Improving Machine Translation with Human Feedback: An Exploration of Quality Estimation as a Reward Model

Anonymous ACL submission

Abstract

Insufficient modeling of human preferences within the reward model is a major obstacle for leveraging human feedback to improve translation quality. Fortunately, quality estimation (QE), which predicts the quality of a given translation without reference, has achieved impressive alignment with human evaluations in the last two years. In this work, we investigate the potential of employing the QE model as the reward model (the QE-based reward model) to predict human preferences for feedback training. We first identify the *overoptimization* problem during QE-based feedback training, manifested as an increase in reward while translation quality declines. We examine the problem and argue that the vulnerability of the QE model might lead to high rewards for incorrect translations, resulting in overoptimization and error propagation. To address the problem, we adopt a simple yet effective method that uses heuristic rules to detect the incorrect translations and assigns a penalty term to the QE-based rewards for the detected incorrect translations. Experimental results show that the proposed QE-based feedback training achieves consistent and significant improvements across various settings, further verified through human preference studies. Our subsequent analysis demonstrates the high data efficiency of the proposed QE-based feedback training: the proposed approach using a small amount of monolingual data can outperform systems using larger parallel corpora.

1 Introduction

Human feedback has greatly contributed to recent advances in large language models (LLMs), aligning model behavior with human preferences and thereby enhancing the helpfulness and harmlessness of LLMs (Dong et al., 2023; Yuan et al., 2023; Zhao et al., 2023; Rafailov et al., 2023). The common practice involves using human evaluation data to train a reward model as a proxy for human preferences, followed by feedback training to fine-tune the LLM and maximize the reward score.

Early efforts in neural machine translation (NMT) also attempted to integrate feedback to improve translation quality. Most works used similarity scores (such as sentence-level BLEU (Papineni et al., 2002)) between the predicted translation and a reference translation to simulate feedback rather than employing feedback from humans (Sokolov et al., 2016a,b; Kreutzer et al., 2017; Sokolov et al., 2017; Lawrence et al., 2017; Nguyen et al., 2017; Wu et al., 2018; Wieting et al., 2019). Few attempts to use real human feedback have been made, and these efforts either used implicit feedback in limited scenarios (e.g., e-commerce) (Kreutzer et al., 2018a) or relied only on a minimal amount of human feedback data (Kreutzer et al., 2018b). Therefore, the integration of real human feedback in NMT has been constrained by inadequate modeling of human preferences.

Fortunately, the field of MT has seen substantial advancements in quality estimation (QE) (Rei et al., 2021, 2022b; Wan et al., 2022). A QE model offers a reference-free estimation of translation quality and has been facilitated by the growing availability of human evaluation data (Specia et al., 2020, 2021; Zerva et al., 2022) and the development of pre-trained language models (Devlin et al., 2019; Conneau et al., 2020). Typically, given a source sentence and its translation, a sentence-level QE model can provide a numerical score to indicate the quality of the translation. The most advanced QE models to date have achieved impressive alignment with human evaluations (Freitag et al., 2022).

In light of this progress, we explore the potential of utilizing QE models as proxies of human preferences and functioning them as reward models in feedback training for the first time. Firstly, we identify the *overoptimization* problem in feedback training, manifested as an increase in reward while translation quality declines. Our analysis reveals that the underlying issue lies in the vulnerability of QE-based reward models, which, in rare instances, assign high scores to patently incorrect translations. As a result, these flawed patterns spread through subsequent training, leading to divergence from human preferences and a training collapse. However, the reward keeps rising throughout the whole process. This phenomenon aligns with the observation from Fernandes et al. (2023) that "overoptimizing" against an imperfect reward model can lead to systems "that receive good feedback from the model, but not humans".

Through manual examination of these flawed patterns, we categorize the most prevalent errors into two groups: length ratio errors and off-target errors (i.e., not the desired target language). Guided by this observation, we propose a simple yet effective solution to mitigate overoptimization. We first detect these errors and then add a negative penalty to the reward for these erroneous translations. We show that this approach significantly alleviates the overoptimization problem and results in notable improvements across various settings, which are further verified by human preference studies.

In summary, the contributions of this work are detailed as follows:

- We identify the *overoptimization* problem when using QE-based reward models for feedback training and verify the ubiquity of this phenomenon across comprehensive settings (12 in total): 3 QE models × 2 model architectures (decoder-only LLM and encoder-decoder NMT models) × 2 resource settings (high-resource and low-resource).

- By addressing the overoptimization with a simple yet effective method, we successfully integrate the QE model into feedback training for the first time, achieving remarkable improvements across various settings.

- Through further analysis, we demonstrate the high data efficiency of using the QE-based reward model for feedback training, showing that it can outperform systems using larger parallel corpora by only a small amount of monolingual data.

- We investigate the influence of the base model on feedback training, finding that stronger base models (larger in size and pretrained) yield greater improvements after feedback training. We also examine the effect of crucial hyperparameters.

2 Feedback Training

2.1 Formulation

We denote an MT model as $M = P(y|x;\theta)$, with model parameters θ. It takes a source sentence x (or a prompt) as input and generates a target sentence y according to the distribution $P_T(y|x;\theta)$, where T is a temperature parameter to control the diversity. We consider the QE-based reward model as $r(x,y)$ whose value indicates the quality of y as the translation of x. Taken $\mathcal{D}$ as the training distribution of x, the optimization objective is:

$$\max_{\theta} \mathbb{E}_{x\sim\mathcal{D},y\sim P(y|x;\theta)} r(x,y). \tag{1}$$

We adopt the local ranking version of Reward rAnked FineTuning (RAFT) (Dong et al., 2023) to train the model M. The basic idea of RAFT is to rate the generated candidates from a prompt using the reward model and just learn from the best one of them. It has been proved to be more stable and efficient than Proximal Policy Optimization (PPO) (Schulman et al., 2017). Algorithm 1 shows the details of RAFT.

Algorithm 1 RAFT

```
Require: Training set $\mathcal{X}$, reward function $r(x,y)$, initial model $M_0 = P(y|x;\theta_0)$, batch size $b$, temperature $T$, the number of candidate $k$
for iteration $i$ in $0, 1, \ldots, N-1$ do
    $D_i \leftarrow \text{SampleBatch}(\mathcal{X}, b)$
    $\mathcal{B} = \emptyset$
    for $x \in D_i$ do
        $y_1, \ldots, y_k \sim P_T(y|x;\theta_i)$
        $y^* = \arg\max_{y_j \in \{y_1,\ldots,y_k\}} r(x, y_j)$
        $\mathcal{B} = \mathcal{B} \cup \{(x, y^*)\}$
    Fine-tune $\theta_i$ on $\mathcal{B}$ to obtain $M_{i+1} = P(y|x;\theta_{i+1})$.
```

2.2 Addressing Overoptimization Problem

Overoptimization Our preliminary experiment observed *that as the reward increases, the translation performance deteriorates* (shown in Figure 1). This phenomenon is dubbed as *overoptimization* by Gao et al. (2023). The underlying reason for overoptimization is that the reward model does not serve as a perfect proxy for human preferences. Hence, overoptimizing rewards could steer the model's behavior away from human preferences. This aligns with Goodhart's Law, which states, "When a measure becomes a target, it ceases to be a good measure."

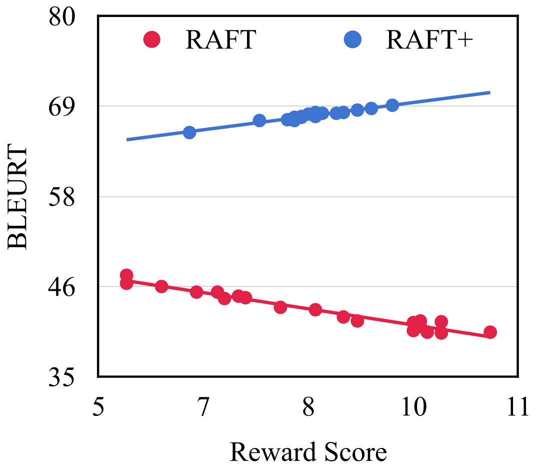


Figure 1: The relationship between reward score and translation quality (in terms of BLEURT). Each point represents the average performance of a checkpoint on the development sets. COMET-QE-DA is used as QE-based reward model.

