NICT Kyoto Submission for the WMT'21 Quality Estimation Task: Multimetric Multilingual Pretraining for Critical Error Detection

Raphael Rubino and **Atsushi Fujita** and **Benjamin Marie**
National Institute of Information and Communications Technology
3-5 Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0289, Japan
raphael.rubino,atsushi.fujita,bmarie@nict.go.jp

Abstract

This paper presents the NICT Kyoto submission for the WMT'21 Quality Estimation (QE) Critical Error Detection shared task (Task 3). Our approach relies mainly on QE model pretraining for which we used 11 language pairs, three sentence-level and three word-level translation quality metrics. Starting from an XLM-R checkpoint, we perform continued training by modifying the learning objective, switching from masked language modeling to QE oriented signals, before finetuning and ensembling the models. Results obtained on the test set in terms of correlation coefficient and F-score show that automatic metrics and synthetic data perform well for pretraining, with our submissions ranked first for two out of four language pairs. A deeper look at the impact of each metric on the downstream task indicates higher performance for token oriented metrics, while an ablation study emphasizes the usefulness of conducting both self-supervised and QE pretraining.

1 Introduction

This paper describes the NICT Kyoto submission to the WMT'21 Quality Estimation (QE) shared task. We participated in Task 3 "Critical Error Detection" involving four language pairs, namely English–Chinese, English–Czech, English–Japanese and English–German. A critical error is defined as a translation error falling into one of the following five categories: toxicity, health or safety risk, named entity, sentiment polarity and number or unit deviation.[1]

The objective of the task is to classify a sequence pair, composed of a sentence in the source language and its automatic translation in the target language, in a binary fashion whether it contains or not at least one of the five types of critical errors. This task differs from the other QE tasks as not all translation errors should be detected but only critical ones. Labels were produced by majority vote over three annotators for each pair leading to two possible classes: *ERR* (or class 1) when at least one critical error is spotted and *NO* (or class 0) when no critical errors are present.

Our approach relies mainly on QE model pretraining leveraging a large amount of synthetic data produced using parallel corpora and MT systems. Because annotating translations for critical error is costly, we propose to pretrain a model on translation quality scores computed with automatic metrics. To capture multiple translation error granularities during pretraining, we employ multiple metrics and evaluate their performance individually on the downstream task. Additionally, we pretrain the QE model jointly on all WMT QE shared tasks language pairs as a data augmentation method. Transfer learning is then conducted for each language pair by finetuning the pretrained model on the downstream task with the officially released training data annotated with critical errors.

The remainder of this paper is organized as follows. In Section 2, we introduce our approach involving multimetric and multilingual pretraining. In Section 3, the data, tools and training procedure are presented, followed by the experimental results and their analysis in Section 4, before the conclusion in Section 5.

2 Multimetric & Multilingual Pretraining

Multilingual pretrained masked language models (LMs) were shown to perform well in several downstream natural language processing tasks (Devlin et al., 2019; Conneau et al., 2020; Liu et al., 2020). Starting from an XLM-R checkpoint (Conneau et al., 2020), we performed continued (or intermediate) training (Phang et al., 2018; Rubino and Sumita, 2020) with large amount of automatically

[1] More details about these categories and the task itself can be found here: http://statmt.org/wmt21/quality-estimation-task.html

Proceedings of the Sixth Conference on Machine Translation (WMT), pages 941–947
November 10–11, 2021.

translated source language texts (thereafter called *synthetic data*), replacing the masked LM objective with QE oriented ones. Because XLM-R is multilingual and all languages in this model share a common vocabulary of sub-words, we decided to conduct QE pretraining on the 11 language pairs from all subtasks of WMT'21 QE. These language pairs all share English, whether on the source or target side, and this method can be seen as a data augmentation approach to increase vocabulary coverage.

The objective of QE Task 3 is to classify sentence pairs in a binary fashion. Formally, given a source sequence s and its translation t, we want to learn a function f: $f_\theta(s,t) \rightarrow y$ where $y \in \{0,1\}$ is the class associated with the sequence pair (s,t) and θ represents the model parameters. While fine-tuning a pretrained model on the official QE task 3 data allows us to directly learn model parameters approximating y given (s,t), we do not have such classes for synthetic data. We decided to use MT automatic metric scores as objective instead, assuming that critical error classes could correlate with translation quality scores at least in extreme cases (e.g. no translation errors also means no critical errors).

