COMET-poly: Machine Translation Metric Grounded in Other Candidates

Maike Züfle[1] ★ **Vilém Zouhar**[2] ★ **Tu Anh Dinh**[1] ★ **Felipe Maia Polo**[3]
Jan Niehues[1] **Mrinmaya Sachan**[2]

[1]Karlsruhe Institute of Technology [2]ETH Zurich [3]University of Michigan
{maike.zuefle,tu.dinh}@kit.edu vzouhar.ethz.ch

Abstract

Automated metrics for machine translation attempt to replicate human judgment. Unlike humans, who often assess a translation in the context of multiple alternatives, these metrics typically consider only the source sentence and a single translation. This discrepancy in the evaluation setup may negatively impact the performance of automated metrics. We propose two automated metrics that incorporate additional information beyond the single translation. $\text{COMET}_{\text{poly-cand}}$ uses alternative translations of the same source sentence to compare and contrast with the translation at hand, thereby providing a more informed assessment of its quality. $\text{COMET}_{\text{poly-ic}}$, inspired by retrieval-based in-context learning, takes in translations of similar source texts along with their human-labeled quality scores to guide the evaluation. We find that including a single additional translation in $\text{COMET}_{\text{poly-cand}}$ improves the segment-level metric performance (0.079→0.118 τ_b), with further gains when more translations are added. Incorporating retrieved examples in $\text{COMET}_{\text{poly-ic}}$ yields similar improvements (0.079→0.116 τ_b). We release our models publicly.[1]

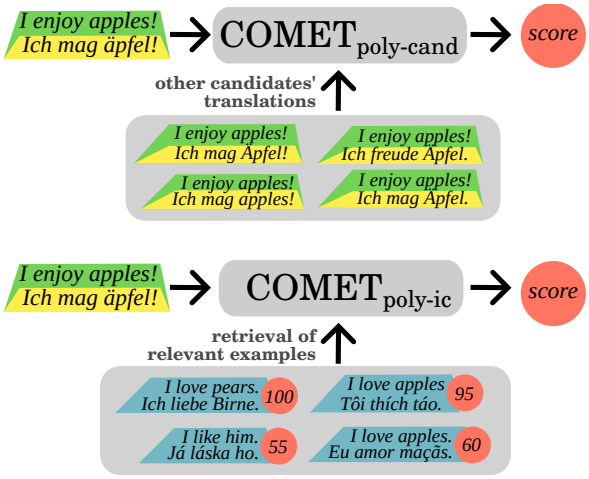

Figure 1: The $\text{COMET}_{\text{poly-ic}}$ model consults a knowledge base of previously human-scored translations before assigning the quality estimation score to the candidate translation. The $\text{COMET}_{\text{poly-cand}}$ considers other possible translations apart from the candidate one. Both metrics work better than just providing the source and the translation.

1 Introduction

There is a gap between how humans and automated metrics score translations. Automated metrics receive the source segment, usually a sentence or a paragraph, a single translation, and optionally a reference translation. They are then tasked with assessing the quality of the translation. In contrast, human evaluation is less episodic. Human raters often assess multiple translations in sequence (Graham et al., 2013; Freitag et al., 2021; Kocmi et al., 2024b), considering them side-by-side. Even though annotations are made for each translation individually, annotators become calibrated (known as sequence effect, Mathur et al., 2017), to common error patterns and their own evaluation criteria as they review multiple translations. As a result, they effectively score each translation in the context of others. Moreover, unlike human annotators, who have a deep understanding of the languages involved and can assess a wide range of translation qualities, automated metrics are limited by the data they were trained on. As a result, their performance tends to degrade when evaluating translations that deviate from their training distributions, such as out-of-domain content (Zouhar et al., 2024).

We present two conceptual approaches to address these two challenges by incorporating additional context into standard automated metrics, such as COMET (Rei et al., 2020). Our main motivation is to narrow the gap between human evaluation and automated metrics, enabling automated metrics to score translations in the context of other

★Equal contribution, sorted anti-alphabetically.

[1]We release the paper code and pre-trained quality estimation models $\text{COMET}_{\text{poly-ic}}$ and $\text{COMET}_{\text{poly-cand}}$.

Proceedings of the Tenth Workshop on Machine Translation (WMT), pages 887–904
November 8–9, 2025.

translations and making them more robust to out-of-domain data. Specifically, we introduce two models trained within this framework, $\text{COMET}_{\text{poly-cand}}$ and $\text{COMET}_{\text{poly-ic}}$:

- In $\text{COMET}_{\text{poly-cand}}$, different translations of the same source sentence are provided to the model as additional context (Figure 1 top). This is suitable for scenarios such as (1) benchmarking, where we evaluate translations of multiple systems on the same source sentence, or (2) reranking, where we need to select the best translation from a pool of candidate translations.
- In $\text{COMET}_{\text{poly-ic}}$, which is inspired by retrieval-based in-context learning, tuples of (*source*, *translation*, *human quality score*) are provided to the model as additional context. The tuples are retrieved based on the source sentence similarity to the evaluation example at hand (Figure 1 bottom). In practice, in-context examples can be obtained from existing, previously scored translations—such as those found in prior WMT annotation datasets (Freitag et al., 2024; Kocmi et al., 2024a).

This paper is structured as follows. In Section 2, we first describe the task of machine translation quality estimation (QE) and COMET (Rei et al., 2020), a popular QE metric. Then, we describe our two proposed model variants, $\text{COMET}_{\text{poly-cand}}$ and $\text{COMET}_{\text{poly-ic}}$. We also apply the same approach to two additional QE systems with contrasting characteristics: a non-parametric k-NN baseline and GEMBA, a large parametric LLM-based evaluator. In Sections 3 and 4, we show that our methods not only improve COMET's segment-level performance but also outperform both the much larger GEMBA model and the k-NN baseline, despite their simplicity. These approaches also show promise for instant on-the-fly domain adaptation. We place our contributions in context with related work in Section 5. Finally, in Section 6, we provide some practical guidance on using these metrics along with potential caveats.

We publicly release our models under open license, and submit our models to the WMT 2025 Metrics Shared Task.

2 Methods

In this section, we introduce the translation quality estimation task, review COMET, and present two extensions for improved quality estimation and domain adaptation.

2.1 Background

Quality estimation (QE). Given a source text s and a model-produced translation (MT) t, which is assessed by a human annotator on a scale from 0% to 100%, the goal of quality estimation (QE) is to develop a metric to predict this score.

