f_k, e_i) \\ & \approx \sum_{k=0}^{K} p(f_k \mid e_i) p(e_j' \mid f_k) \end{aligned} \quad (3)$$

where $f_1^K = (f_1, \ldots, f_K)$ is the source sentence. Moreover, in order to handle the case that two target words are synonyms but neither of them has counter-part in the source sentence, a *null* is introduced on the source side, which is represented by f_0. The last step in (3) assumes that first e_i generates all source words including *null*. Then e_j' is generated by all source words including *null*.

In the common SMT scenario where a large amount of bilingual parallel data is available, we can estimate the translation probabilities from a source word to a target word and vice versa via conventional bilingual word alignment. Then both $p(f_k \mid e_i)$ and $p(e_j' \mid f_k)$ in (3) can be derived:

$$p(e_j' \mid f_k) = p_{s2t}(e_j' \mid f_k)$$

where $p_{s2t}(e_j' \mid f_k)$ is the translation model from the source-to-target word alignment model, and $p(f_k \mid e_i)$, which need to enforce the sum-to-1 constraint over all words in the source sentence, takes the following form,

$$p(f_k \mid e_i) = \frac{p_{t2s}(f_k \mid e_i)}{\sum_{k=0}^{K} p_{t2s}(f_k \mid e_i)}$$

where $p_{t2s}(f_k \mid e_i)$ is the translation model from the target-to-source word alignment model. In our method, $p_{t2s}(null \mid e_i)$ for all target words is simply a constant p_{null}, whose value is optimized on held-out data[2].

The surface similarity model can be estimated in several ways. A very simple model could be based on exact match: the surface similarity model, $p_{sur}(e'_j \mid e_i)$, would take the value 1.0 if $e' = e$, and 0 otherwise[3]. However, a smoothed surface similarity model is used in our method. If the target language uses alphabetic orthography, as English does, we treat words as letter sequences and the similarity measure can be the length of the longest matched prefix (LMP) or the length of the longest common subsequence (LCS) between them. Then, this raw similarity measure is transformed to a surface similarity score between 0 and 1 through an exponential mapping,

$$p_{sur}(e'_j \mid e_i) = \exp\left\{\rho \cdot \left[s(e'_j, e_i) - 1\right]\right\} \qquad (4)$$

where $s(e'_j, e_i)$ is computed as

$$s(e'_j, e_i) = \frac{M(e'_j, e_i)}{\max(|e'_j|, |e_i|)}$$

and $M(e'_j, e_i)$ is the raw similarity measure of e_j' e_i, which is the length of the LMP or LCS of e_j' and e_i. and ρ is a smoothing factor that characterizes the mapping, Thus as ρ approaches infinity, $p_{sur}(e'_j \mid e_i)$ backs off to the exact match model. We found the smoothed similarity model of (4) yields slightly better results than the exact match model. Both LMP- and LCS- based methods achieve similar performance but the computation of LMP is faster. Therefore, we only report results of the LMP-based smoothed similarity model.

3.3 Estimation of the distortion model

The distortion model, which specifies the transition probabilities of the HMM, models the first-order dependencies of word ordering. In bilingual HMM-based word alignment, it is commonly assumed that transition probabilities $p(a_j = i \mid a_{j-1} = i', I)$ depend only on the jump distance (i - i') (Vogel et al., 1996):

$$p(i \mid i', I) = \frac{c(i - i')}{\sum_{l=1}^{I} c(l - i')} \qquad (5)$$

As suggested by Liang et al. (2006), we can group the distortion parameters $\{c(d)\}$, $d = i - i'$, into a few buckets. In our implementation, 11 buckets are used for $c(\leq -4)$, $c(-3)$, ... $c(0)$, ..., $c(5)$, $c(\geq 6)$. The probability mass for transitions with jump distance larger than 6 and less than -4 is uniformly divided. By doing this, only a handful of $c(d)$ parameters need to be estimated. Although it is possible to estimate them using the

[2] The other direction, $p_{s2t}(e'_i \mid null)$, is available from the source-to-target translation model.

[3] Usually a small back-off value is assigned instead of 0.

EM algorithm on a small development set, we found that a particularly simple model, described below, works surprisingly well in our experiments.

Since both the backbone and the hypothesis are in the same language, It seems intuitive that the distortion model should favor monotonic alignment and only allow non-monotonic alignment with a certain penalty. This leads us to use a distortion model of the following form, where *K* is a tuning factor optimized on held-out data.

$$c\left(d\right)=\left(1+\left|d-1\right|\right)^{-K},\ d=-4,\ \ldots,\ 6 \qquad (6)$$

As shown in Fig. 2, the value of distortion score peaks at *d*=1, i.e., the monotonic alignment, and decays for non-monotonic alignments depending on how far it diverges from the monotonic alignment.

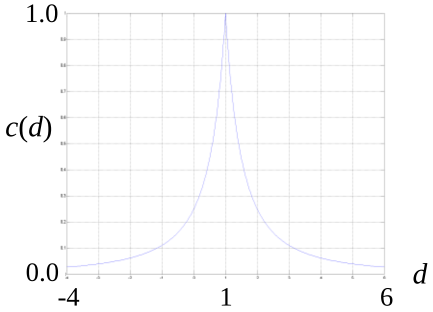


Figure 2, the distance-based distortion parameters computed according to (6), where *K*=2.

