Incremental HMM Alignment for MT System Combination

Chi-Ho Li
Microsoft Research Asia
49 Zhichun Road, Beijing, China
chl@microsoft.com

Xiaodong He
Microsoft Research
One Microsoft Way, Redmond, USA
xiaohe@microsoft.com

Yupeng Liu
Harbin Institute of Technology
92 Xidazhi Street, Harbin, China
ypliu@mtlab.hit.edu.cn

Ning Xi
Nanjing University
8 Hankou Road, Nanjing, China
xin@nlp.nju.edu.cn

Abstract

Inspired by the incremental TER alignment, we re-designed the Indirect HMM (IHMM) alignment, which is one of the best hypothesis alignment methods for conventional MT system combination, in an incremental manner. One crucial problem of incremental alignment is to align a hypothesis to a confusion network (CN). Our incremental IHMM alignment is implemented in three different ways: 1) treat CN spans as HMM states and define state transition as distortion over covered n-grams between two spans; 2) treat CN spans as HMM states and define state transition as distortion over words in component translations in the CN; and 3) use a consensus decoding algorithm over one hypothesis and multiple IHMMs, each of which corresponds to a component translation in the CN. All these three approaches of incremental alignment based on IHMM are shown to be superior to both incremental TER alignment and conventional IHMM alignment in the setting of the Chinese-to-English track of the 2008 NIST Open MT evaluation.

1 Introduction

Word-level combination using confusion network (Matusov et al. (2006) and Rosti et al. (2007)) is a widely adopted approach for combining Machine Translation (MT) systems' output. Word alignment between a backbone (or skeleton) translation and a hypothesis translation is a key problem in this approach. Translation Edit Rate (TER, Snover et al. (2006)) based alignment proposed in Sim et al. (2007) is often taken as the baseline, and a couple of other approaches, such as the Indirect Hidden Markov Model (IHMM, He et al. (2008)) and the ITG-based alignment (Karakos et al. (2008)), were recently proposed with better results reported. With an alignment method, each hypothesis is aligned against the backbone and all the alignments are then used to build a confusion network (CN) for generating a better translation.

However, as pointed out by Rosti et al. (2008), such a *pair-wise* alignment strategy will produce a low-quality CN if there are errors in the alignment of any of the hypotheses, no matter how good the alignments of other hypotheses are. For example, suppose we have the backbone "he buys a computer" and two hypotheses "he bought a laptop computer" and "he buys a laptop". It will be natural for most alignment methods to produce the alignments in Figure 1a. The alignment of hypothesis 2 against the backbone cannot be considered an error if we consider only these two translations; nevertheless, when added with the alignment of another hypothesis, it produces the low-quality CN in Figure 1b, which may generate poor translations like "he bought a laptop laptop". While it could be argued that such poor translations are unlikely to be selected due to language model, this CN does disperse the votes to the word "laptop" to two distinct arcs.

Rosti et al. (2008) showed that this problem can be rectified by incremental alignment. If hypothesis 1 is first aligned against the backbone, the CN thus produced (depicted in Figure 2a) is then aligned to hypothesis 2, giving rise to the good CN as depicted in Figure 2b.[1] On the other hand, the

[1] Note that this CN may generate an incomplete sentence "he bought a", which is nevertheless unlikely to be selected as it leads to low language model score.

Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP, pages 949–957,
Suntec, Singapore, 2-7 August 2009.

