The CASIA Statistical Machine Translation System for IWSLT 2009

Maoxi Li, Jiajun Zhang, Yu Zhou, and Chengqing Zong

National Laboratory of Pattern Recognition, Institute of Automation
Chinese Academy of Sciences, Beijing, 100190, China
{mxli, jjzhang, yzhou, cqzong}@nlpr.ia.ac.cn

Abstract

This paper reports on the participation of CASIA (Institute of Automation Chinese Academy of Sciences) at the evaluation campaign of the International Workshop on Spoken Language Translation 2009. We participated in the challenge tasks for Chinese-to-English and English-to-Chinese translation respectively and the BTEC task for Chinese-to-English translation only. For all of the tasks, system performance is improved with some special methods as follows: 1) combining different results of Chinese word segmentation, 2) combining different results of word alignments, 3) adding reliable bilingual words with high probabilities to the training data, 4) handling named entities including person names, location names, organization names, temporal and numerical expressions additionally, 5) combining and selecting translations from the outputs of multiple translation engines, 6) replacing Chinese character with Chinese Pinyin to train the translation model for Chinese-to-English ASR challenge task. This is a new approach that has never been introduced before.

1. Introduction

This paper describes the statistical machine translation (SMT) system developed by CASIA for evaluation campaign of the International Workshop on Spoken Language Translation (IWSLT) 2009. The tasks that we participated in include:

- Challenge translation tasks:
  - Chinese-to-English: CT_CE (CRR and ASR)
  - English-to-Chinese: CT_EC (CRR and ASR)
- BTEC translation tasks:
  - Chinese-to-English: BTEC_CE

For the Challenge translation tasks, CRR and ASR represent the different input conditions, namely correct recognition results and the outputs of the automatic speech recognizers, respectively. Unlike Challenge translation tasks, the BTEC translation tasks focus on text input only, i.e., no automatic speech recognizer results have to be translated.

Reminder of the paper is organized as follows: Section 2 presents the architecture of CASIA system. In Section 3, we give the details of CASIA system implementation. In Section 4, experimental results are described and the details on the result analyses are also given. The conclusions are drawn in Section 5.

2. System Architecture

The overall architecture of CASIA system is depicted as Figure 1. First, the test data are preprocessed and then passed into multiple statistical machine translation decoders to produce a serial of N-Best lists; we call this process as the decoding module. Second, the N-best lists are collected, which are exploited by a system combination to form a new N-Best list; we call this process as the combining module. Finally, we make use of rich global feature functions to re-score the new N-Best list hypotheses to pick up the best translation; we name this process as the re-scoring module.

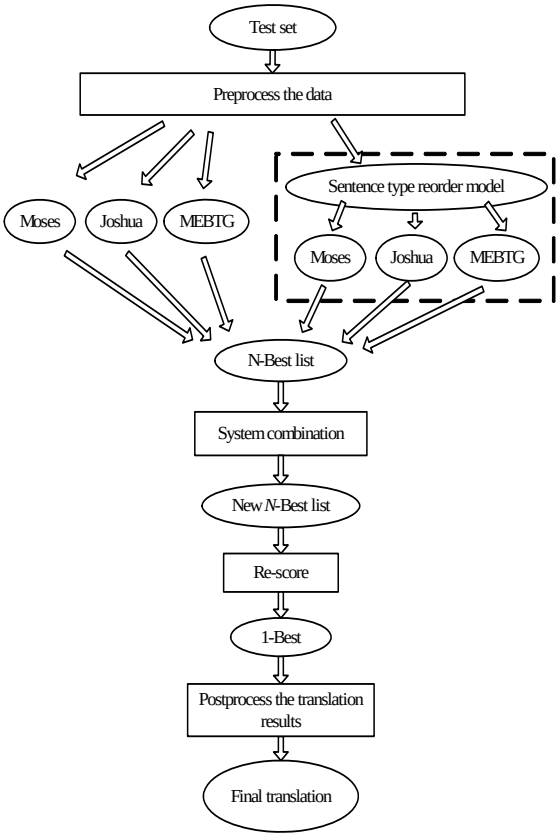

Figure 1. The overall architecture of CASIA system

In the decoding module, three state-of-the-art statistical machine translation decoders are used, which are listed as follows:

- A phrase-based statistical machine translation decoder: Moses[1].
- A hierarchical phrase-based statistical machine translation decoder: Joshua [2].
- An in-home maximum entropy-based reordering model decoder that is similar to [3]: MEBTG [4].

