Quality-Aware Decoding: Unifying Quality Estimation and Decoding

Anonymous ACL submission

Abstract

Quality Estimation (QE) models for Neural Machine Translation (NMT) predict the quality of the hypothesis without having access to the reference. An emerging research direction in NMT involves the use of QE models, which have demonstrated high correlations with human judgment and can enhance translations through Quality-Aware Decoding. Although several approaches have been proposed based on sampling multiple candidate translations and picking the best candidate, none have integrated these models directly into the decoding process. In this paper, we address this by proposing a novel token-level QE model capable of reliably scoring partial translations. We build a uni-directional QE model for this, as decoder models are inherently trained and efficient on partial sequences. We then present a decoding strategy that integrates the QE model for Quality-Aware decoding and demonstrate that the translation quality improves when compared to the N-best list re-ranking with state-of-the-art QE models (up to 1.39 XCOMET-XXL ↑). Finally, we show that our approach provides significant benefits in document translation tasks, where the quality of N-best lists is typically suboptimal[1].

1 Introduction

Large language models (LLMs) have significantly impacted various Natural Language Processing (NLP) tasks (Brown et al., 2020; Jiang et al., 2023; Dubey et al., 2024), including Neural Machine Translation (NMT). The field of NMT is transitioning from using dedicated encoder-decoder transformers (Vaswani, 2017; Team et al., 2024) to leveraging decoder-only LLM-based translation models (Kocmi et al., 2024). This shift is driven by LLMs' ability to retain knowledge, handle large contexts, and follow instructions, learned during extensive pre-training (Xu et al., 2024; Alves et al., 2024). As a result, LLM-based MT models have achieved state-of-the-art translation quality (Kocmi et al., 2024).

In parallel, Quality Estimation (QE) has become a well-researched subfield within NMT. QE models are trained to predict the quality of a translation without requiring access to the reference (Rei et al., 2021, 2022). Interestingly, QE models can achieve performance in assessing translation quality that is comparable to MT evaluation models, which do have access to the reference (Zerva et al., 2024).

This led to the question: "*Can we integrate QE into the current translation process to improve quality?*" Incorporating QE into NMT offers several benefits. First, having a expert QE model guiding the decoding can further improve the quality. Second, by adapting the QE model with feedback from human annotators, we can generate future translations guided with the newly obtained feedback.

Several approaches have been explored to integrate QE into the translation process. These include re-ranking the N-best list (Fernandes et al., 2022), applying minimum Bayes risk (MBR) decoding on a quality-filtered N-best list (Tomani et al., 2024), and training additional models for post-editing based on QE-predicted errors (Treviso et al., 2024). However, all these methods operate on fully generated sequences before the QE model can exert influence. Integrating QE earlier in the decoding process, referred in this paper as *Quality-Aware Decoding*, could enhance translation quality and reduce reliance on the N-best list. This is especially relevant when dealing with long inputs as good translations during decoding are likely to be

[1]We will release the code under Apache License 2.0

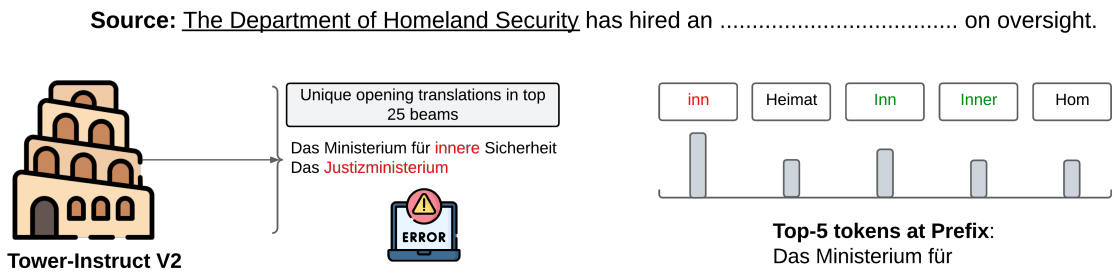


Figure 1: Example from WMT'23 English → German #ID: 10: The paragraph begins with 'Department of Homeland Security,' which should be translated as 'Ministerium für **I**nnere Sicherheit.' However, the top 25 beams do not contain the correct translation and begin with an error, making N-best list re-ranking insufficient. Although the top-5 tokens at the decoding contain the correct forms 'Inn' or 'Inner,' the probabilities split among them giving highest mass to the incorrect token 'inn.' Quality-Aware decoding can prevent errors with earlier integration.

pruned and may need sampling larger number of finished hypothesis. We illustrate this in Figure 1.

To achieve this, a QE model capable of predicting the quality of partial translations is required. However, current leading QE models face challenges in this area, as they are typically not trained to predict scores for incomplete hypotheses. *Therefore, developing QE models that can handle partial translations is essential for implementing Quality-Aware Decoding during the translation process.*

In this work, we propose adapting LLM-based MT models to perform QE on partial translations and incorporating this model into the decoding. We create a token-level synthetic QE dataset using WMT Multidimensional Quality Metrics (MQM) data (Burchardt, 2013; Freitag et al., 2024). We then adapt a uni-directional LLM-based MT model to predict whether a token is *Good* or *Bad*. Training QE models on these token-level tasks alleviates the data challenge and allows us to exploit the MQM data while simultaneously making the task easier for the model compared to predicting a score directly.