Following Och and Ney (2003), we use a fixed value p_0 for the probability of jumping to a *null* state, which can be optimized on held-out data, and the overall distortion model becomes

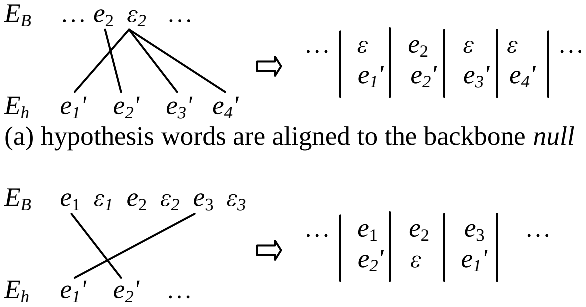
$$\tilde{p}(i\,|\,i',I)=\begin{cases} p_0 & \text{if } i\ =\ \textit{null} \text{ state} \\ (1-p_0)\cdot p(i\,|\,i',I) & \text{otherwise} \end{cases}$$

3.4 Alignment normalization

Given an HMM, the Viterbi alignment algorithm can be applied to find the best alignment between the backbone and the hypothesis,

$$\hat{a}_1^J=\arg\max_{a_1^J}\prod_{j=1}^{J}\left[p(a_j\,|\,a_{j-1},I)\,p(e'_j\,|\,e_{a_j})\right] \qquad (7)$$

However, the alignment produced by the algorithm cannot be used directly to build a confusion network. There are two reasons for this. First, the alignment produced may contain 1-N mappings between the backbone and the hypothesis whereas 1-1 mappings are required in order to build a confusion network. Second, if hypothesis words are aligned to a *null* in the backbone or vice versa, we need to insert actual *null*s into the right places in the hypothesis and the backbone, respectively. Therefore, we need to normalize the alignment produced by Viterbi search.


E_B … e_2 ε_2 …
E_h e_1' e_2' e_3' e_4'
… | ε e_1' | e_2 e_2' | ε e_3' | ε e_4' | …
E_B e_1 ε_1 e_2 ε_2 e_3 ε_3
E_h e_1' e_2' …
… | e_1 e_2' | e_2 ε | e_3 e_1' | …

(a) hypothesis words are aligned to the backbone *null*

(b) a backbone word is aligned to no hypothesis word

Figure 3: illustration of alignment normalization

First, whenever more than one hypothesis words are aligned to one backbone word, we keep the link which gives the highest occupation probability computed via the forward-backward algorithm. The other hypothesis words originally aligned to the backbone word will be aligned to the *null* associated with that backbone word.

Second, for the hypothesis words that are aligned to a particular *null* on the backbone side, a set of *null*s are inserted around that backbone word associated with the *null* such that no links cross each other. As illustrated in Fig. 3 (a), if a hypothesis word e_2' is aligned to the backbone word e_2, a *null* is inserted in front of the backbone word e_2 linked to the hypothesis word e_1' that comes before e_2'. *Null*s are also inserted for other hypothesis words such as e_3' and e_4' after the backbone word e_2. If there is no hypothesis word aligned to that backbone word, all *null*s are inserted after that backbone word .[4]

For a backbone word that is aligned to no hypothesis word, a null is inserted on the hypothesis side, right after the hypothesis word which is aligned to the immediately preceding backbone word. An example is shown in Fig. 3 (b).

3.5 Construction of the confusion network

After all hypotheses being aligned to the backbone, a confusion network can be constructed. This is done in a left-to-right fashion. First, all deletions in the hypotheses are replaced with the empty word *null* such that all deletions are converted to substitutions, i.e., a real word in the backbone substituted by a *null* word in the hypothesis. Then a position point is assigned to each hypothesis. For each hypothesis, the position that the point points to is called the "current" position and word at the current position is called the "current" word. Each of these points is initially assigned to the beginning of the associated hypothesis.

Start from that, we check all the "current" words. If there are insertions, a new column is formed consisting of either a null word for a hypothesis that doesn't have insertion at the "current" position, or the "current" word for a hypothesis that does have insertion. Then for hypotheses that have the insertion, the position points will move forward one step; if there are no insertions, a new column is formed consisting of all the "current" words from all hypotheses and all position points move forward one step. This process goes on until all position points reach the end of the corresponding hypotheses and at the end the confusion network is constructed.

[4] This only happens if no hypothesis word is aligned to a backbone word but some hypothesis words are aligned to the *null* associated with that backbone word.

4. RELATED WORK

The two main hypothesis alignment methods for system combination in the previous literature are GIZA++ and TER-based methods. Matusov et al. (2006) proposed using GIZA++ to align words between different MT hypotheses, where all hypotheses of the test corpus are collected to create hypothesis pairs for GIZA++ training. This approach uses the conventional HMM model bootstrapped from IBM Model-1 as implemented in GIZA++, and heuristically combines results from aligning in both directions. System combination based on this approach gives an improvement over the best single system. However, the number of hypothesis pairs for training is limited by the size of the test corpus. Also, MT hypotheses from the same source sentence are correlated with each other and these hypothesis pairs are not i.i.d. data samples. Therefore, GIZA++ training on such a data set may be unreliable.

