The NiuTrans System for the WMT20 Quality Estimation Shared Task

Chi Hu[†] **Hui Liu**[†] **Kai Feng**[†] **Chen Xu**[†] **Zefan Zhou**[†] **Shiqin Yan**[†]
Yingfeng Luo[†] **Chenglong Wang**[†] **Xia Meng**[†] **Nuo Xu**[†]
Tong Xiao[†‡] **Jingbo Zhu**[†‡]
[†]NLP Lab, School of Computer Science and Engineering
Northeastern University, Shenyang, China
[‡]NiuTrans Reasearch, Shenyang, China
huchinlp@gmail.com, liuhui0717@outlook.com,
{xiaotong,zhujingbo}@mail.neu.edu.cn,

Abstract

This paper describes the submissions of the NiuTrans Team to the WMT 2020 Quality Estimation Shared Task (Specia et al., 2020). We participated in all tasks and all language pairs. We explored the combination of transfer learning, multi-task learning and model ensemble. Results on multiple tasks show that deep transformer machine translation models and multilingual pretraining methods significantly improve translation quality estimation performance. Our system achieved remarkable results in multiple level tasks, e.g., our submissions obtained the best results on all tracks in the sentence-level Direct Assessment task[1].

1 Introduction

Quality estimation (QE) evaluates the quality of machine translation output without human reference translations (Blatz et al., 2004). It has a wide range of applications in post-editing and quality control for machine translation.

We participated in all tasks and language pairs at the WMT 2020 QE shared task[2], including sentence-level Direct Assessment tasks, word and sentence-level post-editing effort tasks, and document-level QE tasks. We investigated transfer learning and ensemble methods using recently proposed multilingual pre-trained models (Devlin et al., 2019; Conneau et al., 2020) as well as deep transformer models (Wang et al., 2019a). Our main contributions are as follows:

- We apply multi-phase pretraining (Gururangan et al., 2020) methods under both high- and low-resource settings to QE tasks.
- We incorporate deep transformer NMT models into QE models.
- We propose a simple strategy to convert document-level tasks into word- and sentence-level tasks.
- We explore effective ensemble methods for both word- and sentence-level predictions.

Results on different level tasks show that our methods are very competitive. Our submissions achieved the best Pearson correlation on all language pairs of the sentence-level Direct Assessment task and the best results on English-Chinese post-editing effort tasks.

We present methods for the sentence-level Direct Assessment task in §2. Then in §3 and §4, we describe our approaches to post-editing tasks and document-level tasks, respectively. System ensemble methods are discussed in §5. We show the detail of our submissions and the results in §6. We conclude and discuss future work in §7.

2 Sentence-level Direct Assessment Task

The sentence-level Direct Assessment task is a new task where sentences are annotated with Direct Assessment (DA) scores by professional translators rather than post-editing labels. DA scores for each sentence are rated from 0 to 100, and participants are required to score sentences according to z-standardized DA scores. The DA task consists of seven tracks for different language pairs and one multilingual track. Submissions were evaluated in terms of Pearson's correlation metric for the DA prediction against human DA (z-standardized mean DA score, i.e., z-mean).

2.1 Datasets and Resources

This task contains 7K sentences for training and 1K sentences for development on each language pair,

[1]Our number of submissions exceeded the daily or total limit.

[2]http://www.statmt.org/wmt20/quality-estimation-task.html

Proceedings of the 5th Conference on Machine Translation (WMT), pages 1018–1023
Online, November 19–20, 2020.

including sentence scores and word probabilities from the NMT models. The organizer also provided parallel data used to train the NMT models except for Russian-English, ranging from high resource (En-De, En-Zh), medium resource (Ro-En), to low-resource (Et-En, Ne-En, Si-En).

In addition to the official data, we also used some multilingual pre-trained models for fine-tuning, including multilingual BERT[3] (mBERT) and XLM-RoBERTa[4] (XLM-R).

