NTT Statistical Machine Translation System for IWSLT 2010

Katsuhito Sudoh, Kevin Duh, Hajime Tsukada

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

Abstract

In this year's IWSLT evaluation campaign (TALK task), we applied three adaptation techniques: (1) training data selection based on information retrieval approach, (2) sub-sentence segmentation, and (3) language model adaptation using source-side of the test set. We also applied a sequential labeling method based on conditional random fields for restoring punctuation markers in the ASR input condition. We present and discuss these techniques in this paper, based on the automatic evaluation results.

1. Introduction

Recent advances in statistical machine translation (SMT) research are based on large-scale bilingual and monolingual language resources in the target domains (mainly news and parliament). On the other hand, for other resource-poor domains, building a good SMT system using only the limited language resources is not easy. Then the problem of *domain adaptation* arises: can *out-of-domain* large-scale language resources compensate for limited *in-domain* resources?

This year's IWSLT TALK task supplied a limited in-domain resource (less than 0.9M tokens) compared to others (260M tokens total). In such a condition, a simplest approach that uses all corpora equally for the translation model may introduce some biases towards out-of-domain vocabulary and linguistic expression. Thus, we tried three adaptation approach in a standard phrase-based SMT framework:

(1) training data selection for the translation model based on information retrieval (IR) [1]

(2) bilingual sentence segmentation into (pseudo-)caption units

(3) language model adaptation using source-side of the test set [2].

An important aspect in IWSLT evaluation is the translation of automatic speech recognition (ASR) results. It introduces two problems to SMT: ASR errors and the lack of punctuation. The first problem is one of the most challenging problem in MT and other spoken language processing applications; ASR errors usually bring more errors in the language processing pipeline. A common approach for the ASR error problem is the use of multiple ASR candidates in n-bests or lattices. The second problem is also important in MT, because MT outputs should be human-readable and therefore require appropriate punctuation. Our system this year tackles the second problem of punctuation restoration.

The remainder of this paper presents and discusses our approaches in detail.

2. Domain Adaptation

2.1. IR-based training data selection

Out-of-domain resources are expected to include domain-independent vocabulary and linguistic expressions that can be useful to compensate for an insufficient amount of resources in the target domain. However, using much larger amount of out-of-domain data might be harmful to domain-specific translation. We then selectively use the out-of-domain data according to their similarity to the target domain, by the following IR-based method [1].

First we collect all n-grams in the in-domain bitext for both sides of the language pair. For each language, a hash table is created where the key represents the n-gram and the value represents the count of this ngram in the training data. Then, an out-of-domain sentence is selected if it contains an ngram in this hash, and the hash value is decremented. We no longer retrieve matches if the hash value becomes zero, similar to the Joshua subsampling technique [3]. The rationale for this is to have a balanced coverage of ngrams. This procedure is performed independently for each language side, and the union of the selected sentences forms the IR bitext (i.e. we take the sentence pair if at least one side is retrieved). Note this method differs from the original IR approach [4] in two important aspects: (1) we sample based on training not test data, and (2) this allows us to sample on both sides of the language pair rather than just the source side. We think working with the training data is a practically efficient solution and allows for new IR approaches.

2.2. Bilingual Sub-sentence Segmentation

Another possible problem in the talk translation is a mismatch in transcript style: transcripts of TED talks are segmented into sub-sentence units (captions) and each line in TED transcripts is not a complete sentence. Indeed, the

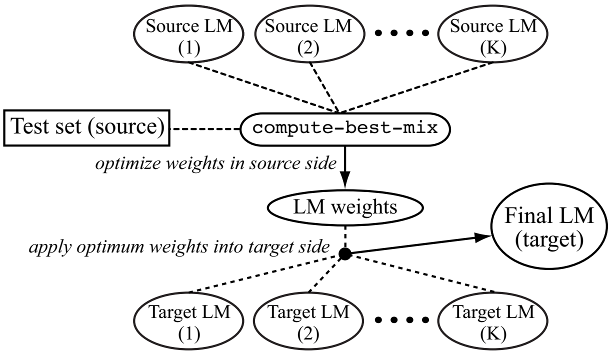

Figure 1: LM weight optimization using source side.

number of tokens in each aligned line is 29 on average in the supplied out-of-domain bitext, but it is 10 in the in-domain bitext. We try to fill this gap by segmenting out-of-domain transcripts into similar sub-sentence units based on in-domain resources.