Furthermore, integrating the QE model into NMT during decoding is not trivial, as we need to combine the QE estimates during decoding. Therefore, we use the decoding strategy from Koneru et al. (2024), and modify it to incorporate token-level predictions efficiently with the adapted QE model to provide real-time feedback during the decoding process.

We summarize our main findings and contributions below.

- We present a novel uni-directional QE model which estimates quality on incomplete hypotheses by averaging the probabilities of each token being classified as *Good*.
- We propose a decoding strategy that combines the token-level QE model on partial hypothesis and the NMT model to perform Quality-Aware Decoding.
- We show through experiments that early integration is essential and the translation quality is improved even when compared to re-ranking the N-best list with state-of-the-art QE models.
- We highlight the significance of our approach in document translation scenarios, where post-generation QE techniques fall short due to their reliance on the quality of the N-best list, a challenge that becomes more difficult as the input length increases.

2 Quality-Aware Decoding

The primary objective of this paper is to achieve Quality-Aware Decoding in MT. To accomplish this, it is essential to predict the quality of partial translations and integrate this information during the decoding process. Our approach proposes using one NMT model for generating translations and another adapted NMT model to predict the quality of the candidate translations produced by the first model.

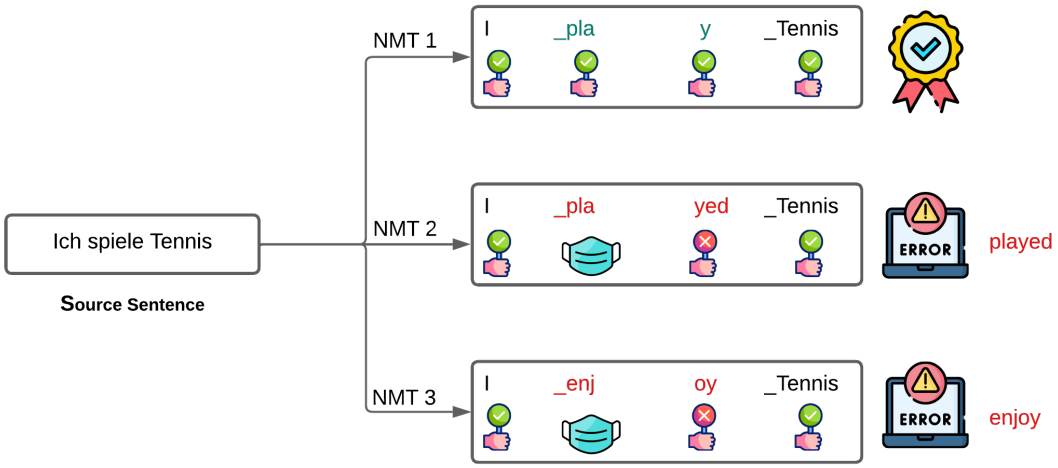


Figure 2: Token-level label annotation scheme using the MQM error tags. *MASK* indicates that this token label will not be used in training to prevent incorrect learning signal.

First, we explain why relying solely on the NMT model to predict the quality of a hypothesis is insufficient and why an additional model is necessary. Next, we outline the adaptation of the NMT model for QE on partial translations, detailing the creation of a token-level QE dataset, the modifications made to the NMT model for this task, and the process of estimating the sentence-level quality score. Finally, we describe the algorithm used to incorporate the QE score into the decoding process.

2.1 Decomposing Decoding: Translation + QE

NMT models generate a token-by-token sequence and provide the probability of each token at the decoding step. The average of the log-probabilities is often used as a proxy to score the candidate during search.

While NMT models are capable of generating high-quality translations, using the average log-probabilities of hypotheses as a scoring metric tends to yield poor correlation with actual translation quality (Eikema and Aziz, 2020; Freitag et al., 2020). In many cases, a translation can continue in several different ways, all of which may be acceptable. If the starting tokens for these continuations differ, the probability mass may be spread across multiple options which is used during the search. However, from a quality perspective, all these continuations could still achieve a high score, as the QE scores are independent and need not sum to 1.

Therefore, we propose a expert model that focuses on quality to estimate the scores better during decoding and improve the search space leading to a better hypothesis.

2.2 Quality Estimation on Partial Sequences

To provide a quality score during decoding, the QE model must be capable of handling incomplete sequences. It should not penalize a sequence if there is a potential extension that could lead to a perfect translation.

Estimating the score in this way is not feasible with current QE models, such as COMET (Rei et al., 2021), as they were not trained for this specific task and cannot provide reliable scores in the context of partial translations. Hence, we need to develop a partial QE system.

When building a partial QE system, several factors need to be considered. First, should the model use a uni-directional or bi-directional architecture? A **uni-directional** model is more efficient, as it allows for caching the hidden states, which can then be used for subsequent steps without re-encoding, unlike a bi-directional model.