2.2 Unsupervised Quality Estimation

Our baseline system was built upon unsupervised quality estimation methods proposed by Fomicheva et al. (2020), which use out-of-box NMT models as sources of information for directly estimating translation quality. We utilized the output sentence probabilities from NMT models as indicators for QE tasks. Given the input sequence $\mathbf{x}$, suppose the decoder generates an output sequence $\mathbf{y} = y_1, \ldots, y_T$ of length T, the probability of generating $\mathbf{y}$ is factorized as:

$$p(\mathbf{y}|\mathbf{x}, \theta) = \prod_{t=1}^{T} p\left(y_t|\mathbf{y}_{<t}, \mathbf{x}, \theta\right) \quad (1)$$

where θ represents model parameters. The output probability distribution $p\left(y_t \mid \mathbf{y}_{<t}, \mathbf{x}, \theta\right)$ is produced by the decoder over the *softmax function*.

We considered the sequence-level translation probability normalized by length:

$$\text{TP} = \frac{1}{T} \sum_{t=1}^{T} \log p\left(y_t|\mathbf{y}_{<t}, \mathbf{x}, \theta\right) \quad (2)$$

And the probability generated from perturbed parameters with dropout, we performed N times inference and used the averaged output:

$$\text{D-TP} = \frac{1}{N} \sum_{n=1}^{N} \text{TP}_{\hat{\theta}^n} \quad (3)$$

2.3 Multi-phase Pretraining

Fine-tuning pre-trained language models have become the foundation of today's NLP (Devlin et al., 2019; Conneau et al., 2020). Recent advances in pre-trained multilingual language models lead to state-of-the-art results on QE tasks (Kim et al., 2019; Kepler et al., 2019a). Similar to Gururangan et al. (2020), we continued training multilingual pre-trained models in both domain- and task-adaptive manners.

Domain-adaptive pretraining uses a straightforward approach–we continue pretraining mBERT and XLM-R on the parallel corpora provided by the organizers, which is used to train the MT systems. Unlike the training data labeled with DA scores, the parallel data for different language pairs vary. The corpus of pre-trained language models also has the problem of data imbalance. In practice, we increased the training frequency of low-resource data.

Task-adaptive pretraining refers to pretraining on the unlabeled training set for a given task. Compared to domain-adaptive pretraining, it uses a far smaller corpus, but the data is much more task-relevant. We used the same models as the domain-adaptive pretraining.

2.4 Fine-tuning

Similar to previous work (Kepler et al., 2019a; Yankovskaya et al., 2019), we used models trained with the above methods as feature extractors for the sentence-level scoring tasks. We treated the scoring task as a regression task. Following standard practice, we added a separator token between source and target sentences and passed the pooled representation from the encoder to a task-specific feed-forward layer for classification. We used the z-standardized mean DA score as the ground truth and minimized the mean squared error during training.

3 Word and Sentence-Level Post-editing Effort Task

This task consists of the word- and sentence-level tracks to evaluate post-editing effort. The word-level tasks predicts OK or BAD tags in both source and target sequences. It evaluates the Matthews correlation coefficient[5] (MCC) for tags. The sentence-level task predicts HTER scores, which is the ratio between the number of edits needed and the reference translation length. It evaluates Pearson's correlation for the HTER prediction. There are two language pairs in both the word- and sentence-level tasks, including English-German (En-De) and English-Chinese (En-Zh).

[3] https://huggingface.co/bert-base-multilingual-cased

[4] https://github.com/facebookresearch/XLM

[5] https://en.wikipedia.org/wiki/Matthews_correlation_coefficient

3.1 Datasets and Resources

The labeled data consists of 7K sentences for training and 1K sentences for development for each language pair. We used the additional parallel data provided by the organizers to train predictors, containing about 20M En-Zh sentence pairs and 23M En-De sentence pairs after pre-processing with the NiuTrans SMT toolkit (Xiao et al., 2012). Pretrained language models include mBERT and XLM-R, were also used for Task 2.

3.2 Predictor-Estimator Models

The predictor-estimator architecture and its variants (Kim et al., 2017; Kepler et al., 2019b) had established state-of-the-art on WMT QE tasks. The system consists of a word prediction module (predictor) trained from additional large-scale parallel corpora and a quality estimation module (estimator) trained from quality-annotated data.

For the sentence-level tasks and target-side word-level tasks, we employed the official bi-RNN predictor-estimator trained with `OpenKiwi` (Kepler et al., 2019b) as the baseline. Similar to Wang et al. (2019b), we used NMT models trained with back-translation as predictors.

The original predictor and estimator use RNNs to encode the source and predict tags or scores. We also implemented two transformer-based predictors which replace the RNN with transformer (Vaswani et al., 2017) or deep transformer architectures (Wang et al., 2019a; Li et al., 2019). We compared different tokenizing strategies such as word segmentation and byte pair encoding (BPE) (Sennrich et al., 2016) for all language pairs.