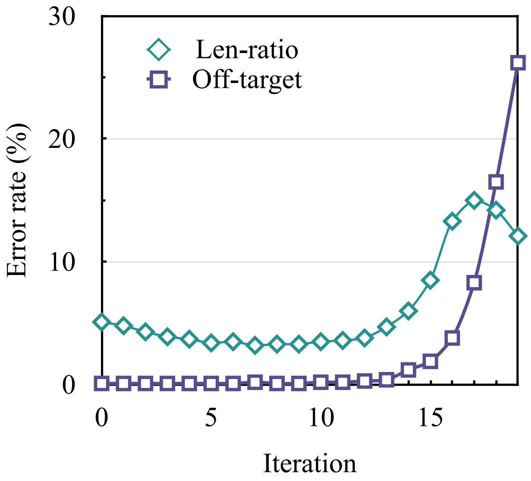


Figure 2: Trends in length-ratio and off-target error rates with the training process. Even though these errors do not manifest significantly during the early and middle phases of training, they may still surge in later stages.

Causes We further found that the reward model may assign high scores to erroneous translations in some cases (as shown in Table 1). These errors encompass common error patterns in MT, such as length-ratio, off-target errors, and hallucinations[1]. While these errors may not be severe initially, they can rapidly propagate to subsequent training stages (see Figure 2) once they are accorded high reward scores, which can lead to disruption of the entire training process.

Error type	Translation	Reward
None	The rule of drinking Red Label Whisky:	2.84
Len-ratio (too long/short translation)	The rule of drinking Red Label Whisky: 1. Always drink responsibly. 2. Never drink alone. 3. Avoid drinking on an empty stomach. 4. Set limits and stick to them. 5. Drink in moderation.	5.60
Off-target (wrong target language)	So trinkt man Red-Label-Whisky:	4.58

Table 1: A case of Chinese⇒English translation where the QE model (COMET-QE-DA) assigns higher scores to length-ratio and off-target errors than an error-free translation. Error spans are highlighted.

Solution To alleviate overoptimization, we monitor length-ratio and off-target errors during training and assign negative, punitive rewards for these errors. Let $C(x, y)$ be true if $\frac{|y|}{|x|} \notin [L, U]$ or $lang(y) \neq$ target language, where $[L, U]$ is an acceptable length ratio interval and $lang(\cdot)$ is a language identification function (detailed in Appendix A). Then the reward modification can be expressed as:

$$r^{+}(x, y) = \begin{cases} r(x, y) - P & \text{if } C(x, y) \\ r(x, y) & \text{otherwise,} \end{cases} \tag{2}$$

where P is the penalty term that we simply set to $-\infty$ since RAFT is an algorithm based on data selection. We refer to this approach as RAFT+. As depicted in Figure 1, RAFT+ facilitates a trend of concurrent improvement in reward and translation quality. Notably, the relationship between reward and translation quality approximates a linear correlation. This suggests that optimizing QE-based reward can be an effective strategy in instances devoid of overoptimization.

3 Experiments

3.1 Experimental Setup

Pipeline We adopt the pipeline of reinforcement learning with human feedback (RLHF) (Ouyang et al., 2022). Starting with the pretrained base model, we carry out the following steps: (1) Supervised fine-tuning (SFT), where we utilize parallel data to fine-tune the pretrained base model, thereby obtaining an initial MT model. (2) Feedback training, where we train the model using RAFT/RAFT+ to maximize the reward from a QE-based reward

[1] For sake of simplicity, we categorize repeating errors as length-ratio errors, and copying-source errors are considered as one kind of off-target errors.

model. Note that this stage only uses monolingual data.

Model For the base models, we adopt:

- Llama-2-7B (Touvron et al., 2023): a decoder-only LLM primarily trained in English, with the objective to predict the next token.

- NLLB-200-1.3B (Costa-jussà et al., 2022): an encoder-decoder model that trained for multilingual translation across 200 languages. After SFT, it will be adapted to the language pairs we considered.

For QE models, we use:

- COMET-QE-DA (Rei et al., 2021): A QE model trained on *Direct Assessments* data.

- COMET-QE-MQM (Rei et al., 2021): Fine-tuned COMET-QE-DA model using *Multidimensional Quality Metric* (MQM) data.

- UniTE-MUP (Wan et al., 2022): A unified translation evaluation framework that is jointly trained for reference-only, source-only, and source-reference-combined evaluation. We only use its source-only evaluation.

Data We consider the settings for both high-resource and low-resource language pairs. In the high-resource setting, we focus on English⇔Chinese and English⇔German. In the low-resource setting, our focus is on English⇔Ukrainian and Ukrainian⇔Czech. It is important to note that both settings are multilingual, where all four translation directions are trained concurrently within one model, and we ensure the balance of all directions to avoid introducing other factors. That is, every direction has an equal number of training samples. We adopt WMT22 (Kocmi et al., 2022) as the test sets for all language pairs. For development sets, high resource setting uses WMT21, and low resource setting uses Flores200 (Goyal et al., 2022). Table 2 lists the detail of the data.

Training details For SFT, we conducted training for one epoch. For RAFT/RAFT+, we trained Llama-2-7B for 10 iterations using a learning rate of 2e-6, and for NLLB-200-1.3B, we trained for 20 iterations with a learning rate of 2e-5. We set the batch size b to 1024, the number of candidates k to 8, and the temperature T to 0.85.

		High-res	**Low-res**
Train	SFT (para)	WMT (2M)	Wiki (200K)
	FB (mono)	CC (85K)	Wiki (84K)
Dev		WMT21	Flores200
Test		WMT22	WMT22

Table 2: Data used in the different phases of our pipeline. SFT uses parallel data, while FB uses monolingual data. Numbers indicate the number of samples. "FB" denotes feedback training. "CC" and "Wiki" denote CCMatrix (Schwenk et al., 2021b) and WikiMatrix (Schwenk et al., 2021a). WMT and CC are both general domain.

