Phrase-level System Combination for Machine Translation Based on Target-to-Target Decoding

Wei-Yun Ma
Department of Computer Science
Columbia University
New York, NY 10027, USA
ma@cs.columbia.edu

Kathleen McKeown
Department of Computer Science
Columbia University
New York, NY 10027, USA
kathy@cs.columbia.edu

Abstract

In this paper, we propose a novel lattice-based MT combination methodology that we call *Target-to-Target Decoding* (TTD). The combination process is carried out as a "translation" from backbone to the combination result. This perspective suggests the use of existing phrase-based MT techniques in the combination framework. We show how phrase extraction rules and confidence estimations inspired from machine translation improve results. We also propose system-specific LMs for estimating N-gram consensus. Our results show that our approach yields a strong improvement over the best single MT system and competes with other state-of-the-art combination systems.

1 Introduction

In the past several years, many machine translation (MT) combination approaches have been developed. Confusion Network (CN) decoding is one of the most successful approaches (Matusov et al., 2006; Rosti et al., 2007a; He et al. 2008; Karakos et al. 2008; Sim et al. 2007; Xu et al. 2011). A CN is a linear word lattice structure, in which the words in all translation hypotheses are aligned against the corresponding words of a selected backbone hypothesis. Each word in the CN is assigned a confidence score and the decoder simply finds the path with the highest sum of these scores.

In addition to word-level combination approaches, such as CN decoding, some phrase-level combination techniques have also recently been presented; their goal is to retain coherence and consistency between the words in a phrase. One successful approach augments a CN (linear word lattice) to a nonlinear phrase lattice which allows several target words to connect with several other target words, i.e., phrase-to-phrase mappings or paraphrases, and then decode over the phrase lattice, searching for the best path (Feng et al 2009; Du and Way 2010). Feng et al (2009) designed heuristic rules to extract paraphrases from word alignments between the backbone and the set of hypotheses. The paraphrases are allowed to be discontinuous but are required to be "minimum" alignment units unless they are generated by adding null words. The lattice was then constructed by adding aligned sentence pairs incrementally. In (Du and Way 2010), TER-Plus (TERp) was employed to carry out the word alignment between the backbone and other hypotheses; a lattice is built by extracting paraphrases based on certain alignment types that TERp indicated, i.e, "stem match", "synonym match" and paraphrases.

In contrast to the above state-of-the-art lattice decoding techniques, in this paper we propose a novel lattice-based MT combination methodology that we call *Target-to-Target Decoding* (TTD). The combination process is carried out as a "translation" from backbone (the first target) to the combination result (the second target). The lattice

is represented in the form of phrase table, composed of monolingual phrase pairs (paraphrases). In other words, the decoding object is no longer the lattice, but the backbone. The combination process can be also interpreted as post-editing the backbone using paraphrases.

Using this perspective of lattice-based MT combination motivates the application of various existing phrase-based MT techniques in the combination framework. For example, bilingual phrase extraction rules (Koehn et al, 2003), which are widely used in MT, can directly map to a target-to-target version for our paraphrase extraction. The simple but efficient rules avoid the complexity of (Feng et al 2009)'s heuristic alignment-unit rules. Moreover, to extract paraphrases that are more than one word, (Feng et al 2009) and (Du and Way 2010)'s rules rely only on many-to-many word alignments that their monolingual word aligners provided, while our rules are capable of utilizing not only many-to-many but also one-to-one monolingual word alignments to form multi-word paraphrases, and this enables us to extract many more paraphrases than (Feng et al 2009) and (Du and Way 2010). For the same reason, even though our implementation uses the Translation Error Rate Plus (TERp) tool as the word aligner, TTD actually can be applied to any kind of monolingual word aligner, including a pure one-to-one word aligner, such as Translation Error Rate (TER). Other benefits of TTD include the fact that the phrase-table based lattice avoids the complexity of lattice construction in (Feng et al 2009), and decoding over the backbone enables us to integrate a reordering model into our combination model directly.

We also adapt the basic translation model in MT to develop our combination model. The confidence score of a hypothesis in our combination model is formulated as a log-linear model including paraphrase confidence scores, lexical weighting, syntactic indicators of whether paraphrases are syntactic constituents, word and phrase penalty, a reordering model, a general language model (LM), and system-specific LMs for employing N-gram consensus information.

Many of these features are unique to our approach. The impact of each major feature is presented in our experiments using the dataset of NIST 2008.

