SAU'S Submission for CCMT 2021 Quality Estimation Task

Yanan Li, Na Ye(✉), Dongfeng Cai

Human-Computer Intelligence Research Center,
Shenyang Aerospace University, Shenyang 110136, China
lynan24@qq.com, yena_1@126.com, caidf@vip.163.com

Abstract. This paper describes our submissions to CCMT 2021 quality estimation sentence-level task for both Chinese-to-English (ZH-EN) and English-to-Chinese (EN-ZH). In this task, We follow TransQuest framework which is based on cross-lingual transformers (XLM-R). In order to make the model pay more attention to key words, we use the attention mechanism and gate module to fuse the last hidden state and pooler output of XLM-R model to generate more accurate prediction. In addition, we use the Predictor-Estimator architecture model to integrate with our model to improve the results. Experiments show that this is a simple and effective ensemble method.

Keywords: Quality estimation, XLM-R, ensemble.

1 Introduction

In recent years, with the development of neural network, the quality of machine translation has been greatly improved. However, it is still a problem whether the translated text needs further post-editing, which needs to be solved by translation quality estimation. Quality Estimation (QE) aims to evaluate the quality of machine translation without reference translation, which saves a lot of manpower and time and is more in line with the actual requirements.

This paper introduces in detail our submission of sentence-level quality estimation task. The sentence-level task aims to predict the Human-targeted Translation Edit Rate (HTER) [1] of the machine translation output, which reflects the editing distance from the translation to the correct reference translation. QE system needs to predict the HTER value, that is, the editing error rate of the translation, which is a regression problem.

Traditional quality estimation methods use time-consuming and expensive artificial features to represent source sentences and machine translations. QuEst++ [2] is a method based on machine learning. Later, researchers began to apply neural networks to generated neural features automatically to quality estimation tasks. However, the scarce quality estimation data can not give full play to the role of neural network. In order to solve this problem, researchers try to transfer bilingual knowledge extracted from parallel corpora to quality estimation tasks. This kind of

work usually adopts the Predictor-Estimator model proposed by Kim et al. [3]. Fan et al. [4] introduced a bidirectional Transformer for predictor to extract features, and used 4-dimensional mis-matching features. Besides, Wang et al. [5] used Transformer-DLCL in predictor. Recently, the emergence of pre-training model has swept the whole field of natural language processing, and more and more researchers have begun to use pre-training model in quality estimation tasks. Pre-training model has been widely used in predictor and combined with appropriate estimator [6-7]. At the same time, the ensemble method has been proved to be very effective to improve the results [8-9].

TransQuest [10] is shown to achieve state-of-the-art results outperforming current open-source quality estimation frameworks when trained on datasets from WMT , so we use it as baseline and improve it. In order to make a better prediction, we have improved the output of this model to predict the translation quality more effectively. In addition, we use the ensemble method to integrate the above two models, which is simple but effective.

2 Methods

In this section, we describe the methods used by our submitted system. We first introduce the basic model we use, and then introduce our improvement methods based on this model.

2.1 Basic Model

We chose TransQuest as our basic model. TransQuest uses cross-language transformers model XLM-R [11], which is different from the previous predictor-estimator framework, because it does not use parallel corpus. Therefore, this model reduces the burden of complex neural networks and the demand for computing resources. TransQuest won the first place in WMT 2020 DA task, and achieved state-of-the-art results in the current open-source quality estimation frameworks in WMT datasets. The authors implement two different architectures, and we chose the MonoTransQuest architecture. The input of this model is to separate the original text and the translated text by [SEP] token and input them into XLM-R model together. Besides, they used the output of the [CLS] token as the input of a softmax layer. XLM-R is a multi-language pre-training model proposed by Facebook, which uses 2.5TB CommonCrawl to filter data, and masked language model pre-trained on text in 100 languages, which obtains state-of-the-art performance on cross-lingual classification, sequence labeling and question answering.