For Chinese-to-English translation tasks, each sentence in the test set is preprocessed by a sentence type-based reordering model [5] and then is decoded by the above three decoders. The deformed decoders are called as Moses-Reorder, Joshua-Reorder, MEBTG-Reorder, respectively. In sum, we have six statistical machine translation decoders.

In the combination module, a word-level system combination approach similar to that given in [6] is exploited, but we enhance the system combination by substituting a word reordering alignment (WRA) [7] for alignment produced by translation error rate (TER) [8].

In the re-scoring module, some global features described in [9] are exploited to re-score the N-best list results from the system combination to get the final submitted translation outputs.

For Chinese-to-English ASR challenge task, a great deal of Out-Of-Vocabulary (OOV) words have emerged, which are caused by the error outputs of the automatic speech recognizers. So we replace Chinese characters with Chinese Pinyin to train the translation model to decrease the negative effect of the errors in recognition results. This approach is the first time to be reported.

3. System Implementation

3.1. Prepare the data

Compared to the previous IWSLT evaluation campaigns, the monolingual and bilingual language resources that are permit to be used to train the translation models for the primary submitted results are limited to the released corpus for each translation task in this year, which aims at eliminating the affects of different language resources that used by different participants. In this case, we carefully preprocess the training corpus, development set and test set to decrease OOV words emerged in the test set.

3.1.1. Chinese word segmentation

Exploiting different Chinese word segmentation approaches have great impact on the performance of the translation systems[10], so we test two Chinese word segmentation approaches, namely ORI (original Chinese word segmentation for the data set), and ICT (the free software toolkit ICTCLAS3.0 [1]). For the Challenge CRR translation task, the BLEU [11] score on the development set is given in Table 1, unless otherwise stated, performance was measured using official evaluation metrics mteval-v13 scripts on case insensitivity:

Table 1: The performances with different Chinese word segmentation approaches

Approaches	ORI	ICT	ORI+ICT
BLEU	35.31	36.24	36.63

The ICT Chinese word segmentation approach has significantly improved the translation performance, and the system that combined the two Chinese word segmentation yields up to 0.39 BLEU points over the system based on ICT word segmentation. In order to get a better result, we combine both the two word segmentation results to train the model and decode the test set with different word segmentations, combine all the N-Best list and pass them to the next step to generate the new N-Best list.

3.1.2. English word Lowercased and tokenized

For English sentences, the first word is written in uppercase and punctuations are followed after the English words, which lead a word in different positions of a sentence may have different morphology.

At the same time, the supplied corpus for each translation task is very small, there is only 19,972 pair-wise bilingual sentences in the training data of BTEC translation tasks and 30,033 pair-wise bilingual sentences in the training data of Challenge translation tasks. To avoid data sparse for each translation task, we make everything lowercase and tokenize each sentence.

We use the lowercased and tokenized scripts of the open source toolkit Moses[2] to do this job.

3.1.3. Named entities process

Since named entities (NE), including person names, location names, organization names, temporal and numerical expressions are very common in spoken language; their translation plays an important role in spoken language translation. In our system, firstly, we adopt a hybrid named entity recognizer [12] to identify Chinese NEs; Secondly, person names and location names are translated word by word, while organization names are translated by a structure-based translation model [13].

We exploit a rule-based approach to translate the temporal and numerical NEs. And the format of the translation results is in accordance with the format of the spoken language tradition or the references of the

[1] http://www.nlp.org.cn

[2] http://www.statmt.org/moses/

development set. For example, we translate the temporal NE "八月十日" into "August tenth", not "August 10"; and we translate the numerical NE "3 4 5 6 7 8 9 0 1 2" into "three four five six seven eight nine zero one two", not "3 4 5 6 7 8 9 0 1 2".

