NICT-2 Translation System for WAT2016: Applying Domain Adaptation to Phrase-based Statistical Machine Translation

Kenji Imamura and **Eiichiro Sumita**
National Institute of Information and Communications Technology
3-5 Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0289, Japan
{kenji.imamura, eiichiro.sumita}@nict.go.jp

Abstract

This paper describes the NICT-2 translation system for the 3rd Workshop on Asian Translation. The proposed system employs a domain adaptation method based on feature augmentation. We regarded the Japan Patent Office Corpus as a mixture of four domain corpora and improved the translation quality of each domain. In addition, we incorporated language models constructed from Google n-grams as external knowledge. Our domain adaptation method can naturally incorporate such external knowledge that contributes to translation quality.

1 Introduction

In this paper, we describe the NICT-2 translation system for the 3rd Workshop on Asian Translation (WAT2016) (Nakazawa et al., 2016a). The proposed system employs Imamura and Sumita (2016)'s domain adaptation technique, which improves translation quality using other domain data when the target domain data is insufficient. The method employed in this paper assumes multiple domains and improves the quality inside the domains (cf., Section 2).

For WAT2016, the Japan Patent Office (JPO) Corpus can be regarded as multi-domain data because it includes chemistry, electricity, machine, and physics patents with their domain ID, and thus it is suitable for observing the effects of domain adaptation. WAT2016 provides the JPO corpora in Japanese and English (Ja-En), Japanese and Chinese (Ja-Zh), and Japanese and Korean (Ja-Ko) pairs. We used Ja-En and Ja-Zh pairs in order to add Asian Scientific Paper Experts Corpus (ASPEC) (Nakazawa et al., 2016b) as a fifth domain.[1] The relationship between the corpora and domains used in this paper is shown in Table 1.

		# of Sentences (Ja-En pair)			# of Sentences (Ja-Zh pair)		
Corpus	Domain	Training	Development	Test	Training	Development	Test
JPC	Chemistry	250,000	500	500	250,000	500	500
	Electricity	250,000	500	500	250,000	500	500
	Machine	250,000	500	500	250,000	500	500
	Physics	250,000	500	500	250,000	500	500
ASPEC	ASPEC	1,000,000	1,790	1,812	672,315	2,090	2,107

Table 1: Bilingual Corpora and Domains

The remainder of this paper is organized as follows. Section 2 briefly reviews our domain adaptation. Section 3 describes the proposed translation system, including preprocessing, training, and translation. Section 4 explains experimental results focusing on the effects of domain adaptation.

[1] The ASPEC corpus is provided in Ja-En and Ja-Zh pairs.

Proceedings of the 3rd Workshop on Asian Translation,
pages 126–132, Osaka, Japan, December 11-17 2016.

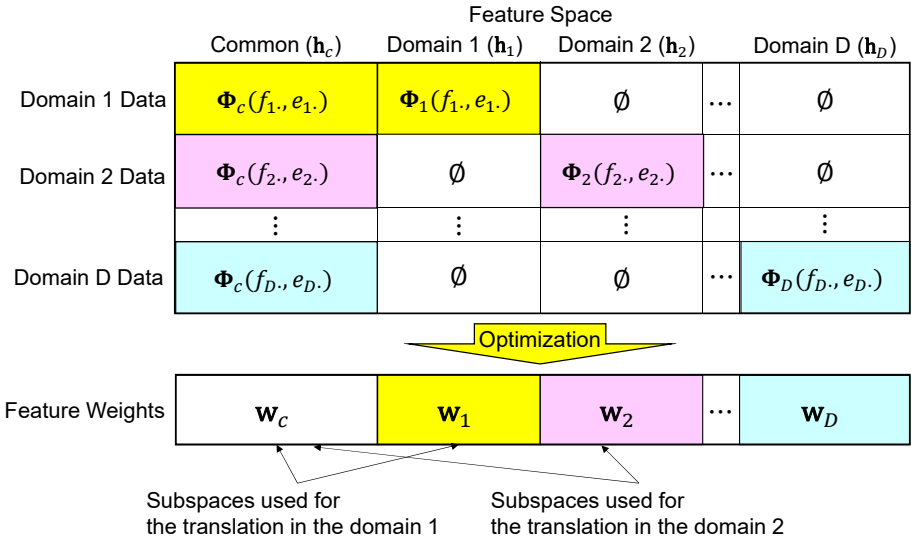

Figure 1: Structure of Augmented Feature Space; $\mathbf{h}_c$ and $\mathbf{h}_i$ denote subvectors of the feature vector $\mathbf{h}(e, f)$. $\mathbf{w}_c$ and $\mathbf{w}_i$ denote subvectors of the weight vector $\mathbf{w}$. $\Phi_c(e, f)$ and $\Phi_i(e, f)$ are feature functions that return feature subvectors (cf., Section 2.2).

