Quality-Aware Neural Machine Translation with Self-evaluation

Jiajia Cui[1], Lingling Mu[1], Qiuhui Liu[2], and Hongfei Xu[1(✉)]

[1] Zhengzhou University, Zhengzhou 450001, Henan, China
jjcui@gs.zzu.edu.cn, iellmu@zzu.edu.cn, hfxunlp@foxmail.com
[2] China Mobile Online Services, Zhengzhou 450001, Henan, China

Abstract. The performance of neural machine translation relies on a large amount of data, but crawled sentence pairs are of different quality. The low-quality sentence pairs may provide helpful translation knowledge but also teach the model to generate low-quality translations. Making the model aware of the quality of training instances may help the model distinguish between good and bad translations while leveraging the translation knowledge. In this paper, we evaluate the quality of training instances with the average per-token loss (negative log-likelihood) from translation models, convert the quality scores into embeddings through vector interpolation and feed the quality embedding into the translation model during its training. We ask the model to decode with the best quality score to generate good translations during inference. Experiments on the IWSLT 14 German to English, WMT 14 English to German and WMT 22 English to Japanese translation tasks show that our method can effectively lead to consistent and significant improvements across multiple metrics.

Keywords: Neural Machine Translation · Quality-aware modeling · Machine translation evaluation

1 Introduction

The Transformer translation model [29] can produce high quality translations with large amounts of parallel data. Most large-scale parallel corpora are automatically extracted from crawled parallel websites and cannot ensure the quality of translation pairs. Training Neural Machine Translation (NMT) models on the unfiltered ParaCrawl leads to degraded translation quality [10,23].

Peter et al. [17] improve the NMT performance by filtering half of the training set with neural Quality Estimation (QE) metrics. However, this approach relies on the availability of high-performance QE models of the target language pair and fully discards the translation knowledge within the low-quality parallel data. Concurrently, Tomani et al. [28] divide the training set into a number of bins based on the MetricX-QE scores and prompt the model with quality tags of the corresponding bins. However, 1) the quality embedding for each bin is only

M. Sun et al. (Eds.): CCL 2025, LNAI 16052, pp. 471–483, 2026.
https://doi.org/10.1007/978-981-95-2725-0_27

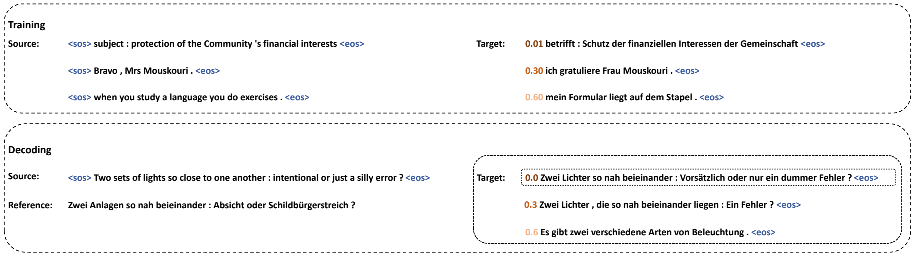

Fig. 1. Quality-aware training and decoding. The left and right sides are for encoder and decoder respectively. The quality score is the first input of the decoder to trigger the auto-regressive decoding. Quality scores are converted into quality embeddings through linear interpolation. The best quality score is used for evaluation. Examples decoded with worse quality scores are only to show that the model follows the quality scores during translation.

trained on part of the training set, 2) the quality differences inside a bin are fully neglected, 3) close quality scores divided into adjacent bins may have different quality embeddings, and 4) the method also relies on QE tools which may not be available for some language pairs.

To avoid the use of QE tools, we train NMT models on the parallel data for both directions, and use the bi-directional average per-token loss from the translation models as the quality measurements of training instances. We derive the quality embedding through vector interpolation with the quality score, and replace the embedding of the special start-of-sentence (<sos>) token with the quality embedding to make the model aware of the quality of training instances. All training data and corresponding translation knowledge are thus kept during the training. During inference, we prompt the model with the best quality score, and ask the model to generate good translations. Our main contributions are as follows:

– We train NMT models on the parallel data, and evaluate the data quality based on bi-directional scores to avoid the use of QE tools.
– We turn quality scores into quality embeddings by vector interpolation, and integrate into the translation model to help distinguish the quality of training instances while leveraging the translation knowledge.
– Our method brings consistent and significant improvements from low-resource to high-resource tasks across several evaluation metrics without negative impacts on the inference speed.