3.2. The SMT decoders

We use three state-of-the-art statistical machine translation decoders and their corresponding deformed decoders, which add a sentence type-based reordering model before translating, to decode the development set and test set.

3.2.1. The base three SMT decoders

Moses: Moses is a cutting-edge machine translation program that reflects the latest developments in the area of statistical machine translation research, which can be trained to translate between any two languages, and yields high quality results. It exploits a log-linear model to search the target sentence with the largest probability given a source sentence.

Joshua: Joshua is a hierarchical phrase-based statistical machine translation decoder, which implements all of the algorithms required for synchronous context free grammars and suffix-array grammar extraction. The best benefit of using Joshua package is that it makes hierarchical phrase-based translation easily and stably run on a large-scale data.

MEBTG: MEBTG is an in-home maximum entropy-based reordering model decoder, which is realized according to the approaches of reference [14] and [3]. In the decoder, the prediction of relative orders of any two adjacent blocks is considered as a problem of classification; and a MaxEnt classifier is trained according to the training data. A CKY algorithm is exploited to decode the test set which limits the phrase table within 40 and the partial hypotheses is within 200.

3.2.2. The deformed three SMT decoders

When translate a sentence from Chinese to English, a sentence type-based reordering model, we call Bandore [5], divides the Chinese sentences into three types and employs different reordering model for each sentence type.

Bandore serves as a preprocessing module for SMT system. Bandore works as followed:

Firstly, a support vector machine is used to classify Chinese sentences into three types: special interrogative sentences, other interrogative sentences and non-question sentences, which directly exploit all the words occurring in the sentence as features.

Secondly, corresponding reordering model is developed for specific sentence types, that is phrase-ahead model is employed for special interrogative sentences and phrase-back model is employed for other sentence types.

Finally, after reordering the Chinese sentences of training set and test set, we pass the reordered sentences into the SMT decoders, which are called as Moses-Reorder, Joshua-Reorder and MEBTG-Reorder, respectively, to get the translation outputs. For more information, please refer to [5].

3.2.3. SMT decoders setting

The Joshua SMT package that we used is version 1.1, while the open source toolkit Moses that we used is version 2009-04-13. The Joshua SMT package version 1.1 is the only publicly available version at that time, but when choice Moses version 2009-04-13, we do some experiments on the development set of the Challenge CRR tasks to test this version to be compared with the old version 2008-07-11. The experimental results are as Table 2:

Table 2. The performance on the Challenge CRR tasks with different Moses version

Tasks	Version 2009-04-13	Version 2008-07-11
CT_CE	35.64	35.37
CT_EC	33.70	33.53

From Table 2, we finally choose the new version 2009-04-13 of Moses as one of our translation decoders. In the initial tuning stage, the parameters are set as default, that is, heuristic grow-diag-final-and alignment, 3gram with the "shortest" tuning option. We also test the "closest" tuning option with Moses and Joshua, The performance is shown as Table 3 and Table 4:

Table 3. The performance on the development set with different Joshua tuning option

Tasks	closest	shortest
CT_CE	38.00	36.58
CT_EC	31.96	31.03
BTEC	47.05	45.66

Table 4. The performance on the development set with different Moses tuning option

Tasks	closest	shortest
CT_CE	36.56	35.64

According to the above experiments, we find that the BLEU score could improve almost 1 point when tuning the parameters with "closest" option, so we use this option in the evaluation campaign. On the other hand, the official evaluation metrics mteval-v13 script takes the closest reference translation length as the effective reference length, so the "closet" option seems better in theory.

3.3. System Combination

We exploit a word-level system combination approach similar to [6] to combine the outputs of multiple SMT decoders, but we improve the system combination performance by substituting a word reordering alignment (WRA) [7] for alignment produced by TER. The 10-Best lists generated by each decoder are used for system combination.

Different from the existing WER[16] and TER monolingual sentence alignment, our WRA approach directly shifts the word sequences of the translation hypothesis to the correct location within the translation hypothesis. In our approach, the continuous word sequences are first found and replaced by some variables. Then we align the variables and words identical to each other in the two sentences and detect the word sequences that should be reordered. Finally, according to some word reordering heuristics, the detected word sequences are shifted to the correct position and dynamic programming are exploited to align the sentences after reordering.

