Multilingual and Multi-Domain Adaptation for Neural Machine Translation

Chenhui Chu[1] **and Raj Dabre**[2]
[1]Institute for Datability Science, Osaka University
[2]Graduate School of Informatics, Kyoto University
chu@ids.osaka-u.ac.jp, raj@nlp.ist.i.kyoto-u.ac.jp

1 Introduction

Neural machine translation (NMT) (Bahdanau et al., 2015) allows one to train an end-to-end system without the need to deal with word alignments, translation rules and complicated decoding algorithms, which are a characteristic of statistical machine translation (SMT) systems. A number of studies have shown that vanilla NMT works better than SMT only when there is an abundance of parallel corpora (Zoph et al., 2016). In a low resource situation, it is important to apply various domain adaptation techniques for NMT to beat SMT (Chu et al., 2017).

Domain adaptation is the process of developing high quality domain specific NLP models by leveraging out-of-domain data or models in order to improve the in-domain performance. In the context of NMT, several domain adaptation approaches have been proposed and shown to be effective in a low resource scenario (Chu et al., 2017). Most domain adaptation approaches focus on using a single resource rich out-of-domain data source to improve the low resource in-domain translations. There are also studies that use multiple out-of-domain data for adaptation (Sajjad et al., 2017).

It may not always be possible to use an out-of-domain parallel corpus in the same language pair and thus it is important to use data from other languages (Johnson et al., 2016). This approach is known as cross-lingual transfer learning, which transfers NMT model parameters among multiple languages. It is well known that a multilingual model, which relies on parameter sharing, helps in improving the translation quality for low resource languages especially when the target language is the same (Zoph et al., 2016).

In this paper, we propose to simultaneously use both, multilingual and multi-domain data for domain adaptation of NMT, which might outperform the methods that use them independently. To the best of our knowledge, this is the first study that uses both multilingual and multi-domain data for domain adaptation. To verify the effective methods in this multilingual and multi-domain adaptation scenario, we compare the different methods in the empirical study of single language pair domain adaptation for NMT (Chu et al., 2017). In particular, we compare *fine tuning*, *multi-domain* and *mixed fine tuning* (Chu et al., 2017).

We study how multilingualism impacts the in-domain translation performance and how transfer learning can be performed by fine tuning multilingual out-of-domain models to learn multilingual in-domain models.

2 Methods for Comparison

All the methods that we compare are simple and do not need any modifications to the NMT system. We study the effects of *fine tuning*, *multi-domain* and *mixed fine tuning* in two different but related scenarios: single-domain, and multilingual and multi-domain adaptation.

2.1 Single-Domain Adaptation Methods

Refer to the original paper (Chu et al., 2017) for details.

Fine Tuning

Fine tuning is the conventional way for domain adaptation where we first train an NMT system on a resource rich out-of-domain corpus till convergence, and then fine tune its parameters on a resource poor in-domain corpus.

Multi-Domain

The *multi-domain* method is motivated by (Johnson et al., 2016). We simply concatenate the corpora of multiple domains by appending artificial