2 Our Method

The training and decoding of quality-aware translation are shown in Fig. 1. In general, we first train source-to-target and target-to-source vanilla translation models (baseline without quality scores). Then we use the baseline models to calculate the average per-token prediction loss of each training instance as quality scores. The quality scores are used for the training of the quality-aware model by replacing the <sos> embedding with the quality embedding generated based on the quality score.

2.1 Bi-directional Self-evaluation

Fomicheva et al. [8] show that NMT models are also strong quality estimators and can achieve good correlation with human quality judgments.

We use the baseline NMT models to compute the average per-token loss of each training instance. The prediction loss of the target token y_n is the negative log-likelihood computed by the NMT model $\overrightarrow{\mathrm{M}}$ based on the source sentence x and all preceding tokens $y_{<n}$ in the reference translation.

$$loss(x, \overrightarrow{\mathrm{M}}, y_n) = -logP(y_n|x, y_{<n}, \overrightarrow{\mathrm{M}}) \tag{1}$$

But the loss of the forward direction may not fully reflect translation quality, especially when the source sentence is inadequately translated. While the target-to-source NMT model $\overleftarrow{\mathrm{M}}$ is likely to produce high loss scores for under-translated source tokens given the target sentence as encoder input. The bi-directional loss ($loss_{bi}$) considers both directions and is computed with Eq. 2.

$$loss_{bi} = \frac{\sum_{t=1}^{|y|} loss(x, \overrightarrow{\mathrm{M}}, y_t) + \sum_{t=1}^{|x|} loss(y, \overleftarrow{\mathrm{M}}, x_t)}{|y| + |x|} \tag{2}$$

where $|x|$ means the number of reference tokens in the source sentence x.

We normalize the bi-directional average per-token loss into the range of $[0, 1]$ as quality score q with Eq. 3.

$$q = \frac{loss_{\mathrm{bi}} - \min(loss_{\mathrm{bi}})}{\max(loss_{\mathrm{bi}}) - \min(loss_{\mathrm{bi}})} \tag{3}$$

There is no existing theoretical basis between the amount of data and the estimation reliability to our knowledge. But even the low-resource machine translation task (e.g., IWSLT 14 De→En) has a much larger training set ($174k$ sentence pairs) than the amount of available training data for the quality estimation task and many other NLP tasks. Our method empirically works well on the low-resource IWSLT 14 De→En task as shown in Table 1.

2.2 Quality-Aware NMT

The NMT model represents discrete tokens with embeddings, but the quality score for each instance is a continuous scalar. To make the NMT model explicitly aware of the training instance quality, we have to turn the quality score q into an embedding and feed the quality embedding into the model.

As the quality score is normalized into a fixed range ($[0, 1]$), we generate quality embeddings by vector interpolation. Specifically, we employ two vectors v_e and v_s as corresponding embeddings for the quality scores of 0 and 1 respectively. The embedding e_q of the other quality score q are computed by the weighted combination of v_s and v_e, where the weights are the distances from q to 0 and 1 respectively, as shown in Eq. 4.

$$e_q = q * v_s + (1 - q) * v_e \tag{4}$$

As the auto-regressive decoder computes in a left-to-right manner, we replace the special <sos> embedding with the quality embedding to provide the model with quality information. We do not append the quality scores as additional word embedding dimensions to avoid changing the embedding size or reducing the number of dimensions for word representation. We also do not insert one more token before the <sos> token which may increase the complexity of decoding when applying the method to the decoder.

During inference, we feed the quality score representing the best quality into the model to generate good translations.