The experiment results show that the overall performance of our TTD-based combination model significantly outperforms the best single MT system of the NIST 2008 participants. We also compare our TTD model with state-of-the-art MT combination models; when following the same requirements of the German-English MT system combination competition held by WMT2011, our model ranks in the top two out of ten MT combination systems.

2 System Overview

The conditions under which we carry out combination are: only the top1 hypotheses of each system are provided, and the MT systems themselves and source sentences are blind to us. Based on these conditions, our combination system involves the following steps:

1. Collect the hypotheses from multiple MT systems.

2. Use a syntactic parser to parse all collected MT system hypotheses. This step is merely to enable determing whether our extracted paraphrases are constituents in step 6.

3. Select the backbone sentence hypothesis. The common strategy is through Minimum Bayes Risk (MBR) decoding (Sim et al., 2007; Rosti et al., 2007a; Feng et al 2009) or system-weighted MBR (Du and Way 2010). These approaches basically only rely on the agreement of system hypotheses. In order to utilize other information, such as a LM, we view the backbone selection as a sentence-based MT combination framework and design the following log-linear model:

$$\log p(E_i) = \sum_{s=1}^{Ns}\left(\lambda_s * \log(1 - TER(E_i, E_s))\right) + \lambda^l * \log(LM(E_i)) + \lambda^w * Length(E_i)$$

Where E is system hypothesis, N_s is system number, λ_s is system weight, λ^l is LM weight and λ^w is word penalty.

4. Get the word alignments between the backbone and all system hypotheses. TTD actually can be applied to any kind of monolingual word aligner. In our implementation, we adopt TERp, one of the state-of-the-art alignment tools, to serve this purpose, described in section 3.

5. Given the word alignments between the backbone and all system hypotheses, we extract paraphrases as phrase table entries, described in section 4.

6. Assign each entry in the phrase table a paraphrase confidence score, lexical weighting and syntactic indicator of whether paraphrases are constituents as described in section 5.1-5.3.

7. In addition to the above confidence estimations for paraphrases, the confidence score of a hypothesis in our model also includes a general LM, and system-specific LMs for determining N-gram consensus across MT systems. The system-specific LM is trained on all hypotheses in the tuning or testing dataset for every MT system. The details are described in section 5.4.

8. Word index the backbone and decode over the modified backbone using the above monolingual phrase table and LMs as described in section 6.

3 Monolingual Word Alignment

Our paraphrases are deduced from monolingual word alignment. Any monolingual word aligner can serve the purpose. Since in our implementation, we adopt TERp as our alignment tool, we briefly review it and use a virtual example to illustrate its alignment output format and how we slightly adjust the format to meet our needs.

TERp (Snover et al. 2009) is an extension of TER (Snover et al. 2006). Both TERp and TER are automatic evaluation metrics for MT, based on measuring the ratio of the number of edit operations between the reference sentence and the MT system hypothesis. TERp uses all the edit operations of TER—Matches, Insertions, Deletions, Substitutions and Shifts—as well as three new edit operations: Stem Matches, Synonym Matches and Paraphrases. TERp identifies the Stem Matches and Synonym Matches using the Porter stemming algorithm (Porter, 1980) and WordNet (Fellbaum, 1998) respectively. Sequences of words in the reference are considered to be paraphrases of a sequence of words in the hypothesis if that phrase pair occurs in the TERp's own paraphrase database.

One valuable characteristic of TERp is that it can produce very high-quality alignments between two given input sentences and identify the alignment types including M (Exact Match), I (Insertion), D (Deletion), S (Substitution), T (Stem Match), Y (Synonym Match) and P (Paraphrase). While P is phrase alignment, all other types are word alignment. A virtual instance is given in the following to illustrate the tool: assume we have a backbone E_b and a system hypothesis E_h as follows:

E_b : w_1 w_2 w_3 w_4 w_5 w_6 w_7 w_8 w_9 w_{10} w_{11}

E_h : $\overline{w}_1$ $\overline{w}_2$ $\overline{w}_3$ $\overline{w}_4$ $\overline{w}_5$ $\overline{w}_6$ $\overline{w}_7$ $\overline{w}_8$ $\overline{w}_9$ $\overline{w}_{10}$

Fig 1. A backbone E_b and a system hypothesis E_h

Where each w_i means a word w in position i in the sentence.