For instance, given two translation hypotheses:

Hyp: this color do you think suits me
Ref: do you think that color suits me

The WER alignment, TER alignment, and WRA alignment are depicted as *Figure 2, 3, 4*.

this	color	do	you	think	null	null	suits	me
null	null	do	you	think	that	color	suits	me

Figure 2. An example of WER alignment

this	do	you	think	null	color	suits	me
null	do	you	think	that	color	suits	me

Figure 3. An example of TER alignment

do	you	think	this	color	suits	me
do	you	think	that	color	suits	me

Figure 4. An example of word reordering alignment

3.4. Re-scoring

Because we have employed several different SMT decoders and system combination technology, the local feature functions of each translation hypothesis cannot be used in the rescoring module. Therefore, we should use the global feature functions to score the new N-Best generated by system combination. The functions that we used are the same as [9]. We merge the 100-Best hypotheses produced by the system combination approach and all the original 10-Best hypotheses generated by each single decoder to produce the final new N-Best list for re-scoring. Note that the 100-Best hypotheses produced by the system combination might include some original hypotheses, so we delete the repeated ones.

3.5. Post-processing

For Chinese to English translation tasks, post-processing includes re-case and de-tokenize. We train a re-caser with Moses and re-case the outputs, while de-tokenizing the outputs is done by the de-tokenizer scripts of Moses package. We also focus on the official evaluation specifications, which require the English sentences being evaluated with punctuation marks tokenized, so we tokenize the final submitted translation with the official tool: "ppEnglish.case+punc.pl" script.

For English to Chinese translation task, the official evaluation specifications require Chinese MT Outputs divide into Chinese characters, we use the official tool: "splitUTF8characters.pl" script to transform segmentation into characters.

3.6. Replace Chinese character with Chinese Pinyin for CT_CE ASR task

Different from English pronunciation, there exist five tones for Chinese character, namely the 1st tone, the 2nd tone, the 3rd tone, the 4th tone, and the 5th tone. It is very difficult for the automatic speech recognizers to distinguish a tone from the others, which often lead the recognizers to make mistakes.

我 的 名字 是 **铃**木 **直**子
wo3 de5 ming2 zi4 shi4 **ling2** mu4 **zhi2** zi5

我 的 名字 是 铃木 **智**子
wo3 de5 ming2 zi4 shi4 ling2 mu4 **zhi4** zi5

我 的 名字 是 铃木 **知**子
wo3 de5 ming2 zi4 shi4 ling2 mu4 **zhi1** zi5

我 的 名字 是 **玲**木 **智**子
wo3 de5 ming2 zi4 shi4 **ling2** mu4 **zhi4** zi5

Figure 5. the N-best list hypotheses of the automatic speech recognizers and the corresponding Chinese Pinyin of the hypotheses.

At the same time, some characters are homophone, for example, the characters "玲" and "铃" have the same pronunciation, this make the situation even worse. For challenge CT_CE ASR task dialog01_13, the N-best list hypotheses of the automatic speech recognizers and the corresponding Chinese Pinyin of the hypotheses are presented in *Figure 5*, where the numbers follow after the alphabets represent the tone. The correct recognition result is "我 的 名字 是 铃木

直子”, we find the main mistakes comes from the different tone , for example “直”, “智”, “知”, and the homophones.

For Challenge CT_CE ASR task, the supplied corpus is very small, we might easily find that “玲木”, “智子”, “知子” are OOV words. The recognizing errors have brought on a great deal of OOV words emerging, which greatly decrease the translation performance. So in order to decrease the negative effect by error recognized outputs, we substitute Chinese Pinyin for Chinese character to train the translation model and exploit this model to decode the test set.

We use the toolkit AddPinyin, which is designed by our laboratory, to transform the Chinese characters in the train corpus and test set to Chinese Pinyin. Note that Pinyin syllables corresponding to a character is similar to an English word, which is segmented by a space. Due to the time limit of the evaluation campaign, we use the parameters that have been tune on the original data to decode the test set.