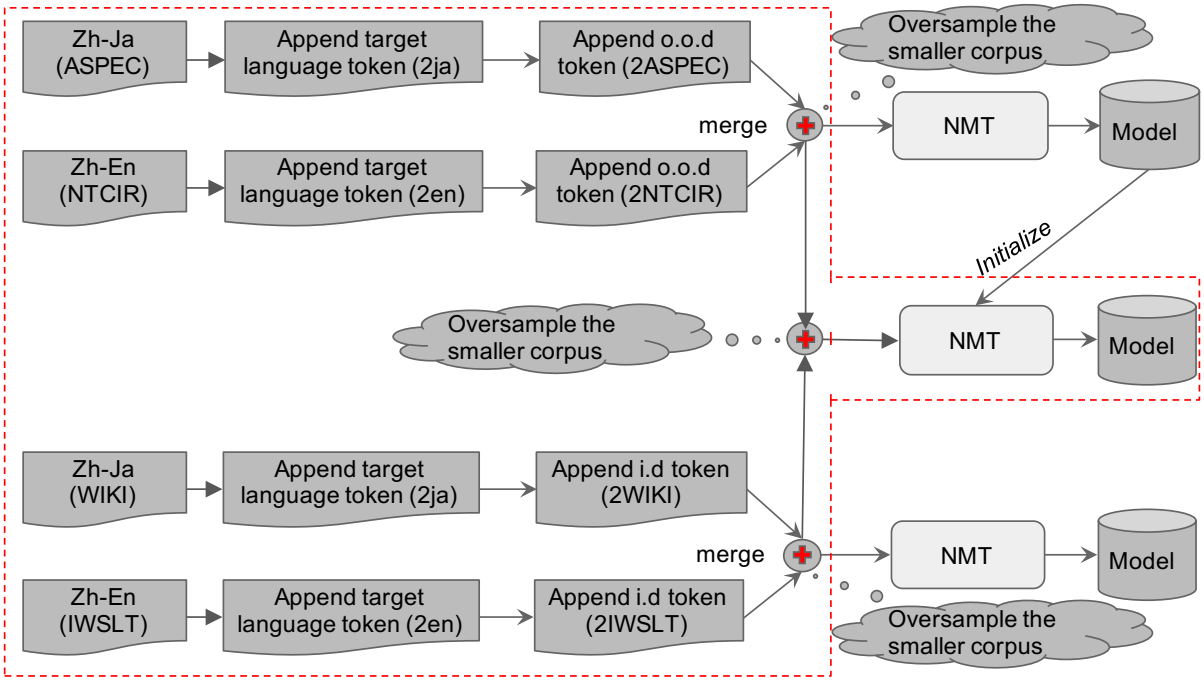

Figure 1: Multilingual and Multi-Domain adaptation for NMT. (The section in the dotted area denotes the multi lingual-domain method. This approach reduces to the originally proposed mixed fine tuning (Chu et al., 2017) when there is only one source and one target language.)

tokens that indicate the domains and by oversampling the corpora of the low resource domain.

Mixed Fine Tuning

Mixed fine tuning (Chu et al., 2017) is a combination of the above methods. Instead of fine tuning out-of-domain model on in-domain data directly, we fine tune on a in-domain and out-of-domain mixed corpus. As such this prevents overfitting and is a kind of domain transition approach.

2.2 Multilingual and Multi-Domain Adaptation Methods

Figure 1 gives an overview our multilingual and multi-domain approach. This is a combination of mixed fine tuning (Chu et al., 2017) and multilingual multiway NMT (Johnson et al., 2016), both of which use artificial tokens to control the target language and domain. Assume that there are two language pairs, Chinese-Japanese (Zh-Ja) and Chinese-English (Zh-En).[1] For each pair, assume that there is one out-of-domain corpus and one in-domain corpus: ASPEC (out-of-domain) for WIKI (in-domain) and NTCIR (out-of-domain) for IWSLT (in-domain).

[1]In this example, the source language is the same but can be different in principle.

Multi Fine Tuning

To train a multilingual out-of-domain model (upper part of figure 1), we append the target language tokens (2ja and 2en) and the domain tokens (2ASPEC and 2NTCIR) to the respective corpora and then merging them by oversampling the smaller corpus and then feed this corpus to the NMT training pipeline. The same approach is used to prepare the multilingual in-domain data (lower part of figure 1) using the in-domain language pairs. We then fine tune the in-domain model with the out-of-domain model.

Multi Lingual-Domain

To train a multilingual and multi-domain model, the merged in-domain and out-of-domain multilingual corpora are further merged into a single corpus by oversampling the smaller corpus. This is then fed to the NMT training pipeline.

Multi Mixed Fine Tuning

Instead of training a model from scratch, we can apply mixed fine tuning by initializing the multilingual and multi-domain model training by using the previously multilingual out-of-domain model. This method can reap the benefits of multilingualism as well as mixed fine tuning for domain adaptation.

3 Experimental Settings

3.1 Multilingual and Multi-Domain Setting

We focused on developing a single model that can translate from Chinese to Japanese and English for two domains of each target language.