Next, we need to decide whether to predict the QE score at the sequence level or at the token level. For **token-level QE**, we can directly use data from MQM annotations, as we already know which tokens are GOOD or BAD. However, for segment-level scoring, we need to consider how to

synthetically create the training data.

Therefore, we decide adapt the uni-directional model into a token-level QE system that predicts whether each token is *Good* or *Bad* (a binary decision) by adding an additional classifier head. This adaptation enables us to estimate the score for a sequence by calculating the average probability that each token is classified as *Good*. We hypothesize that adapting the model in this way, rather than directly predicting the score, provides greater stability, as the last hidden states inherently contain token-level information and do not require mapping the entire sequence to a single score.

For training this model, we leverage the WMT MQM data containing error annotations in MT outputs. We can treat tokens before an error as *Good* and those containing inside an error as *Bad*. Then, we can train in uni-directional manner where each token's label is predicted using only the preceding context in the hypothesis. This is crucial as we only have the preceding context to estimate the quality for partial hypothesis.

2.2.1 Learning the Right Signal

The straightforward approach to creating labels is to assign 1 to all tokens within the error span and 0 otherwise. However, MQM annotations can mark errors from words to phrases, and the starting tokens of an error span may not always be wrong. This is illustrated in Figure 2.

For example, consider the German sentence *"Ich spiele Tennis"* translated by three different NMT systems, each annotated with MQM error labels. In this work, we focus on learning a binary decision: whether an error is present, ignoring error severity.

System 1: No error: The translation *"I play Tennis"* is perfect, and all tokens are labeled as "*Good*."

System 2: Partial error: The translation *"I played Tennis"* has an error in the verb form ("played" instead of "play"). The error is in the token span *"played"*, but not all tokens in this span are incorrect (e.g., "pla" is correct). Assigning a "*Bad*" label to the entire span would lead to incorrect learning. A more refined approach is needed to mark errors accurately at the token level.

System 3: Full error: The translation *"I enjoy Tennis"* contains an error in *"enjoy"*, so all tokens in this span should be labeled as "*Bad*."

It is not trivial to decide when the prefix of an error span is correct/incorrect. To achieve accurate labeling, we propose the following scheme:

- Apply a <MASK> operation to all tokens within the error span.
- Only the last token in the span is assigned the label "*Bad*", as the error is considered complete at the end of the span.

If the error token is in the middle, we still train the model to predict "*Bad*" in the end and let the model determine which tokens should be part of the error span during inference. This approach ensures that errors are identified without explicitly defining the error span.

2.2.2 Sequence-Level Quality Estimation

After fine-tuning a token-level classification model to predict the quality of the tokens, we still need to map these predictions into a sequence-level score that can be integrated during the decoding process. There are several potential ways to achieve this.

One approach is to simply count how many tokens are classified as *Bad* in the current hypothesis. However, this method has limitations. The number of errors should be normalized based on the length of the hypothesis to account for varying sizes. Additionally, converting the probabilities into a fixed number of error tokens would need to account for different error types according to the MQM format, as each error counts differently.

To avoid such strict scoring schemes, we take a simpler approach. We average the log probabilities of all tokens that are classified as *Good*. This method inherently accounts for the length of the hypothesis, and it provides a score on the scale of log probabilities, which aligns with the decoding process. Therefore, we use this averaged log probability as a proxy metric for the QE score, where a higher score indicates better quality (**Line 5** in Algorithm 1).

2.2.3 Fusing Translation and Quality

We can use a token-level QE system to evaluate the quality of a source and partial hypothesis during decoding. However, integrating these probabilities into all candidates is computationally expensive, as each beam considers extensions equal to the vocabulary size.

Algorithm 1 Computing merged score of partial hypothesis with translation and token-level QE models.

1: **procedure** MERGESCORE
2: **Input:** Hypothesis tokens $h_1, h_2, h_3, \ldots, h_n$, Translation Model $\mathcal{M}_{NMT}$, QE model $\mathcal{M}_{QE}$, Source sentence $\mathcal{S}$, Re-ranking weight α,
3: **Output:** $merged_score$
4: $Score_{NMT} \leftarrow \frac{1}{n} \sum \log \mathcal{P}(h_1, h_2, \ldots, h_n | \mathcal{S}; \mathcal{M}_{NMT})$
5: $Score_{QE} \leftarrow \frac{1}{n} \sum \log \mathcal{P}(0_1, 0_2, \ldots, 0_n | h_1, h_2, \ldots, h_n, \mathcal{S}; \mathcal{M}_{QE})$
6: $merged_score \leftarrow (\alpha) \times Score_{NMT} + (1 - \alpha) \times Score_{QE}$
7: **end procedure**

To address this, we adopt a simplified decoding strategy from Koneru et al. (2024), which ensembles models with different vocabularies. By adapting the same MT model for token-level QE, we simplify the merging process, as the vocabularies match. This restriction is reasonable, as it is also beneficial to leverage the knowledge learned by the specialized MT for token-level QE.

