NTT Statistical Machine Translation System for IWSLT 2008

Katsuhito Sudoh, Taro Watanabe, Jun Suzuki, Hajime Tsukada, and Hideki Isozaki

NTT Communication Science Laboratories
2-4 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0237, Japan
{sudoh, taro, jun, tsukada, isozaki}@cslab.kecl.ntt.co.jp

Abstract

The NTT Statistical Machine Translation System consists of two primary components: a statistical machine translation decoder and a reranker. The decoder generates k-best translation canditates using a hierarchical phrase-based translation based on synchronous context-free grammar. The decoder employs a linear feature combination among several real-valued scores on translation and language models. The reranker reorders the k-best translation candidates using Ranking SVMs with a large number of sparse features. This paper describes the two components and presents the results for the evaluation campaign of IWSLT 2008.

1. Introduction

This paper presents NTT Statistical Machine Translation (SMT) System evaluated in the evaluation campaign of the International Workshop on Spoken Language Translation (IWSLT) 2008. The system is composed of two steps: First, k-best translation candidates are generated by a hierarchical phrase-based statistical machine translation decoder, with linear feature combination among several scores on translation and language models. Next, these candidates are reordered and the top-best candidate is chosen, according to approximated BLEU. The reranker is based on Ranking SVMs [1] with a large number of sparse binary features.

A large number of sparse features has been successfully applied to SMT in both decoding [2] and reranking [3, 4]. In this year's IWSLT evaluation, we employ Ranking SVMs with fast optimization algorithm [5] for reranking, and introduce three new types of sparse features: alignment-independent word pairs, skip bigrams, and context-dependent features. Since this year's challenge task focuses on translating utterances in dialogues, we incorporate rich information available from the dialogue context into the reranker.

In the evaluation on the IWSLT 2008 Chinese-to-English challenge task, our primary submission achieved 32.12% for ASR 1-best and 38.47% for clean in BLEU. However, in the official evaluation, an SVM hyperparameter was not optimized. We fixed it and finally achieved 37.41% for ASR 1-best and 44.38% for clean in a post-evaluation experiment using source-target word pair and target-side skip bigram features; Our context-dependent features did not effectively work in the experiments, because they failed to capture useful context information in the current condition. We discuss these features using distinctive examples between reranker selections and decoder 1-bests.

This paper is organized as follows: Section 2 briefly describes our SMT decoder. Section 3 describes our reranking component and sparse features for reranking. Section 4 presents the results for the evaluation campaign of IWSLT 2008, followed by discussion in Section 5.

2. Machine Translation Component

2.1. Statistical Machine Translation

We use a linear feature combination approach [6] in which a foreign language sentence f is translated into another language, for example English, e, by seeking a maximum solution:

$$\hat{e} = \underset{e}{\operatorname{argmax}}\, \mathbf{w}^\top \cdot \mathbf{h}(f, e) \tag{1}$$

where $\mathbf{h}(f, e)$ is a feature vector. $\mathbf{w}$ is a weight vector that scales the contribution from each feature. Feature weights (i.e. elements of $\mathbf{w}$) are optimized based on minimum error rate training [6].

2.2. Hierarchical Phrase-based Approach

Our SMT component employs the hierarchical phrase-based approach [7], in which the translation model is based on a stochastic synchronous context-free grammar (SCFG). A translation is generated by hierarchically combining phrases using non-terminals. Each production rule of SCFG takes the following form.

$$X \rightarrow \langle \gamma, \alpha, \sim \rangle \tag{2}$$

In the notation above, X is a non-terminal symbol, γ is a source-side string of terminal and non-terminal symbols, and α is a target-side one. γ and α share the same number of non-terminals whose one-to-one mapping is defined by $\sim$. Such a quasi-syntactic structure can naturally capture the reordering of phrases that is not directly modeled by a conventional phrase-based approach [8]. The non-terminal embedded phrases are learned from a bilingual corpus without a linguistically motivated syntactic structure.