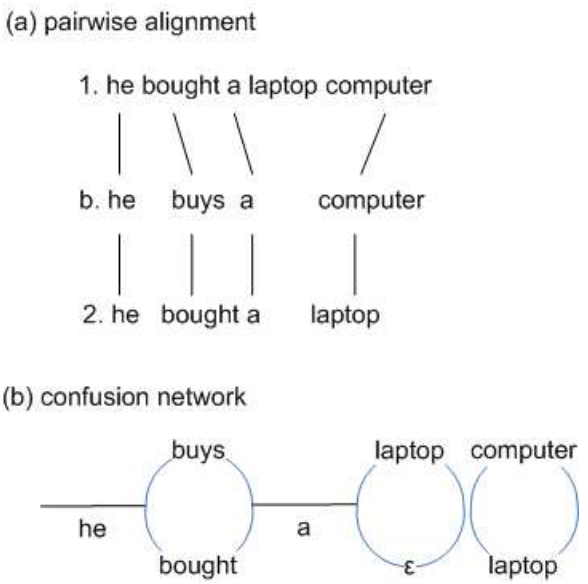

Figure 1: An example bad confusion network due to pair-wise alignment strategy

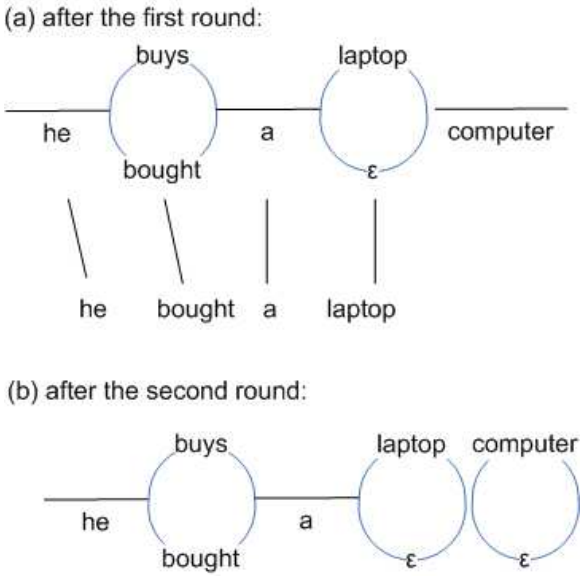

Figure 2: An example good confusion network due to incremental alignment strategy

correct result depends on the order of hypotheses. If hypothesis 2 is aligned before hypothesis 1, the final CN will not be good. Therefore, the observation in Rosti et al. (2008) that different order of hypotheses does not affect translation quality is counter-intuitive.

This paper attempts to answer two questions: 1) as incremental TER alignment gives better performance than pair-wise TER alignment, would the incremental strategy still be better than the pairwise strategy if the TER method is replaced by another alignment method? 2) how does translation quality vary for different orders of hypotheses being incrementally added into a CN? For question 1, we will focus on the IHMM alignment method and propose three different ways of implementing incremental IHMM alignment. Our experiments will also try several orders of hypotheses in response to question 2.

This paper is structured as follows. After setting the notations on CN in section 2, we will first introduce, in section 3, two variations of the basic incremental IHMM model (IncIHMM1 and IncIHMM2). In section 4, a consensus decoding algorithm (CD-IHMM) is proposed as an alternative way to search for the optimal alignment. The issues of alignment normalization and the order of hypotheses being added into a CN are discussed in sections 5 and 6 respectively. Experiment results and analysis are presented in section 7.

2 Preliminaries: Notation on Confusion Network

Before the elaboration of the models, let us first clarify the notation on CN. A CN is usually described as a finite state graph with many spans. Each span corresponds to a word position and contains several arcs, each of which represents an alternative word (could be the empty symbol , ϵ) at that position. Each arc is also associated with M weights in an M-way system combination task. Follow Rosti et al. (2007), the i-th weight of an arc is $\sum_r \frac{1}{1+r}$, where r is the rank of the hypothesis in the i-th system that votes for the word represented by the arc. This conception of CN is called the *conventional* or *compact* form of CN. The networks in Figures 1b and 2b are examples.

On the other hand, as a CN is an integration of the skeleton and all hypotheses, it can be conceived as a list of the component translations. For example, the CN in Figure 2b can be converted to the form in Figure 3. In such an *expanded* or *tabular* form, each row represents a component translation. Each column, which is equivalent to a span in the compact form, comprises the alternative words at a word position. Thus each cell represents an alternative word at certain word position voted by certain translation. Each row is assigned the weight $\frac{1}{1+r}$, where r is the rank of the translation of some MT system. It is assumed that all MT systems are weighted equally and thus the

he	buys	a	ε	computer
he	bought	a	laptop	computer
he	bought	a	laptop	ε

Figure 3: An example of confusion network in tabular form

rank-based weights from different system can be compared to each other without adjustment. The weight of a cell is the same as the weight of the corresponding row. In this paper the elaboration of the incremental IHMM models is based on such tabular form of CN.