Bangalore et al. (2001) used a multiple string-matching algorithm based on Levenshtein edit distance, and later Sim et al. (2007) and Rosti et al. (2007) extended it to a TER-based method for hypothesis alignment. TER (Snover et al., 2006) measures the minimum number of edits, including substitution, insertion, deletion, and shift of blocks of words, that are needed to modify a hypothesis so that it exactly matches the other hypothesis. The best alignment is the one that gives the minimum number of translation edits. TER-based confusion network construction and system combination has demonstrated superior performance on various large-scale MT tasks (Rosti. et al, 2007). However, when searching for the optimal alignment, the TER-based method uses a strict surface hard match for counting edits. Therefore, it is not able to handle synonym matching well. Moreover, although TER-based alignment allows phrase shifts to accommodate the non-monotonic word ordering, all non-monotonic shifts are penalized equally no matter how short or how long the move is, and this penalty is set to be the same as that for substitution, deletion, and insertion edits. Therefore, its modeling of non-monotonic word ordering is very coarse-grained.

Matusov et al. (2006) proposed using GIZA++-based method for hypothesis alignment. The surface similarity information is used to initialize the GIZA++ models and then the models are trained on hypothesis-pairs. However, it is not clear that how much of that information is still kept in the models after following unsupervised maximum likelihood training. In contrast to the GIZA++-based method, our IHMM-based method has a similarity model estimated using bilingual word alignment HMMs that are trained on a large amount of bi-text data. Moreover, the surface similarity information is explicitly incorporated in our model. On the other hand, the TER-based alignment model is similar to a coarse-grained, non-normalized version of our IHMM, in which the similarity model assigns no penalty to an exact surface match and a fixed penalty to all substitutions, insertions, and deletions, and the distortion model simply assigns no penalty to a monotonic jump, and a fixed penalty to all other jumps, equal to the non-exact-match penalty in the similarity model.

There have been other hypothesis alignment methods. Karakos, et al. (2008) proposed using ITGs to find the optimal edit sequence under the restriction that block moves must be properly nested, so as to align hypothesis to the backbone. Rosti et al. (2008) proposed an incremental TER alignment method which allows using a confusion network as the alignment reference. In (Jayaraman and Lavie, 2005), a heuristic-based matching algorithm was proposed.

5. EXPERIMENTAL RESULTS

In this section, we evaluate our IHMM-based hypothesis alignment method on the Chinese-to-English (C2E) test in the constrained training track of the 2008 NIST Open MT Evaluation (NIST, 2008). We compare to the TER-based method used by Rosti et al. (2007). In the following experiments, the NIST BLEU score is used as the evaluation metric (Papineni et al., 2002), which is reported as a percentage in the following sections.

5.1 Implementation details

In our implementation, the backbone is selected with MBR. Only the top hypothesis from each single system is considered as a backbone. A uniform posteriori probability is assigned to all hypotheses. TER is used as loss function in the MBR computation.

Similar to (Rosti et al., 2007), each word in the confusion network is associated with a word posterior probability. Given a system S, each of its hypotheses is assigned with a rank-based score of $1/(1+r)^{\eta}$, where r is the rank of the hypothesis, and η is a rank smoothing parameter. The system specific rank-based score of a word w for a given system S is the sum of all the rank-based scores of the hypotheses in system S that contain the word w at the given position (after hypothesis alignment). This score is then normalized by the sum of the scores of all the alternative words at the same position and from the same system S to generate the system specific word posterior. Then, the total word posterior of w over all systems is a sum of these system specific posteriors weighted by system weights.

Beside the word posteriors, we use language model scores and a word count as features for confusion network decoding.

Therefore, for an M-way system combination that uses N LMs, a total of M+N+1 decoding parameters, including M-1 system weights, one rank smoothing factor, N language model weights, and one weight for the word count feature, are optimized using Powell's method (Brent, 1973) to maximize BLEU score on a development set[5] .

Two language models are used in our experiments. One is a trigram model estimated from the English side of the parallel training data, and the other is a 5-gram model trained on the English GigaWord corpus from LDC using the MSRLM toolkit (Nguyen et al, 2007).

The bilingual translation models used to compute the semantic similarity are from the word-dependent HMMs proposed by He (2007), which are trained on two million parallel sentence-pairs selected from the training corpus allowed by the constrained training condition of MT08.

In order to reduce the fluctuation of BLEU scores caused by the inconsistent translation output length, an unsupervised length adaptation method has been devised. We compute an expected length ratio between the MT output and the source sentences on the development set after maximum- BLEU training. Then during test, we adapt the length of the translation output by adjusting the weight of the word count feature such that the expected output/source length ratio is met. In our experiments, we apply length adaptation to the system combination output at the level of the whole test corpus.

5.2 Development and test data

The development (dev) set used for system combination parameter training contains 1002 sentences sampled from the previous NIST MT Chinese-to-English test sets: 35% from MT04, 55% from MT05, and 10% from MT06-newswire. The test set is the MT08 Chinese-to-English "current" test set, which includes 1357 sentences from both newswire and web-data genres. Both dev and test sets have four references per sentence.

As inputs to the system combination, 10-best hypotheses for each source sentence in the dev and test sets are collected from each of the eight single systems. All outputs on the MT08 test set were true-cased before scoring using a log-linear conditional Markov model proposed by Toutanova et al. (2008). However, to save computation effort, the results on the dev set are reported in case insensitive BLEU (ciBLEU) score instead.

5.3 Description of Individual Systems for System Combination

There are eight individual systems incorporated in the system combination framework. They are named from Sys-1 to Sys-8, respectively. All systems were trained within the confines of the constrained training condition of NIST MT08 evaluation. In the following sub-sections, we give a brief description of each system.

5.3.1 Tree-to-String system

Sys-1 uses a syntax-based decoder (Menezes and Quirk, 2007), informed by a source language dependency parse (Chinese). The Chinese text is segmented using a Semi-CRF Chinese word breaker trained on the Penn Chinese Treebank (Andrew, 2006), then POS-tagged using a feature rich Maximum

[5] The parameters of IHMM are not tuned by maximum-BLEU training.