Baseline COMET. Traditionally, quality estimation relied on static, rule-based metrics, but the field has shifted toward learned, data-driven metrics that can better approximate human judgments (Freitag et al., 2022). Learned automated metrics can be thought of as a function f, taking a source sentence s and a translation t as input and producing a continuous score $f(s,t) \in [0,1]$. f is usually trained in a supervised manner to approximate human judgment $y_{s,t} = \text{human}(s,t)$:

$$f(s,t) \xrightarrow{\text{train}} y_{s,t}$$

A popular recent choice for f is COMET (Rei et al., 2020), which is a combination of a trainable encoder model e_{θ_1} and a multi-layer perception head MLP_{θ_2}. COMET first embeds the source and translation texts, obtaining $s^e = e_{\theta_1}(s)$ and $t^e = e_{\theta_1}(t)$, and then transforms the embeddings into a score prediction using MLP_{θ_2}. We denote the set of trainable weights as $\theta = (\theta_1, \theta_2)$. Specifically, COMET is formulated as:

$$\text{COMET}_\theta(s,t) = \text{MLP}_{\theta_2}\big(g_{\theta_1}(s,t)\big),$$
$$\text{with } g_{\theta_1}(s,t) = \langle s^e, t^e, |s^e - t^e|, s^e * t^e \rangle.$$

Here, g_{θ_1} constructs a feature vector for the pair (s,t) by concatenating their embeddings s^e and t^e with additional element-wise transformations: the absolute difference $|s^e - t^e|$ and the element-wise product $s^e * t^e$. The trainable weights θ are optimized by minimizing the mean squared error between the COMET score and human labels using a variation of the stochastic gradient descent algorithm.

While the baseline COMET framework is effective, it does not support incorporating additional information. Just like for human evaluators, having more information such as (1) multiple candidates' translations for the same source, or (2) ground-truth example quality scores of translations, could improve the performance of COMET further. Thus, we introduce two extensions for COMET, which we train from scratch..

2.2 Multiple Candidates: $\text{COMET}_{\text{poly-cand}}$

Our first variant targets scenarios like benchmarking or reranking MT models, where multiple translations of the same source segment are available. In these cases, we extend the model's context by including additional translations $\{t_i\}_{i=2}^{n}$ of the same source sentence s, allowing the model to leverage multiple candidate translations simultaneously. Figure 1 (top) shows an illustration of this model architecture.

Specifically, we include the embeddings of these additional translations as part of the input to the multi-layer perceptron. Formally, for all $i \in \{2, \cdots, n\}$, we define

$$g_{\theta_1}(t, t_i) = \langle t_i^e, |t_i^e - t^e|, t_i^e * t^e \rangle.$$

We then concatenate $\langle g_{\theta_1}(s,t), g_{\theta_1}(t,t_1), ..., g_{\theta_1}(t,t_n) \rangle$ and pass it to the MLP. During training, we ensure that the additional translations $\{t_i\}_{i=2}^{n}$ differ from the main translation t, and keep n fixed across all training examples.

Joint predictions. To reduce computation time, $\text{COMET}_{\text{poly-cand}}$ can be trained to jointly predict the quality scores of the original translation along with those of the additional translations. The training objective then becomes:

$$f(s, t, t_2, ..., t_n) \xrightarrow{\text{train}} y_{s,t}, y_{s,t_2}, ..., y_{s,t_n}$$

Using scores of other translations. When the human assessment scores for additional translations, $\{y_{s,t_i}\}_{i=2}^{n}$, are available, we can further augment the feature vector using these scores. The input to the MLP would become:

$$\langle g_{\theta_1}(s,t), g_{\theta_1}(t,t_1), y_{s,t_1}, ..., g_{\theta_1}(t,t_n), y_{s,t_n} \rangle$$

This is particularly useful when we wish to evaluate a new system on a pre-existing benchmark with other candidate translations whose qualities are already annotated by humans.

2.3 In-context Learning: $\text{COMET}_{\text{poly-ic}}$

In the previous approach, we used additional translations of the same source sentence, a setup that might be unrealistic outside controlled scenarios such as benchmarking or reranking. An alternative, inspired by the success of *in-context* learning in other domains (Brown et al., 2020), is to provide the model with other, similar examples: by conditioning on human-scored translations, it can learn the mapping between translation patterns and quality judgments on the fly.

$\text{COMET}_{\text{poly-ic}}$ implements this by retrieving source–translation–score triplets from a knowledge base, in our case, prior WMT annotation datasets (Freitag et al., 2024; Kocmi et al., 2024a), and using them as context, enabling the model to adapt its evaluation to different domains. An illustration is shown in Figure 1 (bottom).

Specifically, for each input example (source s, translation t), we retrieve the examples $\{(s_i, t_i, y_{s_i,t_i})\}_{i=2}^{n_{\text{ICL}}+1}$ from a knowledge base $\mathcal{D}$. The new examples are added to the representation vector similar to $\text{COMET}_{\text{poly-cand}}$, considering both embeddings and labels, by appending

$$\langle t_i^e, |t_i^e - t^e|, t_i^e * t^e, s_i^e, |s_i^e - s^e|, |s_i^e * s^e|, y_{s_i,t_i} \rangle$$

to $g_{\theta_1}(s,t)$ for all $i \in \{2, \cdots, n_{\text{ICL}} + 1\}$.

The ICL examples are retrieved using normalized embedding (cosine) similarity computed from either the source s^e (default), the translation t_i^e, their arithmetic combination $s^e + t_i^e$, or their concatenation $\langle s^e, t_i^e \rangle$. We retrieve up to five most similar examples, discarding exact matches during training. We present detailed ablations of different filtering and retrieval setups in Section 4.

2.4 Including Reference Translations

Optionally, COMET can also make use of a reference translation (Rei et al., 2020), though this is no longer part of the standard QE setup. We also report results for $\text{COMET}_{\text{poly-cand}}$ and $\text{COMET}_{\text{poly-ic}}$ in the reference-based setting, by incorporating the reference r in their inputs, i.e., $f(s, t, r)$. However, our primary focus remains on QE, as references are often unavailable in practical scenarios.

2.5 Models Beyond COMET

Since our method is not specific to COMET, we include two models that, like our extensions, can take multiple candidate translations into account.

***k*-nearest neighbors.** As our first baseline method, we propose using a k-nearest-neighbours (k-NN) approach, mirroring methods used in similar contexts (Dinh et al., 2024). k-NN naturally leverages existing high-quality examples by retrieving similar instances, providing a strong non-parametric baseline that complements our model-based approaches. The k-NN baseline is implemented for our two different setups: $k\text{-NN}_{\text{poly-cand}}$ and $k\text{-NN}_{\text{poly-ic}}$.

