BJTU-Toshiba's Submission to WMT22 Quality Estimation Shared Task

Hui Huang[†] **Hui Di**[‡] **Chunyou Li**[†] **Hanming Wu**[†] **Kazushige Oushi**[‡]
Yufeng Chen[†] **Jian Liu**[†] **Jin'an Xu**[†]
[†]Beijing Key Lab of Traffic Data Analysis and Mining, Beijing Jiaotong University
[‡]Research&Development Center, Toshiba (China) Co., Ltd.
{18112023, 21120368, 21120416, chenyf, jianliu, jaxu}@bjtu.edu.cn
dihui@toshiba.com.cn, kazushige.ouchi@toshiba.co.jp

Abstract

This paper presents the BJTU-Toshiba joint submission for WMT 2022 quality estimation shared task. We only participate in Task 1 (quality prediction) of the shared task, focusing on the sentence-level MQM prediction. The techniques we experimented with include the integration of monolingual language models and the pre-finetuning of pre-trained representations. We tried two styles of pre-finetuning, namely Translation Language Modeling and Replaced Token Detection. We demonstrate the competitiveness of our system compared to the widely adopted XLM-RoBERTa baseline. Our system is also the top-ranking system on the Sentence-level MQM Prediction for the English-German language pair[1].

1 Introduction

Machine translation Quality Estimation (QE) aims to evaluate the quality of machine translation automatically without reference. Compared with commonly used machine translation metrics such as BLEU (Papineni et al., 2002), QE can be applicable to the case where references are unavailable. It has a wide range of applications in post-editing and quality control for machine translation.

This paper introduces in detail the joint submission of Beijing Jiaotong University and Toshiba (China) Corporation to the quality estimation shared task in the 7th Conference on Machine Translation (WMT22), and we mainly focus on the Task 1: quality prediction. This year, the quality prediction task consists of two annotations (DA and MQM) and two levels (sentence-level and word-level), and we only participate in the Sentence-level MQM prediction, of which the goal is to predict the MQM score (Freitag et al., 2021) for each source-target sentence pair. Three language pairs are involved: English-German, Chinese-English and English-Russian, with roughly 10K-20k training pairs provided for each direction.

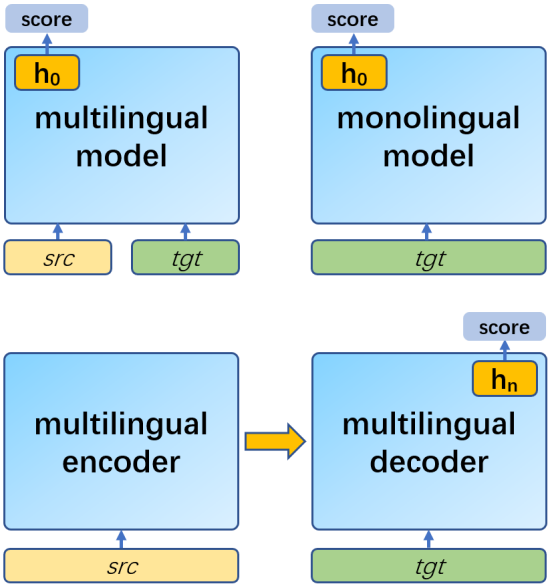

Figure 1: The three QE architectures we adopted.

Our system is mainly based on the ensemble of multiple pre-trained models, both monolingual and multilingual. The monolingual models receive only the target sequence to perform regression (only estimating the target fluency). The multilingual models receive both the source and target sequence to perform regression. We also use in-domain parallel data to pre-finetune the pre-trained models, to adapt their representations to the target language and domain. We try two styles of pre-finetuning, namely Translation Language Model (TLM) and Replaced Token Detection (RTD). The translation language model is to predict the random masked tokens based on the concatenation of source-target pairs. The RTD is to first randomly replace some tokens by another generator, then to detect which token is replaced. Different models are ensembled to get further improvement.

[1]Our codes are openly available at the public repository https://github.com/HuihuiChyan/AwesomeQE.

Proceedings of the Seventh Conference on Machine Translation (WMT), pages 621–626
Abu Dhabi, December 7–8, 2022.