Entropy Markov Model, and parsed using a dependency parser trained on the Chinese Treebank (Corston-Oliver et al., 2006). The English side is segmented to match the internal tokenization of the reference BLEU script. Sentences are word aligned using an HMM with word-based distortion (He, 2007), and the alignments are combined using the grow-diag-final method. Treelets, templates, and order model training instances are extracted from this aligned set; treelets are annotated with relative frequency probabilities and lexical weighting scores.

The decoder uses three language models: a small trigram model built on the target side of the training data, a medium sized LM built on only the Xinhua portion of the English Gigaword corpus, and a large LM built on the whole English Gigaword corpus using a scalable LM toolkit (Nguyen et al., 2007). It also has treelet count, word count, order model logprob, and template logprob features. At decoding time, the 32-best parses for each sentence are packed into a forest; packed forest transduction is used to find the best translation.

5.3.2 Phrase based system

Sys-2 is a single-pass phrase-based system. The decoder uses a beam search to produce translation candidates left-to-right, incorporating future distortion penalty estimation and early pruning to limit the search (Moore and Quirk, 2007). The data is segmented and aligned in the same manner as above. Phrases are extracted and provided with conditional model probabilities of source given target and target given source (estimated with relative frequency), as well as lexical weights in both directions. In addition, word count, phrase count, and a simple distortion penalty are included as features.

5.3.3 Syntactic source reordering system

Sys-3 is essentially the same as Sys-2 except that we apply a syntactic reordering system as a preprocessor to reorder Chinese sentences in training and test data in such a way that the reordered Chinese sentences are much closer to English in terms of word order. For a Chinese sentence, we first parse it using the Stanford Chinese Syntactic Parser (Levy and Manning, 2003), and then reorder it by applying a set of reordering rules, proposed by Wang et al. (2007), to the parse tree of the sentence.

5.3.4 Syntax-based pre-ordering system

Sys-4 is a syntax-based pre-ordering based MT system using a syntax-based pre-ordering model as described in (Li et. al., 2007). Given a source sentence and its parse tree, the method generates, by tree operations, an n-best list of reordered inputs, which are then fed to a standard phrase-based decoder to produce the optimal translation. In implementation, the Stanford parser (Levy and Manning, 2003) is used to parse the input Chinese sentences.

In the system, GIZA++ is used for word alignment and a modified version of MSRSeg tool (Gao et al., 2005) is used to perform Chinese segmentation. Moreover, we recognize certain named entities such as number, data, time, person / location names. For those named entity, translations are generated by rules or lexicon look-up. These translations serve as part of the hypotheses of the translation of the entire sentence. The decoder is a lexicalized maxent-based decoder. Note that non-monotonic translation is used here since the distance-based model is needed for local reordering. A 5-gram language model is used, which is trained on the Xinhua part of English Gigaword version 3. In order to obtain the translation table, GIZA++ is run over the training data in both translation directions, and the two alignment matrices are integrated by the grow-diag-final method into one matrix, from which phrase translation probabilities and lexical weights of both directions are obtained. Regarding to the distortion limit, our experiments show that the optimal distortion limit is 4, which was therefore selected for all our later experiments.

5.3.5 Hierarchical phrase-based system

Sys-5 is a hierarchical phrase-based system as described by Chiang (2007). It uses a statistical phrase-based translation model that uses hierarchical phrases. The model is a synchronous context-free grammar and it is learned from parallel data without any syntactic information.

In this system, the same word segmentation and word alignment process as described in section 5.3.4 were adopted, as well as the language models and the handling of named entities.

5.3.6 Lexicalized re-ordering system

Sys-6 uses a lexicalized re-ordering model similar to the one described by Xiong et al. (2006). It uses a maximum entropy model to predicate reordering of neighbor blocks (phrase pairs). The same word segmentation, word alignment, language model and the handling of named entities were adopted as described in section 5.3.4.

5.3.7 Two-pass phrase-based system

Sys-7 is a two-pass phrase-based system with adapted LM proposed by Foster and Kuhn (2007). This system uses a standard two-pass phrase-based approach. Major features in the first-pass log-linear model include phrase tables derived from symmetrized IBM2 and HMM word alignments, a static 5-gram LM trained on the Giga-word corpus using the SRILM toolkit, and an adapted 5-gram LM derived from the parallel corpus using the technique of Foster and Kuhn (2007). Other features are word count and phrase-displacement distortion. Decoding uses the cube-pruning algorithm of Huang and Chiang (2007), and parameter tuning is performed using Och's max-BLEU algorithm with a closest-match brevity penalty. The rescoring pass uses 5000-best lists, with additional features including various HMM- and IBM- model probabilities; word, phrase, and length posterior probabilities; Google ngrams; reversed and cache LMs; and quote and parenthesis mismatch indicators.

5.3.8 Hierarchical system

Sys-8 is a hierarchical phrase-based system that uses a 4-gram language model in the first pass to generate n-best lists, which are rescored by three additional language models to generate the final translations via re-ranking. The preprocessor performs rule-based translation for number, date and time expressions, as well as some cleanup. The translation engine is a CKY-style decoder, which performs parsing and generation simultaneously guided by a language model and synchronous context free grammars (SCFGs). The SCFGs are extracted from parallel text with word alignments generated by GIZA++, in the similar manner described by Chiang (2007). The three rescoring language models include a count-based LM from Google Tera-word corpus, an almost parsing class LM based on SARV tags, and an approximated parser based LM (Wang et al., 2007).