For k-NN$_{\text{poly-cand}}$ and for a pair (s, t), we retrieve k additional translations for s. These are selected based on the cosine similarity of the target translation t and candidate translations, where embeddings are obtained using the all-MiniLM-L12-v2 (Reimers and Gurevych, 2020), yielding the set $\{t_i\}_{i=2}^{k+1}$. We then rate each candidate using Baseline COMET, obtaining $\{\text{COMET}(s, t_i)\}_{i=2}^{k+1}$. Finally, we use their average as the final prediction, i.e.,

$$\hat{y}_{s,t}^{k\text{-NN}_{\text{poly-cand}}} = \frac{1}{k}\sum_{i=2}^{k+1}\text{COMET}(s, t_i).$$

For k-NN$_{\text{poly-ic}}$, we retrieve $k = n_{\text{ICL}}$ examples, $\{(s_i, t_i, y_{s_i,t_i})\}_{i=2}^{k+1}$, following the retrieval strategies described in Section 2.3, and then average their human scores to obtain the prediction:

$$\hat{y}_{s,t}^{k\text{-NN}_{\text{poly-ic}}} = \frac{1}{k}\sum_{i=2}^{k+1} y_{s_i,t_i}.$$

We further extend the k-NN approaches using weighted averages in Appendix D.

Using LLMs as evaluators. As a second baseline, we use large language models (LLMs) for MT evaluation, leveraging their effectiveness in this task (Kocmi and Federmann, 2023). Specifically, we apply in-context learning (Brown et al., 2020), a standard method for injecting new knowledge into LLMs at inference time. Similar to COMET$_{\text{poly}}$, we provide LLMs with additional contextual information when scoring translations. However, unlike COMET$_{\text{poly}}$ variants, which update model parameters during training, LLMs receive this information only through their prompts at inference time, without any parameter modification.

For prompt creation, we build on top of GEMBA (Kocmi and Federmann, 2023), a framework designed to prompt LLMs to score the quality of translations. Leveraging GEMBA's pre-defined prompts, we extend them to two settings: (1) GEMBA$_{\text{poly-cand}}$, where additional translations of the same source sentence are provided, and (2) GEMBA$_{\text{poly-ic}}$, where full examples (including source, translation, and human quality score) are included. Prompt details are provided in Appendix A.2.

3 Experimental Setup

This section outlines the training and evaluation procedures, as well as the experimental setup.

Data. We use the direct assessment scorings of WMT up to 2023 (inclusive) for training (600k segments). For testing and evaluation, we use WMT 2024 (105 segments), which has been evaluated with the ESA protocol (Kocmi et al., 2024b). This dataset covers eleven language pairs: English to Czech, German, Spanish, Hindi, Icelandic, Japanese, Russian, Chinese, Czech to Ukrainian, and Japanese to Chinese. From ESA, we use the final scores (as opposed to error spans), which have the same scale as direct assessment. For MQM, we convert the error span annotations on a translation to the final score by taking $1 - (5 \cdot major + 1 \cdot minor)/100$, where $major$ is the number of annotated major errors, and $minor$ is the number of minor errors annotated in the translation. In this way, the scores are aligned roughly on the same scale compared to DA scores.

Training. We train the Baseline COMET model, COMET$_{\text{poly-cand}}$ and COMET$_{\text{poly-ic}}$ based on pretrained RoBERTa (Liu et al., 2019) on WMT human judgment data for five epochs. For COMET$_{\text{poly-cand}}$, we retrieve up to five candidate translations, either randomly or based on embedding similarity. For COMET$_{\text{poly-ic}}$, we retrieve up to five in-context examples from the training data based on embedding similarity. The metrics are trained in a maximally comparable model setup, which is detailed in Appendix A.

Evaluation. We evaluate the metrics on the segment level in three ways: Pearson correlation, Kendall's tau-b, and Mean Absolute Error (MAE). In contrast to Freitag et al. (2024) we do not do perform any group-by-item nor group-by-item. Results are macro-averaged across eleven languages.

Pearson correlation measures the linear relationship between metric scores and human ratings: higher values indicate better alignment, though not necessarily on the same scale. Mean Absolute Error (MAE), in contrast, captures the average absolute difference between metric and human scores, with lower values indicating closer agreement in both value and scale. Kendall's tau-b focuses on rank correlation, reflecting how well the metric preserves the relative ordering of translations. While Pearson and Kendall's tau-b range from -1 to 1, MAE is unbounded and depends on the scoring scale.

Experiments. To ensure a controlled evaluation setting, we first train a standard COMET

Model	Reference-less $\rho\uparrow$	Reference-less $\tau_b\uparrow$	Reference-less MAE $\downarrow$	Reference-based $\rho\uparrow$	Reference-based $\tau_b\uparrow$	Reference-based MAE $\downarrow$
standard COMET model $f(s,t) \rightarrow \hat{y}_t$	0.105	0.079	30.2	0.245	0.166	26.6
COMET$_{\text{poly-cand}}$						
additional candidate $f(s,t,t_2^*) \rightarrow \hat{y}_t$	0.160	0.127	28.5	0.281	0.180	26.3
additional candidate, output joint predictions $f(s,t,t_2^*) \rightarrow \hat{y}_t, \hat{y_{t_2^*}}$	0.167	0.113	28.8	0.275	0.172	25.6
additional candidate and its score $f(s,t,t_2^*, y_{t^*2}) \rightarrow \hat{y}_t$	0.267	0.207	21.9	0.374	0.243	20.6
COMET$_{\text{poly-ic}}$						
additional candidate and its score $f(s,t,t_2^*, y_{t^*2}) \rightarrow \hat{y}_t$	0.141	0.116	27.3	0.352	0.247	15.3

Table 1: Results for COMET$_{\text{poly-cand}}$ and COMET$_{\text{poly-ic}}$. The first row shows the standard COMET. The middle and bottom parts show that adding additional translation candidates and in-context examples boosts performance.