2 Domain Adaptation

We used the domain adaptation method proposed by Imamura and Sumita (2016). This method adapts a weight vector by feature augmentation (Daumé, 2007) and a feature vector using a corpus-concatenated model. Since this method only operates in feature space, it can be applied to various translation strategies, such as tree-to-tree translation. In this study, we applied it to phrase-based statistical machine translation (PBSMT) (Koehn et al., 2003; Koehn et al., 2007).

2.1 Adaptation of Weight Vector by Feature Augmentation

Most statistical machine translation employs log-linear models that interpolate feature function values obtained from various submodels, such as phrase tables and language models (LMs). The likelihood of a translation is computed as follows:

$$\log P(e|f) \quad \propto \quad \mathbf{w} \cdot \mathbf{h}(e, f), \tag{1}$$

where $\mathbf{h}(e, f)$ denotes a feature vector and $\mathbf{w}$ denotes its weight vector.

Figure 1 shows a feature space structure of feature augmentation. When we translate texts of D domains, the feature space is segmented into $D + 1$ subspaces: common, domain 1, $\cdots$ domain D. A feature vector (subvector) of each subspace is the same as that of a normal translator, i.e., feature function values obtained from phrase tables and language models.

Features of each translation hypothesis are deployed to different spaces depending on the domain of the input data. For example, features obtained from domain 1 data are deployed to the common and domain 1 spaces. Features obtained from domain 2 data are deployed to the common and domain 2 spaces. In other words, features are always deployed to the common spaces.

We obtain the weight vector $\mathbf{w}$ by optimizing a feature matrix of development data acquired by the above process. This weight vector is optimized to each domain. When we translate test data of domain i, only the subspaces of the common and domain i (i.e., subvectors $\mathbf{w}_c$ and $\mathbf{w}_i$) are used.

2.2 Adaptation of Feature Vector using Corpus-Concatenated Model and Single-Domain Models

Our domain adaptation method adapts the feature function $\mathbf{h}(e, f)$ by changing submodels according to the feature spaces.

- For the common space, where all domain features are located, we use a model trained from a concatenated corpus of all domains (i.e., the corpus-concatenated model) to obtain the features.
- For the domain spaces, where only the domain specific features are located, we use models trained from specific domain data (i.e., single-domain models) to obtain the features.

The procedure is summarized as follows.

1. The training corpora of all domains are concatenated. From this corpus, the corpus-concatenated model is trained. This includes all submodels, such as phrase tables, language models, and lexicalized reordering models. Similarly, the single-domain models are trained from the training corpus of each domain.
2. In feature augmentation, the scores obtained from the corpus-concatenated model are deployed to the common space as the feature function values, while those from the single-domain models are deployed to the domain spaces.

   We represent the augmented feature space as follows:

$$\mathbf{h}(f,e) \quad = \quad \langle \mathbf{h}_c, \mathbf{h}_1, \ldots, \mathbf{h}_i, \ldots, \mathbf{h}_D \rangle, \tag{2}$$

   where $\mathbf{h}_c$ denotes a feature vector of the common space, and $\mathbf{h}_i$ denotes a feature vector of the domain i space. The feature vector $\mathbf{\Phi}_c(f,e)$ obtained from the corpus-concatenated model is always located in the common space. The feature vector $\mathbf{\Phi}_i(f,e)$ is located in the domain-specific space i iff the domain of an input sentence is matched to i.

$$\mathbf{h}_c \quad = \quad \mathbf{\Phi}_c(f,e), \tag{3}$$

$$\mathbf{h}_i \quad = \quad \begin{cases} \mathbf{\Phi}_i(f,e) & \text{if domain}(f) = i \\ \emptyset & \text{otherwise.} \end{cases} \tag{4}$$

3. A feature matrix is obtained by translating a development set, and the weight vector $\mathbf{w}$ is acquired by optimizing the feature matrix.
4. For decoding, phrase pairs are first retrieved from both the corpus-concatenated and single-domain phrase tables. Use of the corpus-concatenated phrase table reduces the number of unknown words because phrase pairs appearing in other domains can be used to generate hypotheses.
5. During search of the best hypothesis, the likelihood of each translation hypothesis is computed using only the common space and domain-specific space of the input sentence.