All eight individual systems are optimized with maximum-BLEU training on different subsets of the previous NIST MT test data.

5.4 Experimental Results

5.4.1 Comparison with TER alignment

In the IHMM-based method, the smoothing factor for surface similarity model is set to ρ = 3, the interpolation factor of the overall similarity model is set to α = 0.3, and the controlling factor of the distance-based distortion parameters is set to K=2. These settings are optimized on the dev set. Individual system results and system combination results using both IHMM and TER alignment, on both the dev and test sets, are presented in Table 1. The TER-based hypothesis alignment tool used in our experiments is the publicly available TER Java program, TERCOM (Snover et al., 2006). Default settings of TERCOM are used in the following experiments.

On the dev set, the case insensitive BLEU score of the IHMM-based 8-way system combination output is about 5.8 points higher than that of the best single system. Compared to the TER-based method, the IHMM-based method is about 1.5 BLEU points better. On the MT08 test set, the IHMM-based system combination gave a case sensitive BLEU score of 30.89%. It outperformed the best single system by 4.7 BLEU points and the TER-based system combination by 1.0 BLEU points. Note that the best single system on the dev set and the test set are different. The different single systems are optimized on different tuning sets, so this discrepancy between dev set and test set results is presumably due to differing degrees of mismatch between the dev and test sets and the various tuning sets.

Table 1. Results of single and combined systems on the dev set and the MT08 test set

System	Dev ciBLEU%	MT08 BLEU%
Sys- 1	34.08	21.75
Sys-2	33.78	20.42
Sys- 3	34.75	21.69
Sys-4	37.85	25.52
Sys- 5	37.80	24.57
Sys- 6	37.28	24.40
Sys- 7	32.37	25.51
Sys- 8	34.98	26.24
TER	42.11	29.89
IHMM	43.62	30.89

In order to evaluate how well our method performs when we combine more systems, we collected MT outputs on MT08 from seven additional single systems as summarized in Table 2. These systems belong to two groups. Sys-9 to Sys-12 are in the first group. They are syntax-augmented hierarchical systems similar to those described by Shen et al. (2008) using different Chinese word segmentation and language models. The second group has Sys-13 to Sys-15. Sys-13 is a phrasal system proposed by Koehn et al. (2003), Sys-14 is a hierarchical system proposed by Chiang (2007), and Sys-15 is a syntax-based system proposed by Galley et al. (2006). All seven systems were trained within the confines of the constrained training condition of NIST MT08 evaluation.

We collected 10-best MT outputs only on the MT08 test set from these seven extra systems. No MT outputs on our dev set are available from them at present. Therefore, we directly adopt system combination parameters trained for the previous 8-way system combination, except the system weights, which are re-set by the following heuristics: First, the total system weight mass 1.0 is evenly divided among the three groups of single systems: {Sys-1~8}, {Sys-9~12}, and {Sys-13~15}. Each group receives a total system weight mass of 1/3. Then the weight mass is further divided in each group: in the first group, the original weights of systems 1~8 are multiplied by 1/3; in the second and third groups, the weight mass is evenly distributed within the group, i.e., 1/12 for each system in group 2, and 1/9 for each system in group 3[6]. Length adaptation is applied to control the final output length, where the same expected length ratio of the previous 8-way system combination is adopted.

The results of the 15-way system combination are presented in Table 3. It shows that the IHMM-based method is still about 1 BLEU point better than the TER-based method. Moreover, combining 15 single systems gives an output that has a NIST BLEU score of 34.82%, which is 3.9 points better than the best submission to the NIST MT08 constrained training track (NIST, 2008). To our knowledge, this is the best result reported on this task.

[6] This is just a rough guess because no dev set is available. We believe a better set of system weights could be obtained if MT outputs on a common dev set were available.

Table 2. Results of seven additional single systems on the NIST MT08 test set

System	MT08 BLEU%
Sys- 9	29.59
Sys- 10	29.57
Sys- 11	29.64
Sys- 12	29.85
Sys- 13	25.53
Sys- 14	26.04
Sys- 15	29.70

Table 3. Results of the 15-way system combination on the NIST MT08 C2E test set

Sys. Comb.	MT08 BLEU%
TER	33.81
IHMM	34.82

5.4.2 Effect of the similarity model

In this section, we evaluate the effect of the semantic similarity model and the surface similarity model by varying the interpolation weight a of (2). The results on both the dev and test sets are reported in Table 4.

Table 4. Effect of the similarity model

	Dev ciBLEU%	Test BLEU%
$a = 1.0$	41.70	28.92
$a = 0.7$	42.86	30.50
$a = 0.5$	43.11	30.94
$a = 0.3$	43.62	30.89
$a = 0.0$	43.35	30.73

In one extreme case, $a = 1$, the overall similarity model is based only on semantic similarity. This gives a case insensitive BLEU score of 41.70% and a case sensitive BLEU score of 28.92% on the dev and test set, respectively. The accuracy is significantly improved to 43.62% on the dev set and 30.89% on test set when $a = 0.3$. In another extreme case, $a = 0$, in which only the surface similarity model is used for the overall similarity model, the performance degrades by about 0.2 point. Therefore, the surface similarity information seems more important for monolingual hypothesis alignment, but both sub-models are useful.