Given the sentence pair as input for the TERp tool, the alignment between E_b and E_h could be produced as follows:

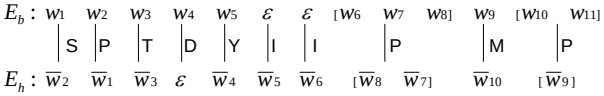

Fig 2. The alignment between E_b and reordered E_h

Note that in this alignment produced by TERp, E_b's word order remains the same but E_h's word order is changed to fit the most reasonable alignment. To extract paraphrases using our extraction rules, we re-order it back to the original word order and keep the alignment links and types. And in order to generate a pure word alignment, for each P, we link every word of E_b to every word of E_h. The adjusted format is as follows:

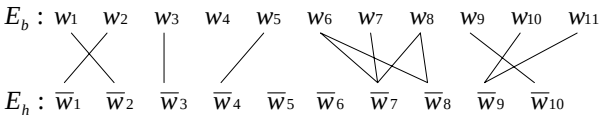

Fig 3. The alignment between E_b and E_h with the original word order

4 Paraphrase Extraction

4.1 Motivation

Before introducing our paraphrase extraction strategy, its motivation is worth mentioning: if we compare the phrase-level combination model with a phrase-based translation model, we see their motivations are quite similar. In translation, it is very common for several words in a foreign language to translate as a whole to several words in the target language. Similarly, in combination of a pair of different translation hypotheses, sometimes

several words substituted as a whole with several other words. For example, “is sick of” and “is disgusted with” basically carry the same meaning and have similar usages. Using the word as the unit to perform combination would face the risk of producing incorrect translations, such as “is sick with” or “is disgusted of”.

Since translation and combination share a similar motivation for using phrases, it is natural for us to apply a similar phrase extraction strategy in our combination framework.

4.2 Strategy

We map the standard bilingual phrase extraction rules (Koehn et al, 2003) to the following target-to-target version for our paraphrase extraction: we extract all phrases that are word-continuous and consistent with the monolingual word alignment. This means that words in a legal paraphrase are not aligned to words outside of the paraphrase, and should include at least one pair of words aligned with each other. The definition of consistency can be formally stated as follows: assume e is a phrase of a backbone and $\bar{e}$ is a phrase of a MT system hypothesis. A pair of phrases (e , $\bar{e}$) is consistent with the monolingual word alignment matrix A if

$$\forall w_i \in e : (w_i, x) \in A \Rightarrow x \in \bar{e}$$
$$\text{and } \forall \bar{w}_j \in \bar{e} : (y, \bar{w}_j) \in A \Rightarrow y \in e$$
$$\text{and } \exists\, w_i \in e, \bar{w}_j \in \bar{e} : (w_i, \bar{w}_j) \in A$$

where w_i is a word of e, $\bar{w}_j$ is a word of $\bar{e}$.

Take the monolingual word alignments in Fig 3 as an example, under the setting of maximum phrase length of 3, the following paraphrases are produced (to save space, only paraphrases starting from w1, w2, w3 and w9 are listed here):

(w_1 , $\bar{w}_2$)	(w_3 , $\bar{w}_3$)
(w_1 w_2 , $\bar{w}_1$ $\bar{w}_2$)	(w_3 w_4 , $\bar{w}_3$)
(w_1 w_2 w_3 , $\bar{w}_1$ $\bar{w}_2$ $\bar{w}_3$)	(w_3 w_4 w_5 , $\bar{w}_3$ $\bar{w}_4$)
	(w_3 w_4 w_5 , $\bar{w}_3$ $\bar{w}_4$ $\bar{w}_5$)
(w_2 , $\bar{w}_1$)	(w_9 , $\bar{w}_{10}$)
	(w_9 w_{10} w_{11} , $\bar{w}_9$ $\bar{w}_{10}$)

Fig 4. Paraphrases starting from w1, w2, w3 and w9, with the maximum phrase length of 3

Because of the need for paraphrase confidence score calculation and decoding, for a paraphrase (e , $\bar{e}$), we make word position information attach to e, while it is not necessary to do so with $\bar{e}$. This results in pairs (is_20 disgusted_21 with_22, is disgusted with) and (is_20 disgusted_21 with_22, is sick of), where 20-22 are the word positions in the backbone. To make the paper more concise, the information of word positions for a paraphrase is not shown in the remainder of the paper unless necessary.