The core idea is to re-rank the top candidates at each decoding step using the QE model. After re-ranking, the translation and QE scores are merged, and the process repeats until the end-of-sentence token is generated, for each beam. This strategy allows us to efficiently incorporate the QE model's estimate, improving translation quality.

During decoding, at each step, we have scores for n beams and V possible extensions from the vocabulary. In typical beam search, we select the top n extensions and expand the hypothesis. To make the decoding process Quality-aware, we estimate the quality of these extensions. Since estimating all extensions is computationally expensive, we limit the candidates by selecting a specified number of top candidates.

To achieve this, we use a hyper-parameter $topk$, which selects the best $topk$ extensions for each beam. For each of these top $topk$ extensions, we compute a combined score, detailed in Algorithm 1. This combined score incorporates both the translation model score and the quality estimation score, ensuring the quality is considered during decoding.

For a top extension at decoding step n, let the current tokens be $h_1, h_2, h_3, \ldots, h_n$. The NMT model score is computed as the average log probabilities of each token (Line 4). For the token-level QE model, we compute the average probability of each token being classified as '*Good*' (Line 5). The merged score is equal to weighted linear combination of these probabilities, with weight α (Line 6).

Thus, to make the decoding process Quality-Aware, we first train a token-level QE system by adapting the same NMT model to ensure vocabulary matching. We then combine the scores from both models to improve the sequence estimates explored during search.

3 Experimental Setup

Datasets: We focus on two language directions given their availability of MQM data: English → German and Chinese → English. To train our token-level QE systems, we use the MQM datasets[2] from WMT (Freitag et al., 2021). Specifically, we use the datasets until 2022 for training, 2024 for validation, and 2023 for testing (Kocmi et al., 2024). This setup is consistent with all the other QE metrics, and we do not use any additional data beyond these datasets.

Models: Our proposed approach achieves Quality-Aware decoding by combining an NMT model with a token-level QE model, where we adapt the same NMT for QE by adding a classification head. We use the state-of-the-art NMT model, Tower 7B[3] (Alves et al., 2024), which provides high-quality translations and has already been exposed to MQM data during instruction-tuning. This ensures that the gains observed in our approach stem from integrating Quality-Aware decoding into the NMT process, rather than introducing new data. Additional details on training and hyper-parameters are provided in Appendix A.1.

[2]https://github.com/google/wmt-mqm-human-evaluation
[3]Unbabel/TowerInstruct-7B-v0.2

Model	Beams	Re-ranking	MetricX (↓)	XCOMET-XXL (↑)
English → German				
Tower	5	_	2.52	86.93
Tower	25	XCOMET-XL QE	2.37	87.79
Tower	25	Tower QE	2.38	87.40
Tower + Tower QE	5 (25* for Tower QE)	_	2.12	88.95
Tower + Tower QE	5 (25* for Tower QE)	XCOMET-XL QE	**2.09**	**89.08**
Chinese → English				
Tower	5	_	2.42	88.91
Tower	25	XCOMET-XL QE	2.30	89.49
Tower	25	Tower QE	2.32	89.51
Tower + Tower QE	5 (25* for Tower QE)	_	2.26	89.82
Tower + Tower QE	5 (25* for Tower QE)	XCOMET-XL QE	**2.24**	**90.00**

Table 1: Translation Quality on WMT23 English → German Test set. Both XCOMET and MetricX columns use reference for reporting translation quality where as XCOMET-XL QE does not use for re-ranking.

	Pearson	Spearmann	Kendall
COMETQE	**44.41**	41.29	31.19
COMETQE-XL	41.23	**42.17**	**31.84**
Tower Avg. Log Prob	32.32	16.74	12.77
Tower QE	40.56	33.96	25.87

Table 2: Correlation on WMT 23 for English → German Test set. The scores are calculated after removing the few sentences labeled for hallucination detection. Best scores according to each coefficient are highlighted in **bold**.

Metrics: For reporting the translation quality, we consistently use XCOMET-XXL[4] (Guerreiro et al., 2024) and MetricX[5] (Juraska et al., 2024) **with the reference**. To compare with N-best list re-ranking, we use the XCOMET-XL QE[6] **without the reference**. This approach allows us to avoid biasing toward a single metric during the re-ranking process and enables us to measure the gains achieved by differently trained metrics.

4 Results

We conduct a series of experiments to validate the effectiveness of Quality-Aware decoding and identify the scenarios where it provides the most benefit. First, we evaluate whether our token-level QE model can better estimate sequence quality compared to the log probabilities of the NMT model. Next, we assess the impact of Quality-Aware decoding by comparing it with other approaches to determine if it improves translation quality. We also perform an ablation study to examine whether training the QE model on errors from the same NMT model enhances its performance. Finally, we explore the impact of source sentence length to highlight the limitations of N-best list re-ranking.

4.1 Quality Estimation Performance

First, we evaluate the agreement between the Tower-based token-level QE model (**Tower QE**) and human scores for a given hypothesis. It is only beneficial if we achieve higher correlation than the average of the NMT model log probabilities to show the need to integrate it during decoding. Therefore, we report the correlation with human scores of different models on WMT 23 English → German in Table 2.

