A Framework for Machine Translation Output Combination

Yi-Chang Chen
Department of Computer Science and Engineering
National Sun Yat-Sen University
m963040046@student.nsysu.edu.tw
Chia-Ping Chen
Department of Computer Science and Engineering
National Sun Yat-Sen University
cpchen@cse.nsysu.edu.tw

摘要

本研究提供一個線上機器翻譯整合系統整合三個不同的線上翻譯引擎。該翻譯整合系統，利用了選擇、替換、插入及刪除等模組，針對線上翻譯假說進行修正。我們實際整合了GOOGLE、YAHOO、譯言堂的翻譯假說。在IWSLT07的測試語料進行中文至英文的翻譯整合。由實驗結果得知，該翻譯整合系統其 BLEU 分數由所整合的最佳翻譯系統的 19.15 進步到 20.55 。該翻譯整合系統相較於所整合的最佳線上翻譯系統進步了 1.4 BLEU 。

Abstract

In this paper, we propose a framework for combining outputs from multiple on-line machine translation systems. This framework consists of several modules, including selection, substitution, insertion, and deletion. We evaluate the combination framework on IWSLT07 in travel domain, for the translation direction from Chinese to English. Three different on-line machine translation systems, Google, Yahoo, and TransWhiz, are used in the investigation. The experimental results show that our proposed combination framework improves BLEU score from 19.15 to 20.55. It achieves an absolute improvement of 1.4 in the BLEU score.

Keyword: Machine translation, System combination

1 Introduction

The on-line machine translation is one application that is becoming popular nowadays. As each on-line translation system has its own strength and weakness, it is reasonable to expect that a framework capable of combining multiple on-line machine translation outputs may have the potential to produce translation results of better quality than the single-system outputs. In fact, this proposition has been shown to be true in certain published works, e.g., [1, 2].

In this paper, we propose such a combination framework. The system is essentially sequential with the following basic components. First, one of the output sentence is selected as the raw best hypothesis. This raw best hypothesis is subjected to further post-processing modules

of substitution, insertion and deletion, based on the information provided by the unselected hypotheses and the source sentence.

This paper is organized as follows. In Section 2, we review related works of system combination for machine translation. In Section 3, we describe our proposed method for this problem. In Section 4, we present our experimental results. In Section 5, we draw conclusions.

2 Review

Our review of machine translation system combination is divided into three different categories: the sentence-level combination, the phrase-level combination, and the word-level combination.

2.1 Sentence-Level Combination

The sentence-level combination simply chooses one of the hypotheses as the combination output. That is, suppose the outputs from systems 1 through N are $H_1, \ldots, H_N$,

$$H^* = \arg \max_{H \in \{H_1, \ldots, H_N\}} S(H), \tag{1}$$

where $S(H)$ is a (re-)scoring function for hypothesis H. The design of the scoring function is the core problem in a sentence-level combination system. For example, different features with weights trained by the minimum error rate training (MERT) [3] can be used [4].

Note H^* is chosen *as is* without further processing. This approach renders the search space very limited. Such deficiency does need to be compensated by a rather sophisticated re-scoring mechanism for good performance. Still, that may not be enough, when the best hypothesis appears to be a *mixed-and-matched* solution.

2.2 Phrase-Level Combination

In the tribe of phrase-level combination, the phrase-level alignments are aggregated. The combined phrase translation table is used to re-decode the source sentence, generating a new hypothesis [2]. A phrase-based machine translation system, such as one based on GIZA++ [5], can be employed to generate phrase-level alignments. Potentially, the phrase-level combination can produce a final output sentence which is better than any of the input sentences.

2.3 Word-Level Combination

In the word-level combination approach, the candidate word for each word position is considered one by one. A consensus network [1] [6] can be constructed. As shown in Figure 1, the counts of word appearance in a given position based on the optimal word-alignment is used as the edge weights. For each section, the word with the maximum weight is then chosen, constituting the final hypothesis[1]. This idea actually comes from the automatic speech recognition [7].

To generate a consensus network, a skeleton (seed) has to be chosen as the reference for the optimal alignment. In [8], using the output of the consensus network as skeleton and re-aligning all hypothesis leads to a better accuracy.

[1]Note the edge weights may be fine-tuned to reflect the scores of each system.