5 Combination Model

We imitate the basic translation model in MT to develop our combination model. The confidence score of a hypothesis in our combination model is formulated as a log-linear model as follows:

$$\begin{aligned}
&\log p(\bar{E} \mid E) = \\
&\sum_{i=1}^{I}\left(\sum_{s=1}^{Ns}\left(\lambda_s^{pc} * pc_s(\bar{e}_i \mid e_i) + \lambda_s^{lex} * lex_s(\bar{e}_i \mid e_i) + \lambda_s^{syn} * syn_s(\bar{e}_i \mid e_i)\right)\right) \\
&+ \sum_{s=1}^{Ns}\left(\lambda_s^{sl} * \log(LM_s(\bar{E}))\right) \\
&+ \sum_{i=1}^{I}\left(\lambda^{d} * d(start_i, end_{i-1})\right) \\
&+ \lambda^{l} * \log(LM(\bar{E})) \\
&+ \lambda^{w} * length(\bar{E}) \\
&+ \lambda^{p} * I
\end{aligned}$$

Where E is the backbone, $\bar{E}$ is the combination output, e_i is a phrase of E, $\bar{e}_i$ is a phrase of $\bar{E}$, I is phrase number and N_s is system number.

The first component of the model is unique to our approach and includes three different estimations for paraphrase confidences of a certain system s: 1. TERp-based paraphrase confidence score (pc_s), 2. Overlap-based lexical weighting (lex_s) and 3. A binary variable (syn_s) indicating whether the paraphrase is a syntactic constituent or not for the system s. They are weighed by λ_s^{pc} , λ_s^{lex} and λ_s^{syn} respectively.

The second component- LM_s , system-specific language model, is also unique to our approach. It is used to determine N-gram consensus across MT systems, and is weighted by λ_s^{sl} .

The third component- d is a reordering model based on distortion cost, weighted by λ^d . The fourth component- LM is a general language

model, weighted by λ^l. The fifth component- λ^w is word penalty, which controls the preference of hypothesis length. And the sixth component- λ^p is phrase penalty, which controls the preference of phrase length.

In this combination model, all weights as well as word and phrase penalty can be trained discriminatively for Bleu score using Minimum Error Rate Training (MERT) procedure (Och 2004).

5.1 TERp-based Paraphrase Confidence Score

As in (Feng et al 2009) and (Du and Way 2010), each paraphrase has different confidences derived from different MT systems. Estimating paraphrase confidence score is basically equal to asking each MT system how confident they feel about the paraphrase. If the paraphrase can be extracted for a MT system, then the system gives a high confidence about the paraphrase; otherwise, zero confidence is given. Basically a binary indicator function is able to serve the purpose no matter what aligner is used.

Because TERp not only provides alignments but also alignment types, we are actually able to know all word alignment types that a paraphrase contains. From our observations, we find M (Exact Match), T (Stem Match), Y (Synonym Match) and P (Paraphrase) can usually be more trusted than I (Insertion), D (Deletion) and S (Substitution) from the perspective of alignment accuracy. To utilize this information, we design the following novel paraphrase confidence score function based on TERp for a certain system s:

$$pc_s(\bar{e} \mid e) = \begin{cases} \dfrac{\text{MTYP\# of } (e, \bar{e})}{\text{MTYP\# of } (e, \bar{e}) + \text{IDS\# of } (e, \bar{e})} & \text{if } (e, \bar{e}) \text{ can be extracted in s} \\ 0 & \text{otherwises} \end{cases}$$

Where MTYP# is the number of word alignment of M, T, Y and P while IDS# is the number of word alignment of I, D and S.

5.2 Overlap-based Lexical Weighting

In MT, infrequent phrase pairs may make it difficult to estimate their translation probabilities and thus, a smoothing algorithm, such as lexical weighting, is often used. The same problem also occurs in the lattice-based combination framework while estimating the paraphrase confidence score. This problem has not been handled in previous related work. So in our combination model, we borrow the idea of lexical weighting from MT (Koehn et. al. 2003) and propose our overlap-based lexical weighting model of a paraphrase for a given system s as follows:

$$lex_s(\bar{e} \mid e) = \frac{\text{Common } \textit{Word\# of } \bar{e} \text{ and } \mathrm{A}_s(e)}{|\bar{e}| + |\mathrm{A}_s(e)|}$$

$\mathrm{A}_s(e_i)$ is the collection of words to which all words of e_i align for the system s. |*| denotes the word number of *. The following example is given to illustrate the formula: assume one entry (e , $\bar{e}$) in the phrase table is "(is disgusted with, is sick of)", and the collection of words to which "is disgusted with" aligns is "is tired of" for the system s. Because "is tired of" is not "is sick of", the paraphrase confidence score would be given 0 by the system s. However, the lexical weighting of the system s is set to 2/6, in which 2 comes from the fact that "is" and "of" are two common words between "(is sick of)" and "(is tired of)", and 6 comes from the sum of the word number of "(is sick of)" and "(is tired of)".