We observe that the best-performing systems are the Comet QE models, which predict a single score using the full hypothesis. This is expected, as these models assess quality after the hypothesis is fully generated. In contrast, both log probabilities and Tower QE scores are based on the predicted token of each decoding step, using only the preceding context. Log probabilities perform poorly in this setup, while our proposed model, Tower QE, achieves twice the correlation with human judg-

[4]Unbabel/XCOMET-XXL
[5]google/metricx-24-hybrid-xl-v2p6
[6]Unbabel/XCOMET-XL

Model	Beams	Re-ranking	MetricX (↓)	XCOMET-XXL (↑)
English → German				
Tower	25	XCOMET-XL QE	2.37	87.79
Tower	25	Tower QE	2.38	87.40
Tower	25	Tower Distill QE	2.38	87.39
Tower + Tower QE	5 (25* for Tower QE)	_	2.12	**88.95**
Tower + Tower QE	5 (25* for Tower Distill QE)	_	**2.11**	88.76

Table 3: Performance of Unidirectional QE trained with/without distillation on WMT23 English → German Test set. Best scores according to each metric are highlighted in **bold**.

Model	Beams	Re-ranking	XCOMET-XL (↑)	XCOMET-XXL (↑)	Impact
Paragraph-Level					
Tower	25	XCOMET-XL QE	**86.56**	87.79	δ = + 1.16 (88.95 - 87.79)
Tower	25	Tower QE	85.40	87.40	
Tower + Tower QE	5 (25* for Tower QE)	_	86.36	**88.95**	
Sentence-Level					
Tower	25	XCOMET-XL QE	**86.42**	87.68	δ = + 0.38 (88.06 - 87.68)
Tower	25	Tower QE	85.23	87.41	
Tower + Tower QE	5 (25* for Tower QE)	_	85.96	**88.06**	

Table 4: Impact of integrating Unidirectional QE during decoding with paragraphs vs sentences on WMT23 English → German Test set. δ denotes the improvement in translation quality from re-ranking N-best list with XCOMET-XL QE to integrating unidirectional Tower QE during the decoding. Best scores according to each metric are highlighted in **bold**.

ments compared to log probabilities, despite scoring token by token with preceding context. This result highlights the potential of integrating our approach into the decoding process.

4.2 Unified Decoding for NMT

To validate the effectiveness of our unified decoding approach, we compare it with several baselines in Table 1. First, we evaluate whether our approach outperforms generating translations with the NMT model alone. Next, we check if the quality of translations improves compared to N-best list re-ranking. To make the setups comparable, we set $topk$ and num_beams to 5 and compare with re-ranking the top 25 beams using XCOMET-XL. Finally, to demonstrate that re-ranking the N-best list remains a viable and complementary approach, we re-rank the top 5 beams obtained from Quality-Aware decoding using the same QE model.

We find that re-ranking with XCOMET-XL and Tower QE yields similar results, indicating that our partial QE model does not over-fit to any specific metric. Furthermore, we observe that the unified decoding approach outperforms N-best list re-ranking across both metrics in both language pairs. For example, the MetricX score improves from 2.37 to 2.12 for English → German. Note that Tower has already seen this data during instruction-tuning and the improvement is not from new data but from Quality-Aware decoding. Moreover, re-ranking the top 5 beams obtained from unified decoding with XCOMET-XL leads to a slight further improvement in quality. This highlights the robustness and generalizability of our approach across different evaluation metrics.

4.3 Adapting for Tower Errors

We use the MQM annotations from WMT to train our Tower QE model, which contains error annotations from other systems. However, a viable alternative would be to adapt Tower QE specifically to the errors it typically makes. To maintain a similar data setup, we first generate translations using Tower on these source sentences. Then, we annotate the generated hypotheses with XCOMET-XL using the reference and fine-tune Tower QE on this synthetic dataset, which we refer to as **Tower Distill QE**. We evaluate the performance of the new

distill QE model and report the results in Table 3.

We observe that the distilled QE model performs very similarly to the QE model trained on errors from other systems. This indicates that there was no significant benefit in adapting the QE model to the specific errors typically made by Tower. However, further analysis on larger datasets and different domains is needed to fully validate the effectiveness of the distillation approach as the current synthetic data generated is small.

4.4 Sentence vs Document-level Translation

From Table 1, we observe that the gains for English → German (paragraph-level) are much higher than for Chinese → English (sentence-level). We hypothesize that this discrepancy arises from the length of the sentences, as the N-best list re-ranking is likely sufficient for shorter sentences. To confirm this, we take the English paragraphs and split them into sentences using a tokenizer while tracking the paragraph IDs. We then perform the entire decoding process similarly, and later join the sentences back using the paragraph IDs before evaluation. We report the results in Table 4.

We define the impact as the improvement in translation quality from re-ranking the N-best list with XCOMET-XL QE to integrating Tower QE. Comparing the results at the paragraph level to those at the sentence level, we observe that the impact decreases, which confirms our hypothesis. Additionally, we obtain better scores at the document level, further highlighting the potential benefits of Quality-Aware Decoding.