Direction	Model	Type	Input	Spearman	Pearson
En-De	mBERT	multilingual understanding	*src-tgt*	0.3621	0.3484
	XLM	multilingual understanding	*src-tgt*	0.3692	0.3682
	XLMR-large	multilingual understanding	*src-tgt*	0.4548	0.4235
	mBART	multilingual encoder-decoder	*src-tgt*	0.3890	0.3946
	OpusMT	multilingual encoder-decoder	*src-tgt*	0.3981	0.4184
	BERT-base	monolingual understanding	*tgt*	0.4620	0.4381
	BERT-large	monolingual understanding	*tgt*	0.4963	0.4574
	Electra-base	monolingual understanding	*tgt*	0.5069	0.4654
	Electra-large	monolingual understanding	*tgt*	**0.5413**	0.4974
Zh-En	XLM	multilingual understanding	*src-tgt*	0.2503	0.1494
	XLMR-large	multilingual understanding	*src-tgt*	0.2614	0.1083
	mBERT	multilingual understanding	*src-tgt*	0.2661	0.1439
	mBART	multilingual encoder-decoder	*src-tgt*	0.2332	0.1021
	OpusMT	multilingual encoder-decoder	*src-tgt*	0.2353	0.1196
	Electra-base	monolingual understanding	*tgt*	0.2337	0.1412
	BERT-large	monolingual understanding	*tgt*	0.2425	0.1149
	Roberta-large	monolingual understanding	*tgt*	0.2523	0.0969
	Deberta-large	monolingual understanding	*tgt*	0.2514	0.1024
	Deberta-v3-large	monolingual understanding	*tgt*	0.2714	0.1486
	Electra-large	monolingual understanding	*tgt*	**0.2829**	0.1475
En-Ru	mBERT	multilingual understanding	*src-tgt*	0.3897	0.3744
	XLM	multilingual understanding	*src-tgt*	0.4281	0.4143
	XLMR-large	multilingual understanding	*src-tgt*	0.4502	0.4144
	mBART	multilingual encoder-decoder	*src-tgt*	0.4174	0.4137
	OpusMT	multilingual encoder-decoder	*src-tgt*	0.4207	0.3884
	BERT-base	monolingual understanding	*tgt*	0.4686	0.3964
	BERT-large	monolingual understanding	*tgt*	0.4899	0.4280
	Roberta-large	monolingual understanding	*tgt*	**0.5175**	0.4265

Table 1: Experiment results on the DEV set of multilingual and monolingual baselines. Results are presented in an ascending order with respect to the spearman's ranking correlation coefficient.

2 Methods

2.1 Architecture

In this work, we perform massive comparison between the multilingual models and monolingual models on QE. Our backbone network is based on several multilingual understanding models, including Multilingual BERT (Devlin et al., 2018), XLM (Lample and Conneau, 2019), XLM-RoBERTa (Ruder et al., 2019), etc. Meanwhile, we integrate several monolingual models, including BERT, RoBERTa (Liu et al., 2020b), DeBERTa (He et al., 2021b), DeBERTa-v3 (He et al., 2021a), Electra (Clark et al., 2020), etc. We also perform estimation on multilingual encoder-decoder models, including Multilingual BART (Liu et al., 2020a) and OpusMT (Tiedemann and Thottingal, 2020)[2].

For multilingual understanding models, we feed the concatenation of *src* (source sentence) and *tgt* (machine translated sentence) to the model, and take the first output hidden state for regression. For monolingual understanding models, we simply feed the *tgt* to the model, and take the first output hidden state for regression. For encoder-decoder

[2]To be specific, we use the released models from https://huggingface.co/Helsinki-NLP/opus-mt-en-de, https://huggingface.co/Helsinki-NLP/opus-mt-zh-en, and https://huggingface.co/Helsinki-NLP/opus-mt-en-ru for En-De, Zh-En and En-Ru, respectively