3.3 Multi-task learning

The word- and sentence-level tasks are highly related to their annotations are commonly based on the HTER measure. We used a linear summation of sentence-level and target word-level objective losses as follows:

$$\mathcal{L} = \mathcal{L}_{mt.word} + \mathcal{L}_{mt.gap} + \mathcal{L}_{HTER} \quad (4)$$

where the components denote the loss of target-word, target-gap, and predictions for HTER score.

We also trained models using source sentence and origin/post-edited MT output to predict the source-side word level tags:

$$\mathcal{L}_{SRC} = \mathcal{L}_{src-mt} + \mathcal{L}_{src-pe} \quad (5)$$

4 Document-Level QE Task

This task aims to predict document-level quality scores as well as fine-grained annotations. Each document is annotated for translation errors with word span, severity, and error type[6]. Additionally, there are document-level scores (MQM scores) generated from the error annotations using the method proposed by Torrón and Koehn (2016). The annotation task evaluates F1 scores on the gold annotations. The scoring task evaluates the Pearson's correlation between the gold and predicted MQM scores.

4.1 Datasets and Resources

We also used 35M WMT14 En-Fr parallel data to train our predictors for the annotation task except for the official 1,448 En-Fr documents. For the scoring task, we used pre-trained language models, including mBERT and XLM-R.

4.2 Document-level Annotating Task

Following Kepler et al. (2019a), we treated the document-level annotation problem as a word-level task, with each sentence processed separately. We tokenized the training set and tagged each token with an OK/BAD tag. Specifically, each token was labeled as BAD if it contains any character in error spans. Besides token tags, we labeled a gap as BAD if a span begins and ends exactly in its borders. Otherwise, it was labeled as OK. During the test time, we mapped BAD tags to annotations in a single scheme: (a) continuous labels were merged into an error annotation; (b) individual labels were directly converted to error annotations. We ignored the severity information and always treated the error as the most frequent 'major'.

We adopt the predictor-estimator architecture for this task. We implemented our predictors with deep transformers with relative position representation. The settings for model training are described in (Hu et al., 2020). We also compared two tokenization schemes, including word-level tokenization and BPE. Similar to Task 2, we jointly trained our models with target-side word-level and word gap tasks.

4.3 Document-level Scoring Task

We treated the document-level scoring task as a sentence-level task with a simple mapping scheme.

[6] http://www.qt21.eu/mqm-definition/definition-2015-12-30.html

We also ignored all critical and minor errors, and thus the MQM score for each document is calculated by:

$$\text{MQM} = 100 \times (1 - \frac{W \times Count_{major}}{Count_{word}}) \quad (6)$$

where $Count_{major}$ and $Count_{word}$ are the count of major errors and total words, respectively. W denotes the weight of major errors, which is fixed at 5 in our experiments.

Then we score each sentence according to the number of errors it contains:

$$\text{Score}_{\text{sent}} = 100 - W \times Count_{major} \quad (7)$$

We applied the same fine-tuning strategies, as mentioned in Sec 2, to this task. During the test time, the count of errors was retrieved from the predicted score of all sentences. A document score is 0 if it has too many errors.

5 System Ensemble

In addition to training models for each task, we also explored effective ensemble methods to combine outputs for different level tasks.

5.1 Word-level ensemble

We used two approaches to ensemble word-level predictions for Task 2 and Task 3.

Voting-Based Ensemble. Voting is the easiest method to combine predictions from multiple models. We chose the label with the most votes for each token as the output.

Averaging-Based Ensemble. Similar to Kepler et al. (2019a), we used Powell's conjugate direction method to optimize the task metric (MCC or F1 score) and learn the weights of different systems on the development set.

5.2 Sentence-level ensemble

We averaged the predicted scores from multiple models associated with different weights. The weights were also learned on the development set using Powell's method. We removed outliers from the candidate pool to make the prediction more stable.

6 Experiments and Results

6.1 Task 1

Below we describe our systems for Task 1.

Unsupervised baseline. As described in §2, our baseline system leverages the output probabilities from NMT models to assess the sentence score. We performed 20 inference passes and set the dropout rate as 0.3 for all language pairs.