Another basic model is QE Brain [12] which follows the predictor-estimator architectures. They use a bidirectional Transformer [13] for predictor and bidirectional LSTM [14] for estimator. QE Brain constructed mis-matching features, and only using this feature to make predictions can get good results. The model can be directly understood as that if the quality of the translated text is very high, the word prediction model based on conditional language model can accurately predict the

current word based on the context of the original sentence and the target sentence. On the contrary, if the translation quality is not high, it is difficult for the model to accurately predict the current words based on the context.

2.2 Proposed Method

TransQuest model follows the standard method of XLM-R classification, and uses the tensor corresponding to the first token [CLS] of the last layer for classification.We want to make full use of the output of XLM-R. Therefore, we adopt the method of fully fusing the information of pooler_output and last_hidden_state, using the attention mechanism [15] and gate module [16].

When dealing with pooler_output, we use the same operation as CLS token of last_hidden_state. We use dropout layer, linear layer and tanh nonlinear function to deal with it. The model structure is shown in Fig. 1.

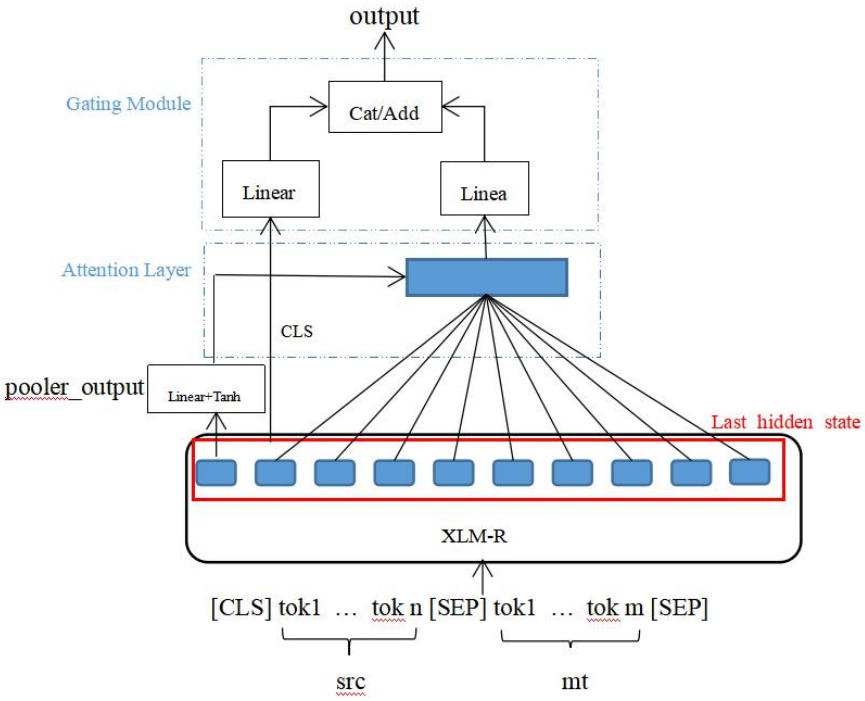

Fig. 1. Fusing pooler_output and last_hidden_state

2.3.1 Attention mechanism

Considering that the method based on attention and weight has proved to be an effective way to selectively use additional information in many tasks, we add an attention layer. According to the contribution of words to tasks in different sentences, words are weighted to further enhance semantic information, and then pay attention to some keyword information. Different from the sequence-to-sequence task, the output

of the translation quality estimation task is not a serialization process, so the attention weight of sentence vectors to word vectors in the current batch can be obtained only by one calculation, which is easy to implement.

We can simply regard the output of last_hidden_state as the word vector of the sequence, and the output of pooler_output as the sentence vector of the whole sequence. The formula for calculating attention is as follows：

$$\alpha_{i,j} = \frac{\exp\left(h_i \cdot e_{i,j}\right)}{\sum_{j=i}^{m} \exp\left(h_i \cdot e_{i,j}\right)} \tag{1}$$

$$v_i = \sum_{j=1}^{m} \alpha_{i,j} e_{i,j} \tag{2}$$

Where h_i denote the output of pooler_output, $e_{i,j}$ denote word embeddings in the output of last_hidden_state, and v_i is the output of the final attention layer.