3 Experiments

3.1 Settings

Datasets. IWSLT 14 [4] using TED talks is the cleanest. Common Crawl is the major part of WMT 14 English (En) to German (De) task [3]and noisy (COMET scores of 11.23% and 1.7% of the training set are below 0.5 and 0.3 respectively). ParaCrawl constitutes the largest part of the training set of the WMT 22 English to Japanese (Ja) task [14]. Larger training sets normally tend to be more noisy. We tokenized and truecased English and German with moses, and tokenized Japanese with MeCab [15]. We applied joint Byte-Pair Encoding (BPE) [26] with $16k$ merge operations for the low-resource IWSLT 14 De→En task ($174k$), $32k$ merge operations for the WMT 14 En→De task ($4.5M$), and independent BPE with $32k$ merge operations for the WMT 22 En→Ja task ($33M$).

Baselines. We compared our method with the Transformer baseline, QE-based filter [17] and quality-aware prompting method [28]. For the Filter method [17], we filtered half of the training set with lower COMET-QE scores. For the Prompt method [28], we used COMET-QE scores to divide the training set into 10 bins and replaced the <sos> token of both source and target sentences with bin tags following their paper. Recurrent decoders may lead to improved translation quality compared to the Transformer decoder [6]. We also tested the performance of our approach with the MHPLSTM decoder [30]. Specifically, We replace the self-attention layers of the Transformer decoder with the MHPLSTM.

Hyperparameters. We followed the Base setting (512/2048, 8 heads) of Transformer [29] for WMT 14 En→De and WMT 22 En→Ja tasks. For IWSLT De→En task, we followed the experiment settings of Araabi and Monz [1]. For the training, we trained for 100k steps with a batch size of around 36k target tokens for the WMT 14 En→De task and the En→Ja task, a batch size of around 6k target tokens for the IWSLT 14 De→En task. The results in Table 1 are reproduced under the same setting. All the experiments used the same training script and hyper-parameters.

Evaluation. we used a beam size of 4 for decoding with the averaged model of the last 5 checkpoints saved with an interval of 1500 training steps. We evaluated with BLEU and chrF implemented by the sacreBLEU toolkit [18] and COMET score [20]. Human evaluation would be valuable, but it is often impractical due to inconsistent standards across studies and neural metrics like COMET have led to high correlation with humans. For our method, we prompt the best quality score for high-quality translations during decoding and evaluation. The best quality score depends on the metric, it is 0 for bi-directional translation loss and MetricX, but 1 for COMET.

We found that the performance of Transformer is quite sensitive to small variations on the quality scores. It can vary for around 5 BLEU on the development and test sets when gradually increasing the decoding quality score from 0 to 0.1. So we treated the quality scores (the concatenation of q and $1 - q$ in Eq. 4) like hidden representations in the neuron network and applied dropout to them during training to prevent the model from over-fitting the quality scores.[1]

Table 1. Main results. Bold scores indicate improvements of our method over corresponding baselines.

Methods	IWSLT 14 De→En			WMT 14 En→De			WMT 22 En→Ja		
	BLEU	chrF	COMET	BLEU	chrF	COMET	BLEU	chrF	COMET
Transformer	29.57	52.45	77.67	27.64	57.26	82.42	21.55	29.47	83.76
Filter [17]	28.39	51.51	75.66	27.68	57.21	82.75	21.92	30.02	84.31
Prompt [28]	29.45	52.66	77.09	27.92	57.43	82.74	22.06	30.05	84.34
Ours	**29.99**	**52.85**	**78.01**	**28.78**	**57.91**	**83.51**	**22.56**	**30.66**	**84.83**
MHPLSTM [30]	29.90	53.20	78.19	28.36	57.57	83.01	22.12	30.10	84.53
Ours	**30.34**	**53.42**	**79.19**	**29.01**	**57.94**	**83.55**	**22.75**	**30.69**	**84.88**

[1] The model leads to BLEU scores of 23.72, 28.43 and 28.36 on the WMT 14 En→De task when decoding with 0, 0.05 and 0.1 as the quality scores respectively without dropout, while obtaining a highest BLEU score of 28.78 with a quality score of 0 with dropout probability of 0.1.