Our decoder and rule extraction procedure is based on Hiero [7]. The decoder is an in-house developed CKY-based

one. Rules in forms of (2) are extracted using phrase pairs obtained by the phrase extraction algorithm [8]. The phrase extraction uses many-to-many word alignment, derived from heuristics on one-to-many word alignment in both directions [9, 10]. Using the extracted phrases, SCFG production rules are accumulated by finding "holes" in extracted contiguous phrases:

- For a phrase pair $(\bar{f}, \bar{e})$, a rule $X \rightarrow \langle \bar{f}, \bar{e} \rangle$ is extracted.
- For a rule $X \rightarrow \langle \gamma, \alpha \rangle$ and a phrase pair $(\bar{f}, \bar{e})$ s.t. $\gamma = \gamma_1 \bar{f} \gamma_2$ and $\alpha = \alpha_1 \bar{e} \alpha_2$, a rule $X \rightarrow \langle \gamma_1 X_{\boxed{k}} \gamma_2, \alpha_1 X_{\boxed{k}} \alpha_2 \rangle$ is extracted.

where boxed indices $\boxed{k}$ indicate one-to-one mapping between non-terminals.

2.3. Decoder features

Features used in our machine translation component are real-valued scores derived from the translation and language models. These features are those used as baseline features in our IWSLT 2006 evaluation [3]:

- Hierarchical phrase translation probabilities
- Lexical translation probabilities in phrase pairs
- Word-based insertion/deletion penalties
- Word 5-gram language model scores
- Reordering penalties
- Length penalties on both words and hierarchical phrases

3. Reranking Component

Our reranking component is based on Ranking SVMs [1]. Each decoder k-best translation candidate is represented by a feature vector, and the reranker chooses the best-scored candidate over k vectors.

3.1. Ranking SVMs

Ranking SVM is a variant of support vector machines (SVMs) for the purpose of ranking samples, not classification. However, its optimization problem is equivalent to that of a classification SVM on pairwise difference vectors (see details in [1]). In our reranking component, we do not define the whole rank order over k-best translation candidates but only distinguish the best candidate among the rest. The best candidate is chosen based on approximated BLEU [3], which will be explained in 3.2. If more than one (k_{top}) candidates has the same value of approximated BLEU, all of them are regarded as the best candidates. In this setting, only $k_{top}(k - k_{top})$ pairwise difference vectors between the best candidates and the rest ones are used as training data.

We employs Pegasos[1], a fast optimization algorithm for linear-kernel SVMs. It uses only k samples to calculate subgradient for optimization, so learning time of Pegasos does *not* depend on the training data size [5].

3.2. Approximated BLEU

We use approximated BLEU [3] to choose the best translation candidates, for optimizing the reranker in terms of BLEU [11]. The approximated BLEU is independently calculated on each translation candidate for each sentence in reranker training data during pre-processing, although original BLEU required document-wise calculation and is not suitable for sentence-level reranking.

Given 1-best translation outputs for T input sentences $O_1^T = \{e_1^1, ..., e_1^T\}$, the approximated BLEU on i-best translation candidate for t-th input sentence e_i^t is calculated by substituting e_1^t with e_i^t, i.e. the BLEU on the sentence set $\{e_1^1, ..., e_1^{t-1}, e_i^t, e_1^{t+1}, ..., e_1^T\}$.

3.3. Reranker features

We use a large number of sparse binary features for reranking, as well as a real-valued feature (decoder score).

3.3.1. Word alignment features

We use source-target word pairs extracted by separately running IBM Model 1 in both direction [4]. In addition to source-target word unigram pairs, we used pairs of target-side word bigram and their corresponding source-side words in terms of the word alignment. We also include POS-based features, target-side word surfaces are replaced with their POS tags in the word alignment features above. Target-side (English) POS tags are automatically annotated by Brill Tagger.

3.3.2. Word pair features

Since the word alignment features highly depends on IBM Model 1 word alignments, the features are influenced by word alignment errors. As alignment independent features, we use all possible word unigram and bigram pairs between source-side words and target-side words. We also use POS-based features in the same way as the word alignment features.

3.3.3. Target-side skip bigram features

We use target-side skip bigrams, which are allowed to skip up to two words, as features. POS-based skip bigrams are also used.