5.3 Syntactic Indicator

In addition to the paraphrase confidence score and lexical weighting, we also investigate the impact of considering syntactic paraphrases in the phrase confidence estimation. If we extract paraphrases using standard bilingual phrase extraction rules, this would include many non-intuitive paraphrases, just as happens in MT. To investigate its impact, Koehn et. al. (2003) weighted syntactic phrases in the phrase table used in their MT experiments, and found that the consideration of syntactic phrases does not bring benefits. In our TTD model, we adopt Koehn et. al. (2003)'s steps and use one binary feature in our log-linear model to represent if a paraphrase is syntactic constituents or not. It is shown as follows:

$$syn_s(\bar{e}_i \mid e_i) = \begin{cases} 1 & \text{if } (e, \bar{e}) \text{ can be extracted in system s and} \\ & e \text{ and } \bar{e} \text{ are both syntactic constituents} \\ 0 & \text{otherwise} \end{cases}$$

Because all MT system hypotheses have been parsed by a syntactic parser before, here we can just check if a phrase is a constituent by looking up the parses.

5.4 System-Specific Language Model

Both the paraphrase confidence score and lexical weighting described above are based on the estimation of the degree of agreement between a phrase with another phrase. Now our question is: how can we estimate and utilize the agreement degree between a set of consecutive phrases with another set of consecutive phrases during decoding? In lattice-based combination, this issue has not been addressed before.

Our solution is simple; –we consider N-gram consensus in addition to the confidence estimations for paraphrases during decoding. This idea of considering N-gram consensus was widely used in N-best list reranking (Chen et al., 2005; Zens and Ney, 2006; Chen et al., 2007). These years the technique has also been presented in some word-based combination schemes and proven effectives. The approaches can be divided into two categories: one is based on a sentence-specific LM, built on translation hypotheses of multiple systems (Zhao and He 2009; Heafield and Lavie 2010); the other one is based on a corpus-based LM, built on the whole tuning/test corpus of all translation hypotheses of multiple systems (Matusov et al, 2008; Leusch et al, 2011). The strength of sentence-specific LM is that it considers the most specific data available while the corpus-based LM has the advantage of gathering more data with which to compare, including document-level matches.

Inspired by these ideas, we propose a system-specific LM, which is a modified corpus-based LM, built on the whole tuning/test corpus of all translation hypotheses of each single system so that each system-specific LM can have its own weight. Through these LMs, system-weighted N-gram consensus is considered during decoding.

6 Decoding

TTD is carried out as a "translation" from the backbone to the combination result. The words in the backbone are not necessarily unique in the entire sentence, so before decoding, they need to be indexed using word positions.

Any standard decoder can be used to decode the format [1]. Take a toy example to illustrate the decoding process as follows. Given an indexed backbone:

… He_19 is_20 disgusted_21 with_22 that_23 …

Assume there are only four entries in our phrase table:

(He_19, He)
(is_20 disgusted_21 with_22, is disgusted with)
(is_20 disgusted_21 with_22, is sick of)
(that_23, that)

Then one of the following hypotheses would be generated by the decoding:

… He is disgusted with that …
… He is sick of that …

7 Experiments

7.1 Settings

The experiments are conducted and reported on two public datasets: One is Chinese-English selected reference and system translations of NIST 2008 (LDC2010T01). The other one is German-English combination shared task held by the WMT in 2011[2]. The two datasets are abbreviated by "CE-NIST" and "GE-WMT" respectively in the reminder of the paper. Because both the datasets consist of human reference translations and corresponding machine translations, we directly use their machine translations for our combination experiments so that our combination system and others can compare with each other by following the same settings. For this reason, we will describe our settings using the same terms used in the two datasets and provide more details for future comparison.

7.1.1 CE-NIST

"CE-NIST" consists of four human reference translations and corresponding machine translations for the NIST Open MT08 test sets, which consist of newswire and web data. The test set contains 105 documents with 1312 sentences and output from 23 machine translation systems. Each system provides the top one translation hypothesis for every sentence. We further divide NIST Open MT08 test set into the tuning set and test set for our experiment. We divide the data in

[1] In our implementation, we use MOSES (http://www.statmt.org/moses/)

[2] http://www.statmt.org/wmt11/system-combination-task.html

this way in order to enable other researchers to compare their approaches with ours in the future: the documents of "AFP", "CNS", "GMW" and "cmn-NG" (the first three are newswire, and the fourth is web data) are collected as tuning set, which includes 524 sentences, and the documents of others are collected as testing set, which includes 788 sentences. Out of 23 MT systems, we manually select Top5 constrained-trained MT systems as our MT systems for our combination experiment. Table 1 lists these systems with their performances on the testset and the performance of our sentence-based combination (perform out of the Top5) results (backbone performance).