Pair	TP Score	D-TP Score
En-De	0.249	0.273 (+10%)
En-Zh	0.330	0.348 (+5%)
Ro-En	0.648	0.693 (+7%)
Et-En	0.497	0.562 (+13%)
Ne-En	0.431	0.490 (+14%)
Si-En	0.423	0.469 (+11%)
Ru-En	0.518	0.535 (+3%)

Table 1: Pearson (r) correlation between unsupervised methods and human DA judgements on the validation data for sentence-level DA tasks. We mark improvements of D-TP by percentage.

Pair	mBERT	XLM-R	Ensemble
En-De	0.516	0.555	0.562
En-Zh	0.512	0.533	0.551
Ro-En	0.888	0.911	0.917
Et-En	0.809	0.820	0.833
Ne-En	0.816	0.821	0.830
Si-En	0.607	0.670	0.698
Ru-En	0.728	0.796	0.816
Multilingual	-	-	0.732

Table 2: Pearson (r) correlation between pretraining methods and human DA judgements on the test data for sentence-level DA tasks. We only present the results of XLM-R-large for the second method.

Pretraining and fine-tuning. We experimented with different pre-trained models for multi-phase pretraining and fine-tuning. Specifically, we used three model settings, including mBERT-base cased (∼200M parameters), XLM-R-base (∼300M parameters), and XLM-R-large (∼600M parameters). Systems for the first six language pairs in Table 2 were pre-trained on the parallel data while the system for Ru-En was only trained on the task data. We combined predictions on the first six language pairs as the submission to the multilingual task.

As shown in Table 1, unsupervised QE indicators obtained competitive results using sequence-level probability from NMT models. Disturbing the model parameters improves the performance of all language pairs. We did not combine the predictions from unsupervised methods into our submissions.

Table 2 lists the results of the system ensemble

System	Target	Source
RNN-word	0.467	-
Transformer-word	0.511	-
Transformer-subword	0.542	0.292
Deep Transformer-subword	0.545	-
Ensemble	0.610	0.308

Table 3: Results of the English-Chinese post-editing task. 'word' denotes the system uses word-level tokenization.

System	Target	Source
RNN-word	0.395	-
Transformer-word	0.413	-
Transformer-subword	0.451	0.285
Ensemble	0.500	0.347

Table 4: Results of the English-German post-editing tasks.

with pretraining and fine-tuning. We combined predictions from 10 pre-trained models with three different settings: mBERT, XLM-R-base, and XLM-R-large. We only report the results with the highest Pearson (r) correlation on the test data. We observe that larger models consistently outperformed small ones for all language pairs. Besides, ensemble methods significantly improved the performance on the test set. It also shows that the quality estimation of high-resource languages performs far worse than low-resource languages.

6.2 Task 2

For En-Zh, we trained 5-10 single models for each setting: token-based bi-RNNs (RNN-Token), token-based transformer (Trans-Token), BPE-based transformer (Trans-BPE), and BPE-based deep transformer with 25 encoder layers (Deep Trans). For En-De, we created three systems using the same architectures as En-Zh except for the deep transformer. We applied the multi-task learning strategies to the target-side word-level and sentence-level tasks described as §3.

Table 3 shows the results on the English-Chinese word-level task. Deep transformer and BPE tokenization bring the most gains to both the target-side MCC. Results on the English-German task are listed in Table 4. It shows that our ensemble methods are effective in boosting performance across different tasks.

System	F1 Score	Pearson
Transformer-word	0.373	-
Transformer-subword	0.400	-
Deep Transformer	0.402	-
mBERT	-	0.446
XLM-R	-	0.489
Ensemble	0.418	0.494

Table 5: Results of the document-level tasks. The deep transformer model contains 24 encoder layers and 6 decoder layers.

6.3 Task 3

Table 5 shows the results obtained by three different models and the ensemble on the annotation task. BPE brings about 0.03 points improvements of F1 scores on both the validation and test sets. The system ensemble further pushes the score by about 0.02. Table 5 also lists the results of the scoring task. We report the results of two pretraining methods and their ensemble on the test data. XLM-R outperformed the mBERT model by 0.04 points in the Pearson correlation, while the ensemble brought a slight benefit.

7 Conclusion

This paper describes the submissions of the NiuTrans Team to the WMT 2020 QE task. We explored the combination of transfer learning, multi-task learning, and model ensemble. Different level tasks show that deep transformer NMT models and multilingual pretraining methods significantly boost QE models' performance.