3.2 Main Results

We used bi-directional NMT losses as quality scores (Sect. 3.5) and added quality scores to the decoder (Sect. 3.6) based on our ablation study.

Results in Table 1 show that: 1) data filtering or prompting based on quality scores can lead to consistent and significant improvements across all metrics only on the high-resource WMT 22 En→Ja task of the largest training set, but their performances are close to the baseline on the WMT 14 En→De task and worse on the low-resource IWSLT 14 De→En task with some metrics, while our method is also effective for low-resource and middle-resource tasks, obtaining consistent and significant improvements with all metrics, and 2) our method can also improve the performance of the stronger recurrent decoder baseline.

3.3 Results on Large Language Model

We tested the performance of our method with the Large Language Model (LLM) on the WMT 22 En→Ja task, which is the largest dataset. We fine-tuned `Qwen3-8B` [27] with LoRA with rank $r = 128$ as our baseline. LoRA was applied to the attention query/key/value projection layer. Experiments were conducted with identical configurations, including the number of training steps and batch size. We incorporated quality scores into the LLM fine-tuning for our method. Specifically, we replaced the embedding of the start tag ($< |im\_start| >$) that triggers the generation with the quality embedding. We used a beam size of 4 for decoding with the best performing model on the development set.

Results in Table 2 show that our method outperforms the baseline across all metrics (+1.25/+1.48/+1.24 on BLEU/chrf/COMET). The BLEU and chrF scores are lower than the results in Table 1, but the COMET score is higher, which suggests that LLM generates more semantically adequate translations compared to the Transformer.

Table 2. LLM Result on WMT 22 test set.

Method	BLEU	chrF	COMET
Baseline	20.49	28.28	85.85
Ours	**21.74**	**29.76**	**87.09**

3.4 Effectiveness with Scaling

We tested the performance of shallower and deeper models on the WMT 14 En→De task. Results in Table 3 show that our method can also obtain consistent improvements with varying depths, and shallow models with our method can perform comparably to deeper baselines.

Table 3. Results with varying depths.

Depth		Method	BLEU	chrF	COMET
Encoder	Decoder				
3	3	Baseline	26.52	56.15	80.58
		Ours	27.36	56.92	81.98
6	6	Baseline	27.64	57.26	82.42
		Ours	28.78	57.91	83.51
12	6	Baseline	28.63	57.81	83.36
		Ours	29.48	58.55	84.37
18	6	Baseline	29.06	58.16	83.50
		Ours	29.84	58.67	84.39

3.5 Effects of Quality Estimation Methods

To verify that the translation losses (of the forward, reverse and bi-direction) are valid quality indicators, we measure the Pearson correlation coefficients between the translation losses and the COMET [21]/MetricX [11] scores. We compare with automatic metrics, because our method only uses quality scores of the training set. The vanilla translation models are trained without quality annotations and cannot overfit to quality scores. The quality estimation performance on the other datasets may not be robust due to the machine translation training data limitation, but our method only requires sufficient quality estimation performance on the training set.

Results in Table 4 show that: 1) translation losses of individual directions lead to similar Pearson coefficients to the COMET scores and the bi-directional translation loss leads to a slightly higher coefficient, and 2) the Pearson coefficient of the bi-directional translation loss (0.80) is close to that of MetricX (0.83), supporting that the translation losses can be used as effective quality measures. The correlation with MetricX scores is lower than with COMET for MetricX scores are somehow problematic (17.9% are 0, 8% are 25).

Table 4. Pearson correlation coefficients between translation losses, MetricX and COMET/MetricX scores.