Several automatic metrics are used by the research community to evaluate the performance of MT systems by measuring translation accuracy against a human-produced reference at different granularity levels. We opted for metrics capturing quality information at the character (chrF (Popović, 2017)), token (TER (Snover et al., 2006)) and token n-gram (BLEU (Papineni et al., 2002)) levels. For the latter, the smoothed sentence-level BLEU was chosen (Chen and Cherry, 2014). In addition to sentence-level metrics, token-level binary tags were also extracted following the usual procedure to determine post-editing effort (Specia et al., 2020).[2]

To allow for sentence-level QE predictions, we added a feed-forward layer on top of XLM-R for each of the three metrics employed without parameter sharing, following:

$$\hat{y}_s = \tanh\left(\phi(h)W_{s1} + b_{s1}\right)W_{s2} + b_{s2} \quad (1)$$

where $\hat{y}_s \in \mathbb{R}^1$ is the sentence-level score, $W_{s1} \in \mathbb{R}^{d\times d}$, $b_{s1} \in \mathbb{R}^d$, $W_{s2} \in \mathbb{R}^{d\times 1}$ and $b_{s2} \in \mathbb{R}^1$ are parameters of the model with dimensionality $d = 1,024$, ϕ is a pooling function and $h \in \mathbb{R}^{n\times d}$ is the set of contextual embeddings corresponding to the n tokens in (s,t). The pooling function is the *class* token added at the beginning of each input sequence. For token-level predictions, we used a linear transformation from contextual embeddings to two-dimensional output (for binary token-level classes): $\hat{y}_t = \text{softmax}\left(hW_t + b_t\right)$, with $\hat{y}_t \in \mathbb{R}^{n\times 2}$ are token-level scores, $W_t \in \mathbb{R}^{d\times 2}$ and $b_t \in \mathbb{R}^2$ are the parameter matrix and bias. Parameters of the model are learned with mini-batch stochastic gradient descent based on losses computed for sentence-level and token-level predictions. For the former loss, we used mean squared error, while cross-entropy was used for the latter. All losses are linearly summed with equal weights before back-propagation. The parameters of the classification and regression heads are optimized along with XLM-R.

3 Data and Tools

This section presents the data used in our experiments, including the synthetic data produced for pretraining and the official QE task 3 corpora, along with the tools required to train our models and the procedure employed for both pretraining and fine-tuning.

3.1 Datasets

In order to gather as much data as possible for many language pairs, we collected all parallel data from the QE shared tasks (from all subtasks). Additionally, we retrieved parallel data from the WMT news translation task (Barrault et al., 2020) and from OPUS (Tiedemann, 2016).[3] The source side of these parallel corpora was translated using publicly available neural MT models based on the Transformer architecture (Vaswani et al., 2017). For Estonian–English (et–en), Nepalese-English (ne–en), Romanian–English (ro–en), Russian–English (ru–en), Sinhala–English (si–en), English–German (en–de) and English–Chinese (en–zh), we used the MT systems made available by the shared task organizers,[4] while for English–Czech (en–cs), English–Japanese (en–ja), Khmer–English (km–en) and Pashto–English (ps–en), we used the mBART50

[2]Scripts and procedure available at https://github.com/deep-spin/qe-corpus-builder

[3]The corpora from OPUS used in our experiments are: Common Crawl, ParaCrawl, OpenSubtitles, DGT, IWSLT, KFTT and XLEnt.