5.4.3 Effect of the distortion model

We investigate the effect of the distance-based distortion model by varying the controlling factor K in (6). For example, setting K=1.0 gives a linear-decay distortion model, and setting K=2.0 gives a quadratic smoothed distance-based distortion model. As shown in Table 5, the optimal result can be achieved using a properly smoothed distance-based distortion model.

Table 5. Effect of the distortion model

	Dev	Test
	ciBLEU%	BLEU%
K=1.0	42.94	30.44
K=2.0	43.62	30.89
K=4.0	43.17	30.30
K=8.0	43.09	30.01

5.4.4 Effect of length adaptation

As described in section 5.1, length adaptation is applied to system combination to reduce the fluctuation of BLEU scores caused by the inconsistent translation output length. In this section, we investigate the effect of length adaptation measured by the BLEU score. The IHMM based method is used in the experiments in this section.

In order to investigate the effect of length adaptation under different dev/test mismatch condition, we collect a different dev set, called *dev_mix*, for system combination model training. Unlike the previous dev set which contains only newswire data, this dev_mix set contains 501 newswire sentences generated in a similar way as *dev*, plus 483 sentences of newsgroup/web data from NIST MT06 test set. Therefore, the *dev_mix* is more consistent with the test data MT08, which also contains both newswire and web data.

The results are tabulated in table 6. Results on the dev set are not reported because that *dev_mix* contains data different from *dev* so the results are not comparable. However, the results on the test set are comparable and are presented.

Table 6. Effect of length adaptation on MT08 C2E "current" test set measured by BLEU scores

Development	w/o Len. Adpt.	w/ Len. Adpt.
set	BLEU% (BP)	BLEU% (BP)
dev *(newswire only)*	29.86 (0.8920)	30.89 (0.9521)
dev_mix *(newswire+web)*	30.48 (0.9281)	30.80 (0.9514)

As shown in table 6, when there is a severe mismatch between development and test sets, the feature weights max-BLEU-trained on the development set may not be suitable for the test data and may lead to biased output length which causes heavy brevity penalty on the BLEU score on the test set. After length adaptation, the system is encouraged to output a translation that has more consistent hyp/src ratio and achieve a better result.

5.4.5 Examples of System Combination for Chinese-to-English Translation

In this section two real examples from our eight-way IHMM based system combination experiment are presented. As shown in the examples, in some cases the algorithm is able to pick the best translation among the multiple hypotheses, and sometimes it can compose a translation better than the best hypotheses from any of the individual systems. Note that the best hypotheses are not always from the same single system. For example, the best single system in example 1 is SYS-5, while in the second example SYS-1 gets most of the translation text right.

Example 1

Source (Chinese) and human translation reference (English)

Source: 前一阵儿，王燕已能在医生的帮助下在床上坚持坐 5 分钟。

Human ref: A while before, with the help of doctors, Wang Yan was able to remain sitting in bed for five minutes.

MT outputs from the eight individual systems

SYS1: the former wang yan, with the help of doctors have been able to stick to sit in bed for five minutes.
SYS2: in a little while ago son, wang yan in bed with the help of a doctor to sit for five minutes.
SYS3: ago, wang yan had been able to sit in bed with the help of doctors in five minutes.
SYS4: former, wang yan had been under the help of doctors in bed persist in sitting for five minutes.
SYS5: ago, wang yan had been able to sit in bed for five minutes with the help of doctors.
SYS6: before, to uphold minutes in bed with the help of doctors.
SYS7: earlier, has been able to persist in bed with the help of doctors sit five minutes.
SYS8: earlier, doctors have been able to uphold the help of the bed minutes.

Confusion network and the output of system combination (in bold)

```
=================================================================================================
-  - -      -     the     -   former wang    yan  , have been able to stick to -       sit
in a little while ago     son ,      wang    yan  - -    -    -    -  -     to -       sit
-  - -      -     ago     -   ,      wang    yan  - had  been able -  -     to -       sit
-  - -      -     former  -   ,      wang    yan  - had  been -    -  -     -  -       -
-  - -      -     ago     -   ,      wang    yan  - had  been able -  -     to -       sit
-  - -      -     before  -   ,      -       -    - -    -    -    -  -     to uphold  minutes
-  - -      -     earlier -   ,      -       -    - has  been able -  -     to -       persist
-  - -      -     earlier -   ,      doctors -    - have been able -  -     to -       uphold
=================================================================================================
                  Earlier     ,      Wang    Yan    had  been able          to         sit

=================================================================================================
-   in -  bed -       -       - for  five minutes with the  help of -   -   doctors .
-   in -  bed -       -       - for  five minutes with the  help of a   -   doctor  .
-   in in bed -       -       - -    five minutes with the  help of -   -   doctors .
-   in in bed persist sitting - for  five minutes underthe  help of -   -   doctors .
-   in -  bed -       -       - for  five minutes with the  help of -   -   doctors .
-   in -  bed -       -       - -    -    -       with the  help of -   -   doctors .
sit in -  bed -       -       - -    five minutes with the  help of -   -   doctors .
-   -  -  -   -       -       - -    -    -       -    the  help of the bed minutes .
=================================================================================================
    in    bed                   for  five minutes with the  help of         doctors .
```

Example 2

Source (Chinese) and human translation reference (English)

Source: 过了三个星期,系主任找我谈话,大意是告诉我他的工作就是要对每个学生包括我负责.

REF: Three weeks later, the dean approached me to have a talk. He told me that his job was be responsible toward every student, including me, something to that effect.

