The ICT,CAS MT Systems for the IWSLT09 Evaluation

Haitao Mi, Yang Liu, Tian Xia, Yang Feng, Xinyan Xiao, Jun Xie,Zhaopeng Tu,
Hao Xiong, Daqi Zheng, Yajuan Lü and Qun Liu
Key Laboratory of Intelligent Information Processing
Institute of Computing Technology, Chinese Academy of Sciences
{htmi,yliu, xiatian, fengyang, xiaoxinyan, lvyajuan, liuqun}@ict.ac.cn

1. Overview

ICT,CAS participated in three tasks:

1. **BTEC task, Chinese-English direction;** **BTEC_CE**
2. **Challenge task, Chinese-English direction;** **CT_CE**
3. **Challenge task, English-Chinese direction.** **CT_EC**

For each task, we finally submitted a single system who achieved a maximum BLEU score on development set among four different single systems.

2. Single Systems

2.1 Silenus

Silenus (Mi et al., 2008; Mi and Huang, 2008) is a forest-based tree-to-string SMT system. A packed parse forest is a compact representation of all derivations (i.e., parse trees) for a given sentence under a context-free grammar. A tree-to-string rule describes the correspondence between a source parse tree and a target string.

Compared with conventional tree-to-string (Liu et al., 2006; Huang et al., 2006) or string-to-tree models (Galley et al., 2006; Marcu et al., 2006), our model replaces single-best tree with a forest at both rule extraction time and decoding time.

In the rule extraction step, we extract tree-to-string rules form a pair of word-aligned source *parse forest* and target string, shown in figure 1 (a). Compared with the rules extracted from the aligned pair of 1-best tree and string, some extra tree-to-string rules, shown on the right corner, can be extracted with forest approach.

In the decoding step, we first parse the input sentence into a source-*parse forest* (Figure 1(a)) and convert it into a *translation forest* (Figure 1(b))with rule set by pattern-matching. Then the decoder searches for the best derivation on the *translation forest* and outputs the target string. Figure 1 (c) shows the correspondence between translation hyperedges and translation rules.

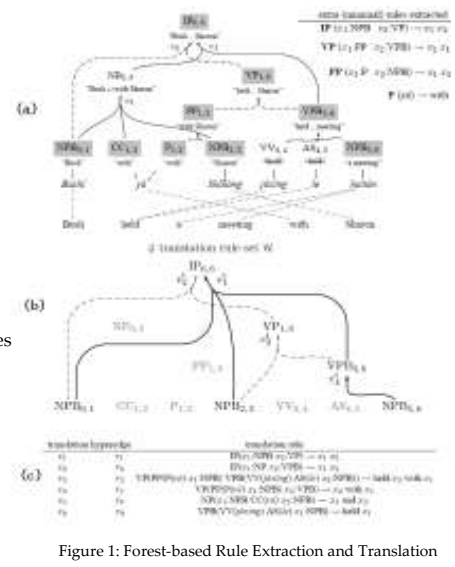

Figure 1: Forest-based Rule Extraction and Translation

2.2 Bruin

Bruin (Xiong et al., 2006) is a formally syntax-based system that implements a maximum entropy based reordering model on BTG rules (Wu 1997). Bruin employs the following three BTG rules to direct translation:

$$A \xrightarrow{\Pi} \langle A^1, A^2 \rangle$$
$$A \xrightarrow{\Box} \langle A^1, A^2 \rangle$$
$$A \rightarrow \langle x, y \rangle$$

The first two rules are used to merge two neighboring blocks into one larger block either in a monotonic or an inverted order. A block is a pair of source and target contiguous sequences of words. The last rule translates a source phrase into a target phrase and generate a block.

Figure 2 gives some blocks. The first block and the second block is connected in a monotonic order. The third and the fourth block is connected in an inverted order.

The reordering problem is a typical two-class classification. So we build a maximum entropy model to predict the merging order of two phrases.

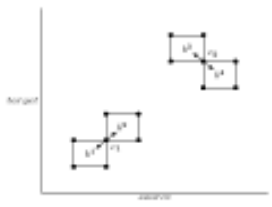

Figure 2: *Blocks and orders. The bold dots are corners and the arrows from the corners are their links.*

2.3 Chiero

Chiero is a re-implementation of the state-of-the-art hierarchical string-to-string translation system (Chiang, 2007). The model can formalized as a synchronous context-free grammar.

2.4 Moses

Moses is a phrase-based model. It is an open source system and uses beam-search to reduce the searching space. We use the default settings for this model.

3. Data Preparation

We only use the data provided by the organizer for each task. We first used the Chinese lexical analysis system ICTCLAS for splitting Chinese characters into words and a rule-based tokenizer for tokenizing English sentences. Then, we convert all alphanumeric characters to their 2-byte representation. Finally, we ran GIZA++ and used the "grow-diagfinal" heuristic to get many-to-many word alignments.

We used the SRI Language Modeling Toolkit to train the Chinese/English 5-gram language model with Kneser-Ney smoothing on the Chinese/English side of the training corpus respectively.

Regarding to Silenus, we used the Chinese parser of Xiong et al.(2006) and English parser of Charniak et al.(2005) to parse the source and target side of the bilingual corpus into packed forests respectively. Then we pruned the forests with the marginal probability based inside-outside algorithm with a pruning threshold $p_e = 3$. At the decoding time, we use a larger pruning threshold $p_d = 12$ to generate the packed forest.