3.3.4. Context-dependent word pair features

To define context-dependent features, we simply use bag-of-words of both source- and target-side words in the *previous*

[1] http://ttic.uchicago.edu/~shai/code/index.html

sentence. Target-side context words are extracted from k-best translation candidates. For each pair of context word in the bag-of-words and *current* target-side word, context word pair feature is defined.

4. Evaluation

We present evaluation results of our system on IWSLT 2008 Chinese-to-English challenge task, for field-experiment data in tourism domain.

4.1. Setup

Training and development data came from only IWSLT supplied data for Chinese-to-English challenge task shown in Table 1, and no extra data resources were used. Chinese sentences are re-tokenized by our in-house developed tool [12].

For estimating feature weights in Eq.(1) by minimum error rate training, we used the development sets 3, 6, and CT_CE with an intermediate phrase-table and word 5-gram language model trained using the other training and development sets (1, 2, 4, and 5). Throughout this experiment we share the same feature scaling, instead of re-running minimum error rate training for each different setting. For training Ranking SVMs, we used 100-best outputs for the development set CT_CE from the decoder with the estimated feature weights and the models trained using the other training and development sets (1-6). For the final decoder, we used all supplied data for training the phrase-table and word 5-gram language model. Source-side test set perplexity by the final language model was 56.2, and 13 (0.5%) Chinese words were not found in the vocabulary. All word 5-gram language models above were trained by SRILM with modified Kneser-Ney smoothing.

We compared four methods with varying reranking features.

(1) Using 1-best decoder output and did not apply reranking.

(2) Using the decoder score and word alignment features, as a baseline of reranking.

(3) Adding the word pair and skip bigram features to the baseline features above (primary).

(4) Using all features described in 3.3.

The number of distinct features extracted from reranker training set (DevCT_CE) are presented in Table 2.

Our experiments were conducted on clean and ASR 1-best inputs, without consideration of ASR N-best or word lattice information. Note that our SMT decoder and reranker were trained using only supplied clean text data and ASR transcriptions were not used.

Table 2: The number of distinct reranking features extracted from reranker training set.

Feature type	# features
Word alignment	53,902
(surface unigram)	3,491
(surface bigram)	26,959
(POS unigram)	2,585
(POS bigram)	20,867
Word pair	201,605
(surface unigram)	16,883
(surface bigram)	117,620
(POS unigram)	5,334
(POS bigram)	61,768
Target-side skip bigram	30,803
(surface skip bigram)	28,496
(POS skip bigram)	2,307
Context word pair	87,653
(with source-side context)	19,122
(with target-side context)	68,531
(2) Decoder score + alignment	53,903
(3) + word pair + skip bigram	286,311
(4) + context	373,964

Table 3: Averaged cross-validation results in BLEU (%) between Dev4 and Dev5.

Reranking Features	BLEU
(1) No reranking	24.64
(2) Decoder score + alignment	25.73
(3) + word pair + skip bigram	25.98
(4) + context	25.36
100-best oracle	46.58

4.2. Pre-evaluation results

Our reranker was tuned based on cross-validation between the development sets 4 and 5 (from IWSLT 2006) [2]. The translation and language models for the cross-validation were trained on the training and development sets 1,2,3, and 6. In this condition, the reranker achieved better BLEU than decoder 1-bests as shown in the cross-validation results on Table 3. The results with context features were slightly worse than those without them in BLEU, so that we abandoned these features in our primary setting.

4.3. Official results

Our results in BLEU [11] are presented in Table 4. We achieved 35.62 % for ASR 1-best input and 42.16 % for clean input without reranking. Our reranking approach showed worse results than decoder 1-bests; the primary results were

[2]These data preserve utterance order and are suit for using context-dependent features.

Table 1: Bilingual data statistics. "ppl." mean test set perplexities on DevCT_CE and Test (clean) by word 5-gram language model trained on each dataset.