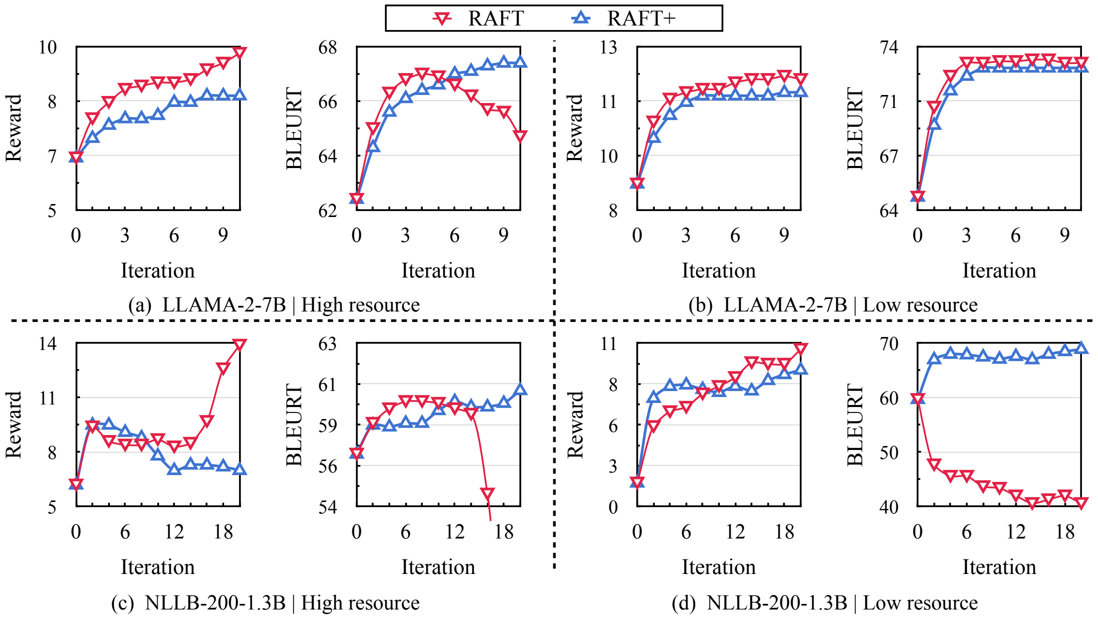
Evaluation We use COMET (Rei et al., 2022a) and BLEURT (Sellam et al., 2020) to assess translation quality. These neural metrics show superiority over string-based metrics like BLEU (Freitag et al., 2022; Kocmi et al., 2021; Bawden and Yvon, 2023). We use `Unbabel/wmt22-comet-da` and `BLEURT-20` checkpoints for these two metrics. It is worth noting that we also observe the overoptimization problem on COMET, that is, COMET increased but the actual translation quality decreased (see § 3.2). Therefore, we use BLEURT as the main metric, but still report COMET as a reference and conduct human evaluation in § 4.1. We select the best checkpoint based on the performance on the dev sets, and report the final results on test sets using beam search (beam size = 4).

3.2 Results

Training curves Figure 3 illustrates the training curves of reward and BLEURT on the development sets when using COMET-QE-DA as the QE-based reward model. Our observations are:

• **Overoptimization is a phenomenon of high frequency when using vanilla RAFT.** Three out of the four settings (Figure 3a, c, d) have the situation that the reward shows an increase while the BLEURT declines. We further verify this phenomenon in Appendix C when using COMET-QE-MQM or UniTE-MUP as the reward model.

• **The severity of overoptimization varies under different settings.** Figure 3d represents the most severe overoptimization, where the BLEURT score starts decreasing from the onset of the training process. Figure 3a and Figure 3c exhibit a trend of initial increase followed by a decrease. In such scenarios, a relatively good checkpoint can be chosen based on the performance on the development set. Conversely, Figure 3b does not display overoptimization. We conjecture that the severity of overoptimization could be related to multiple fac-



(a) LLAMA-2-7B | High resource

(b) LLAMA-2-7B | Low resource

(c) NLLB-200-1.3B | High resource

(d) NLLB-200-1.3B | Low resource

Figure 3: Training curves under various settings. The metrics are average values for all language pairs on the development set. The QE-based reward model is COMET-QE-DA.

tors, including language pairs, the reward model, and the SFT model.

• **RAFT+ alleviates overoptimization effectively.** The BLEURT scores of RAFT+ in the four settings consistently increase as the training progresses. At the same time, the growth rate of the reward scores in RAFT+ is significantly slower than that in RAFT. We even witness a situation where the reward decreases in Figure 3c. Moreover, in Figure 3b where overoptimization did not occur originally, RAFT+ can still achieve a performance close to that of RAFT. However, it is worth noting that the overoptimization problem has not been completely solved, as some errors, like hallucinations, are challenging to identify. Furthermore, we cannot guarantee generalization to all other language pairs since language identification is not reliable for extremely low-resource language (Aji et al., 2022), indicating room for improvement. We leave a more comprehensive study to future work.

Main results Table 3 shows the main results on test sets when using COMET-QE-DA or COMET-QE-MQM as the QE-based reward model. We present the results of UNITE-MUP in Appendix D and use chrF as the evaluation metric in Appendix E.

• **Feedback training brings significant improvements in general.** Regardless of whether the resource setting is high or low and irrespective of the variation in base models, RAFT+ tends to deliver notable enhancements, particularly pronounced in the low resource setting. This suggests that current QE models are already equipped with the capability to function as reward models after addressing the overoptimization problem. In contrast, while achieving good gains in most settings, RAFT can suffer from severe overoptimization that might lead to failed training.

• **The performance of the reward model varies under different resource settings.** In the high-resource setting, COMET-QE-MQM outperforms COMET-QE-DA, while the opposite is true in the low-resource setting, which suggests that QE models still have a lot of room for improvement.

• **Remarkably, even COMET, a reference-based metric, can be overoptimized.** In RAFT training of NLLB-200-1.3B under low-resource setting (bottom of Table 3b), there is a marginal increase in COMET scores, yet a significant drop in BLEURT, which is unusual. By inspecting the outputs, we found that severe off-target errors significantly hamper the model's performance. One plausible reason might be the similarities between the COMET and COMET-QE models, both of which may be susceptible to these off-target errors. This finding underscores the necessity of treating automatic metrics with caution and the importance of employing multiple metrics simultaneously for a more comprehensive evaluation. It also stresses the significance of recognizing a metric model's vulnerabilities during its development, including

Method	De⇒En		En⇒De		Zh⇒En		En⇒Zh		**Average**	
	COMET	BLEURT	COMET	BLEURT	COMET	BLEURT	COMET	BLEURT	**COMET**	**BLEURT**
LLAMA-2-7B										
SFT	82.5	70.5	80.7	68.2	76.1	62.3	84.9	69.3	81.0	67.6
REWARD MODEL: COMET-QE-DA										
RAFT	83.7	72.1	82.8	71.1	78.7	65.3	85.9	70.1	$82.8_{\uparrow 1.7}$	$69.7_{\uparrow 2.1}$
RAFT+	83.6	72.1	84.4	73.9	79.0	66.1	85.4	69.3	**83.1**$_{\uparrow 2.1}$	**70.3**$_{\uparrow 2.7}$
REWARD MODEL: COMET-QE-MQM										
RAFT	83.3	72.0	84.8	75.1	77.8	64.3	86.1	70.4	$83.0_{\uparrow 2.0}$	$70.5_{\uparrow 2.9}$
RAFT+	83.7	72.4	85.6	75.7	78.6	65.6	85.8	70.0	**83.4**$_{\uparrow 2.4}$	**70.9**$_{\uparrow 3.3}$
NLLB-200-1.3B										
SFT	70.9	52.5	85.3	74.8	66.0	48.4	83.7	69.1	76.5	61.2
REWARD MODEL: COMET-QE-DA										
RAFT	73.2	52.2	85.8	75.1	67.9	50.5	84.2	68.9	$77.8_{\uparrow 1.3}$	$61.7_{\uparrow 0.5}$
RAFT+	74.2	56.7	85.8	75.2	69.0	52.6	84.0	67.9	**78.2**$_{\uparrow 1.7}$	**63.1**$_{\uparrow 1.9}$
REWARD MODEL: COMET-QE-MQM										
RAFT	82.8	71.3	83.9	73.4	76.1	62.3	84.6	68.6	$81.8_{\uparrow 5.3}$	$68.9_{\uparrow 7.7}$
RAFT+	83.3	71.8	84.6	74.4	76.7	62.9	84.6	68.4	**82.3**$_{\uparrow 5.8}$	**69.4**$_{\uparrow 8.2}$