2.3 Implementation Notices

There are some notices for applying the proposed method to phrase-based statistical machine translation.

Empty Value In the proposed method, several phrases appear in only one of the phrase tables of the corpus-concatenated and single-domain models. The feature functions are expected to return appropriate values for these phrases. We refer to these as empty values.

Even though an empty value is a type of unknown probability and should be computed from the probability distribution of the phrases, we treat it as a hyper-parameter. In other words, an empty value was set experimentally to maximize the BLEU score of a development corpus. Since the BLEU scores were almost stable between -5 and -10 in our preliminary experiments, we used -7 for all settings. If this value is regarded as a probability, it is $\exp(-7) \approx 0.0009$.

Very Large Monolingual Corpora In machine translation, monolingual corpora are easier to obtain than bilingual corpora. Therefore, language models are sometimes constructed from very large monolingual corpora. They can be regarded as corpus-concatenated models that contain various domains. When we introduce models constructed from external knowledge, they are located in the common space while increasing the dimension. We introduce language models constructed from Google n-grams in Section 4.

		Japanese	English	Chinese
Preprocessing	Character Normalization	NFKC Normalization of Unicode		
	Tokenizer	MeCab	Moses Toolkit	Stanford Segmenter
	TrueCaser	-	Moses Toolkit	-
	PreOrderer	(1) Top-Down BTG (2) Developed by NICT, for Patents (w/ Berkeley Parser)		
Training	Phrase Tables	The same as the baseline system of WAT2016.		
	Lex. Reordering Models	The same as the baseline system of WAT2016.		
	Language Models	(1) 5-gram model built from the target side of the bilingual corpora.		
		(2) Google n-gram	(2) Google n-gram	-
	Optimization	K-Best Batch MIRA		
Translation	Decoder	Clone of Moses Decoder		
	DeTrueCaser	-	Moses Toolkit	-
	DeTokenizer	-	Moses Toolkit	-

Table 2: Summary of Preprocessing, Training, and Translation

Optimization Imamura and Sumita (2016) proposed joint optimization and independent optimization. We employ independent optimization, which can use existing optimizers.

3 System Description

In this section, we describe the preprocessing, training, and translation components of the proposed system (Table 2).

3.1 Preprocessing

Preprocessing is nearly the same as the baseline system provided by the WAT2016 committee. However, preorderers are added because our system is phrase-based with preordering. We used Nakagawa (2015)'s Top-Down Bracketing Transduction Grammar (TDBTG) trained by the JPO corpus as the preorderer without external knowledge. For the preorderer with external knowledge, we used the one developed in-house (Chapter 4.5 of Goto et al. (2015)),[2] which was tuned for patent translation.

3.2 Training and Optimization

We used the Moses toolkit (Koehn et al., 2007) to train the phrase tables and lexicalized reordering models. We used multi-threaded GIZA++ for word alignment.

For the language models of the corpus-concatenated and single-domain models, we constructed 5-gram models from the target side of the bilingual corpora using KenLM (Heafield et al., 2013). In addition, we included the Google n-gram language models for Japanese and English as the external knowledge. These are back-off models estimated using maximum likelihood. The Japanese model was constructed from Web Japanese N-gram Version 1,[3] and the English model was constructed from Web 1T 5-gram Version 1 (LDC2006T13).

For optimization, we used k-best batch MIRA (Cherry and Foster, 2012).

3.3 Translation

The decoder used here is a clone of the Moses PBSMT decoder. It accepts feature augmentation, i.e., it can use multiple submodels and set an empty value.

[2]This preorderer modifies word order based on parse trees output by the Berkeley parser (Petrov et al., 2006; Petrov and Klein, 2007).

