The NICT Translation System for IWSLT 2014

Xiaolin Wang *Andrew Finch* *Masao Utiyama* *Taro Watanabe* *Eiichiro Sumita*

Multilingual Translation Group
National Institute of Information and
Communications Technology
Kyoto, Japan
{first.last}@nict.go.jp, mutiyama@nict.go.jp

Abstract

This paper describes NICT's participation in the IWSLT 2014 evaluation campaign for the TED Chinese-English translation shared-task. Our approach used a combination of phrase-based and hierarchical statistical machine translation (SMT) systems. Our focus was in several areas, specifically system combination, word alignment, and various language modeling techniques including the use of neural network joint models. Our experiments on the test set from the 2013 shared task, showed that an improvement in BLEU score can be gained in translation performance through all of these techniques, with the largest improvements coming from using large data sizes to train the language model.

1. Introduction

In the IWSLT 2014 machine translation evaluation campaign, the NICT team participated in the TED [1] translation shared-task for Chinese-English. This paper describes the machine translation approach adopted for this campaign.

Our system was a combination of phrase-based and hierarchical SMT systems. The combination was performed by reranking the n-best hypotheses from these systems. A log-linear model which used the hypothesis scores of the component systems as features was used to calculate the score used in reranking. Additional features were also added into the log-linear model, for example features from a neural network model, or talk-level language model scores.

In addition to system combination, we put emphasis on language modeling. We used three approaches to improve the language modeling in the system. In the first approach we used a language model that was an interpolation of an in-domain language model, and a language model trained on the GIGAWORD data. In the second approach, we incorporated a language model trained on the machine translations of each talk in the test dataset into the reranking procedure. In the third approach, a bilingual feed-forward neural network [2] was used in the reranker.

Finally, we also improved the word alignment by using combining the alignments from two independent aligners: GIZA++ [3] and a modified version of the CICADA aligner [4].

2. Data

We used same Chinese-English data sets in all of the experiments in this paper. The supplied bilingual data consisted of 179901 sentence pairs. From this data we randomly selected a 3023-pair development set for tuning the decoder, and a 1553-pair development set for tuning the reranker. These development sets consisted of complete talks. All of the remaining talks were used as bilingual training data for the component SMT systems. We used the IWSLT 2013 test set for evaluation. For some of the experiments we used language models trained on the English LDC Gigaword dataset, a collection of approximately 4 billion words of international newswire text.

2.1. Pre-processing

The English data was tokenized by applying the EUROPARL tokenizer [5]. We also removed all case information from the English text to help to minimize issues of data sparseness in the models of the translation system. All punctuation was left in both source and target. We took the decision to generate target punctuation directly using the process of translation, rather than as a punctuation restoration step in post processing based on experiments carried out for the 2010 IWSLT shared evaluation [6].

2.2. Post-processing

The output of the translation system was subject to the following post-processing steps which were carried out in the following order:

1. In all experiments, the out of vocabulary words (OOVs) were passed through the translation process unchanged, some of these OOVs were Chinese and some English. For the primary submission, we took

the decision to delete only those OOVs containing Chinese characters not included in the ASCII character set and leave words containing only ASCII characters in the output.

2. The output was de-tokenized using the de-tokenizer included with the MOSES toolkit [7].

3. The output was re-cased using the re-casing tool supplied with the MOSES toolkit. We trained the re-casing tool on cased text from the TED talk training data.

3. The Base Systems

3.1. Decoders

Our submission used two SMT systems within a system combination framework; these systems were:

1. OCTAVIAN, an in-house phrase-based decoder.

2. A hierarchical version of the MOSES decoder [7].

The OCTAVIAN decoder used in these experiments is an in-house phrase-based statistical machine translation decoder that can operate in a similar manner to the publicly available MOSES decoder [7]. The base decoder used a standard set of features that were integrated into a log-linear model using independent exponential weights for each feature. These features consisted of: a language mode; five translation model features; a word penalty; and a lexicalized re-ordering model with monotone, discontinuous, swap features for the current and previous phrase-pairs. We decoded with a reordering limit of 5 in the OCTAVIAN phrase-based decoder.

3.2. Language Model Training

The language models were built using the SRI Language Modeling Toolkit [8]. A 5-gram model was built for decoding the development and test data for evaluation. The language models were smoothed using modified Knesser-Ney smoothing.

