An Empirical Comparison of Simple Domain Adaptation Methods for Neural Machine Translation

Chenhui Chu[1] **, Raj Dabre**[2]**,** and **Sadao Kurohashi**[2]
[1]Japan Science and Technology Agency
[2]Graduate School of Informatics, Kyoto University
chu@pa.jst.jp, raj@nlp.ist.i.kyoto-u.ac.jp, kuro@i.kyoto-u.ac.jp

Abstract

In this paper, we compare two simple domain adaptation methods for neural machine translation (NMT): (1) We append an artificial token to the source sentences of two parallel corpora (different domains and one of them is resource scarce) to indicate the domain and then mix them to learn a multi domain NMT model; (2) We learn a NMT model on the resource rich domain corpus and then fine tune it using the resource poor domain corpus. We empirically verify fine tuning works better than the artificial token mechanism when the low resource domain corpus is of relatively poor quality (acquired via automatic extraction) but in the case of a high quality (manually created) low resource domain corpus both methods are equally viable.

1 Introduction

One of the most attractive features of neural machine translation (NMT) (Bahdanau et al., 2014; Cho et al., 2014; Sutskever et al., 2014) is that it is possible to train an end to end system without the need to deal with alignments, phrase tables and complicated decoding algorithms, which are a characteristic of statistical machine translation (SMT) systems. A major drawback of NMT is that it works better than SMT only when there is an abundance of parallel corpora. In the case of low resource domains, vanilla NMT is either worse than or comparable to SMT.

Multilingual GNMT (Johnson et al., 2016) has shown that it is possible to mix multiple language pairs into a single model without any modification to the architecture. They simply append a token "<2xx>" to the source sentences to indicate that "xx" is the target language.

Motivated by this approach we experimented with mixing two corpora, for the same language pair but belonging to different domains where one of them is resource scarce. Instead of a target language we use the "<2DOMAIN>" tokens to specify the domain so that the NMT system is primed to generate translation for a particular domain. We compared this against a fine tuning method that is commonly used in-domain adaption for NMT (e.g., (Luong and Manning, 2015; Freitag and Al-Onaizan, 2015) etc.), where we first train a NMT model on the resource rich domain corpus and then fine tune it on the resource poor domain corpus. We tried two different corpora settings:

- Manually translated resource poor corpus: Using the NTCIR corpus (patent domain; resource rich) to improve the translation quality for the IWSLT domain (TED talks; resource poor).
- Automatically extracted resource poor corpus: Using a the ASPEC corpus (scientific domain; resource rich) to improve the translation quality for the Wiki domain (resource poor). The parallel corpus of the latter domain was automatically extracted (Chu et al., 2016).

We observed that the fine tuning works better than the artificial token mechanism when the in-domain corpus is of relatively poor quality (acquired via automatic extraction) but in the case of a high quality (manually created) in-domain corpus both methods are equally viable.

2 Related Work

Domain adaptation for NMT was proven to be possible by using a RNN language model with the NMT decoder (Gülçehre et al., 2015), but this leads to a complicated architecture which is difficult to train. Other related works include domain specialization (Servan et al., 2015) and fast domain adaptation (Freitag and Al-Onaizan, 2015),

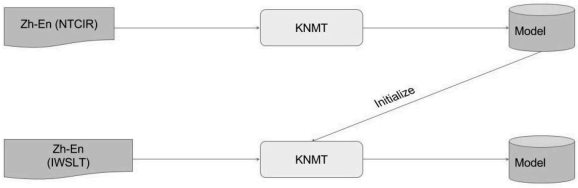

Figure 1: Fine tuning for domain adaptation

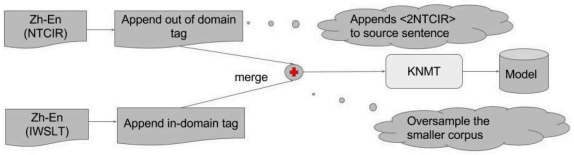

Figure 2: Tag based multi domain NMT

where a NMT system is fine tuned on a smaller in-domain corpus, and control the domain (Kobus et al., 2015) for NMT.

Recently, Google's multilingual NMT system (Johnson et al., 2016) was released, which worked on the principle of appending a tag like "<2xx>" to the source text where "xx" indicated the target language. We utilize this idea to domains instead.

3 Overview of Method

We compare two main domain adaptation techniques: a fine tuning approach (Figure 1) and a tag based multi domain approach (Figure 2). In the former approach, we first train a NMT system on a resource abundant out of domain corpus, and then fine tune its parameters on a resource scarce in-domain corpus.