MT outputs from the eight individual systems

SYS1: after three weeks, the head of the department talked to me told me to the effect that his job is to each of the students, including my responsibility .
SYS2: in the past three weeks, talked to me, that is, head of the department told me that he is responsible for each student like me.
SYS3: after three weeks, the head of my speech, the effect is to inform me that he is for each student, including me.
SYS4: after three weeks, i head, told me that he is to the effect that each student, including me.
SYS5: after three weeks, the head of my speech, to the effect that he is responsible for the work, including my students told me.
SYS6: after three weeks at me and told me that he is fallacious task is to include my responsibility to each student
SYS7: after three weeks, the department found i talk to effect told me his job is to include i am responsible for each student.
SYS8: after three weeks of talks is to find my fallacious told me that he is responsible for including me for each student.

Confusion network and the output of system combination (in bold)

```
==============================================================================================================
after -  -   -    three weeks -  , - -    the  - head of -  -      - the department -     - talked -  to -
-     in the past three weeks -  , , that is   , head of -  -      - the department -     - talked -  to -
after -  -   -    three weeks -  , - -    the  - head of my speech , the effect     -     - is     -  to
inform
after -  -   -    three weeks -  , - -    i    - head ,  -  -      - -   -          -     - -      -  -  -
after -  -   -    three weeks -  , - -    the  - head of -  -      - -   -          -     - -      -  -  -
after -  -   -    three weeks at - - -    -    - -    -  -  -      - -   -          -     - -      -  -  -
after -  -   -    three weeks -  , - -    the  - -    -  -  -      - -   department found i talk   -  to -
after -  -   -    three weeks -  - - -    -    - -    of -  -      - -   -          -     - talks  is to find
==============================================================================================================
After             three weeks    ,        the    head of             the department         talked    to

==============================================================================================================
me     -          told -  -      - me to the effect -    that his job -  -          -    is -          -
me     -          told -  -      - me -  -   -      -    that he  -   -  -          -    is -          -
me     -          -    -  -      - -  -  -   -      -    that he  -   -  -          -    is -          -
-      -          told -  -      - me to the effect that that he  -   -  -          -    is -          -
-      -          told my speech , me to the effect -    that he  -   -  -          -    is -          -
me     and        told -  -      - me -  -   -      -    that he  -   is fallacious task is -          -
effect -          told -  -      - me -  -   -      -    -    his job -  -          -    is -          -
my     fallacious told -  -      - me -  -   -      -    that he  -   -  -          -    is responsible for
==============================================================================================================
me     and        told           me to the effect      that he                         is responsible for

==============================================================================================================
to          -   each of  the students -        , including my   -  responsibility -    -       -       .
responsible for each -   -   student  -        - -         like -  me             -    -       -       .
for         -   each -   -   student  -        , including me   -  -              -    -       -       .
-           -   each -   -   student  -        , including me   -  -              -    -       -       .
responsible -   -    for the work     students , including my   -  -              -    -       -       .
to          to  each -   -   student  -        - include   my   -  responsibility -    -       -       -
to          -   -    -   -   -        -        - include   i    am responsible    for  each    student .
-           -   -    -   -   -        -        - including me   -  for            each student -       .
==============================================================================================================
                each         student           , including me                                          .
```

Figure 4: Confusion-network-based MT system combination

6. DISCUSSION

Synonym matching and word ordering are two central issues for hypothesis alignment in confusion-network-based MT system combination. In this paper, an IHMM-based method is proposed for hypothesis alignment. It uses a similarity model for synonym matching and a distortion model for word ordering. In contrast to previous methods, the similarity model explicitly incorporates both semantic and surface word similarity, which is critical to monolingual word alignment, and a smoothed distance-based distortion model is used to model the first-order dependency of word ordering, which is shown to be better than simpler approaches.

Our experimental results show that the IHMM-based hypothesis alignment method gave superior results on the NIST MT08 C2E test set compared to the TER-based method. Moreover, we show that our system combination method can scale up to combining more systems and produce a better output that has a case sensitive BLEU score of 34.82, which is 3.9 BLEU points better than the best official submission of MT08.

ACKNOWLEDGEMENT

The authors are grateful to Chris Quirk, Arul Menezes, Kristina Toutanova, William Dolan, Mu Li, Chi-Ho Li, Dongdong Zhang, Long Jiang, Ming Zhou, George Foster, Roland Kuhn, Jing Zheng, Wen Wang, Necip Fazil Ayan, Dimitra Vergyri, Nicolas Scheffer, Andreas Stolcke, Kevin Knight, Jens-Soenke Voeckler, Spyros Matsoukas, and Antti-Veikko Rosti for assistance with the MT systems and/or for the valuable suggestions and discussions.

REFERENCES

Srinivas Bangalore, German Bordel, and Giuseppe Riccardi. 2001. Computing Consensus Translation From Multiple Machine Translation Systems. In *Proc. of IEEE ASRU*, pp. 351–354.

Richard Brent, 1973. Algorithms for Minimization Without Derivatives. Prentice-Hall, Chapter 7.

David Chiang. 2007. Hierarchical Phrase-Based Translation. *Computational Linguistics*, 33(2):201–228.

George Foster and Roland Kuhn. 2007. Mixture-Model Adaptation for SMT. In *Proc. of the Second ACL Workshop on Statistical Machine Translation*. pp. 128 – 136.

Michel Galley, Jonathan Graehl, Kevin Knight, Daniel Marcu, Steve DeNeefe, Wei Wang and Ignacio Thayer. 2006. Scalable Inference and Training of Context-Rich Syntactic Translation Models. In *Proc. of COLING-ACL*, pp. 961–968.