	BLEU	TERp	METEOR
System 03	**30.16**	**63.04**	**51.94**
System 15	30.06	62.82	51.80
System 20	28.15	65.39	49.72
System 22	29.94	63.19	51.51
System 31	29.52	61.70	51.89
backbone	**30.89**	**61.28**	**52.65**

Table 1. Performance of best 5 MT system in CE-NIST and backbone

We view system 03 as the best system based on BLEU. Backbone is better than any system.

7.1.2 GE-WMT

In order to compare our approach with the other state-of-the-art combination techniques, we also carry out our experiment on "GE-WMT", in which tuning and testing data of MT system outputs are provided and especially, the outputs of 10 participants in this combination shared task are also provided. So we can compare our system with them by following the same constraints that this shared task specifies.

We decide to work on German-English language pair because this language pair is relatively challenging for MT among the language pairs in the shared task of combination.

In "GE-WMT", one human reference translation and the corresponding machine translations from 26 machine translation systems are provided. It contains 1003 sentences for combination tuning and 2000 sentences for combination testing. Each system provides the top one translation hypothesis for every sentence. Out of 26 MT systems, we manually select the Top 6 MT systems as our MT systems for our combination experiment. The performances of the best 6 systems for the testing set and the performance of our sentence-based combination (perform out of the Top6) results (backbone performance) are listed in the table 2.

	BLEU	TERp	METEOR
cmu-dyer	22.72	60.89	55.09
dfki-xu	22.44	62.31	53.89
kit	22.75	60.82	54.81
online-A	23.16	58.96	56.34
online-B	**24.27**	**57.89**	**56.93**
rwth-fre-c	21.86	62.82	53.46
backbone	**25.38**	**57.05**	**57.72**

Table 2. Performance of best 6 MT system in GE-WMT and backbone

We view system online-B as the best system based on BLEU. Backbone is better than any system.

7.2 Result and Analysis

Three metrics are used for evaluation: BLEU[3], TERp[4] and METEOR[5].

We first use "CE-NIST" to investigate the impact of using "word" and "phrase" in TTD (denoted by "W" and "P". in the "W" setting, the maximum length of each phrase is one while in the "P" setting, the maximum length of phrase is seven.). We also study the impacts of five major features including paraphrase confidence score (cs), lexical weighting (lex), syntactic indicator (syn), system-specific LM (sl) and reordering model (r). Other features other than the five are all used by default in our experiments.

	BLEU	TERp	METEOR
System 03	30.16	63.04	51.94
backbone	30.89	61.28	52.65
W+cs	30.98	60.98	52.90
W+cs+sl	31.29	61.36	52.70
P+cs	31.74	60.11	53.59
P+cs+sl	32.63	60.49	53.53
P+cs+lex	31.81	60.32	53.53
P+cs+syn	31.74	60.22	53.55
P+cs+sl+lex+syn	**32.85**	**60.32**	**53.76**

Table 3. Performance of combination without using reordering model

[3] mteval-v13a.pl (http://www.itl.nist.gov/iad/mig/tests/mt/2009/)
[4] TERp-adq (http://www.umiacs.umd.edu/~snover/terp/)
[5] METEOR-1.3-adq (http://www.cs.cmu.edu/~alavie/METEOR/)

	BLEU	TERp	METEOR
System 03	30.16	63.04	51.94
backbone	30.89	61.28	52.65
W+r+cs	31.13	60.99	53.01
W+r+cs+sl	31.33	61.72	52.55
P+r+cs	31.80	60.21	53.71
P+r+cs+sl	**32.80**	**60.13**	**53.86**
P+r+cs+lex	31.76	60.12	53.54
P+r+cs+syn	31.72	60.37	53.38
P+r+cs+sl+lex+syn	32.75	60.48	53.63

Table 4. Performance of combination with using reordering model

Form Table 3 and Table 4, we can make the following observations: 1. No matter whether reordering model is used, "phrase" as the unit is better than "word", which proves our basic claim about the advantage of phrase. 2. The fact that "W+cs" and "P+cs" are both better than the system 03 and the backbone shows the effectiveness of "cs". 3. Among "sl", "lex" and "syn", we can find the effectiveness of "sl" is obvious and the other two are not if they are not used with "sl". 4. In general, the impact of reordering is not very great, which means the word order of backbone seems pretty trustable. 5. Among the two tables, we find the two settings- "P+cs+sl+lex+syn" and "P+r+cs+sl" provide the best performance.

We used the two settings to perform combination in GE-WMT in order to compare our approach with the other state-of-the-art combination techniques. The results are shown in table 5.