The Chinese-English data comes from the patent (out-of-domain) and spoken language (in-domain) domains. The patent domain MT was conducted on the Chinese-English subtask (NTCIR-CE) of the patent MT task at the NTCIR-10 workshop.[2] The NTCIR-CE task uses 1M, 2k, and 2k sentences for training, development, and testing, respectively. The spoken domain MT was conducted on the Chinese-English subtask (IWSLT-CE) of the TED talk MT task at the IWSLT 2015 workshop. The IWSLT-CE task contains 209,491 sentences for training. We used the dev 2010 set for development, containing 887 sentences. We evaluated all methods on the 2010, 2011, 2012, and 2013 test sets, containing 1570, 1245, 1397, and 1261 sentences, respectively, and reported the average performance on these test sets. Note that both the in-domain and out-of-domain corpora are of a high quality since they were manually created.

The Chinese-Japanese data comes from the scientific (out-of-domain) and Wikipedia (in-domain; essentially open domain) domains. The scientific domain MT was conducted on the Chinese-Japanese paper excerpt corpus (ASPEC-CJ)[3] which is one subtask of the workshop on Asian translation (WAT).[4] The ASPEC-CJ task uses 672315, 2090, and 2107 sentences for training, development, and testing, respectively. The Wikipedia domain task was conducted on a Chinese-Japanese corpus automatically extracted from Wikipedia (WIKI-CJ) using the ASPEC-CJ corpus as a seed.[5] The WIKI-CJ task contains 136013, 198, and 198 sentences for training, development, and testing, respectively.

3.2 MT Systems

The NMT settings were the same as (Chu et al., 2017). The sizes of the source and target vocabularies, the source and target side embeddings, the hidden states, the attention mechanism hidden states, and the deep softmax output with a 2-maxout layer were set to 32000, 620, 1000, 1000, and 500, respectively. We used 2-layer LSTMs for both the source and target sides. ADAM was used as the learning algorithm, with a dropout rate of 20% for the inter-layer dropout, and L2 regularization with a weight decay coefficient of 1e-6. The mini batch size was 64, and sentences longer than 80 tokens were discarded. We early stopped the training process when we observed that the BLEU score of the development set converges. For testing, we ensembled the model checkpoints corresponding to the best development loss, the best development BLEU, and the final parameters in a single training run. The decoding beam size was set to 100. The maximum length of the translation was set to 2, and 1.5 times of the source sentences for Chinese-to-English, and Chinese-to-Japanese, respectively.

For performance comparison, we also conducted experiments on phrase based SMT (PBSMT). We used the Moses PBSMT system[6] for all of our MT experiments. For the respective tasks, we trained 5-gram language models on the target side of the training data using the KenLM toolkit[7] with interpolated Kneser-Ney discounting, respectively. In all of our experiments, we used the GIZA++ toolkit[8] for word alignment; tuning was performed by minimum error rate training, and it was re-run for every experiment.

For both MT systems, we preprocessed the data as follows. For Chinese, we used KyotoMorph[9] for segmentation. For English, we tokenized and lowercased the sentences using the *tokenizer.perl* script in Moses. Japanese was segmented using JUMAN.[10] For NMT, we further split the words into sub-words using byte pair encoding (BPE),[11] which has been shown to be effective for the rare word problem in NMT. We stopped the BPE merging process when the predefined vocabulary size 32000 reached for all our tasks.

4 Results

Tables 1 shows the results. The entries with SMT and NMT are the PBSMT and NMT systems, respectively; others are the different methods described in Section 2. "NTCIR-CE for IWSLT-

[2] http://ntcir.nii.ac.jp/PatentMT-2/
[3] http://lotus.kuee.kyoto-u.ac.jp/ASPEC/
[4] http://orchid.kuee.kyoto-u.ac.jp/WAT/
[5] http://lotus.kuee.kyoto-u.ac.jp/~chu/resource/wiki_zh_ja.tgz
[6] http://www.statmt.org/moses/
[7] https://github.com/kpu/kenlm/
[8] http://code.google.com/p/giza-pp
[9] https://bitbucket.org/msmoshen/kyotomorph-beta
[10] http://nlp.ist.i.kyoto-u.ac.jp/EN/index.php?JUMAN
[11] https://github.com/rsennrich/subword-nmt