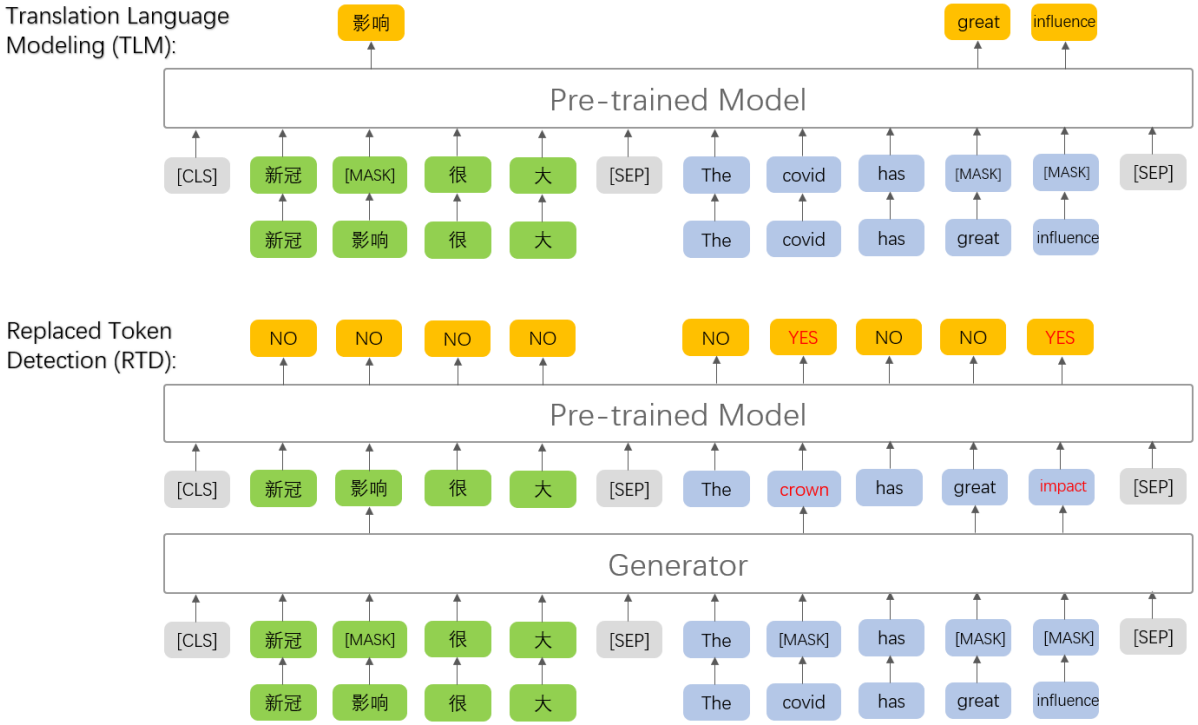

Figure 2: The two different pre-finetuning schemes. Notice for RTD, some masked tokens may be restored correctly by the generator, and we only detect the mismatched tokens.

style models, we feed the *src* to the encoder, the *tgt* to the decoder, and take the last hidden state corresponding to the last token of the *tgt* for regression. All three architectures are depicted in Figure 1.

As shown in Table 1, the monolingual baselines can surpass the multilingual baselines in all directions. Although the alignment information is absent, estimation can still be performed solely on the target text to estimate the fluency. In this year, the MQM prediction data are actually the submissions from the translation evaluation task, therefore most *tgt*s are roughly correct translations aligned with the source sentence, and most translation errors are very subtle. Therefore, it would be easier for the model to estimate the fluency instead of the alignment. With the help of powerful monolingual models, we are able to achieve higher estimation accuracy based solely on the target input.

2.2 Adaptative Pre-finetuning

Fine-tuning pre-trained language models on domain-relevant unlabeled data has become a common strategy to adapt the pretrained parameters to downstream tasks (Gururangan et al., 2020). Previous works also demonstrate the necessity of pre-finetuning when performing QE on pretrained models (Kim et al., 2019; Hu et al., 2020). In this work, we perform two methods to pre-finetune the pre-trained models, namely Translation Language Modeling (TLM) (Lample and Conneau, 2019) and Replaced Token Detection (RTD) (Clark et al., 2020), as shown in Figure 2.