3.3. Translation Model Training

The translation model for the base system was built in the standard manner using a 2-step process. First the training data was word-aligned using a combination of the CICADA and GIZA++ [3] aligners. Two copies of the corpus were aligned independently with each aligner, then the aligned copies were concatenated prior to phrase extraction. Second, the phrase-extraction heuristics from the MOSES [7, 9] machine translation toolkit were used to extract a set of bilingual phrase-pairs using the alignments.

3.4. Parameter Tuning

To tune the values for the log-linear weights in our system, we used the standard minimum error-rate training procedure (MERT) [10]. The weights for the models were tuned using the development data supplied for the task.

Component System	BLEU (%)
OCTAVIAN	14.74
MOSES (hierarchical)	14.95

Table 1: BLEU scores of the component systems

3.5. Evaluation

We evaluated each of these systems on the IWSLT 2013 test set, and the results are shown in Table 3.5. The evaluation in all of the experiments in this report was carried out on tokenized, lowercase data, using the "multi-bleu.perl" evaluation script included in release version 2.1 of the MOSES toolkit. The systems are roughly comparable in performance, and about 1.5 BLEU percentage points higher than the case-insensitive MOSES baseline reported in [11], we believe this can be explained by differences in the tokenization used for evaluation, and also by differences in the development sets used for tuning. We found that when tuned and evaluated on different data sets, the relative rankings of the systems may vary.

4. Methodology

4.1. Language Modeling

4.1.1. Neural Network Model

We implemented the neural network joint models proposed in [2] and used the output as a feature in the reranker. We ran a set of experiments to determine the optimal network architecture. We varied the size of the context on both source and sides, and also the scale of the neural network. We found the settings used in [2] gave rise the highest performance, and we therefore adopted these settings in our system. These settings were: 11-word source context, 3-word target context, 192-unit shared embedding layer, and two additional 512-unit hidden layers. We set both input and output vocabulary size to 32000. The neural network was implemented using the NPLM toolkit [12].

The results are shown in Table 4.3. The gain using from this approach was approximately 0.5 BLEU points. This was lower than the gains reported in [2], however, in their experiments the neural network was directly integrated into the decoding process. We integrated monolingual neural network model into the OCTAVIAN decoder, however, the experiments were not completed due to time limitations.

4.1.2. Gigaword

We combined language models trained on the source of the parallel TED corpus, and the Gigaword newswire corpus by linear interpolation. The interpolated language model was then used directly in the decoding process, and constituted a

SMT System	BLEU (%)
OCTAVIAN TED LM	14.74
OCTAVIAN TED+Gigaword	16.72
MOSES hierarchical TED LM	14.95
MOSES hierarchical TED+Gigaword	16.83

Table 2: Evaluation of the effectiveness of using a large out-of-domain language model.

single feature in the log-linear model. The interpolation was done using the SRI Language Modeling Toolkit [8]. We ran pilot experiments to determine the best interpolation weight by grid search and found a weight of 0.5 to be the most effective. Both of the language models were trained with modified Knesser-Ney smoothing [13, 14].

The results are shown in Table 4.1.2. It is clear that adding a large out-of-domain language model is very effective on our task.

4.1.3. Talk-level Model

This model was a language model built by applying the SRI Language Modeling Toolkit to machine translated output. The talk-level language model was built from the set of 1000-best translation hypotheses obtained by translating the test set using each of the component translation systems. The 1000-best lists from the component systems were merged, into a set of unique word sequences. A different language model was build from each talk in the test set, and applied only to sentences from the same talk. The score of the language model was used as a feature for reranking.

The results are shown in Table 4.1.2 and show a modest improvement in performance over the baseline without this model.

4.2. Alignment

Two copies of the training data were aligned. One copy with GIZA++, and the other with an enhanced version of the CICADA aligner. The SMT models derived from the alignment were trained on the union of this aligned data.

The results are shown in Table 4.2. The largest gain arises from using the CICADA aligner together with the hierarchical SMT system. However we took the decision to use this strategy in our primary submission because in pilot experiments the strategy based on a combination of methods typically outperformed the strategy based on a single method.

4.3. System Combination

The system combination was performed by integrating features from the component SMT systems, together with a set of additional features within the framework of a log-linear model. The log-linear weights of all the features were tuned on a separate development set using the same MERT approach as in tuning the weights in the models used by the decoders. The features using in reranking were:

SMT System	BLEU (%)
OCTAVIAN GIZA++	14.74
OCTAVIAN CICADA	15.21
OCTAVIAN Union	15.22
MOSES hierarchical GIZA++	14.95
MOSES hierarchical CICADA	15.56
MOSES hierarchical Union	15.54

Table 3: Evaluation of the various alignment strategies.