Jianfeng Gao, Mu Li, Andi Wu and Chang-Ning Huang. (2005). Chinese Word Segmentation And Named Entity Recognition: A Pragmatic Approach. *Computational Linguistics,* 31(4).

Xiaodong He. 2007. Using Word-Dependent Transition Models in HMM based Word Alignment for Statistical Machine Translation. In *Proc. of the Second ACL Workshop on Statistical Machine Translation*.

Liang Huang and David Chiang. (2007). Forest Rescoring: Faster Decoding with Integrated Language Models. Proc ACL.

Shyamsundar Jayaraman and Alon Lavie. 2005. Multi-Engine Machine Translation Guided By Explicit Word Matching. In *Proc. of EAMT*. pp. 143 – 152.

Damianos Karakos, Jason Eisner, Sanjeev Khudanpur, and Markus Dreyer. 2008. Machine Translation System Combination using ITG-based Alignments. In *Proc. of ACL-HLT*, pp. 81–84.

Roger Levy and Christopher Manning. 2003. Is It Harder To Parse Chinese, Or The Chinese Treebank? Published in Proceedings of ACL 2003

Chi-Ho Li, Minghui Li, Dongdong Zhang, Mu Li, Ming Zhou, Yi Guan. 2007. A Probabilistic Approach to Syntax-based Reordering for Statistical Machine Translation. In *Proc. of ACL*. pp. 720 – 727.

Percy Liang, Ben Taskar, and Dan Klein. 2006. Alignment by Agreement. In *Proc. of NAACL*. pp 104 – 111.

Evgeny Matusov, Nicola Ueffing, and Hermann Ney. 2006. Computing Consensus Translation From Multiple Machine Translation Systems Using Enhanced Hypotheses Alignment. In *Proc. of EACL*, pp. 33–40.

Robert Moore and Chris Quirk. 2007. Faster Beam-Search Decoding for Phrasal Statistical Machine Translation. In *Proc. of MT Summit XI*.

Patrick Nguyen, Jianfeng Gao and Milind Mahajan. 2007. MSRLM: a Scalable Language Modeling Toolkit. *Microsoft Research Technical Report MSR-TR-2007-144.*

NIST. 2008. The 2008 NIST Open Machine Translation Evaluation. www.nist.gov/speech/tests/mt/2008/doc/

Franz J. Och and Hermann Ney. 2003. A Systematic Comparison Of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–51.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A Method For Automatic Evaluation Of Machine Translation. In *Proc. of ACL*, pp. 311–318.

Koehn, Philipp, Franz Josef Och, and Daniel Marcu. 2003. Statistical Phrase Based Translation. In *Proc. of NAACL*. pp. 48 – 54.

Chris Quirk, Arul Menezes, and Colin Cherry. 2005. Dependency Treelet Translation: Syntactically Informed Phrasal SMT. In *Proc. of ACL*. pp. 271–279.

Antti-Veikko I. Rosti, Bing Xiang, Spyros Matsoukas, Richard Schwartz, Necip Fazil Ayan, and Bonnie J. Dorr. 2007a. Combining Outputs From Multiple Machine Translation Systems. In *Proc. of NAACL-HLT*, pp. 228–235.

Antti-Veikko I. Rosti, Spyros Matsoukas, and Richard Schwartz. 2007b. Improved Word-Level System Combination for Machine Translation. In *Proc. of ACL*, pp. 312–319.

Antti-Veikko I. Rosti, Bing Zhang, Spyros Matsoukas, and Richard Schwartz. 2008. Incremental Hypothesis Alignment for Building Confusion Networks with Application to Machine Translation System Combination, In *Proc. of the Third ACL Workshop on Statistical Machine Translation*, pp. 183–186.

Libin Shen, Jinxi Xu, Ralph Weischedel. 2008. A New String-to-Dependency Machine Translation Algorithm with a Target Dependency Language Model. In *Proc. of ACL-HLT*, pp. 577–585.

Khe Chai Sim, William J. Byrne, Mark J.F. Gales, Hichem Sahbi, and Phil C. Woodland. 2007. Consensus Network Decoding For Statistical Machine Translation System Combination. In *Proc. of ICASSP*, *vol. 4*. pp. 105–108.

Matthew Snover, Bonnie Dorr, Rich Schwartz, Linnea Micciulla, and John Makhoul. 2006. A Study Of Translation Edit Rate With Targeted Human Annotation. In *Proc. of AMTA*.

Kristina Toutanova, Hisami Suzuki and Achim Ruopp. 2008. Applying Morphology Generation Models to Machine Translation. In *Proc. of ACL*. pp. 514 – 522.

Stephan Vogel, Hermann Ney, and Christoph Tillmann. 1996. HMM-based Word Alignment In Statistical Translation. In *Proc. of COLING*. pp. 836-841.

Chao Wang, Michael Collins, and Philipp Koehn. 2007a. Chinese Syntactic Reordering for Statistical Machine Translation. In *Proc. of EMNLP-CoNLL*. pp. 737-745.

Wen Wang, Andreas Stolcke, Jing Zheng. 2007b. Reranking Machine Translation Hypotheses With Structured and Web-based Language Models. In *Proc. of IEEE ASRU*. pp. 159 – 164.

Deyi Xiong, Qun Liu and Shouxun Lin. 2006. Maximum Entropy Based Phrase Reordering Model for Statistical Machine Translation. In *Proc. of ACL*. pp. 521 – 528.