Let $E_1^I = (E_1 \dots E_I)$ denote the backbone CN, and $e_1'^J = (e_1' \dots e_J')$ denote a hypothesis being aligned to the backbone. Each e_j' is simply a word in the target language. However, each E_i is a span, or a column, of the CN. We will also use $E(k)$ to denote the k-th row of the tabular form CN, and $E_i(k)$ to denote the cell at the k-th row and the i-th column. $W(k)$ is the weight for $E(k)$, and $W_i(k) = W(k)$ is the weight for $E_i(k)$. $p_i(k)$ is the normalized weight for the cell $E_i(k)$, such that $p_i(k) = \frac{W_i(k)}{\sum_i W_i(k)}$. Note that $E(k)$ contains the same bag-of-words as the k-th original translation, but may have different word order. Note also that $E(k)$ represents a word sequence with inserted empty symbols; the sequence with all inserted symbols removed is known as the *compact* form of $E(k)$.

3 The Basic IncIHMM Model

A naïve application of the incremental strategy to IHMM is to treat a span in the CN as an HMM state. Like He et al. (2008), the conditional probability of the hypothesis given the backbone CN can be decomposed into similarity model and distortion model in accordance with equation 1

$$p(e_1'^J|E_1^I) = \sum_{a_1^J} \prod_{j=1}^{J} [p(a_j|a_{j-1}, I) p(e_j'|e_{a_j})] \quad (1)$$

The similarity between a hypothesis word e_j' and a span E_i is simply a weighted sum of the similarities between e_j' and each word contained in E_i as equation 2:

$$p(e_j'|E_i) = \sum_{E_i(k) \epsilon E_i} p_i(k) \cdot p(e_j'|E_i(k)) \quad (2)$$

The similarity between two words is estimated in exactly the same way as in conventional IHMM alignment.

As to the distortion model, the incremental IHMM model also groups distortion parameters into a few 'buckets':

$$c(d) = (1 + |d - 1|)^{-K}$$

The problem in incremental IHMM is when to apply a bucket. In conventional IHMM, the transition from state i to j has probability:

$$p'(j|i, I) = \frac{c(j - i)}{\sum_{l=1}^{I} c(l - i)} \quad (3)$$

It is tempting to apply the same formula to the transitions in incremental IHMM. However, the backbone in the incremental IHMM has a special property that it is gradually expanding due to the insertion operator. For example, initially the backbone CN contains the option e_i in the i-th span and the option e_{i+1} in the $(i+1)$-th span. After the first round alignment, perhaps e_i is aligned to the hypothesis word e_j', e_{i+1} to e_{j+2}', and the hypothesis word e_{j+1}' is left unaligned. Then the consequent CN have an extra span containing the option e_{j+1}' inserted between the i-th and $(i + 1)$-th spans of the initial CN. If the distortion buckets are applied as in equation 3, then in the first round alignment, the transition from the span containing e_i to that containing e_{i+1} is based on the bucket $c(1)$, but in the second round alignment, the same transition will be based on the bucket $c(2)$. It is therefore not reasonable to apply equation 3 to such gradually extending backbone as the monotonic alignment assumption behind the equation no longer holds.

There are two possible ways to tackle this problem. The first solution estimates the transition probability as a weighted average of different distortion probabilities, whereas the second solution converts the distortion over spans to the distortion over the words in each hypothesis $E(k)$ in the CN.

3.1 Distortion Model 1: simple weighting of covered n-grams

Distortion Model 1 shifts the monotonic alignment assumption from spans of CN to n-grams covered by state transitions. Let us illustrate this point with the following examples.

In conventional IHMM, the distortion probability $p'(i + 1|i, I)$ is applied to the transition from state i to $i+1$ given I states because such transition

jumps across only one word, viz. the i-th word of the backbone. In incremental IHMM, suppose the i-th span covers two arcs e_a and ϵ, with probabilities p_1 and $p_2 = 1 - p_1$ respectively, then the transition from state i to $i+1$ jumps across one word (e_a) with probability p_1 and jumps across nothing with probability p_2. Thus the transition probability should be $p_1 \cdot p'(i+1|i,I) + p_2 \cdot p'(i|i,I)$.