First we restore sentences from in-domain sub-sentences, based on punctuations: periods (.) and question marks (?). Then we train a linear support vector machine (SVM)[1] classifying sub-sentence boundaries in both language sides, using length-based, word-based, and class-based features. Here, there are too many sub-sentence boundary candidates, so we limit them to ones which satisfy a constraint same as Moses phrase extraction constraint over `grow-diag-final` word alignment. By this constraint, we only consider small number of boundary candidates that are consistent with word alignment. The length-based features consist of lengths of current sub-sentence units and original sentences in both language sides. The word- and class-based features are unigrams and bigrams of words and word classes around the boundary candidate. The word classes are detemined by `mkcls`.

2.3. LM Mixture Weight Optimization towards Test Set

The third problem that we focus on is the language model (LM). A common method for LM adaptation is weighted interpolation of multiple LMs from different resources. The weights are determined by the perplexity on some held-out data in the target language, or, in recent SMT framework, they can be optimized through system-wide optimization like minimum error rate training (MERT). In the last year's IWSLT evaluation, the FBK team optimized the weights of the *target-side* LMs using the *source side* input sentences with promising results [2]. Although their LM weight optimization was sentence-by-sentence, we simply determine one set of LM weights for the whole test set according to the test set perplexity, using SRILM's `compute-best-mix`. The procedure is illustrated in Figure 1.

Table 1: Features for punctuation restoration. l_i is a punctuation labels for the i-th word w_i and c_i is the word class of w_i.

word-based feature	class-based feature
(w_i, l_i)	(c_i, l_i)
(w_{i+1}, l_i)	(c_{i+1}, l_i)
(w_i, w_{i+1}, l_i)	(c_i, c_{i+1}, l_i)
(w_{i-1}, w_i, l_i)	(c_{i-1}, c_i, l_i)
$(w_{i-2}, w_{i-1}, w_i, l_i)$	$(c_{i-2}, c_{i-1}, c_i, l_i)$
$(w_{i-1}, w_i, w_{i+1}, l_i)$	$(c_{i-1}, c_i, c_{i+1}, l_i)$
$(w_i, w_{i+1}, w_{i+2}, l_i)$	$(c_i, c_{i+1}, c_{i+2}, l_i)$
$(w_{i-2}, w_{i-1}, w_i, w_{i+1}, l_i)$	$(c_{i-2}, c_{i-1}, c_i, c_{i+1}, l_i)$
$(w_{i-1}, w_i, w_{i+1}, w_{i+2}, l_i)$	$(c_{i-1}, c_i, c_{i+1}, c_{i+2}, l_i)$
	$(c_{i-2}, c_{i-1}, c_i, c_{i+1}, c_{i+2}, l_i)$

3. Punctuation Restoration

An important problem on this year's task is punctuation restoration for the ASR inputs. We tackle the problem as a sequential labeling problem like part-of-speech tagging and sentence boundary detection, using conditional random fields (CRFs)[2] [5, 6]. We define four punctuation labels for ASR 1-best words: PERIOD (.), COMMA (,), QUESTION (?), and NO_PUNC (no punctuation marks), which represent the punctuation mark following the words. Features used on the i-th word w_i with its label l_i are listed in Table 1, where c_i represents the word class of w_i determined by `mkcls`. Once the labeling model is trained, we can restore punctuation marks in ASR 1-best results by applying the model.

Lu and Ng [7] proposed more sophisticated CRF-based punctuation restoration [3]. Our method is almost the same as their baseline by linear-chain CRFs, with a small difference in features.

4. Experiment

4.1. Resources

4.1.1. In-domain and Out-of-domain Corpora

We used TED (in-domain), Europarl, UN, and News Commentary (out-of-domain, hereafter *OOD*) corpora for training[4]. For the out-of-domain corpora, we applied IR-based sentence selection described in 2.1 and retrieved 364,330 sentences (hereafter *IR*). IR was further segmented into sub-sentence units by the segmenter described in 2.2, and 899,502 sub-sentences were obtained (hereafter *IR-seg*). Corpus statistics are summarized in Table 2, and the data flow is illustrated in Figure 2.

[1] We used LIBOCAS (http://cmp.felk.cvut.cz/˜xfrancv/ocas/html/).

[2] We used CRF++ (http://crfpp.sourceforge.net/).

[3] We independently developed the CRF-based method prior to its publication.

[4] We found some problematic sentences in 10^9 corpus, so we completely omitted it.

Table 2: Corpus statistics.