System	IWSLT-CE	WIKI-CJ	NTCIR-CE	ASPEC-CJ
IWSLT-CE SMT	14.31	-	-	-
IWSLT-CE NMT	7.87	-	-	-
WIKI-CJ SMT	-	34.10	-	-
WIKI-CJ NMT	-	18.29	-	-
NTCIR-CE SMT	-	-	29.54	-
NTCIR-CE NMT	-	-	**37.11**	-
ASPEC-CJ SMT	-	-	-	36.39
ASPEC-CJ NMT	-	-	-	**42.92**
NTCIR-CE for IWSLT-CE (fine tuning)	16.41	-	-	-
NTCIR-CE for IWSLT-CE (multi-domain)	16.34	-	36.40	-
NTCIR-CE for IWSLT-CE (mixed fine tuning)	**18.01**	-	37.01	-
ASPEC-CJ for WIKI-CJ (fine tuning)	-	**37.66**	-	-
ASPEC-CJ for WIKI-CJ (multi-domain)	-	35.79	-	42.52
ASPEC-CJ for WIKI-CJ (mixed fine tuning)	-	37.57	-	**42.56**
NTCIR-CE_ASPEC-CJ for IWSLT-CE_WIKI-CJ (multi fine tuning)	15.00	28.22	-	-
NTCIR-CE_ASPEC-CJ for IWSLT-CE_WIKI-CJ (multi lingual-domain)	10.59	22.50	17.63	20.93
NTCIR-CE_ASPEC-CJ for IWSLT-CE_WIKI-CJ (multi mixed fine tuning)	13.46	27.51	33.53	40.11

Table 1: Domain adaptation results (BLEU-4 scores) for IWSLT-CE and WIKI-CJ using NTCIR-CE and ASPEC-CJ. The numbers in bold indicate the best system and all systems that were not significantly different from the best system.

CE" and "ASPEC-CJ for WIKI-CJ" denote the systems that use single out-of-domain data (i.e., NTCIR-CE or ASPEC-CJ) for adapting single in-domain data using the methods described in Section 2.1. "NTCIR-CE_ASPEC-CJ for IWSLT-CE_WIKI-CJ" denotes the systems that use multilingal and multi-domain data for adaptation with the methods described in Section 2.2.

We can see that, with single out-of-domain data, all the three single domain adaptation methods improve BLEU scores, which also outperforms SMT. Among which, mixed fine tuning shows the best performance. "NTCIR-CE_ASPEC-CJ for IWSLT-CE_WIKI-CJ," however, decreases the translation performance for all the adaptation methods. We think there are two main reasons: small number of parameters and limited vocabulary. Compared to bilingual domain adaptation setting, the training data sizes of multilingual and multi-domain adaptation are 2-4 times larger. Although, the complexity of the task is much higher, we use same model sizes for training the model. The NMT implementation we used was quite simple and thus we could not use it to train models with larger parameters quickly. In our experiments, the target languages are Japanese and English, which almost do not share vocabularies; however, we use same vocabulary sizes for all the systems, which limits the amount of vocabulary space for each language and can be a limitation for the multilingual and multi-domain systems.

5 Conclusion

In this paper, we proposed using both multilingual and multi-domain data for adapting multilingual in-domain NMT. We applied an approach that simply extends the domain adaptation methods that use single out-of-domain data for single in-domain data. Although we get negative results, we believe that we will obtain quality results by using NMT models with more parameters and larger vocabulary sizes along with better early stopping methods that focus on the development set performances for each language and domain equally.

Acknowledgments

This work was supported by Grant-in-Aid for Research Activity Start-up #17H06822, JSPS.

References

D. Bahdanau, K. Cho, and Y. Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of ICLR*.

C. Chu, R. Dabre, and S. Kurohashi. 2017. An empirical comparison of domain adaptation methods for neural machine translation. In *Proceedings of ACL*. pages 385–391.

M. Johnson et al. 2016. Google's multilingual neural machine translation system: Enabling zero-shot translation. *CoRR* abs/1611.04558.

H. Sajjad, N. Durrani, F. Dalvi, Y. Belinkov, and S. Vogel. 2017. Neural machine translation training in a multi-domain scenario. In *Proceedings of IWSLT*.

B. Zoph, D. Yuret, J. May, and K. Knight. 2016. Transfer learning for low-resource neural machine translation. In *Proceedings of EMNLP*. pages 1568–1575.