Target	Forward	Reverse	Bi-direction	MetricX
COMET	0.77	0.77	0.80	0.83
MetricX	0.70	0.71	0.73	–

We conducted experiments on the WMT 14 En→De task to test the effects of different quality scores. Results in Table 5 show that: 1) all quality evaluation methods can lead to consistent improvements except for the MetricX, the poor performance with MetricX may be because more than 1/4 of the training set shares the same quality embeddings (17.9% and 8% of MetricX scores are 0 and 25 respectively) and lacks discrimination in quality, 2) using the bi-directional loss is better than using a single direction loss, and 3) using COMET-QE scores leads to slightly higher COMET score but slightly worse BLEU and chrF scores than the bi-directional translation loss. This may be due to potential bias between COMET-QE and COMET. Using COMET-QE scores can obtain comparable performance to bi-directional loss and avoid the additional training cost of the bi-directional translation models for quality estimation, but self-evaluation with the translation loss can avoid the reliance on the existence of QE tools.

Table 5. Ablation study of quality estimation methods on the development set (newstest 2013) and test set (newstest 2014).

Methods	Dev Set			Test Set		
	BLEU	chrF	COMET	BLEU	chrF	COMET
Baseline	26.13	54.32	82.24	27.64	57.26	82.42
MetricX	25.69	53.93	81.79	27.97	57.17	82.26
COMET-QE	26.31	54.69	**83.64**	28.63	57.84	**83.57**
Forward	26.22	54.44	83.15	28.45	57.64	83.37
Reverse	26.33	54.64	83.31	28.50	57.82	83.35
Bi-direction	**26.43**	**54.76**	83.48	**28.78**	**57.91**	83.51

3.6 Quality-Aware Encoding and Decoding

We tested the effects of providing quality scores to the encoder or the decoder or both on the WMT 14 En→De task. Results in Table 6 show that: 1) providing quality scores to either encoder or decoder can lead to significant improvements than the baseline, showing the importance of integrating quality scores, and 2) using quality scores in the decoder leads to better performance than in both, probably because that it might be better to only let the decoder learn to generate translations of quality than additionally asking the encoder to perform source language understanding for decoding of varying quality.

Table 6. Results for quality-aware encoding/decoding on the development set (newstest 2013) and test set (newstest 2014).

Methods	Dev Set			Test Set		
	BLEU	chrF	COMET	BLEU	chrF	COMET
Baseline	26.13	54.32	82.24	27.64	57.26	82.42
Encoder	26.18	54.60	83.04	28.41	57.54	83.19
Decoder	**26.43**	**54.76**	**83.48**	**28.78**	**57.91**	**83.51**
Both	26.21	54.60	83.08	28.54	57.73	83.13

3.7 Effects of Quality Scores on Inference

To test whether the model is aware of the translation quality after training, we tested the BLEU score of model translations with varying quality scores on the WMT 14 En→De task.

Results in Fig. 2 show that the model learns to generate translations of the given quality score.

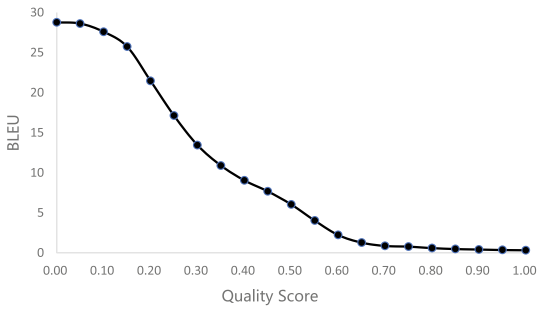

Fig. 2. Translation quality with increasing loss scores.

4 Related Work

The performance of machine translation models heavily depends on the quality of the training data. Traditional approaches often improve data quality by filtering noisy data [5,10,23]. Recently, Peter et al. [17] use QE metrics to filter low-quality sentence pairs in the training data. And they found QE methods focus on selecting the best translation examples and identify more fine grained problems in the training data.

On the other hand, some studies have attempted to incorporate quality signals into the model training or decoding process. Fernandes et al. [7] using QE for reranking or Minimum Bayes Risk decoding lead to significant improvements over beam search decoding. Concurrently, Tomani et al. [28] eliminate the need for an external QE model during decoding by embedding quality signals directly into the model, they divide the training set into a number of bins based on the quality estimation scores and prompt the model with the quality embedding of the corresponding bin.