[4]Links to models available at https://github.com/facebookresearch/mlqe/blob/master/nmt_models/README-models.md

Lang. src	Lang. tgt	Sent.	Token src	Token tgt	Type src	Type tgt
Synthetic Data (pretraining)						
en	cs	14.1M	244.4M	220.2M	2.3M	2.5M
en	de	22.3M	477.5M	442.9M	2.5M	4.6M
en	ja	3.3M	64.7M	86.7M	1.2M	732.1k
en	zh	16.2M	407.2M	350.4M	1.1M	1.1M
et	en	14.8M	143.3M	176.8M	2.3M	0.9M
km	en	3.7M	47.7M	34.8M	1.3M	480.5k
ne	en	0.9M	10.1M	8.5M	307.6k	343.2k
ps	en	1.0M	11.6M	10.2M	332.6k	190.3k
ro	en	2.3M	55.7M	51.9M	331.7k	261.1k
ru	en	5.0M	82.1M	90.1M	1.8M	0.9M
si	en	1.4M	17.6M	12.8M	366.7k	344.4k
Official QE Task 3 Data (finetuning)						
en	cs	7.5k	122.2k	125.9k	23.6k	22.5k
en	de	7.9k	127.7k	154.6k	24.7k	19.6k
en	ja	7.7k	126.3k	213.7k	24.6k	12.8k
en	zh	6.9k	110.7k	122.9k	21.9k	12.9k

Table 1: Number of sentences (*Sent.*), tokens and types in the source (*src*) and target (*tgt*) corpora used in our experiments (M stands for millions and k for thousands).

model (Liu et al., 2020; Tang et al., 2020).[5]

Statistics about the synthetic corpora after translation are presented in Table 1, along with the official QE data for Task 3 released by the shared task organizers. After deduplicating and cleaning the synthetic corpora produced to conduct QE pretraining, the total amount of data reached 72.3M triplets (source, translation and reference sentences).

3.2 Tools

Data preprocessing was conducted using the tokenizer and truecaser from the Moses distribution (Koehn et al., 2007), except for Chinese, Japanese, Nepalese and Sinhala, for which the tokenization was conducted using *jieba*,[6] *KyTea*[7] and FLORES (Goyal et al., 2021) respectively.

To compute the sentence-level and token-level scores, we used automatic metrics implementations available in the tools *SacreBLEU* (Post, 2018) for BLEU and chrF and *tercom* (Snover et al., 2006) for TER and token-level classes.

The XLM-R checkpoint used was the *xlm-roberta-large* from HuggingFace Transformers library (Wolf et al., 2020). We used in-house Pytorch (Paszke et al., 2019) code and V100 GPUs hardware for QE pretraining and finetuning, 8 GPUs for the former step and 1 GPU for the latter.

3.3 Training Procedure

Model pretraining on synthetic data was conducted for one epoch (approx. 500k updates) with batches of 128 source and target sequences for a total training time of 3 days. The AdamW optimizer (Loshchilov and Hutter, 2019) was used with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 1 \times 10^{-6}$, while the weight decay was set to 0. A linear learning rate warmup was used during the first 50k updates to reach a maximum value of 5×10^{-6}, which remained without decay until the end of the first epoch. The dropout rates were set to 0.1 for both the embeddings and the transformer blocks (feed-forward and attention layers). A total of four models were pretrained with different random seeds before being finetuned on the official QE Task 3 data.

To conduct finetuning, we added a classification layer on top of XLM-R following:

$$\hat{y}_e = \text{softmax}(\tanh\left(\phi(h)W_{e1} + b_{e1}\right)W_{e2} + b_{e2}) \quad (2)$$

where $\hat{y}_e \in \mathbb{R}^2$ is the sentence-level probability distribution over the two classes, $W_{e1} \in \mathbb{R}^{d \times d}$, $b_{e1} \in \mathbb{R}^d$, $W_{e2} \in \mathbb{R}^{d \times 2}$ and $b_{e2} \in \mathbb{R}^2$ are parameters of the model with $d = 1,024$. The pooling function ϕ is the same as the one employed during pretraining presented in Section 2. Due to the class imbalance of the critical error dataset, we used the weighted cross-entropy loss function to finetune our models. The weight given to the error class (the least populated) was tuned on the validation set in a grid-search manner, with integer values ranging from 1 to 8.

During finetuning, which lasted 40 minutes per model, we used the validation set to select the best performing models according to the Matthews correlation coefficient (MCC), which is the main metric chosen by the shared task organizers for the final evaluation. One model per seed was selected and a total of four models were ensembled to produce our final submission to the shared task.

4 Results and Analysis

We present in this section the main results obtained on the official shared task test set as reported by the organizers, followed by an analysis with ablation study and various pretraining objectives.