Suppose further that the $(i+1)$-th span covers two arcs e_b and ϵ, with probabilities p_3 and p_4 respectively, then the transition from state i to $i+2$ covers 4 possible cases:

1. nothing ($\epsilon\epsilon$) with probability $p_2 \cdot p_4$;
2. the unigram e_a with probability $p_1 \cdot p_4$;
3. the unigram e_b with probability $p_2 \cdot p_3$;
4. the bigram $e_a e_b$ with probability $p_1 \cdot p_3$.

Accordingly the transition probability should be

$$p_2 p_4 p'(i|i,I) + p_1 p_3 p'(i+2|i,I) + (p_1 p_4 + p_2 p_3) p'(i+1|i,I).$$

The estimation of transition probability can be generalized to any transition from i to i' by expanding all possible n-grams covered by the transition and calculating the corresponding probabilities. We enumerate all possible cell sequences $S(i,i')$ covered by the transition from span i to i'; each sequence is assigned the probability

$$P_i^{i'} = \prod_{q=i}^{i'-1} p_q(k).$$

where the cell at the i'-th span is on some row $E(k)$. Since a cell may represent an empty word, a cell sequence may represent an n-gram where $0 \le n \le i' - i$ (or $0 \le n \le i - i'$ in backward transition). We denote $|S(i,i')|$ to be the length of n-gram represented by a particular cell sequence $S(i,i')$. All the cell sequences $S(i,i')$ can be classified, with respect to the length of corresponding n-grams, into a set of parameters where each element (with a particular value of n) has the probability

$$P_i^{i'}(n;I) = \sum_{|S(i,i')|=n} P_i^{i'}.$$

The probability of the transition from i to i' is:

$$p(i'|i,I) = \sum_n [P_i^{i'}(n;I) \cdot p'(i+n|i,I)]. \quad (4)$$

That is, the transition probability of incremental IHMM is a weighted sum of probabilities of 'n-gram jumping', defined as conventional IHMM distortion probabilities.

However, in practice it is not feasible to expand all possible n-grams covered by any transition since the number of n-grams grows exponentially. Therefore a length limit L is imposed such that for all state transitions where $|i' - i| \le L$, the transition probability is calculated as equation 4, otherwise it is calculated by:

$$p(i'|i,I) = \max_q p(i'|q,I) \cdot p(q|i,I)$$

for some q between i and i'. In other words, the probability of longer state transition is estimated in terms of the probabilities of transitions shorter or equal to the length limit.[2] All the state transitions can be calculated efficiently by dynamic programming.

A fixed value P_0 is assigned to transitions to null state, which can be optimized on held-out data. The overall distortion model is:

$$\tilde{p}(j|i,I) = \begin{cases} P_0 & \text{if } j \text{ is null state} \\ (1-P_0)p(j|i,I) & \text{otherwise} \end{cases}$$

3.2 Distortion Model 2: weighting of distortions of component translations

The cause of the problem of distortion over CN spans is the gradual extension of CN due to the inserted empty words. Therefore, the problem will disappear if the inserted empty words are removed. The rationale of Distortion Model 2 is that the distortion model is defined over the actual word sequence in each component translation $E(k)$.

Distortion Model 2 implements a CN in such a way that the *real* position of the i-th word of the k-th component translation can always be retrieved. The real position of $E_i(k)$, $\delta(i,k)$, refers to the position of the word represented by $E_i(k)$ in the compact form of $E(k)$ (i.e. the form without any inserted empty words), or, if $E_i(k)$ represents an empty word, the position of the nearest preceding non-empty word. For convenience, we also denote by $\delta_\epsilon(i,k)$ the null state associated with the state of the real word $\delta(i,k)$. Similarly, the real length

[2] This limit L is also imposed on the parameter I in distortion probability $p'(i'|i,I)$, because the value of I is growing larger and larger during the incremental alignment process. I is defined as L if $I > L$.

of $E(k)$, $L(k)$, refers to the number of non-empty words of $E(k)$.

The transition from span i' to i is then defined as

$$p(i|i') = \frac{1}{\sum_k W(k)} \sum_k [W(k) \cdot p_k(i|i')] \quad (5)$$

where k is the row index of the tabular form CN.