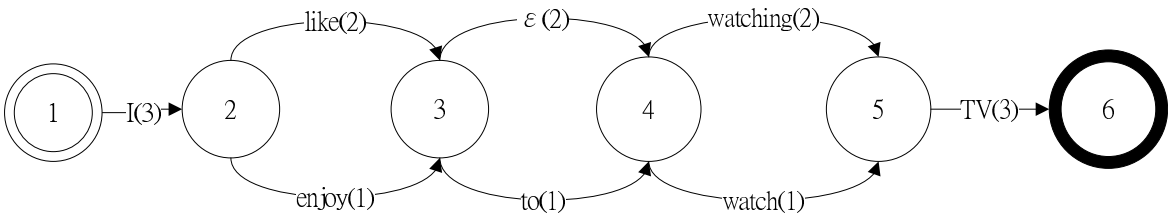

Figure 1: The consensus network of combining “I like watching TV”, “I enjoy watching TV” and “I like to watch TV”.

3 Method

Our proposed system combines three on-line machine translation systems. Let the source sentence be denoted by C. Given C, the target sentences from these systems are denoted by E_G, E_Y, E_{TW}, respectively for *Google*, *Yahoo*, and *TransWhiz*. With E_G, E_Y, E_{TW} and C as input, the combination system performs the following steps.

- **selection**: One of E_G, E_Y, E_{TW} with the highest language-model score is selected. We denote the selected sentence by E, and the unselected hypotheses as F and G.
- **substitution**: Some words in E are considered and may be substituted. The hypothesis after substitution is denoted by E'.
- **insertion**: Each position in E' is considered to insert an extra word. The hypothesis after insertion is denoted by E''.
- **deletion**: Each word in E'' is considered to be deleted. The hypothesis after deletion is denoted by E^*.

E^* is the final output sentence. The overall process is depicted in Figure 2. We next describe the implementation details.

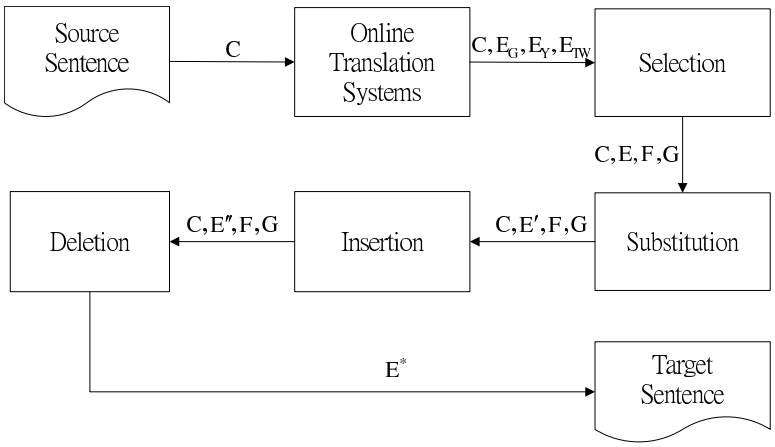

Figure 2: System Organization.

3.1 Selection

The selection is based on a language-model score,

$$E = \arg\max_{H \in \{E_G, E_Y, E_{TW}\}} \frac{1}{|H|} \log p_{5g}(H), \quad (2)$$

where H is the hypothesis, $|H|$ is the length of the hypothesis, and p_{5g} is the 5-gram language model probability. The language model used in the selection module is a 5-gram language model trained from the English side of the IWSLT07 training data. Note that in (2), We use the per-word log probability to avoid the (unfair) preference of short sentences. The per-word log probability is that the langauge model score of H devided by its length.

3.2 Substitution

The substitution of words in E is based on the following idea. If a word w appears in both F and G (the unselected hypotheses) but not in E, it is likely to be better to include w in the output. To safeguard against redundancy, we find a word w' in E to be replaced by w. To make sure that such a replacement is a sound operation, we compare the language model scores before and after the word substitution. A statistical machine translation system using MOSES [9] trained by the IWSLT07 data is used to determine the alignments between source and target sentences. The pseudo code for substitution is given in Algorithm 1, and an example for substitution is given in Example 1.

Algorithm 1 Substitution

Require: C, E, F, G
Ensure: E'

1: extract the set of candidate words for substitution;

$$\mathcal{S} = (\{F\} \cap \{G\}) - \{E\}\ ^{2}$$

2: **for all** $w' \in \mathcal{S}$ **do**
3: **if** find the word $c \in \{C\}$ which is aligned to w' **then**
4: **if** find the word $w \in \{E\}$ which is aligned to c **then**
5: compare the translation-model and bi-gram language-model scores to decide whether to replace w by w';
6: **end if**
7: **end if**
8: **end for**

Example 1 (*Substitution*)