Model (Reference-less)	$\rho\uparrow$					$\tau_b\uparrow$					MAE $\downarrow$				
(+additional)	+1	+2	+3	+4	+5	+1	+2	+3	+4	+5	+1	+2	+3	+4	+5
$f(s,t) \rightarrow \hat{y}_t$	0.105	0.105	0.105	0.105	0.105	0.079	0.079	0.079	0.079	0.079	30.2	30.2	30.2	30.2	30.2
COMET$_{\text{poly-cand}}$															
$f(s,t,t_{...}) \rightarrow \hat{y}_t$	0.160	0.251	0.224	0.202	0.190	0.127	0.145	0.127	0.130	0.120	28.5	27.6	26.8	28.4	27.6
$f(s,t,t_{...}, y_{t_{...}}) \rightarrow \hat{y}_t$	0.267	0.321	0.328	0.327	0.321	0.207	0.229	0.230	0.235	0.233	21.9	17.3	16.0	14.0	13.7
COMET$_{\text{poly-ic}}$															
$f(s,t,t_{...}, y_{t_{...}}) \rightarrow \hat{y}_t$	0.141	0.134	0.148	0.128	0.068	0.116	0.108	0.114	0.105	0.075	27.3	27.2	24.7	27.6	27.4

Table 2: Results for COMET$_{\text{poly-cand}}$ and COMET$_{\text{poly-ic}}$ using different numbers of additional translation candidates. The +1 is equal to the results in Table 1. The +x uses x additional translation candidates, which improves performance especially for COMET$_{\text{poly-cand}}$.

model on the data described before and use it as a baseline. We then investigate COMET$_{\text{poly-cand}}$ by incorporating additional translations into the base model and analysing the impact of different selection strategies. Similarly, we explore COMET$_{\text{poly-ic}}$, experimenting with various retrieval methods and assessing its potential for domain adaptation. We complement our experiments with k-NN$_{\text{poly-cand}}$ and k-NN$_{\text{poly-ic}}$ as non-parametric baselines, and GEMBA$_{\text{poly-cand}}$ and GEMBA$_{\text{poly-ic}}$ as large-parameter LLM baselines.

4 Results and Analysis

In the following, we discuss and analyse the results of COMET$_{\text{poly-cand}}$ and COMET$_{\text{poly-ic}}$, compare them to the non-parametric k-NN and the large parametric GEMBA model, and discuss the runtime impact of our method.

4.1 Results for COMET$_{\text{poly-cand}}$

Additional candidate helps. We begin by evaluating COMET$_{\text{poly-cand}}$ in its simplest setting: adding a single additional translation from the same source as the candidate being scored. We choose the closest additional translation t_2^* as, intuitively, the closer it is to the candidate t, the more relevant it is for assessing its quality. We select t_2^* based on the embedding distance computed between candidate translations (see Appendix A for details on embeddings and distance metrics). The corresponding results are shown in the middle part of Table 1.

Across all evaluation metrics, including an additional translation $f(s,t,t_2)$, considerably improves performance compared to the standard COMET baseline $f(s,t)$. Specifically, Pearson correlation improved by over 50%. The joint translation prediction objective, which scores both the original translation and the additional translation, also yields gains over the baseline, though it performs slightly worse than the single-prediction setup. This suggests that, in scenarios where faster inference is needed, the joint-prediction setup offers a practical trade-off, delivering improved performance with smaller additional cost. Finally, including the gold score y_{t_2} of the additional translation in the input vastly improves the metric performance. However, note that this is an ideal scenario where the gold score y_{t_2} is available, which is not always realistic.

Note that it is not always possible to find additional translations that are similar to the translation

at hand. Therefore, we experiment with using a randomly selected additional candidate to test the robustness of $\text{COMET}_{\text{poly-cand}}$. This still results in notable gains, albeit smaller than with similar candidates. We report these results in Appendix B.

More than one candidate helps. We extend $\text{COMET}_{\text{poly-cand}}$ by increasing the number of additional candidates. The results are shown in Table 2. Having more than one additional candidate further improves the performance of $\text{COMET}_{\text{poly-cand}}$, as we are providing a more global view of possible translations to the model. However, this effect starts to diminish beyond two additional candidates. For comparison, results using random additional candidates are provided in Appendix B.

Additional translation complement reference. Previous experiments focused on reference-free evaluation. To complete the picture, we now explore how $\text{COMET}_{\text{poly-cand}}$ performs when reference translations are available.

The right half of Table 1 shows that using $\text{COMET}_{\text{poly-cand}}$ with reference yields better performance than $\text{COMET}_{\text{poly-cand}}$ in QE mode, though the gain is smaller than for standard COMET. This indicates that additional translations help narrow the gap but cannot fully replace references. Rather, additional translations complement references by providing further improvements on top of them.

4.2 Results for $\text{COMET}_{\text{poly-ic}}$

Building on this idea of leveraging additional context, we next evaluate $\text{COMET}_{\text{poly-ic}}$, which incorporates in-context examples to further enhance evaluation quality.

In-context examples help. We retrieve an in-context example using the source text s^e, embedded via an external embedding model (details in Appendix C). Results in the bottom row of Table 1 show that COMET benefits significantly from these examples, outperforming the baseline without in-context examples. This improvement also holds for $\text{COMET}_{\text{poly-ic}}$ with references. However, compared to $\text{COMET}_{\text{poly-cand}}$, in-context examples appear less informative than additional candidates with the same source, resulting in slightly reduced performance. We also test other embedding types (including COMET's own) and variations using the target or both source and target for retrieval. However, none of these alternatives yields further improvements. Full ablations are presented in Appendix C.

(a) k-$\text{NN}_{\text{poly-cand}}$

k	$\rho\uparrow$	$\tau_b\uparrow$	MAE$\downarrow$
1	0.083	0.064	30.4
2	0.087	0.064	30.3
3	0.086	0.062	30.4
4	0.085	0.059	30.4
5	0.085	0.057	30.4

(b) k-$\text{NN}_{\text{poly-ic}}$

k	$\rho\uparrow$	$\tau_b\uparrow$	MAE$\downarrow$
1	0.029	0.014	31.1
2	0.031	0.017	29.4
3	0.034	0.017	28.7
4	0.036	0.019	28.2
5	0.037	0.020	27.9

Table 3: Results for the k-nearest neighbors baseline using embeddings $\langle s^e, t_i^e \rangle$ in both k-$\text{NN}_{\text{poly-cand}}$ and k-$\text{NN}_{\text{poly-ic}}$ setup. k-NN consistently underperforms $\text{COMET}_{\text{poly-cand}}$ and $\text{COMET}_{\text{poly-ic}}$, showing notably lower correlations despite comparable MAE.