Corpus	#sentences	#tokens (En)	#tokens (Fr)
In-domain			
TED	83,923	841,107	893,381
Out-of-domain			
OOD	9,041,376	262,633,459	300,994,876
(Europarl)	1,726,535	47,955,610	52,722,393
(UN)	7,230,217	212,612,925	245,848,393
(News)	84,624	2,064,924	2,424,090
IR	364,330	8,561,030	9,466,005
IR-seg	899,502	8,561,030	9,466,005

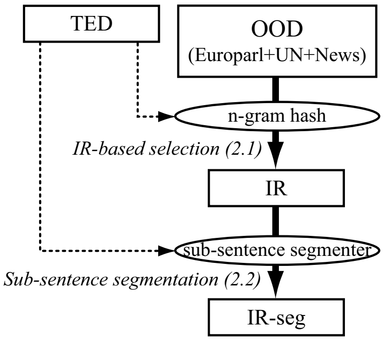

Figure 2: Data flow in the experiment.

4.1.2. Preprocessing

We used `tokenizer.perl` in the WMT baseline system[5], with a slight modification for quotation marks[6]. To avoid underflow in word alignment, long sentence pairs whose number of tokens in either language side exceeded 64 were filtered out.

4.2. Baseline System

Our baseline system almost followed the WMT baseline system, using the corpora above. The word alignment was estimated by mgizapp-0.6.3 with grow-diag-final-and heuristics or berkeleyaligner_unsupervised-2.1 (experimentally chosen according to the cross-validation performance below). The LMs were word 4-gram models trained using SRILM-1.5.9: four corpus-wise LMs from TED, Europarl, UN, and News. The decoder was moses-2010-04-26 and its parameters were optimized to the development set (only Reference, no ASR) by `mert-moses-new.pl`.

4.3. Results

4.3.1. Development Cross Validation Results and Selected Configurations

We tested various combinations of the adaptation approach described so far in our development phase; The supplied development set was divided into two non-overlapping sets[7] and used for cross validation.

Table 3 shows the cross validation results of several configurations, varying i) training data of phrase table, ii) target for LM mixture weight optimization, and iii) word alignment method. Metrics are BLEU, TER, and BLEU - TER + 1 (B-T+1). We chose the best one as our primary configuration, and three contrastive configurations to the primary.

4.3.2. Results

Official automatic evaluation results for our primary and contrastive runs and additional results in BLEU, TER, and B-T+1 are shown in Table 4. The evaluation on ASR was based on automatic sentence segmentation tool supplied by the organizers [8]. Our primary run was the best among our submitted runs but not among all methods compared in 4.3.1.

Through our significance test [9], differences among most methods were not significant – significant differences were observed only between the methods with out-of-domain data (OOD, IR, and IR-seg) and those with only TED data.

4.3.3. Evaluation of Punctuation Restoration

To investigate the effect of our punctuation restoration, we conducted additional experiment using original ASR results (i.e., without punctuation) as decoder inputs. We compared the BLEU scores in three different conditions: 1) *case+punc*: with true casing and punctuation, 2) *case+no_punc*: with true casing but without punctuation, 3) *no_case+no_punc*: without casing and punctuation.

The results are shown in Table 5. The *case+punc* results show our method successfully restored correct punctuation markers. Even in punctuation insensitive evaluation (*case+no_punc* and *no_case+no_punc*), our method achieved significant improvement in TER. On the other hand, the improvement in BLEU was not significant in case-insensitive conditions. These differences may come from *shift* errors – shift errors were reduced by the punctuation restoration (0.107 in *case+no_punc* and 0.115 in *no_case+no_punc*) from those without punctuation (0.116 and 0.126, respectively). This suggests punctuation markers are useful to reduce shift errors (i.e., constraining reordering) in SMT, and restoring punctuation markers in ASR results is important in spoken language translation.

[5] http://www.statmt.org/wmt10/baseline.html

[6] `tokenizer.perl` replaces ‘ (left quotation mark) with ’ (right quotation mark) but we didn’t change them.

[7] One was composed of *docids* 69, 129, 453, and 531 (644 sentences), and the other was composed of 326, 327, 328, 332, 335, 340, and 359 (663 sentences). They were chosen to balance the number of sentences between sets.

Table 3: 2-fold cross validation results in BLEU, TER, and BLEU-TER+1 (B-T+1) on supplied development set (with true casing and punctuation). Scores in **bold** represent the best ones. Rows in gray represents the methods tested after the official evaluation period.