[5] More details about the model available at https://github.com/pytorch/fairseq/tree/master/examples/multilingual

[6] https://github.com/fxsjy/jieba

[7] http://www.phontron.com/kytea/

Lang.	MCC	F1 ERR	F1 NOT	F1 Multi
Official Baseline				
en–cs	0.3875	0.8992	0.4768	0.4287
en–de	0.3974	0.8484	0.5317	0.4511
en–ja	0.2139	0.9505	0.2439	0.2318
en–zh	0.1873	0.8980	0.2694	0.2419
Our Baseline				
en–cs	0.4030	0.8984	0.4985	0.4478
en–de	0.5204	0.8687	0.6495	0.5642
en–ja	0.2523	0.9294	0.3191	0.2966
en–zh	0.2413	0.8667	0.3714	0.3219
Our Ensemble				
en–cs	0.5105	0.9132	0.5949	0.5433
en–de	0.5464	0.8767	0.6667	0.5845
en–ja	0.2375	0.9447	0.2896	0.2736
en–zh	0.3109	0.8833	0.4260	0.3763

Table 2: Results obtained on the test set for the WMT'21 QE shared task, Task 3 "Critical Error Detection". *F1 ERR* denotes the F-score obtained on the error class, *F1 NOT* denotes the F-score obtained on the non-error class, *F1 Multi* stands for the multiplication of *F1 ERR* and *F1 NOT*.

Lang.	MCC	F1 ERR	F1 NOT	F1 Multi
No Checkpoint				
en–cs	0.3844	0.4847	0.8996	0.4360
en–de	0.3796	0.5575	0.8219	0.4582
en–ja	0.1963	0.2047	0.9461	0.1937
en–zh	0.2461	0.3513	0.8948	0.3143
No QE Pretraining				
en–cs	0.4728	0.5593	0.9132	0.5107
en–de	0.5182	0.6192	0.8804	0.5451
en–ja	0.2999	0.3439	0.9441	0.3247
en–zh	0.3649	0.4633	0.8897	0.4122
Checkpoint + QE Pretraining				
en–cs	0.5271	0.6000	0.9266	0.5560
en–de	0.5501	0.6615	0.8829	0.5840
en–ja	0.3286	0.3497	0.9499	0.3322
en–zh	0.3833	0.4784	0.8905	0.4260

Table 3: Results obtained on the WMT'21 QE Task 3 "Critical Error Detection" validation set. All results are obtained with ensemble of 4 models. *No Checkpoint* denotes QE pretraining of randomly initialized XLM-R without usual masked LM pretraining, followed by finetuning, *No QE Pretraining* denotes direct finetuning of an XLM-R checkpoint on the official task specific training data, *Checkpoint + QE Pretraining* is our submission to the shared task based on XLM-R and QE pretraining with finetuning.

4.1 Shared Task Results

The official results reported by the shared task organizers are presented in Table 2. We compare our final ensemble results, obtained with four models trained on different seeds, to our baseline, obtained with a single model. We also include the official baseline provided by the shared task organizers. All our submissions outperform the official baseline and our ensembles reach the highest performance according to the correlation score and F-measure. One exception, however, is for the English–Japanese language pair. Despite several attempts to improve our ensembling method for this pair, we could not improve over our baseline.

A comparison with other shared task participants in terms of MCC and F1 scores shows that our submissions were ranked first for English–Czech and English–German, third for English–Chinese and sixth for English–Japanese. We assume that the smaller amount of synthetic data, as well as a possible preprocessing mismatch between the official data and our synthetically generated corpora, could be the reason behind the low performance of the two latter language pairs. More precisely, the data preprocessing pipeline for English, German and Czech are commonly based on the Moses tokenizer and truecaser, and it is possible to infer the parameters used with these tools by looking at the official training data released for the task. For Chinese and Japanese, however, due to the lack of details given by the shared task organizers, it was not possible to use the same preprocessing tools and parameters with certainty.