	BLEU	TERp	METEOR
Online B	24.27	57.89	56.93
backbone	25.38	57.05	57.72
koc-combo	23.41	61.83	54.08
quaero-combo	23.37	60.86	55.03
rwth-leusch-combo	25.62	57.44	57.20
jhu-combo	25.08	57.81	56.87
jhu-combo-contrastive	24.46	57.20	57.26
bbn-combo	**26.73**	**56.13**	**58.30**
cmu-heafield-combo	25.31	57.27	57.71
cmu-heafield-combo-contrastive	25.24	57.37	57.68
upv-prhlt-combo	24.65	59.25	56.24
uzh-combo	24.55	58.47	56.76
P+r+cs+sl	**25.81**	**56.89**	**57.88**
P+cs+sl+lex+syn	**25.96**	**57.18**	**57.64**

Table 5. Performance comparison of our two settings with other 10 state-of-the-art combination systems

From table 5, we see that our two settings are both in the top two (only worse than "bbn-combo"), proving the effectiveness of our approach and showing TTD is a promising combination framework.

8 Related Work

In addition to lattice-based combination models, another technique used in phrase-level combination, which is quite different from our approach, is re-translation to combine MT outputs: by constructing a new phrase translation table from each system's source-to-target phrase alignments, one can re-decode the source sentence using this new translation table (Rosti et al., 2007b; Huang and Papineni, 2007; Chen et al., 2007; Chen et al., 2009). One challenge for this kind of approach is that the translated word order would rely entirely on the quality of the reordering model of the re-decoder. To solve the problem, Huang and Papineni (2007) collect the information of word orders from all system output paths; the re-decoder then references the information to help decide the final word order. In contrast, in lattice-based frameworks, such as CN, phrase-level lattice decoding or TTD, word order is not a problem because the decoding process follows the word order of the backbone or other system hypotheses. TTD is particularly able to allow more flexibility in choice of word order through integration of a reordering model.

There is another class of phrase-level combination techniques called "confusion forest" proposed by Watanabe and Sumita (2011), in which hypotheses are encoded as a packed forest representing alternative trees. The forest is generated using syntactic consensus among parsed hypotheses and the new hypothesis is produced by searching the best derivation in the forest. Compared with a "confusion forest", our syntactic indicator of whether paraphrases are constituents is relatively simpler and only used for assisting the estimation of the paraphrase confidence score.

Other than phrase-level combination techniques, paraphrases are also used in sentence-level MT combination framework. Ma and McKeown (2011) use paraphrases in their text-to-text generation process. Given source-to-target word alignments provided by MT systems, they applied standard bilingual phrase extraction rules with syntactic

constraints to form syntactic paraphrases. These syntactic paraphrases are used to randomly generate as many as a threshold number of fused hypothesis candidates for the final hypothesis selection based on LM, hypothesis agreement and grammaticality.

9 Conclusion

Our approach to phrase-level system combination features Target-to-Target Decoding and multiple confidence estimations inspired from machine translation. We apply various existing phrase-based MT techniques in our approach, including phrase extraction rules and lexical weighting. We also propose some new confidence estimations such as TERp-based paraphrase confidence scores and system-specific LMs. Our results show that this approach yields a strong improvement over the best single MT system and can compete with other state-of-the-art combination systems.

There are several research directions for our future work. They involve the investigation of the impact of selection strategies for the backbone or even multiple backbones (Rosti et al., 2007a; Matusov et al, 2008), the effectiveness of incremental alignment (Rosti et al., 2008) applied to TTD, and the development of other syntactic confidence estimations.

Acknowledgments

We would like to thank the anonymous reviewers for their helpful comments. This work is supported by the National Science Foundation via Grant No. 0910778 entitled “Richer Representations for Machine Translation”. All views expressed in this paper are those of the authors and do not necessarily represent the view of the National Science Foundation.