More in-context examples improve performance. While a single in-context example already boosts performance, adding up to three examples leads to further improvements. As shown in the bottom half of Table 2, performance increases with the number of retrieved examples using the external embedding model and s^e for retrieval, but declines beyond three examples, likely because additional examples become less similar and less relevant.

We also provide preliminary experiments in Appendix C.3 on how $\text{COMET}_{\text{poly-ic}}$ can leverage in-context examples to adapt its quality estimation to a new domain, and find a slight improvement compared to the base model.

4.3 Adding Candidates to Models Beyond COMET

In order to see whether having additional candidates or examples also helps with other QE methods other than COMET, we look into the performance of two baselines: the non-parametric ***k***-nearest neighbors and large parametric LLM evaluator with GEMBA.

We use ***k***-nearest neighbors in the retrieval setting for both k-$\text{NN}_{\text{poly-cand}}$ and k-$\text{NN}_{\text{poly-ic}}$, i.e., retrieving similar examples along with their gold quality scores, since the gold scores are required for ***k***-nearest neighbors. For GEMBA, we experiment with all $\text{GEMBA}_{\text{poly-cand}}$ variances (random/similar candidate, with/without gold scores) and $\text{GEMBA}_{\text{poly-ic}}$, similar to $\text{COMET}_{\text{poly-cand}}$ and $\text{COMET}_{\text{poly-ic}}$.

***k*-nearest neighbors underperforms COMET.** We present results for the k-nearest neighbors (k-NN) baseline in Table 3, varying k from 1 to 5,

	Input → Output	Reference-less			Reference-based		
		$\rho \uparrow$	$\tau_b \uparrow$	MAE $\downarrow$	$\rho \uparrow$	$\tau_b \uparrow$	MAE $\downarrow$
standard GEMBA	$f(s,t) \rightarrow \hat{y}_t$	0.266	0.199	27.6	0.311	0.200	27.3
GEMBA$_{\text{poly-cand}}$, closest t_2^*							
additional candidate	$f(s,t,t_2^*) \rightarrow \hat{y}_t$	0.245	0.185	28.2	0.277	0.187	27.5
additional candidate, joint predictions	$f(s,t,t_2^*) \rightarrow \hat{y}_t, \hat{y_{t_2^*}}$	0.235	0.149	28.6	0.296	0.181	27.9
additional candidate and its score	$f(s,t,t_2^*,y_{t^*2}) \rightarrow \hat{y}_t$	0.276	0.187	27.4	0.337	0.217	26.8
GEMBA$_{\text{poly-ic}}$							
additional candidate and its score	$f(s,t,s_2,t_2,y_{t_2}) \rightarrow \hat{y}_t$	0.195	0.099	28.3	0.291	0.168	27.4

Table 4: Results for GEMBA$_{\text{poly-cand}}$ and GEMBA$_{\text{poly-ic}}$. The first row shows the standard GEMBA model. In contrast to the COMET models, adding additional translation candidates and in-context examples does not significantly boost performance.

along with the simple average approach. For k-NN$_{\text{poly-ic}}$, neighbors are retrieved using the embedding $\langle s^e, t_i^e \rangle$. k-NN$_{\text{poly-ic}}$ performs markedly worse than our COMET variants (COMET$_{\text{poly-cand}}$ and COMET$_{\text{poly-ic}}$), particularly on correlation metrics, though MAE differences remain small. This is expected, as ***k***-nearest neighbors naively aggregate the scores of the closest datapoints, without actually modeling the underlying relationships between the source and translation to output the quality score. In the cases where the neighbors are not close enough, the output from ***k***-nearest neighbors would be suboptimal. In the poly-cand scenario, k-NN$_{\text{poly-cand}}$ achieves results similar to the naive COMET approach, unsurprising given that k-NN in this case effectively averages COMET scores for similar translations.

A more comprehensive set of results is provided in Appendix D, including a weighted variant of the ***k***-nearest neighbors baseline. The appendix also compares different retrieval strategies for k-NN$_{\text{poly-ic}}$. Among them, retrieval using $\langle s^e, t_i^e \rangle$ performs best; this contrasts with COMET$_{\text{poly-ic}}$, where retrieving based solely on the source yields better results. This difference arises because retrieval based only on source can hurt k-NN$_{\text{poly-ic}}$ by averaging scores from translations that may not align well with the target one.

COMET$_{\text{poly-cand}}$ outperforms GEMBA. We now move on to the parameter-heavy LLM baseline GEMBA. The main results for GEMBA$_{\text{poly-cand}}$ and GEMBA$_{\text{poly-ic}}$ are shown in Table 4.

Due to the large size and large amount of pretraining data of LLMs, the baseline GEMBA model has notably better performance than the baseline COMET (0.266 Pearson versus 0.105 Pearson). However, GEMBA does not benefit from our poly-cand and poly-ic setup. In most configurations, neither method improves over the baseline. Consequently, by better making use of additional examples, the COMET$_{\text{poly-cand}}$ variance outperforms all GEMBA variances. The exception is GEMBA$_{\text{poly-cand}}$ with the closest additional translation and its gold quality score, which yields better performance than baseline GEMBA. This is unsurprising, as the target translation's quality is likely similar to that of its closest neighbor, whose score is provided to the model. We also test adding random or multiple examples; random candidates perform comparably to similar ones, while multiple examples do not consistently yield further gains. Detailed results can be found in Appendix E.

4.4 Comparing Efficiency of COMET-poly Models

While the previous section shows that COMET$_{\text{poly-cand}}$ outperforms GEMBA in certain evaluation settings, this advantage is even more significant in practice due to efficiency. Table 5 shows that overall, running GEMBA is considerably slower and requires more computational resources than COMET. This highlights the benefits of training a small, specialized model (COMET$_{\text{poly-cand}}$) to match the performance of large, general-purpose models (GEMBA), while substantially reducing inference-time computational costs.

On the other hand, compared to k-NN, COMET$_{\text{poly}}$ is less efficient. k-NN is non-parametric, thus its computation time is almost instantaneous when excluding retrieval cost. However, as we have seen in the previous section, k-NN has notably worse performance compared to COMET$_{\text{poly}}$.