Depending on $E_i(k)$ and $E_{i'}(k)$, $p_k(i|i')$ is computed as follows:

1. if both $E_i(k)$ and $E_{i'}(k)$ represent real words, then

   $$p_k(i|i') = p'(\delta(i,k)|\delta(i',k), L(k))$$

   where p' refers to the conventional IHMM distortion probability as defined by equation 3.

2. if $E_i(k)$ represents a real word but $E_{i'}(k)$ the empty word, then

   $$p_k(i|i') = p'(\delta(i,k)|\delta_\epsilon(i',k), L(k))$$

   Like conventional HMM-based word alignment, the probability of the transition from a null state to a real word state is the same as that of the transition from the real word state associated with that null state to the other real word state. Therefore,

   $$p'(\delta(i,k)|\delta_\epsilon(i',k), L(k)) = p'(\delta(i,k)|\delta(i',k), L(k))$$

3. if $E_i(k)$ represents the empty word but $E_{i'}(k)$ a real word, then

   $$p_k(i|i') = \begin{cases} P_0 & \text{if} \delta(i,k) = \delta(i',k) \\ P_0 P_\delta(i|i';k) & \text{otherwise} \end{cases}$$

   where $P_\delta(i|i';k) = p'(\delta(i,k)|\delta(i',k), L(k))$. The second option is due to the constraint that a null state is accessible only to itself or the real word state associated with it. Therefore, the transition from i' to i is in fact composed of the first transition from i' to $\delta(i,k)$ and the second transition from $\delta(i,k)$ to the null state at i.

4. if both $E_i(k)$ and $E_{i'}(k)$ represent the empty word, then, with similar logic as cases 2 and 3,

   $$p_k(i|i') = \begin{cases} P_0 & \text{if} \delta(i,k) = \delta(i',k) \\ P_0 P_\delta(i|i';k) & \text{otherwise} \end{cases}$$

4 Incremental Alignment using Consensus Decoding over Multiple IHMMs

The previous section describes an incremental IHMM model in which the state space is based on the CN taken as a whole. An alternative approach is to conceive the rows (component translations) in the CN as individuals, and transforms the alignment of a hypothesis against an entire network to that against the individual translations. Each individual translation constitutes an IHMM and the optimal alignment is obtained from consensus decoding over these multiple IHMMs.

Alignment over multiple sequential patterns has been investigated in different contexts. For example, Nair and Sreenivas (2007) proposed multi-pattern dynamic time warping (MPDTW) to align multiple speech utterances to each other. However, these methods usually assume that the alignment is monotonic. In this section, a consensus decoding algorithm that searches for the optimal (non-monotonic) alignment between a hypothesis and a set of translations in a CN (which are already aligned to each other) is developed as follows.

A prerequisite of the algorithm is a function for converting a span index to the corresponding HMM state index of a component translation. The two functions δ and δ_ϵ s defined in section 3.2 are used to define a new function:

$$\bar{\delta}(i,k) = \begin{cases} \delta_\epsilon(i,k) & \text{if } E_i(k) \text{ is null} \\ \delta(i,k) & \text{otherwise} \end{cases}$$

Accordingly, given the alignment $a_1^J = a_1 \ldots a_J$ of a hypothesis (with J words) against a CN (where each a_j is an index referring to the span of the CN), we can obtain the alignment $\tilde{a}_k = \bar{\delta}(a_1,k) \ldots \bar{\delta}(a_J,k)$ between the hypothesis and the k-th row of the tabular CN. The real length function $L(k)$ is also used to obtain the number of non-empty words of $E(k)$.