		Train	Dev1	Dev2	Dev3	Dev4	Dev5	Dev6	DevCT_CE	Test (clean)
Chinese	# sentences	19,972	506	500	506	489	500	489	246	504
	# words	159,507	3,152	3,215	3,494	5,592	5,979	2,926	1,369	2,680
	ppl.(DevCT_CE)	74.2	59.7	70.1	71.2	107.9	102.6	64.5	-	30.8
	ppl.(Test)	76.7	50.6	52.4	59.4	88.2	85.4	55.8	29.2	-
English	# sentences	19,972	8,096	8,000	8,096	3,423	3,500	2,934	1,722	n/a
	# words	191,596	68,181	67,850	68,782	47,570	52,587	23,423	14,539	n/a
	ppl.(DevCT_CE)	23.6	25.8	28.6	27.5	56.2	61.7	30.5	-	n/a

Table 4: Official automatic evaluation results in BLEU (%), with casing and punctuations. The primary submissions are indicated by †.

Input	Reranking features	BLEU
ASR 1-best	(1) No reranking	35.62
	(2) Decoder score + alignment	35.12
	(3) + word pair + skip bigram	32.12†
	(4) + context	31.19
Clean	(1) No reranking	42.16
	(2) Decoder score + alignment	40.01
	(3) + word pair + skip bigram	38.47†
	(4) + context	37.10

Table 5: Post-evaluation results in BLEU (%), with casing and punctuations. (The Pegasos's hyperparameter λ was optimized through cross-validation on DevCT_CE.)

Input	Reranking features	BLEU
ASR 1-best	(1) No reranking	35.62
	(2) Decoder score + alignment	37.40
	(3) + word pair + skip bigram	37.41
	(4) + context	36.87
Clean	(1) No reranking	42.16
	(2) Decoder score + alignment	44.13
	(3) + word pair + skip bigram	44.38
	(4) + context	42.62

3.5% worse in BLEU.

4.4. Post-evaluation results

In the official evaluation, a Pegasos's regularization hyperparameter λ ($\lambda = 1/mC$, where m is the number of training samples and C is the soft margin parameter) was not optimized; we used its default value $\lambda = 0.01$. Although we could increase BLEU with the default value in the pre-evaluation experiment, it turned to be inappropriate in the official evaluation.

We conducted another post-evaluation experiment with the hyperparameter λ that was optimized to maximize BLEU averaged over two- and three-fold cross-validation [3] on DevCT_CE. The post-evaluation results in BLEU are presented in Table 5. We achieved 37.41 % for ASR 1-best input and 44.38 % for clean input with our primary setting. Thus, our reranking approach worked well with an appropriate SVM hyperparameter.

5. Discussion

The post-evaluation results show that our reranking approach can improve translation performance in BLEU. To investigate how the reranker chose better translation candidate, we focus on the difference between the features appeared in decoder 1-bests and reranker selections in the cross-validation on DevCT_CE. Figures 1, 2, and 3 shows examples of distinctive features of reranker selections compared to decoder 1-bests, by the rerankers (2), (3), and (4), respectively.

These examples suggest the following:

- Word alignment features (in Fig. 1) captured lexical correspondence and the reranker (2) chose better translation candidates than decoder 1-bests in terms of *adequacy*.
- Bigram and skip bigram features captured target-side natural word order and bigram pairs captured their source-target co-occurence, and therefore the reranker (3) chose slightly better translation candidates than the reranker (2) in terms of *fluency*.
- Context features turned out to capture many general word co-occurence and the reranker (4) failed to distinguish better translation candidate from others.

Our current context-dependent features are simply defined without dialogue boundaries and may not be useful for capturing dialogue context. Since utterances in a dialogue are considered to correlate with each other, we need further feature engineering to incorporate such correlations into rerank-

[3] The sentences were grouped based on their dialogue IDs and the groups were splitted equally into the (two or three) cross-validation datasets.

ing. Another problem may be that the data consist of only one-sided utterances and do not sufficiently hold *dialogue* information.

6. Conclusion

We evaluated the NTT Statistical Machine Translation System for the evaluation campaign of IWSLT 2008. The system is composed of decoding and reranking components. The decoder is based on the hierarchical phrase-based approach and the linear feature combination. The reranker employs Ranking SVMs and a large number of sparse features of alignment-independent word pairs, skip bigrams, and context-dependent word pairs. Experimental results show that our reranker effectively works for choosing better translation candidates than decoder 1-bests. Our future work involves more effective sparse features, especially context-dependent ones.