The evaluation methods for machine translation can be divided into two categories: reference-based and reference-free. Reference-based evaluation (BLEU [16], BLEURT [24] and COMET [20]) requires comparing the model output with reference translations. Compared with lexical overlap based methods sush as BLEU, neural metrics like BLEURT and COMET can capture semantic relationships. Reference-free evaluation (OpenKiwi [12], TransQuest [19], COMET-QE [21] and MetricX-QE [11]), also named quality estimation (QE), mainly indicates that directly predicting a quality score for the translation based on the input and model output.

Adding tags is an effective method to integrate additional signals to neural models, such as formality level [31], politeness [25], domain [13], and so on. Johnson et al. [9] indicate the target language for multilingual NMT. Scarton and Specia [22] indicate the audience for text simplification. Bandel et al. [2] control paraphrasing via semantic similarity, syntactic and lexical variation.

5 Conclusion

In this paper, we facilitate quality-aware NMT by self-estimated quality with bi-directional translation losses and representing quality scores via vector interpolation. Experiments show that our method can lead to consistent and significant improvements on low/middle/high-resource tasks. Our analysis shows that the model is aware of the quality after training and can generate translations of the given quality score.

Acknowledgments. We appreciate our reviewers for their insightful comments and suggestions. This work is partially supported by the National Natural Science Foundation of China (Grant No. 62306284), China Postdoctoral Science Foundation (Grant No. 2023M743189), and the Natural Science Foundation of Henan Province (Grant No. 232300421386).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Araabi, A., Monz, C.: Optimizing transformer for low-resource neural machine translation. In: Scott, D., Bel, N., Zong, C. (eds.) Proceedings of the 28th International Conference on Computational Linguistics, pp. 3429–3435. International Committee on Computational Linguistics, Barcelona (2020). https://doi.org/10.18653/v1/2020.coling-main.304, https://aclanthology.org/2020.coling-main.304
2. Bandel, E., Aharonov, R., Shmueli-Scheuer, M., Shnayderman, I., Slonim, N., Ein-Dor, L.: Quality controlled paraphrase generation. In: Muresan, S., Nakov, P., Villavicencio, A. (eds.) Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 596–609. Association for Computational Linguistics, Dublin (2022). https://doi.org/10.18653/v1/2022.acl-long.45, https://aclanthology.org/2022.acl-long.45/
3. Bojar, O., et al.: Findings of the 2014 workshop on statistical machine translation. In: Proceedings of the Ninth Workshop on Statistical Machine Translation, pp. 12–58. Association for Computational Linguistics, Baltimore (2014). http://www.aclweb.org/anthology/W/W14/W14-3302