(a) High-resource language pairs

Method	En⇒Uk		Uk⇒En		Uk⇒Cs		Cs⇒Uk		**Average**	
	COMET	BLEURT	COMET	BLEURT	COMET	BLEURT	COMET	BLEURT	**COMET**	**BLEURT**
LLAMA-2-7B										
SFT	79.2	64.0	76.7	66.0	70.0	53.2	71.2	51.3	74.3	58.6
REWARD MODEL: COMET-QE-DA										
RAFT	82.3	68.0	81.4	71.1	82.5	69.5	84.3	69.9	**82.6**$_{\uparrow 8.3}$	**69.6**$_{\uparrow 11.0}$
RAFT+	82.0	67.8	81.5	71.2	82.2	68.8	84.5	70.1	**82.6**$_{\uparrow 8.3}$	$69.5_{\uparrow 10.9}$
REWARD MODEL: COMET-QE-MQM										
RAFT	80.7	65.5	76.7	66.0	75.7	59.9	75.2	54.8	$77.1_{\uparrow 2.8}$	$61.5_{\uparrow 2.9}$
RAFT+	81.2	67.0	79.2	68.9	77.3	62.3	78.8	60.7	**79.1**$_{\uparrow 4.8}$	**64.8**$_{\uparrow 6.2}$
NLLB-200-1.3B										
SFT	83.1	70.2	71.1	62.7	73.2	61.5	57.3	43.4	71.2	59.4
REWARD MODEL: COMET-QE-DA										
RAFT	85.2	72.5	64.7	33.2	70.5	29.7	73.8	30.1	$73.6_{\uparrow 2.4}$	$41.4_{\downarrow 18.0}$
RAFT+	84.5	71.3	77.7	67.0	83.1	70.3	72.0	55.1	**79.3**$_{\uparrow 8.1}$	**65.9**$_{\uparrow 6.6}$
REWARD MODEL: COMET-QE-MQM										
RAFT	85.8	73.2	67.5	50.0	71.1	41.6	71.1	42.7	$73.9_{\uparrow 2.7}$	$51.9_{\downarrow 7.5}$
RAFT+	84.5	71.8	76.4	66.1	82.1	69.9	71.4	54.5	**78.6**$_{\uparrow 7.4}$	**65.6**$_{\uparrow 6.2}$

(b) Low-resource language pairs

Table 3: Translation performance on the test sets under various settings, using COMET-QE-DA and COMET-QE-MQM as reward models. Results when UNITE-MUP is used as the reward model are presented in Appendix D. Bold indicates that the average performance of the method exceeds that of SFT and RAFT/RAFT+ within the same QE model. The subscripts indicate the change in performance relative to the SFT.

adjustments for prevalent translation inaccuracies such as length-ratio, off-target errors, hallucinations, and so forth.

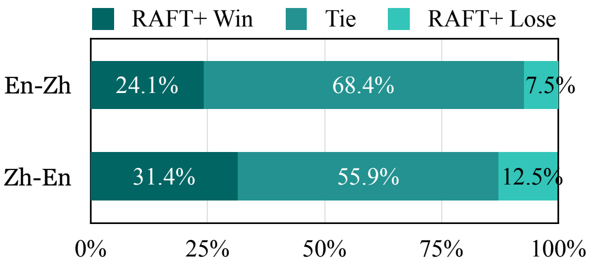


Figure 4: Human preference evaluation, comparing RAFT+ to SFT model on En⇔Zh test sets.

4 Analysis

4.1 Human evaluation

As discussed in § 3.2, automatic metrics may not correlate perfectly with actual translation quality. Therefore, we perform human preference studies on En⇔Zh test sets. For each test sample, our annotators (professional translators) were presented with a source sentence and two possible translations generated by either the SFT or RAFT+ model (based on Llama-2-7B). They were then tasked with selecting the superior translation or determining that neither translation was better than the other. Figure 4 shows the results of human preference studies. We find that RAFT+ achieves better or equal translations in 92.5% of the cases for En-Zh and 87.3% for Zh-En, compared to SFT, confirming the effectiveness of feedback training.

4.2 Data Efficiency of Feedback Training

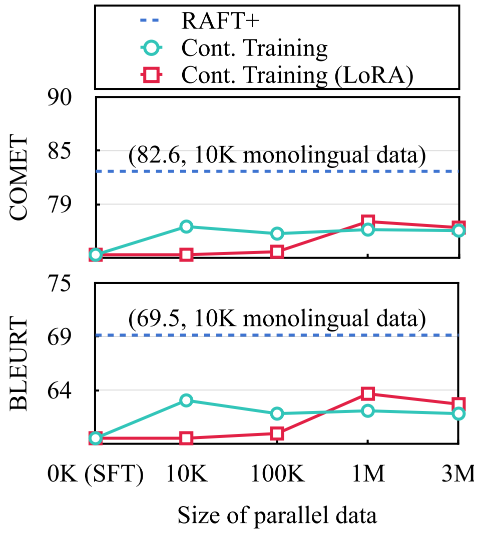


Figure 5: Comparison between RAFT+ and continuous training in the low-resource setting.

Data efficiency refers to the capacity to achieve good performance with less training data. For low-resource language pairs, data efficiency is important since it is costly and labor-intensive to annotate high-quality parallel data. Feedback training provides an alternative path where humans only need to evaluate rather than generate translations.

We examine the data efficiency of feedback training in the low-resource setting. We collected an extra 3M of parallel data from WikiMatrix (Schwenk et al., 2021a) and continued training the SFT model (based on Llama-2-7B) using data of different sizes. We adopted two continued training strategies: full parameter fine-tuning and parameter-efficient fine-tuning, specifically employing LoRA (Hu et al., 2022). To be fair, we filtered samples with length-ratio and off-target errors in the parallel data. On the other hand, RAFT+ only consumes 10K monolingual data (1024 batch size × 10 iterations). Figure 5 depicts their average performance on test sets. Unexpectedly, the continuous training with increasing amounts of parallel data fails to yield consistent improvements. This observation aligns with a similar phenomenon reported by Zhou et al. (2023). A plausible explanation could be the low quality of the crawled data for low-resource languages. Conversely, RAFT+ performs markedly better using merely 10K monolingual data, exhibiting high data efficiency. It is also worth noting that in RAFT+ the model is only trained on self-generated translations. Therefore, we conjecture that the SFT model already has strong translation potential inherently, and RAFT+ can bring this potential to fruition.

4.3 Effects of Scaling Model Size and Pretraining

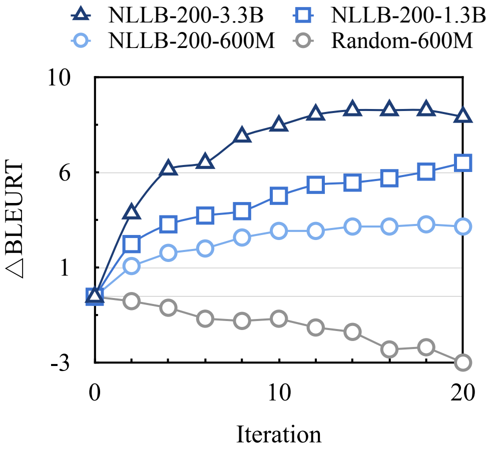


Figure 6: Training curves of RAFT+ (high-resource, COMET-QE-MQM) under different base models. We report the change in BLEURT score for each checkpoint relative to the SFT model.

In this section, we examine the effects of two dimensions of the base model on feedback training: model size and the presence of pretraining. We conduct experiments under the high-resource setting, using COMET-QE-MQM as the reward model. For model size, we consider NLLB-200-(600M, 1.3B, 3.3B) as base models; for pretraining, we randomly initialize NLLB-200-600M and perform the SFT from scratch with 80M parallel data, which we call Random-600M. From Figure 6, we have two obvious phenomena: (1) a larger base model size results in a more significant enhancement from feedback training; (2) feedback training is effective only when the base model has undergone pretraining. Combining the two, we deduce that a stronger base model results in more significant improvements from feedback training. A plausible rationale for this could be that a well-established base model inherently possesses great potential, which can be further unlocked by suitable feedback.