2.3.2 Gate module

Considering that the contribution of CLS token and attention vector in quality estimation task changes in different contexts, we hope to weight this information in the changing context through the gate module. We use a gate to control the information flow by

$$g = \sigma\left(W_1 \cdot h_{\text{cls}} + W_2 \cdot v_i + b_g\right) \tag{3}$$

$$u_i = \left[g \circ h_{\text{cls}}\right] + \left[(1-g) \circ v_i\right] \tag{4}$$

where W_1 and W_2 are trainable matrices and b_g the corresponding bias term. Then g is used to balance the information of CLS token and attention output, where h_{cls} denotes the CLS token, u_i denotes the output of the gate module and ° represents the element-wise multiplication operation. In the fusion mode of gate module, we try the addition and concatenation methods.

2.3 Ensemble

In order to further boost performance, we use the ensemble method. It is worth mentioning that we only use two models. One is the TransQuest model after our improvement, and the other is the QE Brain model. We chose QE Brain model because it works well on WMT data.

For the ensemble methods, due to time constraints, we only used the weighted average ensemble. According to the performance of the two models under CCMT data, we designed different weight ratios for ensemble, and finally chose the weight of the best result on the validation set for the ensemble experiment.

3 Experiment

3.1 Dataset

All the data we used came from CCMT2021, and no other extra data was used. The QE datasets have two language directions of both English-Chinese (EN-ZH) and Chinese-English (ZH-EN). The statistics of QE datasets are shown in Table 1. We don't use extra parallel data when using TransQuest framework, but we use QE Brain framework when we use ensemble methods, and the parallel corpus for training predictor comes from the machine translation task of CCMT2021. The statistics of parallel corpus are shown in Table 2.

Table 1. The statistics of QE datasets.

Direction	Aspect	Train	Dev	Test
EN-ZH	sent	14,789	1,381	2,528
ZH-EN	sent	10,070	1,143	2,412

Table 2. The statistics of parallel corpus.

Dataset	Data	Sentences
Datum2017	train	999,985
Casict2015	train	2,036,833
Casia2015	train	1,050,000
Neu2017	train	2,000,000
CCMT2019-en2zh	dev	1,000

3.2 Settings

For sentence-level task, Pearson correlation coefficient is the main evaluation measure, In addition, we have set other measure: RMSE, MAE and Spearman Correlation.

We use the same settings for the two language directions pairs evaluated in this paper. We follow the default configuration of TransQuest framework, but adjust the learning rate to 2e-6, and other settings remain unchanged. We use Adam optimizer with a batch-size of eight. In the training process, the parameters of xlm-roberta-large model and the parameters of subsequent layers are updated. In the experiment, we used an NVIDIA Tesla T4 GPU.

We use different dropout rates for different language pairs. The final dropout rate is 0.4 in EN-ZH experiment and 0.3 in ZH-EN experiment.

3.3 Results of the single model

The results of CCMT2021 dev2019 are shown in Table 3 and Table 4. In order to get comparative experiments, the effectiveness of attention layer and gate module in translation quality estimation is demonstrated.

It can be seen from the results that after adding the attention output of pool and last, the use of add mode under the gate module has been improved by 3.67% in EN-ZH experiment and 2.76% in ZH-EN experiment, and the effect of using cat mode is not obvious. In addition, if the gate module is not used, the effect of direct addition will decrease, which may be because the attention output affects the semantic vector representation of the whole [CLS]. Therefore, there must be a gate module to control the attention output. If attention layer is not used, the promotion is not obvious, which fully proves that attention layer and gate model are indispensable.

Table 3. Results of the CCMT 2021 EN-ZH dev2019.

Model	Pearson	RMSE	MAE	Spearman
Baseline	0.5063	0.1552	0.1083	0.4439
+attention+cat	0.5069	0.1633	0.1104	0.4146
+attention+add	0.4835	0.1667	0.1152	0.3984
no_attention+gate_cat	0.5046	0.1628	0.1139	0.4328
no_attention+gate_add	0.5099	0.1620	0.1143	0.4409
+attention+gate_cat	0.5097	0.1633	0.1145	0.4355
+attention+gate_add	**0.5249**	0.1526	0.1081	0.4499

Table 4. Results of the CCMT 2021 ZH-EN dev2019.

