Bering Lab's Submissions on WAT 2021 Shared Task

Heesoo Park and Dongjun Lee
Bering Lab, South Korea
{heesoo.park, djlee}@beringlab.com

Abstract

This paper presents the Bering Lab's submission to the shared tasks of the 8th Workshop on Asian Translation (WAT 2021) on JPC2 and NICT–SAP. We participated in all tasks on JPC2 and IT domain tasks on NICT–SAP. Our approach for all tasks mainly focused on building NMT systems in domain-specific corpora. We crawled patent document pairs for English–Japanese, Chinese–Japanese, and Korean–Japanese. After cleaning noisy data, we built parallel corpus by aligning those sentences with the sentence-level similarity scores. Also, for SAP test data, we collected the OPUS dataset including three IT domain corpora. We then trained transformer on the collected dataset. Our submission ranked 1^{st} in eight out of fourteen tasks, achieving up to an improvement of 2.87 for JPC2 and 8.79 for NICT–SAP in BLEU score .

1 Introduction

The WAT 2021 Shared Task (Nakazawa et al., 2021) [1] focuses a comprehensive set of machine translations on Asian languages. They gather and share the resources and knowledge about Asian language translation through a variety of tasks on the broad topics such as document-level translation, multi-modal translation, and domain adaptation. Among those tasks, we participated on two tasks: (1) JPO Patent Corpus (JPC2), a translation task on patent corpus of Japanese ↔ English/Korean/Chinese, and (2) NICT-SAP IT domain, a translation task on software documentation corpus of English ↔ Hindi/Indonesian/Malaysian/Thai.

According to the Table 1, both two corpora mostly consist of technical terms. Specifically, jargon such as "acrylic acid" from JPC2 is not commonly used in everyday life. Similarly, terminology "Virus Scan Interface" from NICT-SAP cannot be easily found on the general corpus. Therefore, we focused on domain adaptation for both tasks.

	JPC2
JP	その中でも、アクリル酸を好適に使用することができる。
EN	Among them, an acrylic acid can be preferably used.
	NICT-SAP IT domain
ID	Spesifikasi Antarmuka Pemindaian Virus (NW-VSI)
EN	Virus Scan Interface (NW-VSI) Specification

Table 1: Sample sentences of JPC2 and NICT-SAP.

Our approach begins with collecting rich and clean sentence pairs from web and public dataset. For JPC2, we crawled the patent documents from web for each language pairs then built parallel corpus by pairing each sentence with the similarity scores between source and target sentence representation vectors. For NICT-SAP IT domain, we collected public dataset, OPUS (Tiedemann, 2012), and weighted the IT corpus among those corpus while training. In addition to the rich and clean additional corpus, we chose transformer (Vaswani et al., 2017), broadly recognized as a strong machine translation system.

Our method obtained the new state-of-the-art results on four out of six JPC2 tasks, especially amounting to 2.87 absolute improvement on BLEU scores for Japanese to Korean translation. To validate the effect of the additional data, we conducted the ablation study on Korean → Japanese data. Furthermore, our models ranked first place on four out of eight NICT-SAP IT domain tasks, achieving 8.79 improvement for Indonesian to English.

[1] https://lotus.kuee.kyoto-u.ac.jp/WAT/WAT2021/index.html

Proceedings of the 8th Workshop on Asian Translation, pages 141–145
Bangkok, Thailand (online), August 5-6, 2021. ©2021 Association for Computational Linguistics

Data	# Sen	Avg. Len
Train$_{JP-EN}$	1,000,000	44.85
Dev$_{JP-EN}$	2,000	53.17
Test$_{JP-EN}$	5,668	58.63
Train$_{JP-KO}$	1,000,000	52.27
Dev$_{JP-KO}$	2,000	83.56
Test$_{JP-KO}$	5,230	82.67
Train$_{JP-ZH}$	1,000,000	53.47
Dev$_{JP-ZH}$	2,000	63.14
Test$_{JP-ZH}$	5,204	62.37

(a) Statistics of JPC2. “Avg. Len” represents the average of the number of characters per Japanese sentence.