4.2 Impact of Pretraining Steps

While our approach relied on a two-step process, QE pretraining on synthetic data followed by finetuning on the task specific training set, we still made use of a pretrained XLM-R model by initiating QE pretraining from a checkpoint. Overall, three steps are thus required to obtain the results presented in Table 2. XLM-R and QE pretraining, as well as producing synthetic data, are the most computationally expensive steps, whereas finetuning is relatively cheap to perform due to the small amount of task specific data. Therefore, we performed an ablation study aiming at evaluating the impact of each pretraining step and ran two sets of experiments following the same experimental setup employed for our main submission to the shared task.

For the first set of experiments, no pretraining of XLM-R was conducted, meaning that we did not start QE pretraining from an existing checkpoint, but instead randomly initialized XLM-R parameters and ran QE pretraining *from scratch* (this setup is noted *No Checkpoint*). For the second set of

Lang.	MCC	F1 ERR	F1 NOT	F1 Multi
TER pretraining				
en–cs	0.4725	0.5605	0.9235	0.5176
en–de	0.5092	0.6378	0.8786	0.5604
en–ja	0.2891	0.3628	0.9490	0.3443
en–zh	0.3284	0.4324	0.9158	0.3960
BLEU pretraining				
en–cs	0.4760	0.5629	0.9266	0.5216
en–de	0.4917	0.6290	0.8725	0.5488
en–ja	0.2982	0.3636	0.9513	0.3459
en–zh	0.3442	0.4450	0.9061	0.4032
chrF pretraining				
en–cs	0.4200	0.4988	0.9210	0.4594
en–de	0.4122	0.5911	0.8540	0.5048
en–ja	0.2375	0.3163	0.9496	0.3004
en–zh	0.2925	0.3838	0.9242	0.3547
All sentence-level pretraining				
en–cs	0.4700	0.5539	0.9258	0.5128
en–de	0.5229	0.6609	0.8726	0.5767
en–ja	0.2982	0.3636	0.9496	0.3453
en–zh	0.3660	0.4639	0.9207	0.4271
All word-level pretraining				
en–cs	0.4697	0.5556	0.9172	0.5096
en–de	**0.5323**	0.6667	0.8728	0.5819
en–ja	0.3100	0.3743	0.9505	0.3558
en–zh	**0.3756**	0.4688	0.9127	0.4279
All metrics pretraining				
en–cs	**0.5015**	0.5796	0.9289	0.5384
en–de	0.5276	0.6431	0.8779	0.5646
en–ja	**0.3131**	0.3824	0.9507	0.3635
en–zh	0.3546	0.4391	0.9112	0.4001

Table 4: Results obtained on the WMT'21 QE Task 3 "Critical Error Detection" validation set with single models (no ensemble) based on various learning objectives used during pretraining. Results in bold indicate the best MCC scores among the pretraining configurations for a given language pair.

experiments, we finetuned the XLM-R checkpoint directly on the task specific data, without conducting QE pretraining. This alleviates the need to produce large amount of synthetic QE data (this setup is noted *No QE Pretraining*). We conducted an additional set of experiments based on XLM-R and QE pretraining without finetuning on the official training set but the obtained results were subpar compared to the baseline, due to the randomly initialized parameters of the classification layer (see eq. (2)) which was not tuned for the task following this configuration. We present the results of the two ablation experiments in Table 3.

While combining both the use of a pretrained XLM-R with masked LM and QE pretraining on synthetic data leads to the best results on the four language pairs, *No QE Pretraining* performs better than the *No Checkpoint* configuration. These results emphasize the usefulness of large self-supervised LM pretraining. The amount of data used for QE pretraining is smaller compared to the large quantity of monolingual and parallel data used to train *xlm-roberta-large*, which could be an explanation for the difference in downstream performances according to the MCC and F1 metrics.

4.3 Impact of Pretraining Objectives

As an additional analysis, we propose to evaluate the impact of different metrics used as pretraining objectives on the downstream critical error detection task. Several independent QE pretraining were conducted for this purpose: one for each sentence-level translation quality metrics, one for the combination of sentence-level metrics and finally one for word-level metrics which includes source, target and gap error predictions as described in Section 2. The finetuning step for each pretrained model is identical, only the learning objective during pretraining differs. The results obtained on the validation set for the critical error detection task are presented in Table 4.