We next examine the general runtime behavior

	COMET	GEMBA
standard model		
$f(s,t) \rightarrow \hat{y_t}$	4.4s/1k	196.1s/1k
poly-cand		
$f(s,t,t_2) \rightarrow \hat{y_t}$	6.9s/1k	254.0s/1k
$f(s,t,t_2) \rightarrow \hat{y_t}, \hat{y_{t_2}}$	3.5s/1k	146.3s/1k
$f(s,t,t_2,y_{t_2}) \rightarrow \hat{y_t}$	6.9s/1k	256.0s/1k
poly-ic		
$f(s,t,s_2,t_2,y_{t_2}) \rightarrow \hat{y_t}$	7.2s/1k	233.0s/1k

Table 5: Inference time of GEMBA models compared to COMET models on the WMT 2024 test set (time per 1000 scores output on a single NVIDIA H100). COMET has ∼0.5B params and GEMBA 70B. GEMBA is run with 4-bit quantization. $\text{COMET}_{\text{poly-ic}}$ introduces an additional cost of retrieving from a vector knowledge base which we exclude for both $\text{COMET}_{\text{poly-ic}}$ and $\text{GEMBA}_{\text{poly-ic}}$.

of our methods across multiple settings. Looking at Table 5, unsurprisingly, integrating additional candidates $f(s,t,t_2)$ is more expensive in comparison to the baseline model with only one translation $f(s,t)$. However, most of the computation is spent on encoding the text sequences, which can be efficiently cached during inference (Rei et al., 2022), making all of the metric variations comparable. Moreover, if both t and t_2 need to be scored, then using a model that predicts both of their scores $\hat{y_t}, \hat{y_{t_2}}$ is faster than computing $f(s,t)$ and $f(s,t_2)$ together.

4.5 Analysis

To better understand the impact of our method, we investigate how additional translations or samples influence COMET's quality predictions.

$\text{COMET}_{\text{poly-cand}}$. We first perform a systematic analysis by categorizing test cases according to the gold quality scores of both the translation under evaluation and its additional translation. Specifically, we consider four combinations: (i) both high-quality, (ii) sample high / additional low, (iii) sample low / additional high, and (iv) both low-quality.

Results show that additional translations are most beneficial when the evaluated output is of lower quality. Interestingly, the quality of the additional translation itself has little impact on QE performance. This suggests that even low-quality additions can aid COMET by introducing complementary error patterns that highlight discrepancies. Detailed results can be found in Appendix F.

We then focus on individual cases where the additional translation yields the largest improvements. To do so, we sort the test samples in descending order by the difference between COMET's absolute error and that of $\text{COMET}_{\text{poly-cand}}$, thereby identifying the samples where $\text{COMET}_{\text{poly-cand}}$ yields the greatest improvement. We then conduct a manual inspection of the top cases, revealing that additional translations help COMET better detect specific failure modes: undertranslation, where the translation is merely a copy of the source; numerical errors, where numeric values in the translation differ from the source; explanations, where unnecessary explanatory text is added; and refusals, where the translation includes statements declining to translate the input. In these cases, the additional translations do not exhibit the same errors as the translation under evaluation. We therefore hypothesize that the additional translations effectively serve as references in such scenarios. We provide specific examples in Appendix B.2 in Appendix F.

$\text{COMET}_{\text{poly-ic}}$. We perform a similar systematic analysis for $\text{COMET}_{\text{poly-ic}}$ to study how in-context examples influence the scoring of high- and low-quality translations. Consistent with $\text{COMET}_{\text{poly-cand}}$, $\text{COMET}_{\text{poly-ic}}$ shows greater benefits when evaluating lower-quality outputs (see Appendix F for details).

In addition, we also investigate the choice of in-context examples, which is critical for $\text{COMET}_{\text{poly-ic}}$'s performance. During training, retrieved examples are drawn from the training set and thus come from the same distribution and have been seen by the model. In contrast, at test time, the examples are unseen and often less similar. We investigate whether the train-test mismatch affects $\text{COMET}_{\text{poly-ic}}$ by training models with different similarity thresholds. However, we find that the train-test mismatch does not significantly impact performance. Details can be found in Appendix C.4.

5 Related Work

This section reviews the broader context of automated metrics and human evaluations that use multiple inputs: either multiple translations or, more commonly, multiple references.

Automated metrics. Early metrics like BLEU (Papineni et al., 2002) and ChrF (Popović, 2015) operate at segment or corpus level and support multiple references but not multiple hypotheses simultaneously. COMET (Rei et al., 2020) trains an

encoder for human-like quality assessment and supports a single reference. Adding more references shows limited gains (Zouhar and Bojar, 2024).

Closest to our work, Dinh et al. (2024) propose a k-NN quality estimator similar to $\text{COMET}_{\text{poly-ic}}$, but aggregate train-test similarity of MT models as a quality indicator rather than having a separate QE model that assesses translations based on similarity and contextual relevance. Moosa et al. (2024) introduce MT-Ranker, which compares translation pairs and outputs a binary preference.

With the rise of Large Language Models (LLMs), an up-to-date approach for Quality Estimation is to use LLM-as-a-Judge. Simply prompting LLMs to output the quality score of a translation has become the state-of-the-art approach, with the most prominent example of GEMBA (Kocmi and Federmann, 2023). This approach has the potential to improve even further, by applying different strategies such as including in-context examples (few-shot judge), chain-of-thought prompting, pairwise comparison, as recommended by Zheng et al. (2023).

Human evaluation. Human evaluation of machine translation takes many forms. For benchmarking, WMT initially used RankME (Novikova et al., 2018), where annotators rank multiple hypotheses simultaneously.

Due to biases and high cognitive load, this shifted to single-hypothesis assessments such as Direct Assessment and its variants (Graham et al., 2013; Kocmi et al., 2022), Multidimensional Quality Metrics (Freitag et al., 2021), and Error Span Annotation and its variants (Kocmi et al., 2024b; Zouhar et al., 2025). Despite judging one hypothesis at a time, annotators gradually see other translations during evaluation, implicitly calibrating their quality judgments. Automated metrics, however, lack this contextual grounding and evaluate translations independently.

6 Discussion and Conclusion

Recommendation. $\text{COMET}_{\text{poly-cand}}$ can be applied in scenarios where multiple translations exist for the same source sentence, such as: (1) enchmarking various competing systems on the same test set (e.g., WMT General shared tasks), (2) comparing outputs from different checkpoints or models during MT development, or (3) cselecting the best translation from a pool of hypotheses during reranking for final output selection.

The intended use of $\text{COMET}_{\text{poly-ic}}$ is for quick domain adaptation without retraining the metric (Appendix C.3). While different retrieval methods can cause slight variations in performance (see Appendix C), it is crucial that the retrieval mechanism is deterministic to ensure reproducible scores. Additionally, changing the retrieval mechanism or the set of previously annotated translations that are being retrieved instantiates a new metric with non-comparable scores to the previous evaluations. Therefore, when using $\text{COMET}_{\text{poly-ic}}$, always disclose the retrieval set and retrieval method.