Finally, we combine the entire N-Best list generated by the character-based and pinyin-based combining modules to generate the new N-Best list.

4. Experimental Results

We carry out the experiments on each track task and test the performance with different word alignment approaches mainly on Challenge CT_CE CRR task.

4.1. Corpus statistics

We carry out the experiments only on the supplied corpus, and the table 5, 6 and 7 give the corpus statistics for each task respectively. In order to get stable parameters for the test set, we merge all individual development sets of a task when tuning parameters. Note that different development sets have different number of reference translations given a translation task; we copy the references to obtain the same number of references for each development sentence. For example, some development sets of BTEC task have 16 reference translations and another has 7 reference translations, we copy the first 2 reference translations 3 times and the last 5 reference translations 2 times (2*3+5*2 = 16) to extend the 7 reference translations to 16. According to the BLEU score calculating, this does not change the score value.

For the development sets of different tasks, we use the same pre-processing approach to deal with the source sentences and the reference translations.

The IWSLT official announcements that the participants are free to use the developments sets as they wish for tuning of model parameters or as training bitext, so we add the development set to the training corpus to re-train the models to generate the translation phrase table or rules for decoding the test set under the parameters tuned on all the development set.

Table 5. The corpus statistics for BTEC task

corpus	Size
Train corpus	19,972 sentence pairs
Development set	2,508 sentence with 16 references
Test set	469 sentence

Table 6. The corpus statistics for Challenge CT_CE task

corpus	Size
Train corpus	30,033 sentence pairs
Development set	4,447 sentence with 16 references
Test set	405 sentence

Table 7. The corpus statistics for Challenge CT_EC task

corpus	Size
Train corpus	30,033 sentence pairs
Development set	1,465 sentence with 7 references
Test set	393 sentence

4.2. The performance with different word alignment approaches

We carry out the experiments to test the performance with different word alignment approaches on Challenge CT_CE CRR task. We use Moses system with the following setting: extracted maximum phrase length is 10, language model is 4gram.

4.2.1. Combine word alignments produced by GIZA++[17] and BerkeleyAligner[18]

To balance the word alignment performance between precision and recall, we combine the word alignment produced by GIZA++ and BerkeleyAligner.

We use GIZA++ and BerkeleyAligner to generate different word alignment files, and then merge the two files to a big word alignment file by concatenating one alignment file to the other; the big word alignment file is exploited to produce the phrase table and reordering table by Moses decoder. At the same time, the big word alignment file can be exploited by Joshua decoder.

Table 8. The translation performance on Moses and Joshua by combining word alignment

Challenge CT_CE	Moses	Joshua
Baseline	36.24	36.83
Combining word alignment	38.09	39.24

Table 8 shows the translation performance on Moses and Joshua by combining the word alignments produced by GIZA++ and BerkeleyAligner. We find that this approach yields up to almost 1.9 BLEU points on Moses and 2.5 BLEU points on Joshua over the baseline system. In *table 8*, all the scores are used the BLEU scores under the toolkit released by Moses or Joshua.

4.2.2. A two-step word alignment approach

In IWSLT'09 evaluation campaigns, the monolingual and bilingual language resources are limited to the supplied corpus for each translation task, so we cannot add additional dictionary to correct the word alignment. We present a new approach, namely two-step word alignment, to construct a dictionary and add the dictionary words to the training data to produce a new word alignment.

First, we use GIZA++ to produce word alignment and phrase table; we set a threshold value, such as 0.5, to filter the phrase table and the phrase pairs with the probability larger than the threshold are reserved as the dictionary entries. In the second step, we add the reliable bilingual words generated by GIZA++ with high probabilities into the training data and re-train the phrase table and reordering model.

Table 9 shows the translation performance on Moses with a two-step word alignment approach. We find that the two-step word alignment approach improves the translation performance about 0.6 BLEU points. Table 9 also gives the BLEU score on Moses with the two-step word alignment approach and the combining word alignment.