4.4 Effects of Hyperparameters

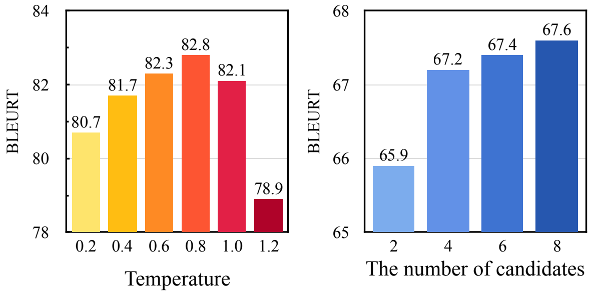


Figure 7: Effects of temperature and the number of candidates on the final performance (high-resource setting, LLAMA-2-7B, COMET-QE-DA).

The sampling temperature and the number of candidates are two crucial hyperparameters for RAFT+. Both influence the diversity of the candidates. Intuitively, having a broader range of candidates may provide richer feedback signals, potentially leading to better final performance. However, as illustrated in Figure 7, (1) continuously increasing the temperature leads to performance degradation; (2) the boost from continuously increasing the number of candidates approaches saturation. This means that the increase in candidate diversity has reached its upper limit.

5 Related Work

5.1 Feedback Training in MT

Many early works have attempted to use various forms of feedback to improve MT, where the model has no access to the gold references but receives partial feedback on its output. Depending on the source of the feedback, they can be categorized into simulated or human feedback.

Simulated feedback This type of work utilizes a measure of similarity between the model's output and the reference translation to simulate feedback, e.g., sentence-BLEU and cross-entropy loss (Sokolov et al., 2016a,b). Kreutzer et al. (2017) lifts linear bandit learning to neural sequence-to-sequence learning. Lawrence et al. (2017) demonstrates the possibility of counterfactual learning from deterministic bandit logs. Nguyen et al. (2017) propose a training algorithm combining the advantage actor-critic algorithm with the attention-based neural encoder-decoder architecture. Wieting et al. (2019) proposes to use semantic similarity rather than BLEU as simulated feedback. Wu et al. (2018) discusses how to train the MT model using feedback effectively. The main drawback of simulated feedback is that there is no real human involvement, so the metrics being optimized, such as BLEU, do not correlate well with human preferences (Freitag et al., 2022). In addition, the need for references presents a challenge for low-resource language pairs.

Human feedback Kreutzer et al. (2018a) utilizes implicit human feedback in a constrained e-commerce scenario, which cannot be applied to general MT. Kreutzer et al. (2018b) investigates how to improve translation using human evaluation data but is limited by the small data size (1K).

5.2 Aligning LLMs with Human

Pretrained on raw text from the internet, LLMs might generate toxic, inaccurate, and unhelpful content (Fernandes et al., 2023). To mitigate these issues, researchers have employed human feedback to better align the behavior of LLMs with human preferences, thereby enhancing their helpfulness and reducing potential harm (Ouyang et al., 2022; Stiennon et al., 2020; Bai et al., 2022). The basic idea involves learning a reward function that capturing human preferences and optimizing the LLM using proximal policy optimization (PPO) (Schulman et al., 2017). Lighter weight training strategies, such as RAFT (Dong et al., 2023) and RRHF (Yuan et al., 2023), use training data ranking based on rewards to align the LLM. LIMA (Zhou et al., 2023) underscores the importance of high-quality supervised data for effective alignment. For a comprehensive understanding of alignment, we recommend readers refer to survey papers: Fernandes et al. (2023) and Wang et al. (2023).

6 Conclusion

We explore the potential of using the current QE model as a reward model. By identifying, analyzing, and mitigating the overoptimization problem, we successfully integrate the QE model into feedback training to refine translation quality. We validate its effectiveness across various settings, including human evaluation. Further analysis demonstrates the high data efficiency of feedback training using the QE-based reward model. Lastly, we delve into the impact of the base model and hyperparameters on feedback training.

Limitations

Training Algorithm This work only considers a simple yet stable training algorithm, RAFT, in favor of its ease of use and reduced computational requirements. We also implemented the Minimum Risk Training (MRT) algorithm (Shen et al., 2016) in Appendix B, but found it difficult to train stably. While other more commonly used reinforcement learning (RL) algorithms such as Proximal Policy Optimization (PPO) (Schulman et al., 2017) exist, their training stability can be influenced by a variety of factors and often demands substantial GPU memory (Dong et al., 2023; Zheng et al., 2023). Hyperparameters, stability, and computational constraints led us to consciously limit our exploration to RAFT, thereby not investigating other potentially effective but more complex training algorithms. This choice simplifies the training process but might limit the general applicability of the approach in different contexts. Future work may explore these alternative algorithms, acknowledging the trade-offs between complexity, stability, and performance.

Granularity of Quality Estimation This work exclusively focuses on sentence-level QE, neglecting other granularities, such as word-level or document-level QE. The choice of granularity inherently limits the scope of our insights and applications. Understanding how word-level QE feedback might be utilized, or how QE could be employed to enhance document translation quality, presents exciting avenues for future research.

Imbalance in Multilingual NMT - Data This work does not consider the imbalances present in multilingual NMT. In real-world scenarios, the quantity of available corpus varies significantly across different languages, which might lead to biased or suboptimal results. However, considering unbalanced scenarios introduces many additional factors, such as different ways of sampling training data. Therefore, we constrain our focus to the balanced situation in this work, recognizing the need for future research to address this complexity and explore methods that can better handle the imbalances inherent in multilingual contexts.

Imbalance in Multilingual NMT - Reward Although we kept the amount of data the same for the different language pairs, we still observed training imbalance problems due to "uneven" distribution of rewards. Specifically, in Table 3a, neither RAFT nor RAFT+ consistently improves NLLB in En→De and En→Zh (both are *from-English* direction) at the high resource setting. However, on the other hand, the performance of the other two *to-English* directions (De→En and Zh→En) has been significantly improved by notable margins (BLEURT | De→En: 53.5→71.3, Zh→En: 48.4→62.9). This means that during training, the model allocates more "capacity" to the *to-English* directions than to the *from-English* directions, i.e., they are not balanced. The intuitive reason is that the model "finds" optimizing the *to-English* directions offers a quick boost in reward, thereby "ignoring" the *from-English* directions which have less room for improvement. This involves *how the reward can be reasonably distributed among different language pairs*, which is less discussed in this work, but a meaningful and underexplored future direction.

Applicability of the Method to Mitigate Overoptimization The current method for mitigating overoptimization focuses on detecting relatively easy-to-identify errors such as len-ratio and off-target errors. More elusive mistakes, such as hallucinations, remain unaddressed and might potentially lead to overoptimization as well (Guerreiro et al., 2023). In addition, current language detectors are not reliable for extremely low-resource languages (Aji et al., 2022), limiting the applicability in these contexts.

References

Alham Fikri Aji, Genta Indra Winata, Fajri Koto, Samuel Cahyawijaya, Ade Romadhony, Rahmad Mahendra, Kemal Kurniawan, David Moeljadi, Radityo Eko Prasojo, Timothy Baldwin, Jey Han Lau, and Sebastian Ruder. 2022. One country, 700+ languages: NLP challenges for underrepresented languages and dialects in Indonesia. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7226–7249, Dublin, Ireland. Association for Computational Linguistics.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Rachel Bawden and François Yvon. 2023. Investigating the translation performance of a large multilingual language model: the case of bloom. *arXiv preprint arXiv:2303.01911*.

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

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

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

Hanze Dong, Wei Xiong, Deepanshu Goyal, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. 2023. Raft: Reward ranked finetuning for generative foundation model alignment. *arXiv preprint arXiv:2304.06767*.

Patrick Fernandes, Aman Madaan, Emmy Liu, António Farinhas, Pedro Henrique Martins, Amanda Bertsch, José GC de Souza, Shuyan Zhou, Tongshuang Wu, Graham Neubig, et al. 2023. Bridging the gap: A survey on integrating (human) feedback for natural language generation. *arXiv preprint arXiv:2305.00955*.

Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and André F. T. Martins. 2022. Results of WMT22 metrics shared task: Stop using BLEU – neural metrics are better and more robust. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 46–68, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Leo Gao, John Schulman, and Jacob Hilton. 2023. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning*, pages 10835–10866. PMLR.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Nuno M. Guerreiro, Elena Voita, and André Martins. 2023. Looking for a needle in a haystack: A comprehensive study of hallucinations in neural machine translation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1059–1075, Dubrovnik, Croatia. Association for Computational Linguistics.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Tom Kocmi, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Thamme Gowda, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Rebecca Knowles, Philipp Koehn, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Michal Novák, Martin Popel, and Maja Popović. 2022. Findings of the 2022 conference on machine translation (WMT22). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 1–45, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Tom Kocmi, Christian Federmann, Roman Grundkiewicz, Marcin Junczys-Dowmunt, Hitokazu Matsushita, and Arul Menezes. 2021. To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 478–494, Online. Association for Computational Linguistics.

Julia Kreutzer, Shahram Khadivi, Evgeny Matusov, and Stefan Riezler. 2018a. Can neural machine translation be improved with user feedback? In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*, pages 92–105, New Orleans - Louisiana. Association for Computational Linguistics.

Julia Kreutzer, Artem Sokolov, and Stefan Riezler. 2017. Bandit structured prediction for neural sequence-to-sequence learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1503–1513, Vancouver, Canada. Association for Computational Linguistics.

Julia Kreutzer, Joshua Uyheng, and Stefan Riezler. 2018b. Reliability and learnability of human bandit feedback for sequence-to-sequence reinforcement learning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1777–1788, Melbourne, Australia. Association for Computational Linguistics.

Carolin Lawrence, Artem Sokolov, and Stefan Riezler. 2017. Counterfactual learning from bandit feedback under deterministic logging : A case study in statistical machine translation. In *Proceedings of the 2017*

Conference on Empirical Methods in Natural Language Processing, pages 2566–2576, Copenhagen, Denmark. Association for Computational Linguistics.

Khanh Nguyen, Hal Daumé III, and Jordan Boyd-Graber. 2017. Reinforcement learning for bandit neural machine translation with simulated human feedback. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1464–1474, Copenhagen, Denmark. Association for Computational Linguistics.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Gray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*.

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.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*.

Ricardo Rei, José G. C. de Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André F. T. Martins. 2022a. COMET-22: Unbabel-IST 2022 submission for the metrics shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 578–585, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Ricardo Rei, Ana C Farinha, Chrysoula Zerva, Daan van Stigt, Craig Stewart, Pedro Ramos, Taisiya Glushkova, André F. T. Martins, and Alon Lavie. 2021. Are references really needed? unbabel-IST 2021 submission for the metrics shared task. In *Proceedings of the Sixth Conference on Machine Translation*, pages 1030–1040, Online. Association for Computational Linguistics.

Ricardo Rei, Marcos Treviso, Nuno M. Guerreiro, Chrysoula Zerva, Ana C Farinha, Christine Maroti, José G. C. de Souza, Taisiya Glushkova, Duarte Alves, Luisa Coheur, Alon Lavie, and André F. T. Martins. 2022b. CometKiwi: IST-unbabel 2022 submission for the quality estimation shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 634–645, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2021a. WikiMatrix: Mining 135M parallel sentences in 1620 language pairs from Wikipedia. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1351–1361, Online. Association for Computational Linguistics.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, Armand Joulin, and Angela Fan. 2021b. CCMatrix: Mining billions of high-quality parallel sentences on the web. In *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)*, pages 6490–6500, Online. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Rico Sennrich, Orhan Firat, Kyunghyun Cho, Alexandra Birch, Barry Haddow, Julian Hitschler, Marcin Junczys-Dowmunt, Samuel Läubli, Antonio Valerio Miceli Barone, Jozef Mokry, and Maria Nădejde. 2017. Nematus: a toolkit for neural machine translation. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 65–68, Valencia, Spain. Association for Computational Linguistics.

Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Minimum risk training for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1683–1692, Berlin, Germany. Association for Computational Linguistics.

Artem Sokolov, Julia Kreutzer, Christopher Lo, and Stefan Riezler. 2016a. Learning structured predictors from bandit feedback for interactive NLP. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1610–1620, Berlin, Germany. Association for Computational Linguistics.

Artem Sokolov, Julia Kreutzer, Stefan Riezler, and Christopher Lo. 2016b. Stochastic structured prediction under bandit feedback. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

Artem Sokolov, Julia Kreutzer, Kellen Sunderland, Pavel Danchenko, Witold Szymaniak, Hagen Fürstenau, and Stefan Riezler. 2017. A shared task on bandit learning for machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 514–524, Copenhagen, Denmark. Association for Computational Linguistics.

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

Lucia Specia, Frédéric Blain, Marina Fomicheva, Chrysoula Zerva, Zhenhao Li, Vishrav Chaudhary, and André F. T. Martins. 2021. Findings of the WMT 2021 shared task on quality estimation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 684–725, Online. Association for Computational Linguistics.

Peter M. Stahl. 2023. lingua-py.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Yu Wan, Dayiheng Liu, Baosong Yang, Haibo Zhang, Boxing Chen, Derek Wong, and Lidia Chao. 2022. UniTE: Unified translation evaluation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8117–8127, Dublin, Ireland. Association for Computational Linguistics.

Yufei Wang, Wanjun Zhong, Liangyou Li, Fei Mi, Xingshan Zeng, Wenyong Huang, Lifeng Shang, Xin Jiang, and Qun Liu. 2023. Aligning large language models with human: A survey. *arXiv preprint arXiv:2307.12966*.

John Wieting, Taylor Berg-Kirkpatrick, Kevin Gimpel, and Graham Neubig. 2019. Beyond BLEU:training neural machine translation with semantic similarity. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4344–4355, Florence, Italy. Association for Computational Linguistics.

Lijun Wu, Fei Tian, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. 2018. A study of reinforcement learning for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3612–3621, Brussels, Belgium. Association for Computational Linguistics.

Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. 2023. Rrhf: Rank responses to align language models with human feedback without tears. *arXiv preprint arXiv:2304.05302*.

Chrysoula Zerva, Frédéric Blain, Ricardo Rei, Piyawat Lertvittayakumjorn, José G. C. de Souza, Steffen Eger, Diptesh Kanojia, Duarte Alves, Constantin Orăsan, Marina Fomicheva, André F. T. Martins, and Lucia Specia. 2022. Findings of the WMT 2022 shared task on quality estimation. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 69–99, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Yao Zhao, Mikhail Khalman, Rishabh Joshi, Shashi Narayan, Mohammad Saleh, and Peter J Liu. 2023. Calibrating sequence likelihood improves conditional language generation. In *The Eleventh International Conference on Learning Representations*.

Rui Zheng, Shihan Dou, Songyang Gao, Wei Shen, Binghai Wang, Yan Liu, Senjie Jin, Qin Liu, Limao Xiong, Lu Chen, et al. 2023. Secrets of rlhf in large language models part i: Ppo. *arXiv preprint arXiv:2307.04964*.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. 2023. Lima: Less is more for alignment. *arXiv preprint arXiv:2305.11206*.

A Error Detection and Penalty

We penalize for len-ratio errors and off-target errors. We consider translations whose length ratios do not belong to $[L, U]$ as len-ratio errors. We computed the distribution of length ratios of the SFT data for each language pair (using the tokenizer of the corresponding model), taking $[L, U]$ so that it covers the top 50% of the most frequent length ratios. We adopt language detector from Stahl (2023) to detect off-target errors.