Training a smaller, specialized module with some tweaks ($\text{COMET}_{\text{poly-cand}}$) can be beneficial compared to directly using large, general-purpose language models (GEMBA). We have shown that $\text{COMET}_{\text{poly-cand}}$ can reach the performance of GEMBA, while being much more efficient in terms of inference time.

Submitted models. We submit the following models to the WMT Metrics Shared Task 2025 and make them publicly available under open license (Apache License 2.0) on Hugging Face. The models are trained on WMT data up to 2024 (inclusive).

- COMET-poly-base-wmt25: baseline
- COMET-poly-cand1-wmt25: one additional translation
- COMET-poly-cand2-wmt25: two additional translations
- COMET-poly-ic1-wmt25: one in-context example
- COMET-poly-ic3-wmt25: three in-context examples
- knn-poly-cand3: three additional translations, scored with COMET-poly-base-wmt25
- knn-poly-ic3: three in-context examples

Conclusion. In this work, we introduced two new paradigms for machine translation quality estimation: (1) evaluating a translation with the context of other translations of the same source, and (2) quality estimation with retrieval for in-context examples. We showed that these approaches show potential in being more adaptable and outperforming the baseline COMET, while also offering practical advantages in efficiency by matching the performance of larger models at lower computational cost.

Limitations

$\text{COMET}_{\text{poly-cand}}$ is entirely constrained to setups where we are scoring multiple translations at the same time. This is by design and thus mostly suited for WMT-style benchmarking competitions or model development where we wish to find which translation model is the best one. It is not useful for scenarios where a single model is being evaluated without the context of other existing translations.

Both $\text{COMET}_{\text{poly-cand}}$ and $\text{COMET}_{\text{poly-ic}}$ are not exempt on the reliance on the quality of previously human-annotated translations. In some cases, the quality of the collected data might be subpar (Kocmi et al., 2024a), which is then then further exemplifies its bias in $\text{COMET}_{\text{poly-cand}}$ and $\text{COMET}_{\text{poly-ic}}$.

Our investigation in this paper omits various tricks used to further boost COMET's performance for the purpose of clarity of the core methodological contributions of $\text{COMET}_{\text{poly-cand}}$ and $\text{COMET}_{\text{poly-ic}}$.

Ethics Statement

Vilém Zouhar declares a potential conflict of interest as an organizer of the WMT 2025 Metrics Shared Task. No privileged information has been used in this work.

Acknowledgements

This research has been funded in part by a Swiss National Science Foundation award (project 201009) and a Responsible AI grant by the Haslerstiftung. Part of this work received support from the European Union's Horizon research and innovation programme under grant agreement No 101135798, project Meetween (My Personal AI Mediator for Virtual MEETtings BetWEEN People). This work was also supported by the Helmholtz Programme-oriented Funding, with project number 46.24.01, project name AI for Language Technologies. We acknowledge the HoreKa supercomputer funded by the Ministry of Science, Research and the Arts Baden-Wurttemberg and by the Federal Ministry of Education and Research.

References

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

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.

Tu Anh Dinh, Tobias Palzer, and Jan Niehues. 2024. Quality estimation with k-nearest neighbors and automatic evaluation for model-specific quality estimation. In *Proceedings of the 25th Annual Conference of the European Association for Machine Translation (Volume 1)*, 133–146, Sheffield, UK. European Association for Machine Translation (EAMT).

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. *Transactions of the Association for Computational Linguistics*, 9:1460–1474.

Markus Freitag, Nitika Mathur, Daniel Deutsch, Chi-Kiu Lo, Eleftherios Avramidis, Ricardo Rei, Brian Thompson, Frederic Blain, Tom Kocmi, Jiayi Wang, David Ifeoluwa Adelani, Marianna Buchicchio, Chrysoula Zerva, and Alon Lavie. 2024. Are LLMs breaking MT metrics? results of the WMT24 metrics shared task. In *Proceedings of the Ninth Conference on Machine Translation*, 47–81. Association for Computational Linguistics.

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)*, 46–68. Association for Computational Linguistics.

Yvette Graham, Timothy Baldwin, Alistair Moffat, and Justin Zobel. 2013. Continuous measurement scales in human evaluation of machine translation. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, 33–41. Association for Computational Linguistics.

Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Marzena Karpinska, Philipp Koehn, Benjamin Marie, Christof Monz, Kenton Murray, Masaaki Nagata, Martin Popel, Maja Popović, Mariya Shmatova, Steinthór Steingrímsson, and Vilém Zouhar. 2024a. 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*, 1–46. Association for Computational Linguistics.

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)*, 1–45. Association for Computational Linguistics.

Tom Kocmi and Christian Federmann. 2023. Large language models are state-of-the-art evaluators of translation quality. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, 193–203. European Association for Machine Translation.

Tom Kocmi, Vilém Zouhar, Eleftherios Avramidis, Roman Grundkiewicz, Marzena Karpinska, Maja Popović, Mrinmaya Sachan, and Mariya Shmatova. 2024b. Error span annotation: A balanced approach for human evaluation of machine translation. In *Proceedings of the Ninth Conference on Machine Translation*, 1440–1453. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach.

Nitika Mathur, Timothy Baldwin, and Trevor Cohn. 2017. Sequence effects in crowdsourced annotations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2860–2865. Association for Computational Linguistics.

Ibraheem Muhammad Moosa, Rui Zhang, and Wenpeng Yin. 2024. MT-ranker: Reference-free machine translation evaluation by inter-system ranking. In *The Twelfth International Conference on Learning Representations*.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2018. RankME: Reliable human ratings for natural language generation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 72–78. Association for Computational Linguistics.

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*, 311–318. Association for Computational Linguistics.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, 392–395. Association for Computational Linguistics.

Ricardo Rei, Ana C Farinha, José G.C. de Souza, Pedro G. Ramos, André F.T. Martins, Luisa Coheur, and Alon Lavie. 2022. Searching for COMETINHO: The little metric that could. In *Proceedings of the 23rd Annual Conference of the European Association for Machine Translation*, 61–70. European Association for Machine Translation.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2685–2702. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 4512–4525. Association for Computational Linguistics.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with MT-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.

Vilém Zouhar and Ondřej Bojar. 2024. Quality and quantity of machine translation references for automatic metrics. In *Proceedings of the Fourth Workshop on Human Evaluation of NLP Systems (HumEval) @ LREC-COLING 2024*, 1–11. ELRA and ICCL.