In the latter approach, we simply append the corpora of multiple domains with two small modifications: a. Appending the tag "<2DOMAIN>" to the source sentences of the respective corpora where "DOMAIN" indicates the domain. This primes the NMT decoder to generate sentences for the specified domain. b. Oversampling the smaller corpora so that the training procedure pays equal attention to each domain. In both cases, we do not need any modifications to the NMT system nor the general training procedure itself.

4 Experimental Settings

We conducted MT domain adaptation experiments on two different quality in-domain corpus settings, i.e., Chinese-to-English high quality and Chinese-to-Japanese poor quality in-domain corpus settings.

4.1 High Quality In-domain Corpus Setting

For Chinese-to-English translation, we adapted a resource rich patent domain task to a resource poor spoken domain task. The patent domain MT task was conducted on the Chinese-English subtask (NTCIR-CE) of the patent MT task at the NTCIR-10 workshop.[1] The NTCIR-CE task uses 1,000,000, 2,000, and 2,000 sentences for training, development, and testing, respectively. The spoken domain task was conducted on the Chinese-English subtask (IWSLT-CE) of the TED talk MT task at the IWSLT 2015 workshop.[2] The IWSLT-CE task contains 209,491 sentences for training. We used the dev 2010 set for development, containing 887 sentences. We tested on the test 2010, 2011, 2012, and 2013 test sets, containing 1,570, 1,245, 1,397, and 1,261 sentences, respectively.

4.2 Poor Quality In-domain Corpus Setting

For Chinese-to-Japanese translation, we adapted a resource rich scientific domain task to a resource poor Wikipedia (essentially open) domain task. The scientific domain MT task 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 672,315, 2,090, and 2,107 sentences for training, development, and testing, respectively. The Wikipedia domain task was conducted on a Chinese-Japanese corpus automatically extracted from Wikipedia (WIKI-CJ) (Chu et al., 2016). The WIKI-CJ task contains 136,013, 198, and 198 sentences for training, development, and testing, respectively.

4.3 MT Systems

For NMT, we used the KyotoNMT system[5] (Cromieres et al., 2016). The settings essentially followed those of the best systems that participated in WAT 2016. 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 32,000, 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

[1] http://ntcir.nii.ac.jp/PatentMT-2/
[2] http://workshop2015.iwslt.org
[3] http://lotus.kuee.kyoto-u.ac.jp/ASPEC/
[4] http://orchid.kuee.kyoto-u.ac.jp/WAT/
[5] https://github.com/fabiencro/knmt

System	NTCIR-CE	IWSLT-CE test 2010	test 2011	test 2012	test 2013	average
IWSLT-CE SMT	N/A	12.73	16.27	14.01	14.67	14.31
IWSLT-CE NMT	N/A	6.75	9.08	9.05	7.29	7.87
NTCIR-CE SMT	29.54	3.57	4.70	4.21	4.74	4.33
NTCIR-CE NMT	37.11	2.23	2.83	2.55	2.85	2.60
Fine tuning	N/A	**13.93**	18.99	16.12	17.12	**16.41**
Multi domain	36.40	13.42	**19.07**	**16.56**	**17.54**	16.34
Multi domain w/o tags	**37.32**	12.57	17.40	15.02	15.96	14.97
Multi domain + fine tuning	N/A	13.18	18.03	16.41	16.80	15.82

Table 1: Domain adaptation results (BLEU-4 scores) for IWSLT-CE using NTCIR-CE.

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 perplexity of the development set converges. For testing, we self ensembled the three parameters of the best development loss, the best development BLEU, and the final parameters. 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. We used the Moses phrase based SMT 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 of the two 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 script in Moses. Japanese was segmented using JUMAN.[10]

For NMT, we further split the words into sub-words using byte pair encoding (BPE) (Sennrich et al., 2016), which has been shown to be effective for the rare word problem in NMT. Another motivation of using sub-words is that it makes the different domain tasks share more vocabulary, which is important especially for the resource poor domain. For the Chinese-to-English tasks, we trained two BPE models on the Chinese and English vocabularies, respectively. For the Chinese-to-Japanese tasks, we trained a joint BPE model on both of the Chinese and Japanese vocabularies, because Chinese and Japanese could share some vocabularies of Chinese characters. The number of merge operations was set to 30,000 for all the tasks.