References

Yu Chen, Andreas Eisele, Christian Federmann, Eva Hasler, Michael Jellinghaus, and Silke Theison. 2007. Multi-engine machine translation with an open-source SMT decoder. In Proceedings of WMT07

Yu Chen, Michael Jellinghaus, Andreas Eisele, Yi Zhang, Sabine Hunsicker, Silke Theison, Christian Federmann, Hans Uszkoreit. 2009. Combining Multi-Engine Translations with Moses. In Proceedings of the Fourth Workshop on Statistical Machine Translation

Boxing Chen, M. Federico and M. Cettolo. 2007. Better N-best Translations through Generative n-gram Language Models. In Proceeding of MT Summit XI

Boxing Chen, Roldano Cattoni, Nicola Bertoldi, Mauro Cettolo, and Marcello Federico. 2005. The ITC-irst SMT System for IWSLT-2005. In Proceedings of IWSLT

Jinhua Du and Andy Way. 2010. Using TERp to Augment the System Combination for SMT. In Proceedings of the Ninth Conference of the Association for Machine Translation (AMTA2010)

Christiane Fellbaum. 1998. WordNet: An Electronic Lexical Database. MIT Press. http://wordnet.princeton.edu/

Yang Feng, Yang Liu, Haitao Mi, Qun Liu, and Yajuan Lu. 2009 Lattice-based System Combination for Statistical Machine Translation. In Proceedings of ACL

Xiaodong He, Mei Yang, Jianfeng Gao, Patrick Nguyen, and Robert Moore. 2008. Indirect-hmm-based hypothesis alignment for computing outputs from machine translation systems. In Proceedings of EMNLP

Kenneth Heafield and Alon Lavie. 2010. Voting on N-grams for Machine Translation System Combination. In Proceedings of Ninth Conference of the Association for Machine Translation in the Americas

Fei Huang and Kishore Papineni. 2007. Hierarchical System Combination for Machine Translation. In Proceedings of EMNLP-CoNLL

Damianos Karakos, Jason Eisner, Sanjeev Khudanpur, and Markus Dreyer. 2008. Machine translation system combination using ITG-based alignments. In Proceedings of ACL-HLT

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical Phrase-Based Translation. In Proceedings of Human Language Technology Conference and Meeting of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)

Wei-Yun Ma and Kathleen McKeown. 2011. System Combination for Machine Translation Based on Text-to-Text Generation. In Proceedings of Machine Translation Summit XIII

Gregor Leusch, Markus Freitag, and Hermann Ney. The RWTH System Combination System for WMT 2011. 2011. In Proceedings of the Sixth Workshop on Statistical Machine Translation

Evgeny Matusov, Nicola Ueffing, and Hermann Ney 2006. Computing consensus translation from multiple machine translation systems using enhanced hypotheses alignment. In Proceedings of EACL

E. Matusov, G. Leusch, R. E. Banchs, N. Bertoldi, D. Dechelotte, M. Federico, M. Kolss, Y. S. Lee, J. B. Marino, M. Paulik, S. Roukos, H. Schwenk, and H. Ney. 2008. System combination for machine translation of spoken and written language. IEEE Transactions on Audio, Speech and Language Processing, 16(7):1222–1237, September.

Sushant Narsale JHU System Combination Scheme for WMT 2010. 2010. In Proceedings of the Fifth Workshop on Statistical Machine Translation

Franz Josef Och. 2004. Minimum Error Rate Training in Statistical Machine Translation. In Proceedings of ACL

Martin F. Porter. 1980. An algorithm for suffix stripping. Program, 14(3):130–137.

Antti-Veikko I. Rosti, Spyros Matsoukas, and Richard Schwartz. 2007a. Improved word-level system combination for machine translation. In Proceedings of ACL

Antti-Veikko I. Rosti, Necip F. Ayan, Bing Xiang, Spyros Matsoukas, Richard Schwartz, and Bonnie J. Dorr. 2007b. Combining outputs from multiple machine translation systems. In Proceedings of NAACL-HLT

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

K.C. Sim, W.J. Byrne, M.J.F. Gales, H. Sahbi and P.C. Woodland .2007. Consensus Network Decoding for Statistical Machine Translation System Combination. In Proceedings of ICASSP.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In Proceedings of Association for Machine Translation in the Americas

Matthew Snover, Nitin Madnani, Bonnie Dorr, and Richard Schwartz, 2009. Fluency, Adequacy, or HTER? Exploring Different Human Judgments with a Tunable MT Metric. In Proceedings of the Fourth Workshop on Statistical Machine Translation

Taro Watanabe, Eiichiro Sumita. 2011. Machine Translation System Combination by Confusion Forest. In Proceedings of ACL

Daguang Xu, Yuan Cao, Damianos Karakos. 2011. Description of the JHU System Combination Scheme for WMT 2011. In Proceedings of the Sixth Workshop on Statistical Machine Translation

Yong Zhao and Xiaodong He. 2009. Using n-gram based features for machine translation system combination. In Proceedings of the North American Chapter of the Association for Computational Linguistics

Richard Zens and Hermann Ney. 2006. N-Gram Posterior Probabilities for Statistical Machine Translation. In Proceedings of the NAACL Workshop on SMT