Given the k-th row of a CN, $E(k)$, an IHMM $\lambda(k)$ is formed and the cost of the pair-wise alignment, $\tilde{a}_k$, between a hypothesis h and $\lambda(k)$ is defined as:

$$C(\tilde{a_k}; h, \lambda(k)) = -\log P(\tilde{a}_k|h, \lambda(k)) \quad (6)$$

The cost of the alignment of h against a CN is then defined as the weighted sum of the costs of the K alignments $\tilde{a}_k$:

$$C(a; h, \Lambda) = \sum_k W(k) C(\tilde{a}_k; h, \lambda(k))$$

$$= -\sum_k W(k) \log P(\tilde{a}_k|h, \lambda(k))$$

where $\Lambda = \{\lambda(k)\}$ is the set of pair-wise IHMMs, and $W(k)$ is the weight of the k-th row. The optimal alignment $\hat{a}$ is the one that minimizes this cost:

$$\begin{aligned}
\hat{a} &= \arg\max_a \sum_k W(k) \log P(\tilde{a}_k|h, \lambda(k)) \\
&= \arg\max_a \sum_k W(k)[\sum_j [\\
&\quad \log P(\bar{\delta}(a_j, k)|\bar{\delta}(a_{j-1}, k), L(k)) + \\
&\quad \log P(e_j|E_i(k))]] \\
&= \arg\max_a \sum_j [\\
&\quad \sum_k W(k) \log P(\bar{\delta}(a_j, k)|\bar{\delta}(a_{j-1}, k), L(k)) + \\
&\quad \sum_k W(k) \log P(e_j|E_i(k))] \\
&= \arg\max_a \sum_j [\log P'(a_j|a_{j-1}) + \\
&\quad \log P'(e_j|E_{a_j})]
\end{aligned}$$

A Viterbi-like dynamic programming algorithm can be developed to search for $\hat{a}$ by treating CN spans as HMM states, with a pseudo emission probability as

$$P'(e_j|E_{a_j}) = \prod_{k=1}^{K} P(e_j|E_{a_j}(k))^{W(k)}$$

and a pseudo transition probability as

$$P'(j|i) = \prod_{k=1}^{K} P(\bar{\delta}(j, k)|\bar{\delta}(i, k), L(k))^{W(k)}$$

Note that $P'(e_j|E_{a_j})$ and $P'(j|i)$ are not true probabilities and do not have the sum-to-one property.

5 Alignment Normalization

After alignment, the backbone CN and the hypothesis can be combined to form an even larger CN. The same principles and heuristics for the construction of CN in conventional system combination approaches can be applied. Our incremental alignment approaches adopt the same heuristics for alignment normalization stated in He et al. (2008). There is one exception, though. All 1-N mappings are not converted to $N-1$ ϵ-1 mappings since this conversion leads to $N-1$ insertion in the CN and therefore extending the network to an unreasonable length. The Viterbi alignment is abandoned if it contains an 1-N mapping. The best alignment which contains no 1-N mapping is searched in the N-Best alignments in a way inspired by Nilsson and Goldberger (2001). For example, if both hypothesis words e'_1 and e'_2 are aligned to the same backbone span E_1, then all alignments $a_{j=\{1,2\}} = i$ (where $i \neq 1$) will be examined. The alignment leading to the least reduction of Viterbi probability when replacing the alignment $a_{j=\{1,2\}} = 1$ will be selected.

6 Order of Hypotheses

The default order of hypotheses in Rosti et al. (2008) is to rank the hypotheses in descending of their TER scores against the backbone. This paper attempts several other orders. The first one is *system-based* order, i.e. assume an arbitrary order of the MT systems and feeds all the translations (in their original order) from a system before the translations from the next system. The rationale behind the system-based order is that the translations from the same system are much more similar to each other than to the translations from other systems, and it might be better to build CN by incorporating similar translations first. The second one is *N-best rank-based* order, which means, rather than keeping the translations from the same system as a block, we feed the top-1 translations from all systems in some order of systems, and then the second best translations from all systems, and so on. The presumption of the rank-based order is that top-ranked hypotheses are more reliable and it seemed beneficial to incorporate more reliable hypotheses as early as possible. These two kinds of order of hypotheses involve a certain degree of randomness as the order of systems is arbitrary. Such randomness can be removed by imposing a *Bayes Risk* order on MT systems, i.e. arrange the MT systems in ascending order of the Bayes Risk of their top-1 translations. These four orders of hypotheses are summarized in Table 1. We also tried some intuitively bad orders of hypotheses, including the *reversal* of these four orders and the random order.