SMT System	BLEU (%)
OCTAVIAN baseline	17.09
MOSES hierarchical baseline	17.56
Combination	17.65
Combination with neural network joint model	17.88
Combination with talk-level LM	17.68
Combination with all features	17.92

Table 4: Evaluation of the combination systems.

1. The decoder score from the OCTAVIAN decoder;
2. The decoder score from the hierarchical MOSES decoder;
3. The output from the joint neural language model;
4. The talk-level language model score.

1000-best lists from the 2-component systems were merged in the following manner:

1. The n-best lists of each component system were made unique; only the best scoring hypotheses was kept from a set of duplicate hypotheses which gave rise to the same target word sequence.
2. Hypotheses with the target text were merged across systems into a single hypothesis, receiving the respective decoder scores in features 1. and 2.
3. If the hypothesis was only generated by one of the component systems, it received zero for the feature corresponding to the decoder that did not generate it.
4. Features 3. and 4. were then calculated for each hypothesis.

The results are shown in Table 4.3. Both of the component systems used in the combination were trained using the enhanced alignment method proposed in Section 4.2, and included the interpolated language model described in Section 4.1.2.

5. Conclusions

This paper described NICT's system for the IWSLT 2014 evaluation campaign for the TED Chinese-English translation shared-task. Our approach was based on a combination of hierarchical and phrase-based statistical machine translation systems integrated with other features within the framework of a single log-linear model. We augmented the base systems using multiple alignment strategies, a neural network joint model, and a talk-level language model. We were able to improve the translation performance over a phrase-based MOSES baseline without these features by 2.96 BLEU points.

6. References

[1] M. Cettolo, C. Girardi, and M. Federico, "Wit^3: Web inventory of transcribed and translated talks," in *Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT)*, Trento, Italy, May 2012, pp. 261–268.

[2] J. Devlin, R. Zbib, Z. Huang, T. Lamar, R. M. Schwartz, and J. Makhoul, "Fast and robust neural network joint models for statistical machine translation," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers*, 2014, pp. 1370–1380. [Online]. Available: http://aclweb.org/anthology/P/P14/P14-1129.pdf

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

[4] T. Watanabe, "The cicada open source aligner, http://www2.nict.go.jp/univ-com/multi_trans/cicada/."

[5] P. Koehn, "Europarl: A parallel corpus for statistical machine translation," in *MT summit*, vol. 5, 2005, pp. 79–86.

[6] C.-L. Goh, T. Watanabe, M. Paul, A. Finch, and E. Sumita, "The NICT Translation System for IWSLT 2010," in *Proceedings of the seventh International Workshop on Spoken Language Translation (IWSLT)*, M. Federico, I. Lane, M. Paul, and F. Yvon, Eds., 2010, pp. 139–146.

[7] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowa, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "Moses: open source toolkit for statistical machine translation," in *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL 2007): demo and poster sessions*, Prague, Czeck Republic, June 2007, pp. 177–180.

[8] A. Stolcke, "SRILM - An Extensible Language Modeling Toolkit," in *Proceedings of the International Conference on Spoken Language Processing*, vol. 2, Denver, 2002, pp. 901–904.

[9] P. Koehn, "Pharaoh: a beam search decoder for phrase-based statistical machine translation models," in *Machine translation: from real users to research: 6th conference of AMTA*, Washington, DC, 2004, pp. 115–124.

[10] F. J. Och, "Minimum error rate training for statistical machine translation," in *Proceedings of the 41st Meeting of the Association for Computational Linguistics (ACL 2003)*, Sapporo, Japan, 2003.

[11] M. Cettolo, J. Niehues, S. Stüker, L. Bentivogli, and M. Federico, "Report on the 10th iwslt evaluation campaign," in *Proceedings of the International Workshop on Spoken Language Translation*, Heidelberg, Germany, December 2013, pp. 29–38.

[12] A. Vaswani, Y. Zhao, V. Fossum, and D. Chiang, "Decoding with large-scale neural language models improves translation," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, 2013, pp. 1387–1392. [Online]. Available: http://aclweb.org/anthology/D/D13/D13-1140.pdf

[13] R. Kneser and H. Ney, "Improved backing-off for m-gram language modeling," in *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, vol. 1. IEEE, 1995, pp. 181–184.

[14] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," in *Proceedings of the 34th annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 1996, pp. 310–318.