4. Cettolo, M., Niehues, J., Stüker, S., Bentivogli, L., Federico, M.: Report on the 11th IWSLT evaluation campaign. In: Federico, M., Stüker, S., Yvon, F. (eds.) Proceedings of the 11th International Workshop on Spoken Language Translation: Evaluation Campaign, Lake Tahoe, California, pp. 2–17 (2014). https://aclanthology.org/2014.iwslt-evaluation.1
5. Chaudhary, V., Tang, Y., Guzmán, F., Schwenk, H., Koehn, P.: Low-resource corpus filtering using multilingual sentence embeddings. In: Bojar, O., et al. (eds.) Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2), pp. 261–266. Association for Computational Linguistics, Florence (2019). https://doi.org/10.18653/v1/W19-5435, https://aclanthology.org/W19-5435/
6. Chen, M.X., et al.: The best of both worlds: Combining recent advances in neural machine translation. In: Gurevych, I., Miyao, Y. (eds.) Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 76–86. Association for Computational Linguistics, Melbourne (2018). https://doi.org/10.18653/v1/P18-1008, https://aclanthology.org/P18-1008
7. Fernandes, P., et al.: Quality-aware decoding for neural machine translation. In: Carpuat, M., de Marneffe, M.C., Meza Ruiz, I.V. (eds.) Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 1396–1412. Association for Computational Linguistics, Seattle (2022). https://doi.org/10.18653/v1/2022.naacl-main.100, https://aclanthology.org/2022.naacl-main.100
8. Fomicheva, M., et al.: Unsupervised quality estimation for neural machine translation. Trans. Assoc. Comput. Linguist. **8**, 539–555 (2020) https://doi.org/10.1162/tacl_a_00330, https://aclanthology.org/2020.tacl-1.35
9. Johnson, M., et al.: Google's multilingual neural machine translation system: enabling zero-shot translation. Trans. Assoc. Comput. Linguist. **5**, 339–351 (2017). https://doi.org/10.1162/tacl_a_00065, https://aclanthology.org/Q17-1024
10. Junczys-Dowmunt, M.: Microsoft's submission to the WMT2018 news translation task: How I learned to stop worrying and love the data. In: Bojar, O., et al. (eds.) Proceedings of the Third Conference on Machine Translation: Shared Task Papers. pp. 425–430. Association for Computational Linguistics, Belgium (2018). https://doi.org/10.18653/v1/W18-6415, https://aclanthology.org/W18-6415
11. Juraska, J., Finkelstein, M., Deutsch, D., Siddhant, A., Mirzazadeh, M., Freitag, M.: MetricX-23: the Google submission to the WMT 2023 metrics shared task. In: Koehn, P., Haddow, B., Kocmi, T., Monz, C. (eds.) Proceedings of the Eighth Conference on Machine Translation, pp. 756–767. Association for Computational Linguistics, Singapore (2023). https://doi.org/10.18653/v1/2023.wmt-1.63, https://aclanthology.org/2023.wmt-1.63
12. Kepler, F., Trénous, J., Treviso, M., Vera, M., Martins, A.F.T.: OpenKiwi: an open source framework for quality estimation. In: Costa-jussà, M.R., Alfonseca, E. (eds.) Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 117–122. Association for Computational Linguistics, Florence (2019). https://doi.org/10.18653/v1/P19-3020, https://aclanthology.org/P19-3020/
13. Kobus, C., Crego, J., Senellart, J.: Domain control for neural machine translation. In: Mitkov, R., Angelova, G. (eds.) Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017, pp. 372–378. INCOMA Ltd., Varna (2017). https://doi.org/10.26615/978-954-452-049-6_049, https://aclanthology.org/R17-1049/

14. Kocmi, T., et al.: Findings of the 2022 conference on machine translation (WMT22). In: Koehn, P., et al. (eds.) Proceedings of the Seventh Conference on Machine Translation (WMT), pp. 1–45. Association for Computational Linguistics, Abu Dhabi (2022). https://aclanthology.org/2022.wmt-1.1
15. KUDO, T.: Mecab: yet another part-of-speech and morphological analyzer (2010). http://mecab.sourceforge.net/, https://cir.nii.ac.jp/crid/1571980075722574592
16. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: BLEU: a method for automatic evaluation of machine translation. In: Isabelle, P., Charniak, E., Lin, D. (eds.) Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, pp. 311–318. Association for Computational Linguistics, Philadelphia (2002). https://doi.org/10.3115/1073083.1073135, https://aclanthology.org/P02-1040
17. Peter, J.T., Vilar, D., Deutsch, D., Finkelstein, M., Juraska, J., Freitag, M.: There's no data like better data: using QE metrics for MT data filtering. In: Koehn, P., Haddow, B., Kocmi, T., Monz, C. (eds.) Proceedings of the Eighth Conference on Machine Translation, pp. 561–577. Association for Computational Linguistics, Singapore (2023). https://doi.org/10.18653/v1/2023.wmt-1.50, https://aclanthology.org/2023.wmt-1.50
18. Post, M.: A call for clarity in reporting BLEU scores. In: Bojar, O., et al. (eds.) Proceedings of the Third Conference on Machine Translation: Research Papers, pp. 186–191. Association for Computational Linguistics, Brussels (2018). https://doi.org/10.18653/v1/W18-6319, https://aclanthology.org/W18-6319
19. Ranasinghe, T., Orasan, C., Mitkov, R.: TransQuest: translation quality estimation with cross-lingual transformers. In: Scott, D., Bel, N., Zong, C. (eds.) Proceedings of the 28th International Conference on Computational Linguistics, pp. 5070–5081. International Committee on Computational Linguistics, Barcelona (2020). https://doi.org/10.18653/v1/2020.coling-main.445, https://aclanthology.org/2020.coling-main.445/
20. Rei, R., et al.: COMET-22: Unbabel-IST 2022 submission for the metrics shared task. In: Koehn, P., et al. (eds.) Proceedings of the Seventh Conference on Machine Translation (WMT), pp. 578–585. Association for Computational Linguistics, Abu Dhabi (2022). https://aclanthology.org/2022.wmt-1.52/
21. Rei, R., et al.: CometKiwi: IST-unbabel 2022 submission for the quality estimation shared task. In: Koehn, P., et al. (eds.) Proceedings of the Seventh Conference on Machine Translation (WMT), pp. 634–645. Association for Computational Linguistics, Abu Dhabi (2022). https://aclanthology.org/2022.wmt-1.60
22. Scarton, C., Specia, L.: Learning simplifications for specific target audiences. In: Gurevych, I., Miyao, Y. (eds.) Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 712–718. Association for Computational Linguistics, Melbourne (2018). https://doi.org/10.18653/v1/P18-2113, https://aclanthology.org/P18-2113/
23. Schamper, J., Rosendahl, J., Bahar, P., Kim, Y., Nix, A., Ney, H.: The RWTH Aachen University supervised machine translation systems for WMT 2018. In: Bojar, O., et al. (eds.) Proceedings of the Third Conference on Machine Translation: Shared Task Papers, pp. 496–503. Association for Computational Linguistics, Belgium (2018). https://doi.org/10.18653/v1/W18-6426, https://aclanthology.org/W18-6426
24. Sellam, T., Das, D., Parikh, A.: BLEURT: learning robust metrics for text generation. In: Jurafsky, D., Chai, J., Schluter, N., Tetreault, J. (eds.) Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp.