Although our system achieved impressive results in all tasks and language pairs, there are still many problems. For instance, the translation quality estimation of low-resource languages performs much better than that of high-resource. It raises the concern of whether our model learns the evaluation criteria instead of memorizing data, as suggested in Sun et al. (2020). Besides, strong NMT models help quality estimation, but can we use QE models to improve NMT systems' learning? We plan to answer these questions in the future and promote the joint improvement of QE and NMT models.

Acknowledgements

This work was supported in part by the National Science Foundation of China (Nos. 61876035 and 61732005) and the National Key R&D Program of

China (No.2019QY1801). The authors would like to thank anonymous reviewers for their comments.

References

John Blatz, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2004. Confidence estimation for machine translation. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 315–321, Geneva, Switzerland. COLING.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, F. Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *ACL*.

J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Francisco Guzmán, Mark Fishel, Nikolaos Aletras, Vishrav Chaudhary, and Lucia Specia. 2020. Unsupervised quality estimation for neural machine translation. *ArXiv*, abs/2005.10608.

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.

Chi Hu, Bei Li, Yinqiao Li, Ye Lin, Yanyang Li, Chenglong Wang, Tong Xiao, and Jingbo Zhu. 2020. The NiuTrans system for WNGT 2020 efficiency task. In *Proceedings of the Fourth Workshop on Neural Generation and Translation*, pages 204–210, Online. Association for Computational Linguistics.

Fabio Kepler, Jonay Trénous, Marcos Treviso, Miguel Vera, António Góis, M. Amin Farajian, António V. Lopes, and André F. T. Martins. 2019a. Unbabel's participation in the WMT19 translation quality estimation shared task. In *WMT*.

Fabio Kepler, Jonay Trénous, Marcos Treviso, Miguel Vera, and André F. T. Martins. 2019b. OpenKiwi: An open source framework for quality estimation. In *ACL*.

Hyun Kim, Jong-Hyeok Lee, and Seung-Hoon Na. 2017. Predictor-estimator using multilevel task learning with stack propagation for neural quality estimation. In *Proceedings of the Second Conference on Machine Translation*, pages 562–568, Copenhagen, Denmark. 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 85–89, Florence, Italy. Association for Computational Linguistics.

Bei Li, Yinqiao Li, Chen Xu, Y. Lin, Jiqiang Liu, H. Liu, Ziyang Wang, Y. Zhang, N. Xu, Zeyang Wang, Kai Feng, Hexuan Chen, Tengbo Liu, Yanyang Li, Qiang Wang, Tong Xiao, and Jingbo Zhu. 2019. The niutrans machine translation systems for wmt19. In *WMT*.

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

Lucia Specia, Frédéric Blain, Marina Fomicheva, Erick Fonseca, Vishrav Chaudhary, Francisco Guzmán, and André FT Martins. 2020. Findings of the wmt 2020 shared task on quality estimation. In *Proceedings of the Fifth Conference on Machine Translation: Shared Task Papers*.

Shuo Sun, Francisco Guzmán, and Lucia Specia. 2020. Are we estimating or guesstimating translation quality? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6262–6267, Online. Association for Computational Linguistics.

Marina Sánchez Torrón and Philipp Koehn. 2016. Machine translation quality and post-editor productivity. In *In Proceedings of AMTA*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, L. Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *ArXiv*, abs/1706.03762.

Qiang Wang, Bei Li, Tong Xiao, Jingbo Zhu, Changliang Li, Derek F. Wong, and Lidia S. Chao. 2019a. Learning deep transformer models for machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1810–1822, Florence, Italy. Association for Computational Linguistics.

Ziyang Wang, Hui Liu, Hexuan Chen, Kai Feng, Zeyang Wang, Bei Li, Chen Xu, Tong Xiao, and Jingbo Zhu. 2019b. Niutrans submission for ccmt19 quality estimation task. In *CCMT*.

Tong Xiao, Jingbo Zhu, Hao Zhang, and Qiang Li. 2012. Niutrans: An open source toolkit for phrase-based and syntax-based machine translation. In *ACL*.

Elizaveta Yankovskaya, Andre Tättar, and Mark Fishel. 2019. Quality estimation and translation metrics via pre-trained word and sentence embeddings. In *ACL*.