Model	Pearson	RMSE	MAE	Spearman
Baseline	0.5204	0.1526	0.1070	0.4506
+attention+cat	0.5131	0.1513	0.1141	0.4527
+attention+add	0.4989	0.1654	0.1149	0.4376
no_attention+gate_cat	0.5211	0.1611	0.1130	0.4628
no_attention+gate_add	0.5215	0.1609	0.1124	0.4633
+attention+gate_cat	0.5228	0.1615	0.1104	0.4614
+attention+gate_add	**0.5348**	0.1501	0.1012	0.4927

3.4 Results of the ensemble methods

Through the experiment of different proportions of fusion, we obtained best results when we used the weights 0.7:0.3 in EN-ZH task and the weights 0.6:0.4 in ZH-EN task. The results of ensemble model are shown in Table 5.

According to the ensemble results, our improved TransQuest model has been improved by 6.4% in EN-ZH experiment and 7.8% in ZH-EN experiment, which fully reflects the effectiveness of the ensemble models. It can be concluded that integration is indeed a good way to improve the prediction accuracy in translation quality estimation.

Table 5. The results of ensemble model.

Model	EN-ZH	ZH-EN
Our improved best TransQuest	0.5249	0.5348
QE Brain	0.3995	0.4639
ensemble	**0.5587**	**0.5765**

A good model and a relatively bad model will produce better results. This is an interesting phenomenon, and perhaps it is also a question worth considering. QE Brain has been performing well under our previous WMT QE datasets, but its effect is not good under CCMT QE datasets, probably because the quality of CCMT translation is relatively good and involves a wide range of fields. The strength of the pre-training model is unexplainable to some extent, and the mis-matching feature is a feature that we think is very reasonable, so we will design some features more deeply in the future.

Although the prediction effect of QE Brain was not good on CCMT, after our analysis, we found that it could complement the prediction of TransQuest. We selected the first 400 examples of CCMT validation set to draw a line chart. As shown in the Fig. 2. , we can see, the predictions of TransQuest are mostly distributed above the golden HTER value, while the predictions of QE Brain are mostly distributed below the golden HTER value. Therefore, the fusion of the two models can improve the prediction results of the ensemble methods.

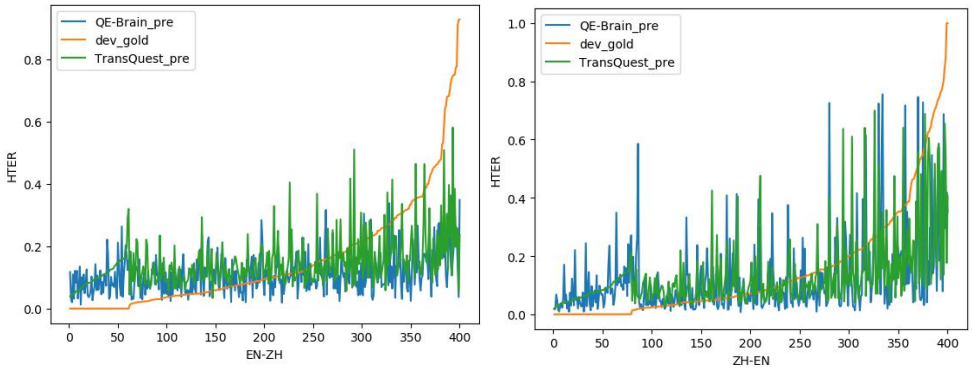

Fig. 2. The yellow line represents dev golden HTER, the blue line represents the predicted value of QE Brain and the green line represents the predicted value of our improved TransQuest. The left picture shows EN-ZH experiment, and the right picture shows ZH-EN experiment.