Based on MCC scores, using sentence-level metrics during pretraining is not leading to the best downstream performance compared to using word-level metrics or combining both sentence and word-level quality indicators. From the three sentence-level metrics used as learning objectives during pretraining, TER and BLEU outperform chrF. For English–German and English–Chinese, using word-level metrics outperforms the combination of all metrics, while it is the opposite for English–Czech and English–Japanese. These results show that the optimal quality indicator for QE pretraining depends on the language pair and the translation direction, and should therefore be considered as a hyper-parameter to be optimized. However, due to the costly nature of large model pretraining, combining multiple translation quality indicators in a multi-task learning fashion appears to be an efficient solution, in addition to using masked LM pretrained model as shown in the results presented in Section 4.2.

5 Conclusion

This paper presented the NICT Kyoto submission for the WMT'21 QE Task 3 "Critical Error Detection". Our submissions were ranked first for two out of four language pairs. Our approach relies mainly on model pretraining with large amount of

synthetic data, followed by finetuning on the official data released for the shared task. We proposed a novel QE pretraining approach which allows for a multimetric learning objective based on relatively cheap to compute MT automatic metrics. An analysis of each automatic metric used during QE pretraining shows the complementarity of metrics both at level of sentences and words. The ablation study emphasized the usefulness of both self-supervised and QE pretraining. Future work focuses on exploring additional metrics and their performance on various downstream QE tasks.

Acknowledgements

We would like to thank the reviewers for their insightful comments and suggestions. A part of this work was conducted under the commissioned research program "Research and Development of Advanced Multilingual Translation Technology" in the "R&D Project for Information and Communications Technology (JPMI00316)" of the Ministry of Internal Affairs and Communications (MIC), Japan, and supported by JSPS KAKENHI grant numbers 20K19879 and 19H05660.

References

Loïc Barrault, Magdalena Biesialska, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Matthias Huck, Eric Joanis, Tom Kocmi, Philipp Koehn, Chi-kiu Lo, Nikola Ljubešić, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Santanu Pal, Matt Post, and Marcos Zampieri. 2020. Findings of the 2020 Conference on Machine Translation (WMT20). In *Proceedings of the Fifth Conference on Machine Translation*, pages 1–55, Online. Association for Computational Linguistics.

Boxing Chen and Colin Cherry. 2014. A Systematic Comparison of Smoothing Techniques for Sentence-Level BLEU. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 362–367, Baltimore, USA. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised Cross-lingual Representation Learning at Scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, USA. Association for Computational Linguistics.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzman, and Angela Fan. 2021. The FLORES–101 Evaluation Benchmark for Low-Resource and Multilingual Machine Translation. *arXiv preprint arXiv:2106.03193*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 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. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual Denoising Pre-training for Neural Machine Translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.

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, Philadelphia, USA. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems*, pages 8026–8037. Curran Associates, Inc.

Jason Phang, Thibault Févry, and Samuel R Bowman. 2018. Sentence Encoders on STILTs: Supplementary Training on Intermediate Labeled-data Tasks. *arXiv preprint arXiv:1811.01088*.

Maja Popović. 2017. chrF++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618, Copenhagen, Denmark. Association for Computational Linguistics.

Matt Post. 2018. A Call for Clarity in Reporting BLEU Scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

Raphael Rubino and Eiichiro Sumita. 2020. Intermediate Self-supervised Learning for Machine Translation Quality Estimation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4355–4360, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231, Cambridge, USA. Association for Machine Translation in the Americas.

Lucia Specia, Frédéric Blain, Marina Fomicheva, Erick Fonseca, Vishrav Chaudhary, Francisco Guzmán, and André F. T. Martins. 2020. Findings of the WMT 2020 Shared Task on Quality Estimation. In *Proceedings of the Fifth Conference on Machine Translation*, pages 743–764, Online. Association for Computational Linguistics.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual Translation with Extensible Multilingual Pretraining and Finetuning. *arXiv preprint arXiv:2008.00401*.

Jörg Tiedemann. 2016. OPUS – Parallel Corpora for Everyone. In *Proceedings of the 19th Annual Conference of the European Association for Machine Translation: Projects/Products*, Riga, Latvia. Baltic Journal of Modern Computing.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *Advances in Neural Information Processing Systems*, pages 5998–6008. Curran Associates, Inc.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.