Table 9. The translation performance on Moses with a two-step word alignment approach

	BLEU
Baseline	36.24
Two-step word alignment	36.83
Combining word alignment+ two-step word alignment	38.26

4.3. The performance on the development set and test set

We exploit all the special methods that described above to tune the development set and decode the test set.

The experimental results are given on table 10, 11 and 12. Note that the BLEU scores on case insensitivity on the test set are coming from the official releasing results. For each task, we submit three system running results, that is re-score result (primary), system combination result (contrastive 1), the result of the best individual system on the development set (contrastive 2). The results show that the improvement extent on the test set by the system combination module or re-score module is slightly larger than the development set. For example, the system combination module for BTEC_CE task get 2.56 BLEU points improvement over the best individual system on the development set, but get 3.29 BLEU points on the test set. This is because of the development set for each task is very large, about 3.73 times (CT_EC task) to 10.98 times (CT_CE task) larger than the test set, the translation performance on the development set is very stable. On the other hand, adding the development set to the training corpus when decoding the test sets is very effective to improve the system performance on the test set.

For challenge ASR task, the SMT decoders decode the 5-Best hypotheses of the automatic speech recognizers. To eliminate the noises induced by error recognized results, we do not tune the parameters on the development set. We use the parameters that are optimized on the corresponding CRR development set.

Table 10. The translation performance on the development set and the test set for BTEC CE task

	DEV	TST
MEBTG	45.40	Not submitted
MEBTG_Reorder	46.71	
Joshua	46.01	
Joshua_Reorder	47.05	
Moses	47.41	
Moses_Reorder	47.52	42.39
SysComb	50.08	46.68
Re-score	52.15	48.97

Table 11. The translation performance on the development set and the test set for Challenge CT_CE CRR task

	DEV	TST
MEBTG	38.04	Not submitted
MEBTG_Reorder	39.02	
Joshua	38.00	
Joshua_Reorder	37.95	
Moses	39.00	
Moses_Reorder	39.60	33.04
SysComb	41.39	36.44
Re-score	43.78	38.08

Table 12. The translation performance on the development set and the test set for Challenge CT_EC CRR task

	DEV	TST
MEBTG	29.85	Not submitted
Joshua	31.96	
Moses	31.80	39.10
SysComb	32.28	40.03
Re-score	34.06	43.04

For challenge CT_CE ASR task, we are surprised to find that the BLEU score on sensitivity of the best individual system reach 29.81, which is only 0.14 BLEU points lower than the best individual system of challenge CT_CE CRR task. It proves that the approach replacing Chinese character with Chinese Pinyin is promising to improve the performance of the spoken

language translation. Our future work may be doing more experiments to verify this fact.

5. Conclusion

This paper describes our work on improving the performance of spoken language translation. Our system use three state-of-the-art SMT decoders to translate each task that we participated, then a word-level system combination approach is exploited to combine the multiple outputs of the SMT decoders, finally we use rich global feature functions to re-score the new hypotheses to pick up the best translation. According to our experimental results and the evaluation results, we draw the following conclusions:

1) The translation performance on the development set and the test set proves that the combination module and rescoring module are effective for SMT systems. The combination module and rescoring module can yield up to 3~6 BLEU points improvement over the best individual system.

2) For Chinese-to-English ASR challenge task, the approach, which replaces Chinese character with Chinese Pinyin to train the translation model and decodes the test set, improves the system performance greatly. The best individual system performance for CT_CE ASR task is only lower than the translation performance for CT_CE CRR task about 0.20 BLEU points.

3) Combining different word alignments, which are produced by GIZA++ and BerkeleyAligner, can effectively improve the translation performance.

4) The two-step word alignment approach by adding larger probability bilingual words to correct word alignment results can also improve the translation performance about 0.6 BLEU points.

5) Combining different word alignments significantly improve the system performance for the spoken language translation on the condition that only supplying corpus could be used.

6) Processing NE, including person names, location names, organization names, temporal and numerical expressions, to the correct formats improves the translation quality.