Data	# Sen	Avg. Len
Dev$_{EN-HI}$	2,016	10.25
Test$_{EN-HI}$	2,073	8.74
Dev$_{EN-ID}$	2,023	10.46
Test$_{EN-ID}$	2,037	8.92
Dev$_{EN-MS}$	2,050	13.00
Test$_{EN-MS}$	2,050	13.05
Dev$_{EN-TH}$	2,049	12.57
Test$_{EN-TH}$	2,050	12.40

(b) Statistics of NICT-SAP (IT domain). “Avg. Len” represents the average of the number of words per English sentence.

Table 2: Data statistics.

2 Task Description

We participate JPO Patent Corpus (JPC2) and SAP’s IT translation tasks.

2.1 Parallel Corpus

JPO Patent Corpus JPC2 consists of Chinese-Japanese, Korean-Japanese, and English-Japanese patent description parallel corpus (Nakazawa et al., 2021). Each corpus consists of 1M parallel sentences with four sections (chemistry, electricity, mechanical engineering, and physics).

SAP’s IT Corpus SAP software documentation corpus (Buschbeck and Exel, 2020) is designed to test the performance of multilingual NMT systems in extremely low-resource conditions (Nakazawa et al., 2021). The dataset consists of Hindi(Hi) / Thai(Th) / Malay(Ms) / Indonesian(Id) ↔ English software documentation parallel corpus. The number of parallel sentences of each corpus is described in Table 2.

Language	# Sen	Avg. Len
JP – EN	21,254,269	215.31
JP – KO	13,916,372	110.29
JP – ZH	13,881,444	144.44

Table 3: Statistics of additional parallel sentences. “Avg. Len” represents the average of the number of characters per Japanese sentence.

2.2 Evaluation metric

The official evaluation metrics are BLEU (Papineni et al., 2002), RIBES (Isozaki et al., 2010), and AMFM (Banchs et al., 2015).

3 System Overview

In this section we introduce our approach for two tasks.

3.1 Data crawling and preprocessing

For JPC2 tasks, we trained the models on combination of the given train dataset (Table 2) and web-crawled dataset (Table 3). For NICT-SAP tasks, we trained the models on OPUS dataset with IT domain corpus weighted (Table 4). For both tasks, the models were evaluated on the given test dataset (Table 2).

Patent crawling data Additional data for JPC2 was obtained from WIPO [2] through website crawling. The JPC2 data (including the evaluation data) consists only of description section in each document. Since our approach is to collect the data which is very close to the task domain, we filtered out all sections but the description section to avoid the redundant noise while training the model.

To pair each sentence, we first split the whole description into sentences and encoded each sentence to a representation vector. As a sentence encoder, we used LASER [3] for Ko–Ja and Universal Sentence Encoder [4] (Cer et al., 2018) for the other pairs. We then measured the cosine similarity between each sentence pair and filtered out the pairs whose score was under threshold.

OPUS data (Tiedemann, 2012) Since the NICT-SAP IT domain translation task does not provide the train dataset, we collected it from public dataset including GNOME, KDE4, Ubuntu,

[2] https://patentscope.wipo.int/search/en/search.jsf

[3] https://github.com/facebookresearch/LASER

[4] https://tfhub.dev/google/universal-sentence-encoder/3

En–X	GNOME	KDE4	Ubuntu	ELRC	TANZIL	Opensubtitles	tico-19	QED	Tatoeba
HI	145,706	97,227	11,309	245	187,080	93,016	3,071	11,314	10,900
ID	47,234	14,782	96,456	2,679	.	9,268,181	3,071	274,581	9,967
MS	299,601	87,122	120,016	1,697	.	1,928,345	3,071	79,697	.
TH	78	70,634	3,785	.	.	3,281,533	.	264,677	1,162

Table 4: Statistics of additional parallel sentences.

Tateoba, Tanzil, QED (Abdelali et al., 2014), tico-19, OpenSubtitles, ELRC. We downloaded all the dataset from OPUS site. Table 4 shows the statistics of the data obtained from the site.