7881–7892. Association for Computational Linguistics (2020). https://doi.org/10.18653/v1/2020.acl-main.704, https://aclanthology.org/2020.acl-main.704

25. Sennrich, R., Haddow, B., Birch, A.: Controlling politeness in neural machine translation via side constraints. In: Knight, K., Nenkova, A., Rambow, O. (eds.) Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 35–40. Association for Computational Linguistics, San Diego (2016). https://doi.org/10.18653/v1/N16-1005, https://aclanthology.org/N16-1005/
26. Sennrich, R., Haddow, B., Birch, A.: Neural machine translation of rare words with subword units. In: Erk, K., Smith, N.A. (eds.) Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1715–1725. Association for Computational Linguistics, Berlin (2016). https://doi.org/10.18653/v1/P16-1162, https://aclanthology.org/P16-1162
27. Team, Q.: Qwen3 technical report (2025). https://arxiv.org/abs/2505.09388
28. Tomani, C., et al.: Quality-aware translation models: efficient generation and quality estimation in a single model. In: Ku, L.W., Martins, A., Srikumar, V. (eds.) Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 15660–15679. Association for Computational Linguistics, Bangkok (2024). https://aclanthology.org/2024.acl-long.836
29. Vaswani, A., et al.: Attention is all you need. In: Guyon, I., et al. (eds.) Advances in Neural Information Processing Systems, vol. 30, pp. 5998–6008. Curran Associates, Inc. (2017). http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf
30. Xu, H., Liu, Q., van Genabith, J., Xiong, D., Zhang, M.: Multi-head highly parallelized LSTM decoder for neural machine translation. In: Zong, C., Xia, F., Li, W., Navigli, R. (eds.) Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 273–282. Association for Computational Linguistics (2021). https://doi.org/10.18653/v1/2021.acl-long.23, https://aclanthology.org/2021.acl-long.23
31. Yamagishi, H., Kanouchi, S., Sato, T., Komachi, M.: Controlling the voice of a sentence in Japanese-to-English neural machine translation. In: Nakazawa, T., et al. (eds.) Proceedings of the 3rd Workshop on Asian Translation (WAT2016), pp. 203–210. The COLING 2016 Organizing Committee, Osaka (2016). https://aclanthology.org/W16-4620/