5 Related Work

Integrating QE in NMT: Several advancements have been made in improving QE for NMT over the years (Rei et al., 2021, 2022; Blain et al., 2023; Zerva et al., 2024; Guerreiro et al., 2024). These developments have led to the integration of QE in various ways. One common approach involves applying QE after generating multiple sequences through techniques such as QE re-ranking (Fernandes et al., 2022; Faria et al., 2024) or Minimum Bayes Risk (MBR) decoding (Tomani et al., 2024). Another direction focuses on removing noisy data using QE models, followed by fine-tuning on high-quality data (Xu et al., 2024; Finkelstein et al., 2024). Vernikos and Popescu-Belis (2024) proposes to generate diverse translations as a first step and then combine them. We perform this explicitly by integrating the QE directly into decoding. Recently, Zhang et al. (2024) exploited the MQM data by training models to penalize tokens within an error span, improving translation quality. In contrast, our approach adopts a modular framework, where we propose an expert QE model that is trained independently for targeted training. This modular approach aims to improve performance by decomposing the task into separate translation and QE components.

Reward Modeling in NLG: Quality-Aware decoding shares several similarities with controllable text generation methods, particularly in the use of an additional "Quality/Reward" model that guides the decoding. A well-explored approach for controlling text is altering the decoding with a reward model (Weighted Decoding) (Yang and Klein, 2021). This method modifies the decoding by adjusting token probabilities based on the reward model, allowing for more controlled generation. Similarly, Deng and Raffel (2023) also used a uni-directional reward model, with the aim of maintaining efficiency during generation. This approach minimizes computational complexity while still benefiting from the guiding influence of the reward model. Moreover, recent work by Li et al. (2024) introduced a token-level reinforcement learning-based reward model, providing more fine-grained feedback that enhances control over text generation at a granular level. While similar, the key contribution in our work lies in the development of the first uni-directional QE model for translation.

6 Conclusion

We have shown the importance of Quality-Aware decoding to improve translation quality, rather than relying on post-generation techniques. In this work, we demonstrated how MQM data can be used to build a uni-directional token-level QE model, which is then integrated into the decoding process. Through a series of experiments, we showed that our Quality-Aware decoding approach results in measurable improvements in translation quality. Notably, we did not introduce new training data to the NMT model, and show that the gains stem from Quality-Aware decoding.

7 Limitations

While our Quality-Aware decoding improves translation quality, it adds considerable computational complexity to the inference process. Theoretically, this approach would double the time needed to generate a translation and require additional memory to utilize the token-level QE model. One potential solution to mitigate this issue could be to use token-level QE as a reward model for training via Reinforcement Learning.

Additionally, we trained our model on a limited set of human-annotated MQM data. However, current QE models, such as XCOMET, are capable of predicting error tags using the reference with reasonable quality. This suggests that further improvements could be achieved if these models were trained on larger-scale datasets, providing more nuanced feedback and refining translation quality even further.

Lastly, our proposed token-level QE model does not account for error severity. Ideally, it should be able to predict the category of errors, allowing for more nuanced feedback and enabling the model to generate translations with only minor errors when necessary.

References

Duarte M Alves, José Pombal, Nuno M Guerreiro, Pedro H Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, et al. 2024. Tower: An open multilingual large language model for translation-related tasks. *arXiv preprint arXiv:2402.17733*.

Frederic Blain, Chrysoula Zerva, Ricardo Rei, Nuno M Guerreiro, Diptesh Kanojia, José GC de Souza, Beatriz Silva, Tânia Vaz, Yan Jingxuan, Fatemeh Azadi, et al. 2023. Findings of the wmt 2023 shared task on quality estimation. In *Proceedings of the Eighth Conference on Machine Translation*, pages 629–653.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Aljoscha Burchardt. 2013. Multidimensional quality metrics: a flexible system for assessing translation quality. In *Proceedings of Translating and the Computer 35*.

Haikang Deng and Colin Raffel. 2023. Reward-augmented decoding: Efficient controlled text generation with a unidirectional reward model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11781–11791, Singapore. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Bryan Eikema and Wilker Aziz. 2020. Is map decoding all you need? the inadequacy of the mode in neural machine translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4506–4520.

Gonçalo RA Faria, Sweta Agrawal, António Farinhas, Ricardo Rei, José GC de Souza, and André FT Martins. 2024. Quest: Quality-aware metropolis-hastings sampling for machine translation. *arXiv preprint arXiv:2406.00049*.

Patrick Fernandes, António Farinhas, Ricardo Rei, José GC de Souza, Perez Ogayo, Graham Neubig, and André FT Martins. 2022. Quality-aware decoding for neural machine translation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1396–1412.

Mara Finkelstein, David Vilar, and Markus Freitag. 2024. Introducing the newspalm mbr and qe dataset: Llm-generated high-quality parallel data outperforms traditional web-crawled data. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1355–1372.

Markus Freitag, George Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021. Experts, errors, and context: A large-scale study of human evaluation for machine translation. *Preprint*, arXiv:2104.14478.