The input is

- C : 我想要送這個特快專遞到日本。
- E : *I want to deliver this special delivery to Japan.*
- F : *I want to send this to fast and particularly pass Japan especially.*
- G : *I'd like to send this Speedpost to Japan.*

[2] We use notation $\{E\}$ to denote the set of words in sentence E.

$$\mathcal{S} = \{send\},\ w' = send,\ c = 送,\ w = deliver.$$

The system checks the translation-model score

$$p_t(send|送) > p_t(deliver|送),\ ^{3}$$

and the language-model score

$$\log\ p_{bg}(send|to) + \log\ p_{bg}(this|send) > \log\ p_{bg}(deliver|to) + \log\ p_{bg}(this|deliver),$$

and decides

- *E′ : I want to* send *this special delivery to Japan.*

The reference is

- *R: I want to send this by special delivery to Japan.*

3.3 Insertion

The insertion of words into E' is based on the following idea. If a word w in E' also appears in F or G, we check the adjacent words of w in F or G for possible insertion. The pseudo code for insertion is given in Algorithm 2. An example for insertion is given in Example 2.

Example 2 (*Insertion*)

The input is

- *C :* 你有地鐵地圖嗎？
- *E′ : Do you have subway map?*
- *F : You have subway map?*
- *G : You have a subway map?*

The set of words $\mathcal{I}$ in this example is

$$\mathcal{I} = \{you, have, subway, map\}.$$

The system checks language-model score

$$\log\ p_{bg}(a|have) + \log\ p_{bg}(subway|a) > 2\log\ p_{bg}(subway|have),$$

and decides

- *E″ : Do you have a subway map?*

The reference is

- *R: Do you have a subway map?*

[3] p_t is the translation probability.

Algorithm 2 Insertion

Require: C, E', F, G
Ensure: E''

extract the set of words;

$$\mathcal{I} = \{E'\} \cap (\{F\} \cup \{G\})$$ [4]

for all $w \in \mathcal{I}$ **do**
  if find the word u immediately before w in F or G **then**
    if the bi-gram language-model scores of inserting u before w in E' is larger than the orginal **then**
      decide inserting u before w in E';
    else
      consider replacing the word before w in E' by u;
    end if
  end if
  if find the word v immediately after w in F or G **then**
    if the bi-gram language-model scores of inserting v after w in E' is larger than the orginal **then**
      decide inserting v after w in E';
    else
      consider replacing the word after w in E' by v;
    end if
  end if
end for

3.4 Deletion

The deletion of words in $\{E''\}$ is based on the following idea. A word $w \in \{E''\}$ is a candidate for deletion if there is no word $c \in \{C\}$ with nonzero translation probability ($p_t(w \mid c)$). To avoid the deletion of the word in phrases, a candidate word w is deleted only when none of the bigrams formed by w and its immediate neighbors appear in the training data. The pseudo code for deletion is given in Algorithm 3. An example for deletion is given in Example 3.

Algorithm 3 Deletion

Require: C, E'', F, G
Ensure: E^*

extract the set of candidate words for deletion;

$$\mathcal{D} = \{w \in \{E''\} \mid t(w \mid c_j) = 0,\ \forall\ j\}$$

for all $w \in \mathcal{D}$ **do**
  if none of the bigrams formed by w and its immediate neighbors in the training data **then**
    w is to be deleted;
  end if
end for

[4] we use the adjacent words of $\mathcal{I}$ as the candidate set for insertion.

Example 3 (*Deletion*)

The input is

- *C :* 那裡有手工藝品商店？
- *E″ : Where is the handicraft article store?*

The set of words $\mathcal{D}$ is

$$\mathcal{D} = \{article\}.$$

The system checks that "handicraft article" and "article store" are neither in the training data and decides

- E^* *: Where is the handicraft store?*

The reference is

- *R: Where is the handicraft store?*

4 Experiments

4.1 Setup

We use IWSLT07 C_E task to run this experiment. IWSLT07 contains tourism-related sentences. The test set consists of 489 Chinese sentences, each of which is accompanied by six reference sentences. Note that the Chinese sentences are word-segmented. The IWSLT07 C_E task we present in the Table 4.1.

Table 1: IWSLT07 C_E task.

	Sentences
Train	39953
Dev	2501
Test	489

We use the on-line machine translation systems of Google[5], Yahoo[6] and TransWhiz[7]. We input the 489 Chinese sentences of the test set to these engines, and get 1, 467 English sentences back.

We use the training data in IWSLT07 task to train our 5-gram language model with SRILM [10]. We use MOSES to train the translation model from the training data in IWSLT07 task.