3.2 Model configuration

For the NMT system, we used OpenNMT-py (Klein et al., 2017) [5] to train Transformer (Vaswani et al., 2017) architecture models with several different parameter configurations for each task. Our models have 6 encoder layers, 6 decoder layers, a sequence length of 512 for both source and target side, 8 attention heads with an attention dropout of 0.1. Each model was trained on Nvidia RTX 3090 Ti (24GB). We used an effective batch size of 2048 tokens. We chose Adam (Kingma and Ba, 2014) optimizer with a learning rate of 1, warm-up steps 8000, label smoothing 0.1 and token-level layer normalization. We set the data type to the floating point 32 and applied relative positional encoding (Shaw et al., 2018) to consider the pair-wise relationships between the input elements. We changed the hidden layer size from 512 to 2048 and the feed forward networks from 2048 to 4096 for finding the model to perform best. We saved the checkpoint every 20,000 steps and choose the model which performed best on the validation set.

We used google sentencepiece library [6] to train separate SentencePiece models (Kudo and Richardson, 2018) on the source and target sides, for each language. We trained a regularized unigram model (Kudo, 2018). For JPC2, we set a vocabulary size of 32,000 for Japanese and Chinese and 16,000 for Korean and English. We set a character coverage to 0.995. For NICT-SAP, we set a vocabulary size of 8,000 for English and Malaysian and 16,000 for Hindi, Indonesian and Thai. We set a character coverage to 0.995. While training sentence piece models, we used only given train dataset and only IT domain (Ubuntu, GNOME, KDE4) for JPC2 and NICT-SAP, respectively.

Sub-task	Tokenizer	BLEU	Rank
En → Ja	mecab	47.44	3 of 15
Ja → En	moses	45.13	1 of 10
Ko → Ja	mecab	75.82	1 of 15
Ja → Ko	mecab	76.68	1 of 10
Zh → Ja	mecab	51.28	2 of 11
Ja → Zh	kytea	42.92	1 of 10

Table 5: Official rank and BLEU scores for JPC2 tasks on Test-n dataset.

Sub-task	BLEU	AMFM	Rank
En → Hi	37.23	0.81	1 of 9
Hi → En	34.48	0.80	4 of 9
En → Id	53.22	0.85	1 of 9
Id → En	53.49	0.85	1 of 9
En → Ms	45.96	0.86	1 of 9
Ms → En	38.42	0.81	2 of 9
En → Th	34.52	0.70	5 of 9
Th → En	25.07	0.73	2 of 9

Table 6: Rank and BLEU/AMFM scores for NICT-SAP IT tasks on leader-board. The rank is scored by BLEU score.

4 Result

We participated in JPC2 and NICT-SAP (IT domain) tasks. JPC2 consists of English–Japanese (En–Ja), Chinese–Japanese (Zh–Ja) and Korean–Japanese (Ko–Ja). NICT-SAP consists of English–Hindi (En–Hi), English–Indonesian (En–Id), English–Malaysian (En–Ms) and English–Thai (En–Th).

4.1 JPC2 patent translation task

Table 5 shows overall results on JPC2 dataset. Our models ranked first in all the tasks whose input is Japanese. Across overall process, we weighted the given dataset to the crawled dataset *by oversampling*.

English – Japanese We collected the additional

[5] https://github.com/OpenNMT/OpenNMT-py

[6] https://github.com/google/sentencepiece

Subtask	# Sen	Avg. Len	w	wo
Test-n	5,230	82.67	76.68	74.60
Test-n1	2,000	85.60	75.90	75.11
Test-n2	3,000	80.32	78.13	74.86
Test-n3	230	87.8	64.47	66.25

Table 7: Ablation studies for JPC2 Ja → Ko subtask."w" and "wo" represents the BLEU score of the model trained **with** and **without** the additional dataset, repectively. "Avg. Len" represents the average of the number of characters per Japanese sentence.

data 20 times more than the given training dataset. We noticed that the average of the sentence length in the collected dataset is much longer than the given dataset. This represents that the collected dataset is quite different from original data. Therefore, we weighted the given train dataset five times for Ja → En and two times for En → Ja task.

In the inference time, we used the seven independent models ensemble for Ja → En and the six independent models for En → Ja task. We selected each model's checkpoint which performed best in the validation data. We set the beam size to 7. The model ensemble method led to a performance improvement by 1.25 and 0.85 of the BLEU score for Ja → En and En → Ja, respectively. The best performance of our model was a BLEU score of 47.44 in the En → Ja and 45.13 in the Ja → En task.