System	ASPEC-CJ	WIKI-CJ
WIKI-CJ SMT	N/A	36.83
WIKI-CJ NMT	N/A	18.29
ASPEC-CJ SMT	36.39	17.43
ASPEC-CJ NMT	**42.92**	20.01
Fine tuning	N/A	**37.66**
Multi domain	42.52	35.79
Multi domain w/o tags	40.78	33.74
Multi domain + fine tuning	N/A	34.61

Table 2: Domain adaptation results (BLEU-4 scores) for WIKI-CJ using ASPEC-CJ.

5 Results

Table 1 and 2 show the translation results on the Chinese-to-English and Chinese-to-Japanese tasks, respectively. Where "DOMAIN SMT" and "DOMAIN NMT" are the phrase based SMT and NMT systems, respectively, trained on different training data DOMAIN; "Fine tuning" denotes the systems that used the parameters obtained from the resource rich domain as the initial parameters for training the resource poor domain; "Multi domain" denotes the systems that mixed both the resource rich and poor domains, and added the domain tags for each domain; "Multi domain w/o tags" denotes the systems that mixed both the resource rich and poor domains, but did not specify the domain tags; "Multi domain + Fine tuning" denotes the systems first trained with the "Multi domain" method, and then fine tuned on the resource poor domain data.

[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

We can see that without domain adaptation, the SMT systems perform significantly better than the NMT system on the resource poor domains, i.e., IWSLT-CE and WIKI-CJ; while on the resource rich domains, i.e., NTCIR-CE and ASPEC-CJ, NMT outperforms SMT.

Directly using the SMT/NMT models trained on the resource rich domain data to translate the resource poor domain data shows bad performance. However, on the WIKI-CJ test set, "ASPEC-CJ NMT" outperforms "WIKI-CJ NMT." We suspect the reason for this is that the WIKI-CJ data was automatically extracted using the ASPEC-CJ data as the seed parallel corpus, and thus they share many vocabularies.

Comparing different domain adaptation methods, we can see that "Fine tuning" performs the best on the WIKI-CJ test set; while on the IWSLT-CE test sets, "Multi domain" outperforms "Fine tuning" on the test 2011, 2012, and 2013 sets, but performs worse on the test 2010 set. On the resource rich test sets, "Multi domain" slightly decreases the performance, compared to the NMT systems that trained on the resource rich data only. "Multi domain w/o tags" decreases the performance on both the IWSLT-CE and WIKI-CJ test sets, indicating that the importance of the domain tags. "Multi domain + Fine tuning" also slightly decreases the performance compared to "Multi domain," we think the reason for this is that the "Multi domain" method already trained on the data that contains the resource poor data used for fine tuning, and thus further fine tuning on it does not help.

6 Conclusion

In this paper, we have empirically compared two simple domain adaptation methods namely, fine tuning and tag based multi domain approaches for NMT. We have shown that both methods have their respective merits where the former works almost as well as the latter in the case of an automatically extracted, poor quality parallel corpus for a low resource domain; but the latter outperforms the former when the low resource domain corpus is manually created and thus is of a higher quality as compared to an automatically extracted one.

In the future, we would like to further experiment on how we can effectively combine the tag and fine tuning based methods, and obtain even better domain specific translations. We also plan to experiment with multiple recurrent neural language models into our current architecture to leverage abundant monolingual corpora.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR*, abs/1406.1078.

Chenhui Chu, Raj Dabre, and Sadao Kurohashi. 2016. Parallel sentence extraction from comparable corpora with neural network features. In *Proceedings of LREC 2016*.

Fabien Cromieres, Chenhui Chu, Toshiaki Nakazawa, and Sadao Kurohashi. 2016. Kyoto university participation to wat 2016. In *Proceedings of the WAT 2016*.

G Markus Freitag and Yaser Al-Onaizan. 2015. Fast domain adaptation for neural machine translation. *arXiv preprint arXiv:1612.06897*.

Çaglar Gülçehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loïc Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On using monolingual corpora in neural machine translation. *CoRR*, abs/1503.03535.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda B. Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's multilingual neural machine translation system: Enabling zero-shot translation. *CoRR*, abs/1611.04558.

Catherine Kobus, Josep Crego, and Jean Senellart. 2015. Domain control for neural machine translation. *arXiv preprint arXiv:1612.06140*.

Minh-Thang Luong and Christopher D Manning. 2015. Stanford neural machine translation systems for spoken language domains. In *Proceedings of IWSLT 2015*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of ACL 2016*.

Christophe Servan, Josep Crego, and Jean Senellart. 2015. Domain specialization: a post-training domain adaptation for neural machine translation. *arXiv preprint arXiv:1612.06141*.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. *CoRR*, abs/1409.3215.