Run	phrase table training data	LM mixture weight target	word alignment	Reference (cross validation)		
				BLEU	TER	B-T+1
primary	TED+IR	test (source)	berkeley	0.2540	0.6548	0.5992
	TED+IR	test (source)	mgiza	0.2504	0.6548	0.5956
contrastive-2	TED+IR	dev (target)	berkeley	0.2482	0.6611	0.5871
	TED+IR	dev (target)	mgiza	0.2482	0.6603	0.5879
	TED+IR	MERT	berkeley	0.2504	0.6591	0.5913
	TED+IR	MERT	mgiza	0.2410	0.6694	0.5716
	TED+OOD	test (source)	berkeley	**0.2564**	**0.6544**	**0.6020**
contrastive-1	TED+OOD	test (source)	mgiza	0.2561	0.6582	0.5979
	TED+OOD	dev (target)	berkeley	0.2554	0.6556	0.5998
	TED+OOD	dev (target)	mgiza	0.2539	0.6570	0.5969
	TED+OOD	MERT	berkeley	0.2494	0.6594	0.5900
	TED+OOD	MERT	mgiza	0.2521	0.6637	0.5884
	TED+IR-seg	test (source)	berkeley	0.2445	0.6615	0.5830
contrastive-3	TED+IR-seg	test (source)	mgiza	0.2466	0.6602	0.5864
	TED+IR-seg	dev (target)	berkeley	0.2492	0.6566	0.5926
	TED+IR-seg	dev (target)	mgiza	0.2479	0.6624	0.5855
	TED+IR-seg	MERT	berkeley	0.2478	0.6630	0.5848
	TED+IR-seg	MERT	mgiza	0.2426	0.6636	0.5790
	TED	test (source)	berkeley	0.2407	0.6698	0.5709
	TED	test (source)	mgiza	0.2351	0.6761	0.5590
	TED	dev (target)	berkeley	0.2426	0.6637	0.5789
	TED	dev (target)	mgiza	0.2310	0.6776	0.5534
	TED	MERT	berkeley	0.2324	0.6761	0.5563
	TED	MERT	mgiza	0.2300	0.6832	0.5468

Table 4: Official and additional automatic evaluation results in BLEU and TER, and BLEU-TER+1 (B-T+1) (with true casing and punctuation). Scores in **bold** represent the best ones. Rows in gray represents the methods tested after the official evaluation period.

Run	phrase table training data	LM weight mixture target	word alignment	Reference			ASR 1-best		
				BLEU	TER	B-T+1	BLEU	TER	B-T+1
primary	TED+IR	test (source)	berkeley	0.2598	0.5784	0.6814	0.1623	0.6923	0.4700
	TED+IR	test (source)	mgiza	0.2572	0.5765	0.6807	0.1603	0.6905	0.4698
contrastive-2	TED+IR	dev (target)	berkeley	0.2594	0.5788	0.6806	0.1621	0.6922	0.4699
	TED+IR	dev (target)	mgiza	0.2623	0.5749	**0.6874**	0.1617	**0.6899**	**0.4718**
	TED+IR	MERT	berkeley	0.2599	0.5866	0.6733	**0.1645**	0.6967	0.4678
	TED+IR	MERT	mgiza	0.2541	0.5826	0.6715	0.1587	0.6945	0.4642
	TED+OOD	test (source)	berkeley	0.2541	0.5832	0.6709	0.1603	0.6939	0.4664
contrastive-1	TED+OOD	test (source)	mgiza	0.2504	0.5817	0.6687	0.1591	0.6931	0.4660
	TED+OOD	dev (target)	berkeley	0.2551	0.5853	0.6698	0.1611	0.6937	0.4674
	TED+OOD	dev (target)	mgiza	0.2537	0.5796	0.6741	0.1592	0.6937	0.4655
	TED+OOD	MERT	berkeley	0.2464	0.5998	0.6466	0.1575	0.7030	0.4545
	TED+OOD	MERT	mgiza	0.2587	0.5938	0.6649	0.1636	0.7071	0.4565
	TED+IR-seg	test (source)	berkeley	**0.2633**	0.5770	0.6863	0.1627	0.6929	0.4698
contrastive-3	TED+IR-seg	test (source)	mgiza	0.2562	0.5788	0.6774	0.1612	0.6933	0.4679
	TED+IR-seg	dev (target)	berkeley	0.2597	**0.5748**	0.6849	0.1608	0.6920	0.4688
	TED+IR-seg	dev (target)	mgiza	0.2615	0.5758	0.6857	0.1619	0.6925	0.4694
	TED+IR-seg	MERT	berkeley	0.2539	0.5844	0.6695	0.1598	0.6964	0.4634
	TED+IR-seg	MERT	mgiza	0.2530	0.5843	0.6687	0.1559	0.6980	0.4579
	TED	test (source)	berkeley	0.2515	0.5876	0.6639	0.1555	0.7012	0.4543
	TED	test (source)	mgiza	0.2464	0.5921	0.6543	0.1538	0.7017	0.4521
	TED	dev (target)	berkeley	0.2513	0.5945	0.6568	0.1556	0.7066	0.4490
	TED	dev (target)	mgiza	0.2480	0.5954	0.6526	0.1553	0.7051	0.4502
	TED	MERT	berkeley	0.2482	0.5913	0.6569	0.1535	0.7027	0.4508
	TED	MERT	mgiza	0.2295	0.6011	0.6284	0.1414	0.7066	0.4348