4 Development Set Selection

Our development set for each task is selected automatically from all the development sentences according to the n-gram similarity, which is calculated against the current test set sentences

Our method works as follows: First, we gather every n-gram(up to 10) in the test set into a map W. and assign a score S_w for each n-gram w in W, which is calculated as

$$S_w(w) = n \cdot count(w)$$

where $count(w)$ is the number of occurrence of w in test set. Then, we assign a sentence score S_s to each candidate sentence s in development set, which is calculated as:

$$S_s(s) = \frac{\sum_{w \in W} S_w(w) \cdot count_s(w)}{length(s)}$$

where the $count_s(w)$ is the number of occurrence of w in s, and the $length(s)$ is the number of words in s. Finally, we choose the top k sentences as our new development set by using different thresholds.

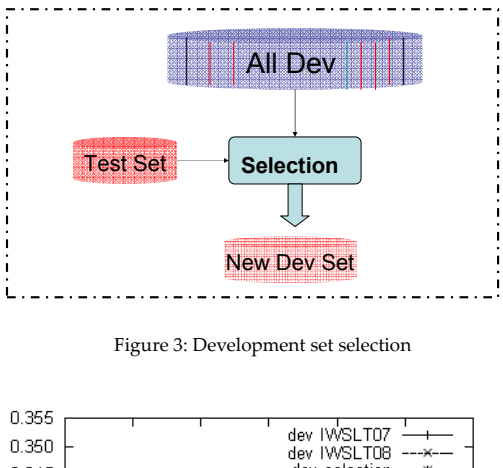

Figure 3: Development set selection

4.1 Results on IWSLT08

We first test our development set selection method on the test set of IWSLT08. The running single system in this section is Chiero. The thresholds are integers from 1 to 5.

The final results are shown on Figure 4. The bottom line is the BLEU scores when we tune feature weights on IWSLT07, while the top line is the performances when we tune weights on test set of IWSLT08. Then the results of our dev selection method are shown on the middle line, whose points are associated with the sentence numbers in each dev set. So we can conclude that our selection method improves the performance of our single system.

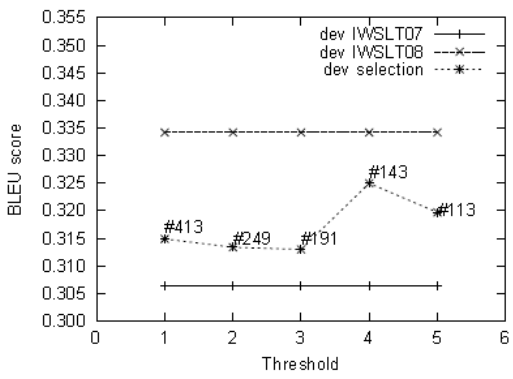

Figure 4: BLEU scores and sentence # on different dev sets

4.2 Results on IWSLT09

Table 1 gives the BLEU scores (case-insensitive, with punctuations) of our four single systems achieved on the dev sets we selected, where "BTEC_CE" denotes Chinese-English direction of BTEC task, "CT_CE" denotes Chinese-English direction of challenge task, and "CT_EC" denotes English-Chinese direction of challenge task.

For each task of this year's evaluation, the final primary system is the system, who achieves the MAX BLEU score on dev set. So we chose Moses for BTEC CE task, Chiero for CT CE task and Silenus for CT EC task, accordingly.

From Table 2, although Silenus achieves a higher BLEU score of 0.3886 and wins the third place on CT _EC CRR task, the correspondent score on ASR.20 task is very low, which is only 0.2901. The main reason lies in the different parsing quality on two set. With too much noise in ASR results, the parser failed to generate good forest, which will hurt the performance inevitably.

System \ Task	BTEC_CE	CT_CE	CT_EC
Bruin	0.4204	0.3521	0.4623
Chiero	0.4359	0.3732	0.4369
Moses	0.4683	0.3645	0.4734
Silenus	0.4489	0.3649	0.4775

Table 1: BLEU scores of our single systems on Dev sets

Task	Input	System	BLEU
BTEC	CRR	Moses	0.3563
CT_CE	CRR	Chiero	0.3078
	ASR.20		0.2859
CT_EC	CRR	Silenus	0.3886
	ASR.20		0.2901

Table 2: The BLEU scores of each primary single system on test sets

5. Additional Experiments

We also conducted several experiments of system combination after the Evaluation Campaign. Firstly, we applied two kinds of word level combination systems, which are based on the techniques of IHMM(He et al., 2008) and TER(Snover et al., 2006) respectively. But both systems are failed due to the poor hypothesis alignments, on which the Oracle BLEU score is only 6 points higher than the score of single best system. Finally, we reranked the merged n-best lists of all single systems with our sentence level combination system, which is global linear model with a series of simple features, we obtain significant improvements of +4 BLEU on BTEC_CE track. What we can conclude is that sentence-level combination method is more suitable than word-level approach on spoken language translation.

6. Conclusion

In this poster, we describe the ICT SMT systems for the evaluation campaign of IWSLT 2009. For each task, we first used the selection method to construct a development set, on which we tuned all the single systems with MERT. Finally, we chose the system with maximum BLEU score as our primary system. Since we didn't use any rescoring or system combination techniques for the final submitions, we got a relatively lower rank. Another problem we doubt is the small training set, which includes only 30K sentence pairs. The small training set will inevitably introduce much errors to SMT pipeline, such as word segmentation, parsing, word-alignment etc.. As a result, on one hand, good translation models are failed to explore their potential strengths; on the other hand, the pre- and post-processing techniques will attract more and more attentions, since they can reduce the negative effects of noise significantly. As last, the additional experiments, we carried out after the Evaluation Campaign, also suggested that our sentence-level combination system performs better than the word-level combiners on tasks of spoken language translation.