7 Evaluation

The proposed approaches of incremental IHMM are evaluated with respect to the constrained Chinese-to-English track of 2008 NIST Open MT

Order	Example
System-based	1:1 . . . 1:N 2:1 . . . 2:N . . . M:1 . . . M:N
N-best Rank-based	1:1 2:1 . . . M:1 . . . 1:2 2:2 . . . M:2 . . . 1:N . . . M:N
Bayes Risk + System-based	4:1 4:2 . . . 4:N . . . 1:1 1:2 . . . 1:N . . . 5:1 5:2 . . . 5:N
Bayes Risk + Rank-based	4:1 . . . 1:1 . . . 5:1 4:2 . . . 1:2 . . . 5:2 . . . 4:N . . . 1:N . . . 5:N

Table 1: The list of order of hypothesis and examples. Note that 'm:n' refers to the n-th translation from the m-th system.

Evaluation (NIST (2008)). In the following sections, the incremental IHMM approaches using distortion model 1 and 2 are named as IncIHMM1 and IncIHMM2 respectively, and the consensus decoding of multiple IHMMs as CD-IHMM. The baselines include the TER-based method in Rosti et al. (2007), the incremental TER method in Rosti et al. (2008), and the IHMM approach in He et al. (2008). The development (dev) set comprises the newswire and newsgroup sections of MT06, whereas the test set is the entire MT08. The 10-best translations for every source sentence in the dev and test sets are collected from eight MT systems. Case-insensitive BLEU-4, presented in percentage, is used as evaluation metric.

The various parameters in the IHMM model are set as the optimal values found in He et al. (2008). The lexical translation probabilities used in the semantic similarity model are estimated from a small portion (FBIS + GALE) of the constrained track training data, using standard HMM alignment model (Och and Ney (2003)). The backbone of CN is selected by MBR. The loss function used for TER-based approaches is TER and that for IHMM-based approaches is BLEU. As to the incremental systems, the default order of hypotheses is the ascending order of TER score against the backbone, which is the order proposed in Rosti et al. (2008). The default order of hypotheses for our three incremental IHMM approaches is N-best rank order with Bayes Risk system order, which is empirically found to be giving the highest BLEU score. Once the CN is built, the final system combination output can be obtained by decoding it with a set of features and decoding parameters. The features we used include word confidences, language model score, word penalty and empty word penalty. The decoding parameters are trained by maximum BLEU training on the dev set. The training and decoding processes are the same as described by Rosti et al. (2007).

Method	dev	test
best single system	32.60	27.75
pair-wise TER	37.90	30.96
incremental TER	38.10	31.23
pair-wise IHMM	38.52	31.65
incremental IHMM	39.22	32.63

Table 2: Comparison between IncIHMM2 and the three baselines

7.1 Comparison against Baselines

Table 2 lists the BLEU scores achieved by the three baseline combination methods and IncIHMM2. The comparison between pairwise and incremental TER methods justifies the superiority of the incremental strategy. However, the benefit of incremental TER over pair-wise TER is smaller than that mentioned in Rosti et al. (2008), which may be because of the difference between test sets and other experimental conditions. The comparison between the two pair-wise alignment methods shows that IHMM gives a 0.7 BLEU point gain over TER, which is a bit smaller than the difference reported in He et al. (2008). The possible causes of such discrepancy include the different dev set and the smaller training set for estimating semantic similarity parameters. Despite that, the pair-wise IHMM method is still a strong baseline. Table 2 also shows the performance of IncIHMM2, our best incremental IHMM approach. It is almost one BLEU point higher than the pair-wise IHMM baseline and much higher than the two TER baselines.

7.2 Comparison among the Incremental IHMM Models

Table 3 lists the BLEU scores achieved by the three incremental IHMM approaches. The two distortion models for IncIHMM approach lead to almost the same performance, whereas CD-IHMM is much less satisfactory.

For IncIHMM, the gist of both distortion mod-

Method	dev	test
IncIHMM1	39.06	32.60
IncIHMM2	39.22	32.63
CD-IHMM	38.64	31.87

Table 3: Comparison between the three incremental IHMM approaches

els is to shift the distortion over spans to the distortion over word sequences. In distortion model 2 the word sequences are those sequences available in one of the component translations in the CN. Distortion model 1 is more encompassing as it also considers the word sequences which are combined from subsequences from various component translations. However, as mentioned in section 3.1, the number of sequences grows exponentially and there is therefore a limit L to the length of sequences. In general the limit $L \geq 8$ would render the tuning/decoding process intolerably slow. We tried the values 5 to 8 for L and the variation of performance is less than 0.1 BLEU point. That is, distortion model 1 cannot be improved by tuning L. The similar BLEU scores as shown in Table 3 implies that the incorporation of more word sequences in distortion model 1 does not lead to extra improvement.