Table 5: Results with and without punctuation restoration (PR) by primary configuration. Results are from three conditions: *case+punc*: with true casing and punctuation, *case+no_punc*: with true casing but without punctuation, *no_case+no_punc*: without casing and punctuation. The results in **bold** are statistically significant (**: $p < .01$, *: $p < .05$).

	case+punc			case+no_punc			no_case+no_punc		
	BLEU	TER	B-T+1	BLEU	TER	B-T+1	BLEU	TER	B-T+1
w/ PR	**0.1623****	**0.6923***	**0.4700****	**0.1735**	**0.7243****	**0.4492**	**0.1850**	**0.7059****	**0.4791***
w/o PR	0.1428	0.7029	0.4399	0.1719	0.7407	0.4312	0.1802	0.7270	0.4532

5. Discussion

Among three adaptation approaches, the performance of IR-based adaptation (TED+IR) was comparable with the use of all out-of-domain data (TED+OOD), using only 4% of bitexts. We expected the IR-based adaptation worked better, but could not achieve significant improvements. The problem is further discussed in our technical paper [1].

The LM weight optimization toward test set was a bit better than the others in development, but did not in test. It may be related to data similarity among the data sets but is not clear in current results.

The sub-sentence segmentation did not work in most cases, while we expected that it can help to reduce word alignment ambiguity in training translation models by shortening sentences (as suggested in [10]). One possibility we think is that our sub-sentence segmentation was naive and not sufficient for decreasing alignment ambiguity. Another possibility is that word alignment is easier between English and French, compared to distant languages like English and Japanese.

In summary, the results suggests our current phrase table and LM adaptation methods do not clearly work and need further studies in various conditions.

6. Conclusion

We applied three adaptation methods to this year's TALK task. They sometimes work, but sometimes do not. For more effective adaptation methods, we need further studies on how adaptation works, from detailed analyses such as the comparison among corpus-, document-, and sentence-wise adaptation.

7. Acknowledgment

We would like to thank the evaluation committee for their efforts in the evaluation campaign. We also thank two anonymous reviewers for their helpful comments.

8. References

[1] K. Duh, K. Sudoh, and H. Tsukada, "Analysis of translation model adaptation in statistical machine translation," in *Proc. IWSLT 2010*.

[2] N. Bertoldi, A. Bisazza, M. Cettolo, G. Sanchis-Trilles, and M. Federico, "FBK @ IWSLT-2009," in *Proc. IWSLT 2009*, pp. 37–44.

[3] Z. Li, C. Callison-Burch, C. Dyer, S. Khudanpur, L. Schwartz, W. Thornton, J. Weese, and O. Zaidan, "Joshua: An open source toolkit for parsing-based machine translation," in *Proc. WMT 2009*, pp. 135–139.

[4] A. S. Hildebrand, M. Eck, S. Vogel, and A. Waibel, "Adaptation of the translation model for statistical machine translation based on information retrieval," in *Proc. EAMT 2005*, pp. 135–142.

[5] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. ICML*, 2001, pp. 282–289.

[6] Y. Liu, A. Stolcke, E. Shriberg, and M. Harper, "Using conditional random fields for sentence boundary detection in speech," in *Proc. ACL*, 2005, pp. 451–458.

[7] W. Lu and H. T. Ng, "Better punctuation prediction with dynamic conditional random fields," in *Proc. EMNLP 2010*, October, pp. 177–186.

[8] E. Matusov, G. Leusch, O. Bender, and H. Ney, "Evaluating machine translation output with automatic sentence segmentation," in *Proc. IWSLT 2005*.

[9] Y. Zhang, S. Vogel, and A. Waibel, "Interpreting BLEU/NIST scores: How much improvement do we need to have a better system?" in *Proc. LREC 2004*, pp. 2051–2054.

[10] K. Sudoh, K. Duh, H. Tsukada, T. Hirao, and M. Nagata, "Divide and translate: Improving long distance rerordering in statistical machine translation," in *Proc. WMT-MetricsMATR*, 2010.