7. Acknowledgements

We would like to thank the IWSLT 2008 organizing committee for their efforts to coordinate the evaluation campaign. This work is partly supported by MEXT Grant-Aid for Scientific Research of Priority Areas.

8. References

[1] T. Joachims, "Optimizing search engines using clickthrough data," in *Proc. ACM SIGKDD*, 2002, pp. 133–142.

[2] T. Watanabe, J. Suzuki, H. Tsukada, and H. Isozaki, "Online large-margin training for statistical machine translation," in *Proc. EMNLP-CoNLL*, 2007, pp. 764–773.

[3] ——, "NTT statistical machine translation for IWSLT 2006," in *Proc. IWSLT*, 2006, pp. 95–102.

[4] T. Watanabe, J. Suzuki, K. Sudoh, H. Tsukada, and H. Isozaki, "Larger feature set approach for machine translation in IWSLT 2007," in *Proc. IWSLT*, 2007.

[5] S. Shalev-Shwartz, Y. Singer, and N. Srebro, "Pegasos: Primal Estimated sub-GrAdient SOlver for SVM," in *Proc. ICML*, 2007, pp. 807–814.

[6] F. J. Och, "Minimum error rate training in statistical machine translation," in *Proc. ACL*, 2003, pp. 160–167.

[7] D. Chiang, "Hierarchical phrase-based translation," *Computational Linguistics*, vol. 33, no. 2, pp. 201–228, 2007.

[8] P. Koehn, F. J. Och, and D. Marcu, "Statistical phrase-based translation," in *Proc. NAACL*, 2003, pp. 263–270.

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

[10] ——, "The alignment template approach to statistical machine translation," *Computational Linguistics*, vol. 30, no. 4, pp. 417–449, 2004.

[11] K. Papineni, S. Roukos, T. Ward, and W. Jing Zhu, "BLEU: a method for automatic evaluation of machine translation," in *Proc. ACL*, 2002, pp. 311–318.

[12] K. Saito and M. Nagata, "Multi-language named entity recognition system based on HMM," in *Proc. ACL Workshop on Multilingual and Mixed-Language Named Entity Recognition*, 2003, pp. 41–48.

ST: ?-<EOS> / 吗　　TS: 吗-<EOS> / <$.$>
ST: 可以 / can　　TS: ? / 吗
ST: tell-me / 请问　　TS: 吗-<EOS> / ?
ST: i-would / 我-想　　TS: 我-想 / i*like
ST: would-like / 想　　TS: 在-哪里 / where
ST: you-have / 有　　TS: 最近-的 / nearest^the
ST: <BOS>-i / 我

Figure 1: Examples of distinctive features in the sentences chosen by the reranker (2) on the cross-validation over DevCT_CE. "ST" means source-to-target direction and "TS" means target-to-source direction.

Bigram: ?-<EOS>
Bigram: .-<EOS>
Bigram: me-the
BigramPair: <BOS>-我 / <BOS>-i
BigramPair: <BOS>-我 / would-like
BigramPair: 吗-<EOS> / <BOS>-can
BigramPair :吗-<EOS> / ?-<EOS>
BigramPair: 多少-钱 / how-much
BigramPair: 多少-钱 / ?-<EOS>
BigramPair: <BOS>-能 / <BOS>-can
BigramPair: 给-我 / give-me
SkipBigram: would-*-to
SkipBigram: <BOS>-*-would
SkipBigram: <BOS>-*-can
SkipBigram: do-*-have
SkipBigram: tell-*-the

Figure 2: Examples of distinctive features in the sentences chosen by the reranker (3) on the cross-validation over DevCT_CE.

TargetContext: for -> ?
TargetContext: is -> ?
TargetContext: a -> you
TargetContext: i -> you
TargetContext: . -> is
TargetContext: ? -> can
TargetContext: please -> ?
TargetContext: , -> can
SourceContext: 的 -> ?
SourceContext: 一 -> me
SourceContext: 吗 -> me
SourceContext: 我 -> .

Figure 3: Examples of distinctive features in the sentences chosen by the reranker (4) on the cross-validation over DevCT_CE.