Markus Freitag, David Grangier, and Isaac Caswell. 2020. Bleu might be guilty but references are not innocent. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 61–71.

Markus Freitag, Nitika Mathur, Daniel Deutsch, Chi-Kiu Lo, Eleftherios Avramidis, Ricardo Rei, Brian Thompson, Frederic Blain, Tom Kocmi, Jiayi Wang, et al. 2024. Are llms breaking mt metrics? results of the wmt24 metrics shared task. In *Proceedings of the Ninth Conference on Machine Translation*, pages 47–81.

Nuno M Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, and André FT Martins.

2024. xcomet: Transparent machine translation evaluation through fine-grained error detection. *Transactions of the Association for Computational Linguistics*, 12:979–995.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2021. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Juraj Juraska, Daniel Deutsch, Mara Finkelstein, and Markus Freitag. 2024. Metricx-24: The google submission to the wmt 2024 metrics shared task. *arXiv preprint arXiv:2410.03983*.

Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, et al. 2024. Findings of the wmt24 general machine translation shared task: The llm era is here but mt is not solved yet. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1–46.

Sai Koneru, Matthias Huck, Miriam Exel, and Jan Niehues. 2024. Plug, play, and fuse: Zero-shot joint decoding via word-level re-ranking across diverse vocabularies. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1467–1481.

Wendi Li, Wei Wei, Kaihe Xu, Wenfeng Xie, Dangyang Chen, and Yu Cheng. 2024. Reinforcement learning with token-level feedback for controllable text generation. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1704–1719, Mexico City, Mexico. Association for Computational Linguistics.

Ricardo Rei, Ana C Farinha, Chrysoula Zerva, Daan van Stigt, Craig Stewart, Pedro Ramos, Taisiya Glushkova, André FT 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.

Ricardo Rei, Marcos Treviso, Nuno M Guerreiro, Chrysoula Zerva, Ana C Farinha, Christine Maroti, José GC de Souza, Taisiya Glushkova, Duarte Alves, Luísa Coheur, et al. 2022. Cometkiwi: Ist-unbabel 2022 submission for the quality estimation shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 634–645.

NLLB Team et al. 2024. Scaling neural machine translation to 200 languages. *Nature*, 630(8018):841.

Christian Tomani, David Vilar, Markus Freitag, Colin Cherry, Subhajit Naskar, Mara Finkelstein, Xavier Garcia, and Daniel Cremers. 2024. Quality-aware translation models: Efficient generation and quality estimation in a single model. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15660–15679.

Marcos Treviso, Nuno Guerreiro, Sweta Agrawal, Ricardo Rei, José Pombal, Tânia Vaz, Helena Wu, Beatriz Silva, Daan Stigt, and André FT Martins. 2024. xtower: A multilingual llm for explaining and correcting translation errors. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 15222–15239.

A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.

Giorgos Vernikos and Andrei Popescu-Belis. 2024. Don't rank, combine! combining machine translation hypotheses using quality estimation. *arXiv preprint arXiv:2401.06688*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024. Contrastive preference optimization: Pushing the boundaries of llm performance in machine translation. *arXiv preprint arXiv:2401.08417*.

Kevin Yang and Dan Klein. 2021. Fudge: Controlled text generation with future discriminators. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3511–3535.

Chrysoula Zerva, Frédéric Blain, José GC De Souza, Diptesh Kanojia, Sourabh Deoghare, Nuno M Guerreiro, Giuseppe Attanasio, Ricardo Rei, Constantin Orasan, Matteo Negri, et al. 2024. Findings of the quality estimation shared task at wmt 2024 are llms closing the gap in qe? In *Proceedings of the Ninth Conference on Machine Translation,*, pages 82–109. Association for Computational Linguistics.

Lily H Zhang, Hamid Dadkhahi, Mara Finkelstein, Firas Trabelsi, Jiaming Luo, and Markus Freitag. 2024.

Learning from others' mistakes: Finetuning machine translation models with span-level error annotations. *arXiv preprint arXiv:2410.16509*.

A Appendix

A.1 Training details

We use the transformers library (Wolf et al., 2020) for training and inference with Tower-Instruct V2. For adapting Tower to token-level QE, we use LoRA (Hu et al., 2021) based fine-tuning with an additional classifier head. Therefore, we only train the adapters and the weights for classification head.

We add the adapters to the modules *q_proj,k_proj,v_proj,gate_proj,up_proj* and *down_proj*. We set a batch size for each device to 12 initially and enable *auto_find_batch_size* to *True* on 4 NVIDIA RTX A6000 GPU's. For having a larger batch size during training, we set *gradient_accumulation_steps* to 6. We use a *learning_rate* of $1e^{-5}$. We set the *eval_steps* to 50 and *num_train_epochs* to 10. The other parameters are set to default.

Using the cross-entropy loss for token-level QE directly is insufficient due to the fact that the majority of tokens are classified as '*Good*'. Hence, we find that the weighted cross-entropy loss is essential when fine-tuning the model. For the training on human MQM data, we set the weights to $0.05, 0.95$ to '*Good*' and '*Bad*' labels respectively. In the case of distilling from XCOMET, we observed more errors. Therefore, we find that setting them $0.2, 0.8$ to '*Good*' and '*Bad*' labels respectively provided stable training.