The TLM simply takes the concatenation of parallel sentence pairs as input, and perform masked language modeling. Therefore, when predicting the masked tokens in one side, the model could utilize its context in the parallel side, learning the bilingual alignment.

On the contrary, instead of masking, RTD corrupts the input by replacing some tokens with samples from the output of a smaller masked language model (Specifically, we use the first 1/3 layers of the pre-trained model to initilize the generator). Then the model is trained as a discriminator that predicts for every token whether it is an original or a replacement, learning to distinguish real input tokens from plausible replacements.

Compared with TLM, RTD mainly has three benefits: 1) The corruption procedure solves a mismatch in MLM (or TLM) where the network sees artificial [MASK] tokens during pre-training but not when being fine-tuned on downstream tasks. 2) The loss is calculated on all tokens instead of a subset, therefore improving the pre-finetuning efficiency. 3) The mismatch produced by a language

model is more subtle than random masking or replacement, therefore the pre-finetuning naturally fits the final objective, whitch is to detect subtle semantic mismatch.

Direction	Model	Spearman	Pearson
En-De	XLM-R-large	0.4548	0.4235
	w/ TLM	0.5084↑	0.4959
	w/ RTD	0.5109↑	0.5024
	BERT-large	0.4963	0.4574
	w/ TLM	0.5033↑	0.4593
	w/ RTD	0.5127↑	0.4704
	Electra-large	0.5413	0.4974
	w/ TLM	0.4748↓	0.4396
	w/ RTD	0.5220↓	0.4871
	Ensemble	0.5809	0.5313
Zh-En	XLM-R-large	0.2614	0.1083
	w/ TLM	0.2590↓	0.1167
	w/ RTD	0.2888↑	0.1332
	mBERT	0.2661	0.1439
	w/ TLM	0.2912↑	0.1360
	w/ RTD	0.2649↓	0.1254
	Deberta-v3-large	0.2714	0.1486
	w/ TLM	0.2561↓	0.1227
	w/ RTD	0.3076↑	0.1787
	Electra-large	0.2829	0.1475
	w/ TLM	0.2361↓	0.1051
	w/ RTD	0.2493↓	0.1190
	Ensemble	0.3231	0.1692
En-Ru	XLM-R-large	0.4502	0.4144
	w/ TLM	0.4956↑	0.3963
	w/ RTD	0.5092↑	0.3954
	BERT-large	0.4986	0.3964
	w/ TLM	0.5030↑	0.4189
	w/ RTD	0.5170↑	0.4453
	Roberta-large	0.5175	0.4265
	w/ TLM	0.5129↓	0.3979
	w/ RTD	0.5321↑	0.4171
	Ensemble	0.5799	0.4544

Table 2: Experiment results on the DEV set of different pre-finetuning methods and ensemble result.

Both methods are performed on millions of parallel sentence pairs. We firstly train a BERT-based domain classifier to select the in-domain parallel data.

Direction	Model	Input	Spearman
Zh-En	Deberta-v3-large	*tgt*	0.2892
	Deberta-v3-large	*src-tgt*	0.3076↑
En-Ru	Roberta-large	*tgt*	0.5245
	Roberta-large	*src-tgt*	0.5321↑

Table 3: Experiment results on the DEV set of pre-finetuned models with bilingual or monolingual input.

Here we use the parallel data from the general translation task of WMT22[3], which contains roughly 20 million pairs for Zh-En and En-De, and 10 million for En-Ru. Specifically, the sentence pairs in the QE training set are deemed as in-domain data, and we randomly sample the same size of data as the general-domain data, and the BERT model is fine-tuned on them as a binary classifier. After that, we select roughly 1 million sentence pairs for each direction.

Notice that for monolingual models we also perform TLM with bilingual input, expecting to introduce further gain with the help of extra information.

As shown in Table 2, both TLM and RTD can improve the estimation accuracy significantly. The multilingual pre-trained model is trained on hundreds of languages simultaneously without any cross-lingual supervision. The monolingual pre-trained model is trained only on the target language. Therefore, adaptation is necessary for both models to solve the language and domain mismatch. Also, the RTD outperforms TLM in most cases, verifying that RTD is more suitable as the pre-finetuning scheme for QE task. Since QE is also targeted at detecting mismatched and disfluent tokens, therefore RTD is more in line with the QE objective.