4 Conclusion

We describe our submissions to CCMT2021 QE sentence-level task. Our systems are based on TransQuest architecture and use QE Brain to make ensemble experiments. In order to make full use of the output of XLM-R and pay more attention to some key words, we use attention mechanism and gate module to fuse the output of XLM-R about last_hidden_state and pooler_output. Experiments show that this is effective. In addition, we also try to split last_hidden_state and add some external knowledge, but the effect is not good. On the other hand, the ensemble method is very effective in the task of translation quality estimation.

In the future work, although the pre-training model represents the source information and the target information in the same feature space, the source information is completely correct, while the target information contains wrong information. How to link the two more effectively is our next work. We want to introduce some external features to further enhance the performance. And we will also try some other ensemble methods in later experiments.

Acknowledgements

This work is supported by the Humanity and Social Science Foundation for the Youth researchers of Ministry of Education of China (19YJC740107), the National Natural Science Foundation of China (U1908216) and the Key Research and Development Plan of Liaoning Province (2019JH2/10100020).

References

1. Snover, M., Dorr, B., Schwartz, R., Micciulla, L., Makhoul, J.: A Study of Translation Edit Rate with Targeted Human Annotation. In: Proceedings of Association for Machine Translation in the Americas, pp. 223-231. AMTA, Cambridge (2006).
2. Specia, L., Paetzold, G., Scarton, C.: Multi-level Translation Quality Prediction with QuEst++. In: Proceedings of ACL-IJCNLP 2015 System Demonstrations, pp.115-120. ACL-IJCNLP, Beijing (2015).
3. Kim, H., Jung, H.Y., Kwon, H., Lee, J.H., Na, S.H.: Predictor-Estimator: Neural Quality Estimation based on Target Word Prediction for Machine Translation. In: ACM Transactions on Asian and Low-Resource Language Information Processing, 17(1), pp. 1-22 (2017).
4. Fan, K., Wang, J., Li, B., Zhou, F., Chen, B., Si, L.: “Bilingual Expert” Can Find Translation Errors. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 6367-6374 (2019).
5. Wang, Z., Liu, H., Chen, H., Feng, K., Wang, Z., Li, B., Xu, C., Xiao, T., Zhu, J.: NiuTrans Submission for CCMT19 Quality Estimation Task. In: China Conference on Machine Translation, pp. 82-92. Springer, Singapore (2019).
6. Wang, Z., Wu, H., Ma, Q., Wen, X., Wang, R., Wang, X., ... & Yao, Z. (2020, August). Tencent Submissions for the CCMT 2020 Quality Estimation Task. In China Conference on Machine Translation (pp. 123-131). Springer, Singapore.

7. Kim, H., Lim, J. H., Kim, H. K., & Na, S. H. (2019, August). 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) (pp. 85-89).
8. Hu, C., Liu, H., Feng, K., Xu, C., Xu, N., Zhou, Z., ... & Zhu, J. (2020, November). The niutrans system for the wmt20 quality estimation shared task. In Proceedings of the Fifth Conference on Machine Translation (pp. 1018-1023).
9. Cui, Q., Geng, X., Huang, S., & Chen, J. (2020, August). NJUNLP ' s Submission for CCMT20 Quality Estimation Task. In China Conference on Machine Translation (pp. 114-122). Springer, Singapore.
10. Ranasinghe T, Orasan C, Mitkov R. TransQuest: Translation quality estimation with cross-lingual transformers[J]. arXiv preprint arXiv:2011.01536, 2020.
11. Alexis Conneau, Kartikay Khandelwal, Naman Goyal et al. Unsupervised Cross-lingual Representation Learning at Scale[C]//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020: 8440-8451.
12. Fan K , Wang J , Li B , et al. "Bilingual Expert" Can Find Translation Errors[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2019, 33:6367-6374.
13. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. arXiv preprint, arXiv:1810.04805, 2018.
14. Graves, A., Schmidhuber, J.: Framewise phoneme classification with bidirectional lstm and other neural network architectures. Neural Networks (2005).
15. Luong, M. T., Pham, H., & Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. arXiv preprint arXiv:1508.04025.
16. Nie Y , Tian Y , Wan X , et al. Named Entity Recognition for Social Media Texts with Semantic Augmentation[J]. 2020.