6. Acknowledgements

The authors thank Yufeng Chen, Rui Xia, Kun Wang, Peng Liu, Tao Zhuang, Ping Jian and Zhiguo Wang for their great help on data pre-processing. The research work described in this paper has also been partially funded by the Natural Science Foundation of China under grant No.60736014, 60723005 and 90820303, the National Key Technology R&D Program under grant No. 2006BAH03B02, the Hi-Tech Research and Development Program (863 Program) of China under grant No. 2006AA010108-4, and also was supported by the China-Singapore Institute of Digital Media as well.

7. Reference

[1] P. Koehn, H. Hoang, A. Birch *et al.*, "Moses: Open Source Toolkit for Statistical Machine Translation," in Annual Meeting of the Association for Computational Linguistics (ACL), demonstration session, Prague, Czech Republic, 2007.

[2] Z. Li, C. Callison-Burch, C. Dyer *et al.*, "Joshua: An Open Source Toolkit for Parsing-based Machine Translation," in Proceedings of the Workshop on Statistical Machine Translation (WMT09), 2009.

[3] D. Xiong, Q. Liu, and S. Lin, "Maximum entropy based phrase reordering model for statistical machine translation," in Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, Sydney, Australia, 2006.

[4] Y. He, and C. Zong, " A Generalized Reordering Model for Phrase-Based Statistical Machine Translation," in Proceedings of the 8th Conference of the Association for Machine Translation in the Americas (AMTA), Waikiki, Hawai'i, USA, 2008.

[5] J. Zhang, C. Zong, and S. Li, "Sentence Type Based Reordering Model for Statistical Machine Translation," in Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), Manchester, UK, 2008, pp. 1089--1096.

[6] A.-V. I. Rosti, S. Matsoukas, and R. Schwartz, "Improved Word-Level System Combination for Machine Translation," in Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, Prague, Czech Republic, 2007, pp. 312–319.

[7] M. Li, and C. Zong, "Word reordering alignment for combination of statistical machine translation systems," in the International Symposium on Chinese Spoken Language Processing (ISCSLP), Kunming, China, 2008.

[8] M. Snover, B. Dorr, R. Schwartz *et al.*, "A study of translation edit rate with targeted human annotation," in Proceedings of the 7th Conference of the Association for Machine Translation in the Americas, Cambridge, 2006, pp. 223-231.

[9] Y. He, J. Zhang, M. Li *et al.*, "The CASIA Statistical Machine Translation System for IWSLT 2008," in Proc. of the International

Workshop on Spoken Language Translation, Hawaii, USA, 2008, pp. 85-91.

[10] H. Wang, H. Wu, X. Hu *et al.*, "The TCH Machine Translation System for IWSLT 2008," in Proc. of the International Workshop on Spoken Language Translation, Hawaii, USA, 2008, pp. 124-131.

[11] K. Papineni, S. Roukos, T. Ward *et al.*, "B LEU: a method for automatic evaluation of machine translation," in Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), Philadelphia, PA, 2002, pp. 311-318.

[12] Y. Wu, J. Zhao, and B. Xu, "Chinese Named Entity Recognition Model Based on Multiple Features," in Proceedings of HLT/EMNLP 2005, Vancouver, B.C. Canada, 2005, pp. 427-434.

[13] Y. Chen, and C. Zong, "A Structural-Based Model for Chinese Organization Name Translation," *ACM Transactions on Asian Language Information Processing (ACM TALIP)*, vol. 7, no. 1, pp. 1-30, 2008.

[14] D. Wu, "Stochastic inversion transduction grammars and bilingual parsing of parallel corpora," *Comput. Linguist.*, vol. 23, no. 3, pp. 377-403, 1997.

[15] S. Kumar, and W. Byrne, "Minimum bayes-risk decoding for statistical machine translation," in Proc. HLT-NAACL, Boston, MA, USA, 2004, pp. 196-176.

[16] S. Bangalore, F. 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.

[17] F. J. Och, and H. Ney, "Discriminative training and maximum entropy models for statistical machine translation," in Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Philadelphia, Pennsylvania, 2001.

[18] J. DeNero, and D. Klein, "Tailoring Word Alignments to Syntactic Machine Translation," in Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL'07), Prague, Czech Republic, 2007, pp. 17-24.