We also found that after the pre-finetuning step, it would be helpful to feed the bilingual input to the monolingual models, as shown in Table 3. Although monolingual models did not see any text from the source language during pre-training, the knowledge between different languages is transferrable (Artetxe et al., 2020), therefore the fine-tuned model on the target side can also be used to model the semantics of the source side. Besides, subword segmentation also enables the model to represent sequences from unseen language.

The only exception is on Electra, where pre-finetuning brings degradation in all cases. It is possibly because we use the released generator instead

[3]https://www.statmt.org/wmt22/translation-task.html

of using the first few layers to initialize a generator, but it is still confusing why their released generator (which is also used to perform replacement during pre-training stage) would lead to degradation.

2.3 Model Ensemble

Till now, we have obtained different QE models trained with different data and strategies, which can capture different information from the same text. While previous work resort to statistical learning methods to perform model ensemble (Kepler et al., 2019), we think their methods might be overfitting. Therefore, we simple take the average of different predictions (normalized between 0 and 1) as the ensemble result. More specifically, we try different combinations of all available predictions (which are all listed in the Table 2), and make submissions based on the best ensemble result on the DEV set. The performance gain compared to single model is significant as can be seen in Table 2.

3 Conclusion

In this paper, we present our WMT22 QE shared task submission to the sentence-level MQM prediction. We perform massive comparison and demonstrate the effectiveness of monolingual language model. We verify that the pre-trained models can be further improved on target language and target domain via pre-finetuning, and we propose different strategies to pre-finetune the model.

As the machine translation has been developing rapidly, the translation errors current MT system makes have also become more than shallow disalignment. While MT systems are mostly trained with massive parallel data, using the same amount of parallel data to train another QE model seems inefficient, and the monolingual knowledge contained in monolingual models can be more helpful than we expected. While previous work mainly rely on the semantic alignment to perform QE, we think it might be a better option to rely more on monolingual fluency in real applications.

Acknowledgements

The research work descried in this paper has been supported by the National Key RD Program of China (2020AAA0108001) and the National Nature Science Foundation of China (No. 61976015, 61976016, 61876198 and 61370130). The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve this paper.

References

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *ICLR*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Markus Freitag, George Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021. Experts, errors, and context: A large-scale study of human evaluation for machine translation. *Transactions of the Association for Computational Linguistics*, 9:1460–1474.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021a. Debertav3: Improving deberta using electra-style pretraining with gradient-disentangled embedding sharing.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021b. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

Chi Hu, Hui Liu, Kai Feng, Chen Xu, Nuo Xu, Zefan Zhou, Shiqin Yan, Yingfeng Luo, Chenglong Wang, Xia Meng, Tong Xiao, and Jingbo Zhu. 2020. The niutrans system for the wmt20 quality estimation shared task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1018–1023, Online. Association for Computational Linguistics.

Fabio Kepler, Jonay Trnous, Marcos Treviso, Miguel Vera, Antnio Gis, M. Amin Farajian, Antnio V. Lopes, and Andr F. T. Martins. 2019. Unbabel participation in the wmt19 translation quality estimation shared task. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 80–86, Florence, Italy. Association for Computational Linguistics.

Hyun Kim, Joon-Ho Lim, Hyun-Ki Kim, and Seung-Hoon Na. 2019. Qe bert: Bilingual bert using multi-task learning for neural quality estimation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 87–91, Florence, Italy. Association for Computational Linguistics.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *Advances in Neural Information Processing Systems (NeurIPS)*.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020a. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2020b. Ro{bert}a: A robustly optimized {bert} pretraining approach.

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, Pennsylvania, USA. Association for Computational Linguistics.

Sebastian Ruder, Anders Søgaard, and Ivan Vulić. 2019. Unsupervised cross-lingual representation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 31–38, Florence, Italy. Association for Computational Linguistics.

Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT — Building open translation services for the World. In *Proceedings of the 22nd Annual Conferenec of the European Association for Machine Translation (EAMT)*, Lisbon, Portugal.