[3]http://www.gsk.or.jp/catalog/gsk2007-c/

Method	JPC			
	Ja-En	En-Ja	Ja-Zh	Zh-Ja
Single-Domain Model	34.58	38.06	33.35	39.54
Corpus Concatenation	35.64	38.61	34.27	40.96
Domain Adaptation	**35.68**	**39.03**	**34.64**	**41.09**

Table 3: BLEU Scores on JPO Corpus (official scores)

Method	JPC			
	Ja-En	En-Ja	Ja-Zh	Zh-Ja
Single-Domain Model	35.12(-)	37.40(-)	31.96(-)	38.15(-)
Corpus Concatenation	36.22	38.03(-)	32.92(-)	39.68(-)
Domain Adaptation	**36.29**	**38.48**	**33.36**	**39.85**

Table 4: BLEU Scores on JPO Corpus (MultEval scores)

4 Experimental Results

For evaluation, we used two toolkits based on BLEU (Papineni et al., 2002). One is the official BLEU scores provided by the WAT2016 committee. Because the official tool cannot measure a significance level of two systems, we also used the MultEval tool (Clark et al., 2011), which can measure significance levels based on bootstrap resampling. Since we represent the mean scores of three optimizations, the MultEval scores differ from the official scores.

4.1 JPO Corpus (without External Knowledge)

For JPO corpus experiments, we did not use external knowledge and compared translations of the single-domain model, corpus concatenation, and domain adaptation. The JPO corpus was divided into four domains (chemistry, electricity, machine, and physics). Tables 3 and 4 show the results evaluated by the official scorer and MultEval tools, respectively. The symbol (-) indicates that the score was significantly degraded compared to that of the domain adaptation ($p < 0.05$). Note that test sentences of each domain were translated using the corresponding models, and the BLEU score was computed by concatenating all test sentences as a document.

Results are presented in Table 4. Corpus concatenation corresponds to typical translation quality where only the JPO corpus was used. The single-domain model scores were inferior to the corpus concatenation scores because the corpus sizes were reduced by one-quarter. In contrast, the domain adaptation scores for most language pairs improved significantly and the domain adaptation was successful.

4.2 JPO and ASPEC Corpora (with External Knowledge)

Next, we conducted experiments using five domains with the JPO and ASPEC corpora. In these experiments, we evaluated the effects of external knowledge using the Google n-gram language model. The results are shown in Tables 5 and 6.

We first describe the effects of external knowledge, as shown in Table 6. In Table 6, the upper and lower halves show the BLEU scores before and after adding the Google n-gram language model, respectively. By adding the Google n-gram LMs, 0.27, 0.82, and 0.12 BLEU scores were improved on average in the JPO domains of Ja-En, En-Ja and Zh-Ja pairs, respectively. In the ASPEC domain, -0.03, 0.56, and 0.67 BLEU scores were improved. Except for the Ja-En pair of the ASPEC domain, the Google n-gram language model contributed to translation quality. The Japanese model tends to be suitable for JPO and ASPEC domains compared to the English model.

Next, we focused on the effect of domain adaptation with the Google n-gram LMs. In most cases, domain adaptation worked effectively except for the Ja-En pair of the ASPEC domain because the BLEU scores improved or were maintained the same level compared to those of the single-domain model and

LM	Method	JPC				ASPEC			
		Ja-En	En-Ja	Ja-Zh	Zh-Ja	Ja-En	En-Ja	Ja-Zh	Zh-Ja
w/o GN	Single-Domain Model	33.67	38.75	33.27	40.06	21.54	33.97	**30.12**	39.33
	Corpus Concatenation	35.49	39.18	33.94	41.08	20.90	33.11	29.66	37.84
	Domain Adaptation	35.96	40.14	**34.64**	**41.93**	21.34	34.21	29.97	39.51
w/ GN	Single-Domain Model	33.99	39.63		40.47	**21.64**	34.59		40.01
	Corpus Concatenation	35.73	40.23		41.31	20.80	33.78		38.30
	Domain Adaptation	**36.06**	**40.90**		41.87	21.54	**34.67**		**40.02**

Table 5: BLEU Scores on JPO and ASPEC Corpora (official scores)