The BLEU [11] measure with six references per sentence is used in our evaluation. The answers are treated as case-insensitive.

[5]http://translate.google.com.tw/translate_t

[6]http://tw.babelfish.yahoo.com/

[7]http://www.mytrans.com.tw/mytrans/freesent.aspx

4.2 Results

The experimental results are presented in Table 4.2. The progressive improvements can be clearly seen in this table. Systems A to C are the three on-line machine translation systems ordered by their performance in BLEU.

- **selection (sel)**: The selection module leads to an absolute improvement of 0.58 BLEU score. Using the language model to select an output from multiple hypotheses is effective, as the selection module selects the most fluent sentence according the 5-gram language model. We think that the selection module can be further improved by joining other features to select the hypothesis.

- **substitution (sub)**: The substitution module leads to a small absolute improvement of 0.07 BLEU score. In this module, a rare word can be replaced by the common word. The candidate set of substitution is small, so we cannot achieve much improvement in this module. Yet it still fixes certain errors in the output. We think that the substitution module can be further improved by replaceing words not only from other hyptoheses but also from dictionaries.

- **insertion (ins)**: The insertion module leads to an absolute improvement of 0.29 BLEU score. It inserts the articles and the adjectives. Given E already contains most of the correct words, the improvement is somewhat limited. We think that the insertion module can be further improved by joining words from other source. For exmaple, phrase tables, dictionaries, and others.

- **deletion (del)**: The deletion module leads to an absolute improvement of 0.46 BLEU score. It deletes the redundant words, incorrect words, and out of travel domain words in the output. These words are error sources of our combination hypotheses.

The total improvement over the single best system is 1.4 BLEU absolute.

Table 2: Experimental results.

System	BLEU
System A	19.15
System B	12.39
System C	10.51
+sel	19.73
+sel+sub	19.80
+sel+sub+ins	20.09
+sel+sub+ins+del	20.55

5 Conclusion and Further Work

In this paper, we propose a combination framework that combines the outputs of multiple on-line translation systems. It uses selection module, substitution module, insertion module,

and deletion module. We evaluate out method with the IWSLT07 C_E corpus. The experiments show an overall improvement of 1.4 BLEU absolute.

Our proposed framework changes the hypothesis only locally. In the future, we plan to consider long-range information for better performance. Moreover, our system uses unselected hypotheses to decide which is the incorrect word in the selected hypothesis, but sometimes the wrong words are chosen. Therefore, we may work directly on the words in the selected hypothesis, and only use the unselected hypotheses after problematic words are spotted.

References

[1] B. Bangalore, G. Bordel, and G. Riccardi, "Computing consensus translation from multiple machine translation systems," in *IEEE Workshop on Automatic Speech Recognition and Understanding, 2001. ASRU'01*, 2001, pp. 351–354.

[2] A. Rosti, N. Ayan, B. Xiang, S. Matsoukas, R. Schwartz, and B. Dorr, "Combining outputs from multiple machine translation systems," in *Proceedings of NAACL HLT*, 2007, pp. 228–235.

[3] F. Och, "Minimum error rate training in statistical machine translation," in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*. Association for Computational Linguistics Morristown, NJ, USA, 2003, pp. 160–167.

[4] A. Stolcke, "Combination of machine translation systems via hypothesis selection from combined n-best lists," in *Proceedings of the Eighth Conference of the Association for Machine Translation*, 2008, pp. 254–261.

[5] F. J. Och and H. Ney, "A systematic comparison of various statistical alignment models," *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, 2003.

[6] K. Sim, W. Byrne, M. Gales, H. Sahbi, and P. Woodland, "Consensus network decoding for statistical machine translation system combination," in *Proc. ICASSP*, vol. 4, 2007, pp. 105–108.

[7] J. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER)," in *1997 IEEE Workshop on Automatic Speech Recognition and Understanding, 1997. Proceedings.*, 1997, pp. 347–354.

[8] N. Ayan, J. Zheng, and W. Wang, "Improving alignments for better confusion networks for combining machine translation systems," in *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, 2008, pp. 33–40.

[9] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens *et al.*, "Moses: Open source toolkit for statistical machine translation," in *ANNUAL MEETING-ASSOCIATION FOR COMPUTATIONAL LINGUISTICS*, vol. 45, no. 2, 2007, p. 2.

[10] A. Stolcke, "SRILM-an extensible language modeling toolkit," in *Seventh International Conference on Spoken Language Processing*. ISCA, 2002.

[11] K. Papineni, S. Roukos, T. Ward, and W. Zhu, "BLEU: a method for automatic evaluation of machine translation."