We train on data until WMT'22 for training and use WMT'24 for validation. We calculate the macro '*F1*' on token-level predictions as the validation metric and stop training if it does not improve for 10 consecutive *eval_steps*.

A.2 Partial vs Full Sequence Quality Estimation

We also compare the difference in performance between our proposed token-level QE for partial sequences with Tower trained for full sequence QE. We achieve this by adding a regression head to predict the score at the end-of-sentence token. Hence, the model uses the source and hypothesis to predict the score using regression head at the end.

We fine-tune the model using only direct assesment data (Zerva et al., 2024) (**Tower Full DA**). Furthermore, we use this as initialisation and continue fine-tuning on the MQM data (**Tower Full DA + MQM**). We also use LoRA similarly to the previous model with a regression head to adapt the model. We report the scores in Table 5.

We see that the both Tower QE models based on full sentences outperforms the partial model. However, this is expected as it has seen the entire context and was also trained on larger amounts of data. Nonetheless, the partial model still achieves much higher correlaiton that the log probabilities showcasing its potential for Quality-Aware decoding.

A.3 Robustness to re-ranking weight

In our method, we introduce a hyperparameter, α, to merge the probabilities from the token-level QE model and the translation model. This section analyzes the impact of α on the final translation quality.

To efficiently evaluate its effect, we re-rank the N-best list using different values of α. This approach allows us to estimate the ideal value of α without the need for joint decoding multiple times. If the re-ranking model (in this case, Tower QE) is beneficial, we expect that any α less than 1 will improve translation quality, as it demonstrates that incorporating the probabilities from the QE model is helpful.

We visualize this impact in Figure 4. The results show that using an α less than 1 leads to improved translation quality in both scenarios. This indicates that relying entirely on the NMT model does not yield the best results and highlights the importance of the Tower QE model.

Thus, we emphasize that re-ranking the N-best list provides an effective way to tune the value of α, and it remains robust to different values.

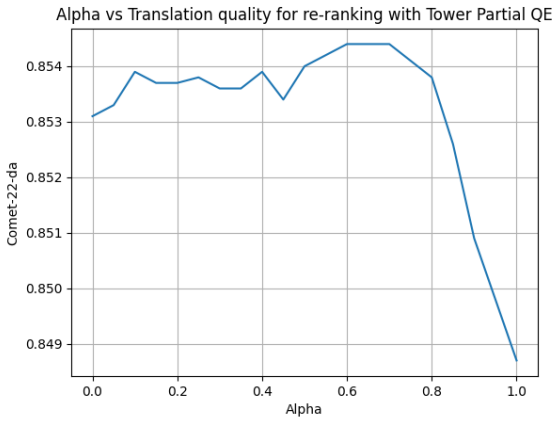
Tower Translation Prompt

<|im_start|>user
Translate the sentence from English into German.
English: {src_sent}
German:
<|im_end|>
<|im_start|>assistant

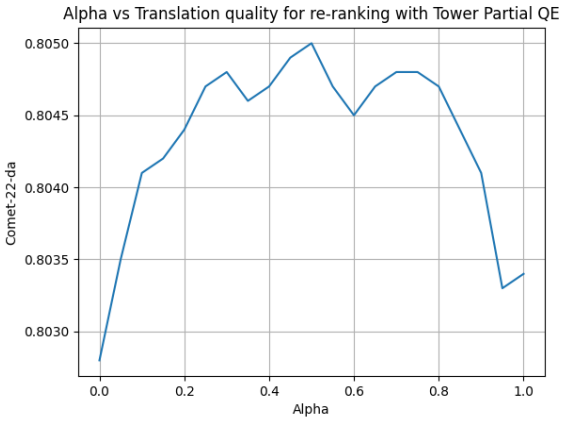
Tower Token-Level QE Prompt

English:{src_sent}
German: {tgt_sent}

Figure 3: Prompts used in our experiments for translation and QE model. {src_sent} and {tgt_sent} represent the source and target sentence. We replace the language with Chinese and English when experimenting with that language pair.



(a) English → German



(b) Chinese → English

Figure 4: Impact of α when re-ranking with token-level Tower QE on WMT'23 Test sets.

	Pearson	Spearmann	Kendall
COMETQE	**44.41**	41.29	31.19
COMETQE-XL	41.23	**42.17**	**31.84**
COMETQE Scratch Fine-tuned (ours)	36.32	33.66	25.24
Tower Log Prob	32.32	16.74	12.77
Tower Partial QE	40.56	33.96	25.87
Tower Full DA	33.67	36.46	27.38
Tower Full DA + MQM	32.03	40.85	30.38

Table 5: Full Correlation results on WMT 23 for English → German Test set. Partial indicates that the QE model predict scores via token-level where as full indicates predicting the score at the end-of-sentence token. The scores are calculated after removing the few sentences labelled for hallucination detection. Best scores according to each coefficient are highlighted in **bold**.