LM	Method	JPC				ASPEC			
		Ja-En	En-Ja	Ja-Zh	Zh-Ja	Ja-En	En-Ja	Ja-Zh	Zh-Ja
w/o GN	Single-Domain Model	33.90(-)	38.19(-)	31.78(-)	38.74(-)	22.79	34.80	**29.47**(+)	38.96(-)
	Corpus Concatenation	35.81(-)	38.62(-)	32.76(-)	39.96(-)	22.20(-)	33.94(-)	28.95(-)	37.62(-)
	Domain Adaptation	36.25	39.58	**33.53**	40.76	22.80	34.91	29.28	39.18
w/ GN	Single-Domain Model	34.35(-)	39.04(-)		38.90(-)	**22.87**(+)	**35.42**		39.74(-)
	Corpus Concatenation	36.03(-)	39.48(-)		40.14(-)	22.10(-)	34.55(-)		38.15(-)
	Domain Adaptation	**36.40**	**40.32**		**40.77**	22.74	35.36		**39.87**

Table 6: BLEU Scores on JPO and ASPEC Corpora (MultEval scores)

corpus concatenation. However, we confirmed that the effects of the ASPEC domain were less than those of the JPO domains because the score did not improve significantly. This is because the ASPEC domain uses one million bilingual sentences; thus, domain adaptation could not contribute to the high-resource domains.

5 Conclusions

We have described the NICT-2 translation system. The proposed system employs Imamura and Sumita (2016)'s domain adaptation. In this study, we regarded the JPO corpus as a mixture of four domains and improved the translation quality. Although we added the ASPEC corpus as a fifth domain, the effects were not significant. Our domain adaptation can incorporate external knowledge, such as Google n-gram language models. The proposed domain adaptation can be applied to existing translation systems with little modification.

Acknowledgments

This work was supported by "Promotion of Global Communications Plan — Research and Development and Social Demonstration of Multilingual Speech Translation Technology," a program of the Ministry of Internal Affairs and Communications, Japan.

References

Colin Cherry and George Foster. 2012. Batch tuning strategies for statistical machine translation. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 427–436, Montréal, Canada, June.

Jonathan H. Clark, Chris Dyer, Alon Lavie, and Noah A. Smith. 2011. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 176–181, Portland, Oregon, USA, June.

Hal Daumé, III. 2007. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 256–263, Prague, Czech Republic, June.

Isao Goto, Masao Utiyama, Eiichiro Sumita, and Sadao Kurohashi. 2015. Preordering using a target-language parser via cross-language syntactic projection for statistical machine translation. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 14(3):13:1–13:23, June.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. 2013. Scalable modified Kneser-Ney language model estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 690–696, Sofia, Bulgaria, August.

Kenji Imamura and Eiichiro Sumita. 2016. Multi-domain adaptation for statistical machine translation based on feature augmentation. In *Proceedings of AMTA 2016: Association for Machine Translation in the Americas*, Austin, Texas, USA, October.

Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *HLT-NAACL 2003: Main Proceedings*, pages 127–133, Edmonton, Alberta, Canada, May 27 - June 1.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic, June.

Tetsuji Nakagawa. 2015. Efficient top-down BTG parsing for machine translation preordering. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-2015) (Volume 1: Long Papers)*, pages 208–218, Beijing, China, July.

Toshiaki Nakazawa, Hideya Mino, Chenchen Ding, Isao Goto, Graham Neubig, Sadao Kurohashi, and Eiichiro Sumita. 2016a. Overview of the 3rd workshop on Asian translation. In *Proceedings of the 3rd Workshop on Asian Translation (WAT2016)*, Osaka, Japan, December.

Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchimoto, Masao Utiyama, Eiichiro Sumita, Sadao Kurohashi, and Hitoshi Isahara. 2016b. ASPEC: Asian scientific paper excerpt corpus. In *Proceedings of the 10th Conference on International Language Resources and Evaluation (LREC-2016)*, Portoroz, Slovenia, May.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318, Philadephia, Pennsylvania, USA, July.

Slav Petrov and Dan Klein. 2007. Improved inference for unlexicalized parsing. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 404–411, Rochester, New York, April. Association for Computational Linguistics.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 433–440, Sydney, Australia, July. Association for Computational Linguistics.