B MRT Implementation

MRT (Shen et al., 2016) is designed to directly optimize the model with respect to evaluation metrics. The *risk* of MRT can be formulate as:

$$\mathcal{R}(\theta) = \sum_{(x,y^*)\in\mathcal{D}} \mathbb{E}_{y\sim P(y|x;\theta)} \left[\Delta\left(y, y^*\right)\right], \tag{3}$$

where y^* represents the ground-truth translation. The Δ function serves as the loss function and can be instantiated using various evaluation metrics, for example, $1 - \mathrm{BLEU}(y, y^*)$. The objective or MRT is to minimize $\mathcal{R}(\theta)$.

In out setting, $\mathcal{D}$ is the training distribution of x, i.e., monolingual data, and we use QE-based reward $r(x, y)$ to indicate the quality of y as the translation of x. Therefore, we modify the *risk* as:

$$\mathcal{R}(\theta) = \sum_{x\in\mathcal{D}} \mathbb{E}_{y\sim P(y|x;\theta)} \left[1 - r(x, y)\right]. \tag{4}$$

In practice, we sample k candidates and normalize the probabilities to approximate the expectation. Let S_k denote a set containing k candidates drawn from the distribution $P_T(y|x;\theta)$, where T is the sampling temperature. $R(\theta)$ is approximated as:

$$\tilde{R}(\theta) = \sum_{x\in D}\sum_{y\in S_k} \tilde{P}(y|x;\theta)[1 - r(x, y)], \tag{5}$$

where

$$\tilde{P}(y|x;\theta) = \frac{P(y|x;\theta)^\alpha}{\sum_{y'\in S_k} P(y'|x;\theta)^\alpha}, \tag{6}$$

and α is a hyperparameter. Algorithm 2 shows the details of MRT. Similar to RAFT+, when adding a penalty term to reward, we call it MRT+. We set the penalty term $P = 1$ and $\alpha = 0.005$[2], and leave other hyperparameters the same as RAFT.

Algorithm 2 MRT

```
Require: Training set $\mathcal{X}$, reward function $r(x, y)$,
    initial model $M_0 = P(y|x;\theta_0)$, batch size $b$,
    temperature $T$, the number of candidate $k$
for iteration $i$ in $0, 1, \ldots, N-1$ do
    $D_i \leftarrow$ SampleBatch$(\mathcal{X}, b)$
    $\mathcal{B} = \emptyset$
    for $x \in D_i$ do
        $y_1, \ldots, y_k \sim P_T(y|x;\theta_i)$
        $S_k = \{y_1, \ldots, y_k\}$
        $\mathcal{B} = \mathcal{B} \cup \{(x, S_k)\}$
    Fine-tune $\theta_i$ on $\mathcal{B}$ to obtain $M_{i+1} =$
$P(y|x;\theta_{i+1})$ using MRT risk: $\tilde{R}(\theta_i)$.
```

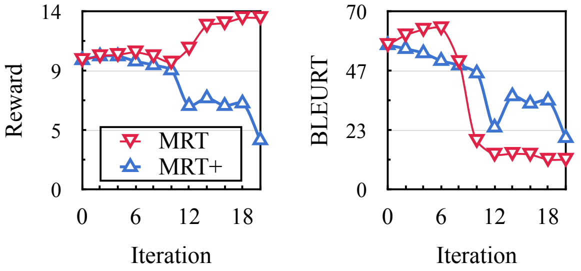


Figure 8: Training curves of MRT and MRT+ under high-resource setting. NLLB-200-1.3B and COMET-QE-MQM are used as base model and QE-based reward model, respectively.

Figure 8 shows that vanilla MRT suffers from the overoptimization problem, manifested as an increase in reward while translation quality declines. Additionally, MRT+ also poses challenges in achieving stable convergence.

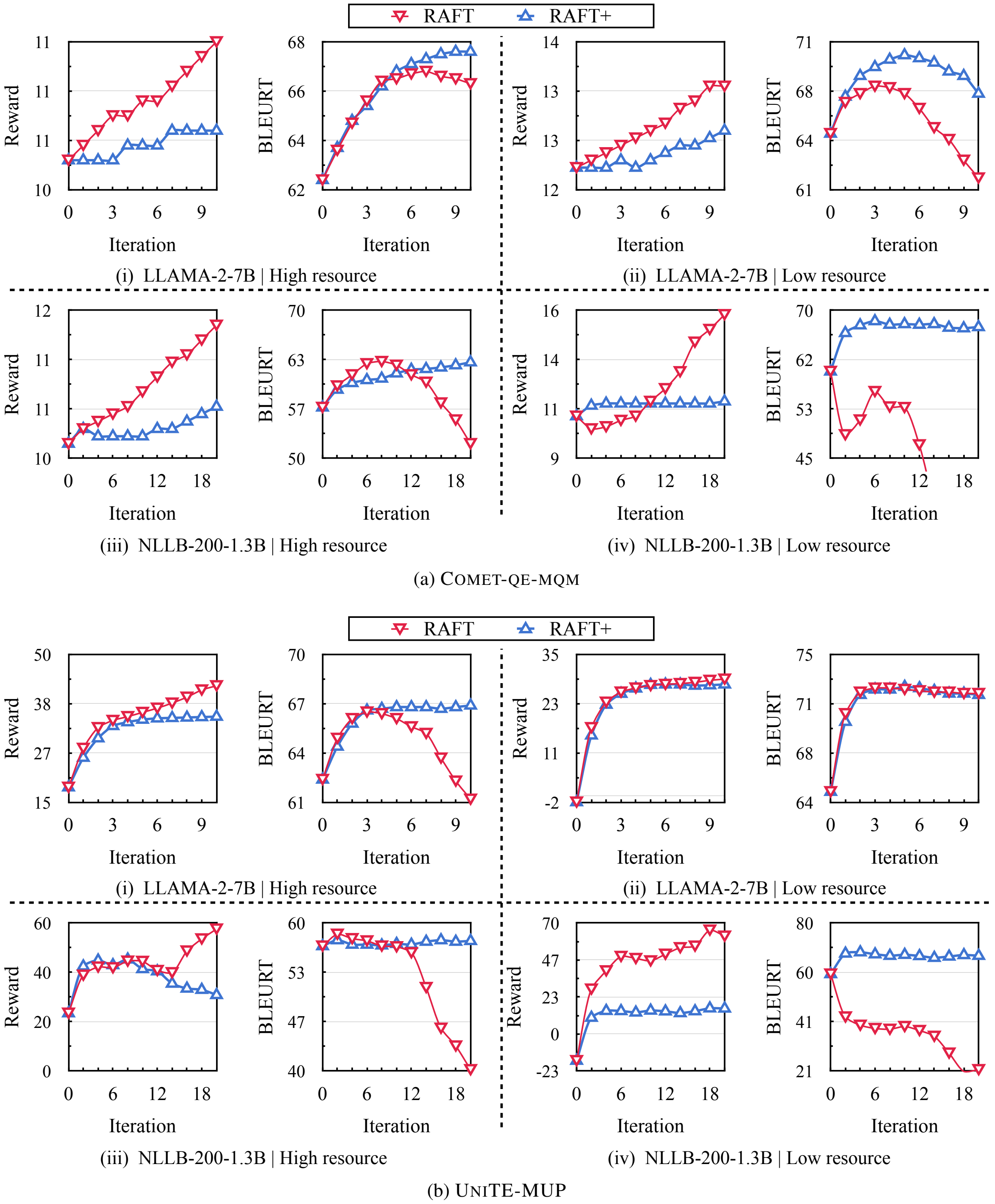
C More Training Curves

Figure 9 shows the training curves when using COMET-QE-MQM and UNITE-MUP as the reward models. Consistent with Figure 3, vanilla RAFT suffers from severe overoptimization problems in most cases, which are greatly alleviated by RAFT+.