Although consensus decoding is conceptually different from both variations of IncIHMM, it can indeed be transformed into a form similar to IncIHMM2. IncIHMM2 calculates the parameters of the IHMM as a weighted sum of various probabilities of the component translations. In contrast, the equations in section 4 shows that CD-IHMM calculates the weighted sum of the logarithm of those probabilities of the component translations. In other words, IncIHMM2 makes use of the sum of probabilities whereas CD-IHMM makes use of the product of probabilities. The experiment results indicate that the interaction between the weights and the probabilities is more fragile in the product case than in the summation case.

7.3 Impact of Order of Hypotheses

Table 4 lists the BLEU scores on the test set achieved by IncIHMM1 using different orders of hypotheses. The column 'reversal' shows the impact of deliberately bad order, viz. more than one BLEU point lower than the best order. The random order is a baseline for not caring about order of hypotheses at all, which is about 0.7 BLEU point lower than the best order. Among the orders with good performance, it is observed that N-best rank order leads to about 0.2 to 0.3 BLEU point improvement, and that the Bayes Risk order of systems does not improve performance very much. In sum, the performance of incremental alignment is sensitive to the order of hypotheses, and the optimal order is defined in terms of the rank of each hypothesis on some system's n-best list.

	normal	reversal
System	32.36	31.46
Rank	32.53	31.56
BR+System	32.37	31.44
BR+Rank	32.6	31.47
random	31.94	

Table 4: Comparison between various orders of hypotheses. 'System' means system-based order; 'Rank' means N-best rank-based order; 'BR' means Bayes Risk order of systems. The numbers are the BLEU scores on the test set.

8 Conclusions

This paper investigates the application of the incremental strategy to IHMM, one of the state-of-the-art alignment methods for MT output combination. Such a task is subject to the problem of how to define state transitions on a gradually expanding CN. We proposed three different solutions, which share the principle that transition over CN spans must be converted to the transition over word sequences provided by the component translations. While the consensus decoding approach does not improve performance much, the two distortion models for incremental IHMM (IncIHMM1 and IncIHMM2) give superb performance in comparison with pair-wise TER, pair-wise IHMM, and incremental TER. We also showed that the order of hypotheses is important as a deliberately bad order would reduce translation quality by one BLEU point.

References

Xiaodong He, Mei Yang, Jianfeng Gao, Patrick Nguyen, and Robert Moore 2008. Indirect-HMM-based Hypothesis Alignment for Combining Outputs from Machine Translation Systems. *Proceedings of EMNLP 2008*.

Damianos Karakos, Jason Eisner, Sanjeev Khudanpur, and Markus Dreyer 2008. Machine Translation

System Combination using ITG-based Alignments. *Proceedings of ACL 2008*.

Evgeny Matusov, Nicola Ueffing and Hermann Ney. 2006. Computing Consensus Translation from Multiple Machine Translation Systems using Enhanced Hypothesis Alignment. *Proceedings of EACL*.

Nishanth Ulhas Nair and T.V. Sreenivas. 2007. Joint Decoding of Multiple Speech Patterns for Robust Speech Recognition. *Proceedings of ASRU*.

Dennis Nilsson and Jacob Goldberger 2001. Sequentially Finding the N-Best List in Hidden Markov Models. *Proceedings of IJCAI 2001*.

NIST 2008. The 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):pp 19-51

Kishore Papineni, Salim Roukos, Todd Ward and Wei-Jing Zhu 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. *Proceedings of ACL 2002*

Antti-Veikko I. Rosti, Spyros Matsoukas, and Richard Schwartz 2007. Improved Word-level System Combination for Machine Translation. *Proceedings of ACL 2007*.

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. *Proceedings of the 3rd ACL Workshop on SMT*.

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. *Proceedings of ICASSP* vol. 4.

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