Korean – Japanese Our collected data 13 times more than the given one. Similar to En ↔ Ja, we weighted the original dataset three times for both Ja → Ko and Ko → Ja. In the inference time, we used the five independent models ensemble for both Ja → Ko and for Ko → Ja. We set the beam size to 7. The best performance of our model was a BLEU score of 75.82 for the Ko → Ja task and 76.68 for the Ja → Ko task.

To validate the effect of additional data, we conducted an ablation studies on the Ja → Ko task. Table 7 shows the sub-tasks in the JPC2 dataset. Each test data in JPC2 can be split according to the publish year and the way they were collected. Test-n1 consists of the patent documents published between 2011 and 2013. Test-n2 and test-n3 consist of patent documents between 2016 and 2017, but test-n3 are manually created by translating source sentences. While the model trained with additional data outperforms the other model in test-n1 and test-n2, it shows poor performance on test-n3 which consists of manual translations.

Chinese – Japanese Similar to En ↔ Ja and Ko ↔ Ja, we weighted the original dataset two times for both Ja → Zh and three times for Zh → Ja. In the inference time, we used the five independent models ensemble for Ja → Zh and seven models for Zh → Ja. We set the beam size to 7. The best performance of our model was a BLEU score of 51.28 in the Zh → Ja dataset and 42.92 in the Ja → Zh dataset.

4.2 NICT-SAP IT domain translation task

Table 6 shows the overall results on NICT-SAP IT domain. While we trained transformer on OPUS dataset from scratch, most of the high-ranked models used the pre-trained mBART (Chipman et al., 2021) and finetuned it. Therefore, others got benefit from the multilingualism and gigantic additional corpus. Even though we used relatively small data, we achieved the state-of-the-art scores on the four out of eight tasks.

For all language pairs, we weighted IT dataset (Ubuntu, GNOME, KDE4) 2.5 times to the general one. We saved the checkpoint at every 20000 step, then submitted the models which showed the best performance for validation set. Except for Thai, our models ranked first on the sub-tasks whose input is English. Furthermore, our models outperformed competitors on En ↔ Id, achieving an improvement of 7.83 for En → Id and 8.79 for Id → En dataset. We used relatively rich amount of dataset in this subtask. In contrast, on the En ↔ Th sub-task, our model performed relatively poor since we used small amount of data to train it.

5 Conclusion

In this work, we described the Bering Lab's submission to the WAT 2021 shared tasks. We collected the in-domain dataset for both JPC2 and NICT–SAP tasks and built transformer-based MT systems on those corpora. which were trained on given train dataset and additional crawled patent data. Our models ranked first place in eight out of fourteen tasks, amounting a high improvements for both tasks.

References

Ahmed Abdelali, Francisco Guzman, Hassan Sajjad, and Stephan Vogel. 2014. The amara corpus: Building parallel language resources for the educational domain. In *LREC*, volume 14, pages 1044–1054.

Rafael E Banchs, Luis F D'Haro, and Haizhou Li. 2015. Adequacy–fluency metrics: Evaluating mt in the continuous space model framework. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(3):472–482.

Bianka Buschbeck and Miriam Exel. 2020. A parallel evaluation data set of software documentation with document structure annotation. *arXiv preprint arXiv:2008.04550*.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Céspedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*.

Hugh A Chipman, Edward I George, Robert E McCulloch, and Thomas S Shively. 2021. mbart: Multidimensional monotone bart. *Bayesian Analysis*, 1(1):1–30.

Hideki Isozaki, Tsutomu Hirao, Kevin Duh, Katsuhito Sudoh, and Hajime Tsukada. 2010. Automatic evaluation of translation quality for distant language pairs. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 944–952.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. *arXiv preprint arXiv:1804.10959*.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

Toshiaki Nakazawa, Hideki Nakayama, Chenchen Ding, Raj Dabre, Shohei Higashiyama, Hideya Mino, Isao Goto, Win Pa Pa, Anoop Kunchukuttan, Shantipriya Parida, Ondřej Bojar, Chenhui Chu, Akiko Eriguchi, Kaori Abe, and Sadao Oda, Yusuke Kurohashi. 2021. Overview of the 8th workshop on Asian translation. In *Proceedings of the 8th Workshop on Asian Translation*, Bangkok, Thailand. Association for Computational Linguistics.

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*, pages 311–318.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in opus. In *Lrec*, volume 2012, pages 2214–2218.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.