D UNITE-MUP as the Reward Model

Table 4 presents the test results when UNITE-MUP functions as the reward model. The main trends are consistent with those shown in Table 3. We also observe that neither RAFT nor RAFT+ achieves positive improvement in high-resource language pairs when NLLB-200-1.3B is the base model. We speculate that this lack of improvement is due to overoptimization problems caused by the

[2] We follow the default setting in Sennrich et al. (2017).



(i) LLAMA-2-7B | High resource

(ii) LLAMA-2-7B | Low resource

(iii) NLLB-200-1.3B | High resource

(iv) NLLB-200-1.3B | Low resource

(a) COMET-QE-MQM

(i) LLAMA-2-7B | High resource

(ii) LLAMA-2-7B | Low resource

(iii) NLLB-200-1.3B | High resource

(iv) NLLB-200-1.3B | Low resource

(b) UNITE-MUP

Figure 9: Training curves under various settings when using COMET-QE-MQM and UNITE-MUP as reward models. The metrics are average values for all language pairs on the development set.

presence of errors that are not related to length ratio and off-target errors.

E chrF as the Evaluation Metric

Table 5 shows the chrF values for main results.

Method	De⇒En		En⇒De		Zh⇒En		En⇒Zh		**Average**	
	COMET	BLEURT	COMET	BLEURT	COMET	BLEURT	COMET	BLEURT	**COMET**	**BLEURT**
					LLAMA-2-7B					
SFT	82.5	70.5	80.7	68.2	76.1	62.3	84.9	69.3	81.0	67.6
RAFT	83.3	71.6	82.8	71.4	78.3	64.8	85.4	69.6	$82.4_{\uparrow 1.4}$	$69.4_{\uparrow 1.8}$
RAFT+	83.4	71.6	84.2	73.6	78.9	66.1	85.0	69.0	$\mathbf{82.9}_{\uparrow \mathbf{1.9}}$	$\mathbf{70.1}_{\uparrow \mathbf{2.5}}$
					NLLB-200-1.3B					
SFT	70.9	52.5	85.3	74.8	66.0	48.4	83.7	69.1	76.5	**61.2**
RAFT	72.7	50.9	85.8	75.4	66.5	48.6	84.5	69.1	$\mathbf{77.4}_{\uparrow \mathbf{0.9}}$	$61.0_{\downarrow 0.2}$
RAFT+	72.8	50.3	85.8	75.5	65.5	47.1	84.4	69.0	$77.1_{\uparrow 0.6}$	$60.5_{\downarrow 0.7}$

(a) High-resource language pairs

Method	En⇒Uk		Uk⇒En		Uk⇒Cs		Cs⇒Uk		**Average**	
	COMET	BLEURT	COMET	BLEURT	COMET	BLEURT	COMET	BLEURT	**COMET**	**BLEURT**
					LLAMA-2-7B					
SFT	79.2	64.0	76.7	66.0	70.0	53.2	71.2	51.3	74.3	58.6
RAFT	80.9	66.4	80.7	70.2	81.6	68.9	83.6	69.1	$81.7_{\uparrow 7.4}$	$68.6_{\uparrow 10.0}$
RAFT+	81.3	67.1	81.0	70.5	81.5	68.7	84.0	69.6	$\mathbf{81.9}_{\uparrow \mathbf{7.6}}$	$\mathbf{69.0}_{\uparrow \mathbf{10.4}}$
					NLLB-200-1.3B					
SFT	83.1	70.2	71.1	62.7	73.2	61.5	57.3	43.4	71.2	59.4
RAFT	85.1	72.4	64.5	30.9	70.5	26.9	74.1	27.4	$73.5_{\uparrow 2.3}$	$39.4_{\downarrow 19.2}$
RAFT+	84.3	71.4	77.0	66.5	82.6	70.2	71.6	54.9	$\mathbf{78.9}_{\uparrow \mathbf{7.7}}$	$\mathbf{65.8}_{\uparrow \mathbf{6.4}}$

(b) Low-resource language pairs

Table 4: Test results under various settings when UNITE-MUP functions as the reward model. Bold indicates that the average performance of the method exceeds that of SFT and RAFT/RAFT+ within the same QE model. The subscripts indicate the change in performance relative to the SFT.

Method	De⇒En	En⇒De	Zh⇒En	En⇒Zh	**Average**
		LLAMA-2-7B			
SFT	52.1	51.5	46.3	34.4	46.0
REWARD MODEL: COMET-QE-DA					
RAFT	55.4	56.1	50.2	35.5	$49.3_{\uparrow 3.3}$
RAFT+	56.1	58.8	51.7	34.5	$50.3_{\uparrow \mathbf{4.3}}$
REWARD MODEL: COMET-QE-MQM					
RAFT	53.4	56.8	47.2	34.9	$48.1_{\uparrow 2.1}$
RAFT+	54.5	58.5	49.2	34.8	$49.3_{\uparrow \mathbf{3.3}}$
		NLLB-200-1.3B			
SFT	35.3	60.4	15.2	30.7	35.4
REWARD MODEL: COMET-QE-DA					
RAFT	35.1	60.8	22.8	30.9	$37.4_{\uparrow 2.0}$
RAFT+	42.6	60.6	27.8	30.7	$40.4_{\uparrow \mathbf{5.0}}$
REWARD MODEL: COMET-QE-MQM					
RAFT	48.3	60.7	35.5	30.7	$43.8_{\uparrow 8.4}$
RAFT+	49.7	60.6	43.2	30.6	$46.1_{\uparrow \mathbf{10.7}}$

(a) High-resource language pairs

Method	En⇒Uk	Uk⇒En	Uk⇒Cs	Cs⇒Uk	**Average**
		LLAMA-2-7B			
SFT	43.1	51.1	28.3	30.2	38.2
REWARD MODEL: COMET-QE-DA					
RAFT	46.2	56.4	43.9	47.5	$48.5_{\uparrow 10.3}$
RAFT+	46.5	56.9	44.5	48.0	$49.0_{\uparrow \mathbf{10.8}}$
REWARD MODEL: COMET-QE-MQM					
RAFT	40.6	48.1	30.0	30.3	$37.3_{\downarrow 0.9}$
RAFT+	44.3	53.5	36.1	38.2	$43.0_{\uparrow \mathbf{4.8}}$
		NLLB-200-1.3B			
SFT	50.6	45.3	36.6	25.7	39.5
REWARD MODEL: COMET-QE-DA					
RAFT	52.1	10.2	10.2	10.3	$20.7_{\downarrow 18.8}$
RAFT+	51.2	55.1	49.7	43.3	$49.8_{\uparrow \mathbf{10.3}}$
REWARD MODEL: COMET-QE-MQM					
RAFT	51.7	31.5	22.5	26.0	$33.0_{\downarrow 6.5}$
RAFT+	51.5	54.1	48.0	41.5	$48.8_{\uparrow \mathbf{9.3}}$

(b) Low-resource language pairs

Table 5: chrF results of Table 3

Method	De⇒En		En⇒De		Zh⇒En		En⇒Zh		Average	
	COMET	BLEURT	COMET	BLEURT	COMET	BLEURT	COMET	BLEURT	COMET	BLEURT
WMT22 Best	85.0	73.8	87.4	77.9	81.0	68.9	86.8	72.8	85.1	73.4

(a) High-resource language pairs

Method	En⇒Uk		Uk⇒En		Uk⇒Cs		Cs⇒Uk		Average	
	COMET	BLEURT	COMET	BLEURT	COMET	BLEURT	COMET	BLEURT	COMET	BLEURT
WMT22 Best	87.8	76.5	85.9	76.6	92.2	82.8	91.6	80.3	89.4	79.1

(b) Low-resource language pairs

Table 6: WMT22 best submissions.