Self-supervised and Supervised Joint Training for Resource-rich Machine Translation – Supplementary Materials

Yong Cheng [1] **Wei Wang** [†] **Lu Jiang** [1 2] **Wolfgang Macherey** [1]

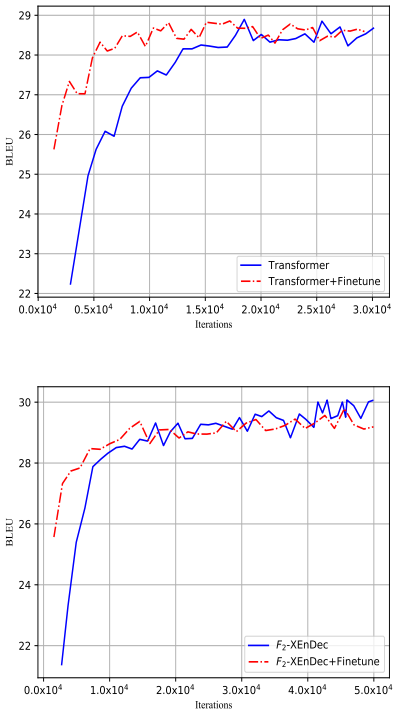

Figure 1. Comparison of finetuning and training from scratch using Transformer and F_2-*XEnDec*. In both methods, pre-training leads to faster convergence but fails to improve the final performance after the convergence. The comparison between the figures shows our joint training approach on the left (the blue curve) significantly outperforms against the two-stage training on the right. Final BLEU numbers are reported in Table 5 in the main paper.

[1]Google Research, Google LLC, USA [2]Language Technologies Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania. [†] Work done while at Google Research. Correspondence to: Yong Cheng <chengyong@google.com>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

1. Training Details

Data Pre-processing We mainly follows the pre-processing pipeline [1] which is also adopted by (Ott et al., 2018), (Edunov et al., 2018) and (Zhu et al., 2020), except for the sub-word tool. To verify the consistency between the word piece model (Schuster & Nakajima, 2012) and the BPE model (Sennrich et al., 2016), we conduct a comparison experiment to train two standard Transformer models using the same data set processed by the word piece model and the BPE model respectively. The BLEU difference between them is about ±0.2, which suggests there is no significant difference between them.

Batching Data Transformer groups training examples of similar lengths together with a varying batch size for training efficiency (Vaswani et al., 2017). In our approach, when interpolating two source sentences, $\mathbf{x}^p$ and $\mathbf{y}^\diamond$, it is better if the lengths of $\mathbf{x}^p$ and $\mathbf{y}^\diamond$ are similar, which can reduce the chance of wasting positions over padding tokens. To this end, in the first round, we search for monolingual sentences with exactly the same length of the source sentence in a parallel sentence pair. After the first traversal of the entire parallel data set, we relax the length difference to 1. This process is repeated by relaxing the constraint until all the parallel data are paired with their own monolingual data.

2. A Prior Alignment Matrix

When $\mathcal{L}_{F_1}$ is removed, we can not obtain $\mathbf{A}'$ according to Algorithm 1 in the main paper which leads to the failure of calculating $\mathcal{L}_{F_2}$. Thus we propose a prior alignment to tackle this issue. For simplicity, we set $n(\cdot)$ to be a copy function when doing the first *XEnDec*, which means that we just randomly mask some words in the first round of *XEnDec*. In the second *XEnDec*, we want to combine $(\mathbf{x}^p, \mathbf{y}^p)$ and $(\mathbf{y}^\diamond, \mathbf{y})$. The alignment matrix $\mathbf{A}'$ for $(\mathbf{y}^\diamond, \mathbf{y})$ is constructed as follows.

If a word y_j in the target sentence $\mathbf{y}$ is picked in the source side which indicates $y_j^\diamond$ is picked and $m_j = 0$, its attention value A'_{ji} if $m_i = 0$ is assigned to $\frac{p}{\|1-\mathbf{m}\|_1}$, otherwise it is

[1]https://github.com/pytorch/fairseq/tree/master/examples/translation

assigned to $\frac{1-p}{\|\mathbf{m}\|_1}$ if $m_i = 1$. Conversely, If a word y_j is not picked which indicates $m_j = 1$, its attention value A'_{ji} is assigned to $\frac{p}{\|\mathbf{m}\|_1}$ if $m_i = 0$, otherwise it is $\frac{1-p}{\|1-\mathbf{m}\|_1}$ if $m_i = 1$.

References

Edunov, S., Ott, M., Auli, M., and Grangier, D. Understanding back-translation at scale. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

Ott, M., Edunov, S., Grangier, D., and Auli, M. Scaling neural machine translation. *arXiv preprint arXiv:1806.00187*, 2018.

Schuster, M. and Nakajima, K. Japanese and korean voice search. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012.

Sennrich, R., Haddow, B., and Birch, A. Neural machine translation of rare words with subword units. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2016.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Zhu, J., Xia, Y., Wu, L., He, D., Qin, T., Zhou, W., Li, H., and Liu, T.-Y. Incorporating bert into neural machine translation. *arXiv preprint arXiv:2002.06823*, 2020.