Vilém Zouhar, Shuoyang Ding, Anna Currey, Tatyana Badeka, Jenyuan Wang, and Brian Thompson. 2024. Fine-tuned machine translation metrics struggle in unseen domains. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 488–500. Association for Computational Linguistics.

Vilém Zouhar, Tom Kocmi, and Mrinmaya Sachan. 2025. AI-assisted human evaluation of machine translation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 4936–4950, Albuquerque, New Mexico. Association for Computational Linguistics.

Appendix Overview

The appendix includes the following information:

- Implementation Details (§A)
- $\text{COMET}_{\text{poly-cand}}$ Ablations and Analysis (§B)
- $\text{COMET}_{\text{poly-ic}}$ Ablations and Analysis (§C)
- k-NN Ablations and Analysis (§D)
- GEMBA Ablations and Analysis (§E)
- Analysis of Impact of Additional Translations and In-Context Examples (§F)

A Implementation details

A.1 COMET

The model details are shown in Table 6. For computing embeddings to retrieve similar examples, by default we use the cosine distance from all-MiniLM-L12-v2 (Reimers and Gurevych, 2020). However, we also experiment in ablations with using the xlm-roberta-large (Conneau et al., 2020) embeddings and embeddings from a trained baseline COMET.

Encoder	xlm-roberta-large (24 layers)
Embeddings	Layerwise attention & CLS
Encoder frozen	30% of first epoch
Regression head	#features $\times$ 2048$\times$ $1024 \times (1 + \text{\#additional})$
Optimizer	AdamW
Learning rate	1.5×10^{-5}, encoder 10^{-6}
Batch size	256 (aggregated)
Loss	Average MSE across all targets
Training epochs	5

Table 6: COMET architecture and training details.

A.2 GEMBA

As the underlying LLM for GEMBA, we use Llama 3.3 70B with 4 bit quantization. All experiments with GEMBA are run on one H100 GPU with 80 GB of memory. The prompts we used for $\text{GEMBA}_{\text{poly-cand}}$ and $\text{GEMBA}_{\text{poly-ic}}$ are show in Table 7. Depending on the setting, the human reference and the gold score of the additional translation can be omitted, and more than one additional translations can be included.

B $\text{COMET}_{\text{poly-cand}}$ Ablations and Analysis

B.1 Robustness towards choice of additional candidate.

In a realistic usage, it might not always be possible to have additional candidate that is close to the original translation. Therefore, we experiment using $\text{COMET}_{\text{poly-cand}}$ with randomly selected additional candidate. The results are shown in the bottom half of Table 8 (5-7). As can be seen, even randomly selected additional translations significantly improve performance compared to the standard COMET model. However, compared to the setting with the closest candidate, random selection worsen the performance of $\text{COMET}_{\text{poly-cand}}$, albeit by a small margin. The largest performance drop occurs when the model uses the additional translation's score as input (7 vs 4). This is expected, as having the gold score of a similar candidate to the original translation is more informative than a score for a random one. This also holds when adding more translation candidates, as can be seen in Table 9. More detailed experiment on different levels of candidate similarity are provided below.

B.2 Effect of candidate similarity level

We examine the relationship between the additional translation's similarity to the one at hand. As can be seen in Table 10, the more similar the candidate, the more helpful it is to improve the performance of $\text{COMET}_{\text{poly-cand}}$. This is even more notable in the setting where we include the gold score of the candidate, $f(s, t, t_2, y_{t_2})$. However, in all settings, $\text{COMET}_{\text{poly-cand}}$ is still considerably improved compared to the baseline COMET model.

C $\text{COMET}_{\text{poly-ic}}$ Ablations and Analysis

C.1 Comparing different retrieval strategies.

We investigate different retrieval strategies: We retrieve using the embeddings derived only from the source text s^e, only the translation t_i^e, the sum of the two $s^e + t_i^e$, and the concatenation $\langle s^e, t_i^e \rangle$. We use all-MiniLM-L12-v2 (Reimers and Gurevych, 2020) as an embedding model. Table 11 shows that the simplest approach, only embedding the source yields the best performance across all metrics.

C.2 Testing different embedding models.

In previous experiments, we used an external embedding model (all-MiniLM-L12-v2 (Reimers and Gurevych, 2020)) to retrieve in-context examples. However, one could alternatively use the COMET model's own embeddings or its untrained xlm-roberta-large (Conneau et al., 2020) backbone. We continue using the source text for generating embeddings, as this consistently yielded the best results. Nonetheless, we find that the external embedding model achieves the strongest performance (Table 12), likely because it was explicitly trained

GEMBA$_{\text{poly-cand}}$
Score the translation provided at the end of this prompt from <*source lang*> to <*target lang*> with respect to human reference on a continuous scale from 0 to 100, where a score of zero means "no meaning preserved" and score of one hundred means "perfect meaning and grammar". Keep your explanation as short as possible. Provide the final score at the end of your answer; do not output anything else afterward.
<*source lang*> source: <*source sentence*>
<*target lang*> human reference: <*human translation*>

Below is an example translation along with its score:
<*target lang*> translation: "<*additional translation*>"
Score: <*score of additional translation*>

Now score this translation (remember to output the final score only at the end of your answer):
<*target lang*> translation: <*MT output*>

Score:

GEMBA$_{\text{poly-ic}}$
Score the translation provided at the end of this prompt from <*source lang*> to <*target lang*> with respect to human reference on a continuous scale from 0 to 100, where a score of zero means "no meaning preserved" and score of one hundred means "perfect meaning and grammar". Keep your explanation as short as possible. Provide the final score at the end of your answer, do not output anything else afterward.

Below is an example translation along with its score:
Source: <*additional source sentence*>
Translation: "<*additional translation*>"
Score: <*score of additional translation*>

Now score this translation (remember to output the final score only at the end of your answer):
<*source lang*> source: <*source sentence*>
<*target lang*> human reference: <*human translation*>
<*target lang*> translation: <*MT output*>

Score:

Table 7: Prompts for GEMBA$_{\text{poly-cand}}$ and GEMBA$_{\text{poly-ic}}$.

for cross-lingual sentence representation. This suggests that COMET$_{\text{poly-ic}}$'s performance is closely tied to the quality and suitability of the embedding model used for retrieval.

C.3 Adaption to the Biomedical Domain using COMET$_{\text{poly-ic}}$

In-Context Enables Domain Transfer. Table 13 presents results from testing our models on in-domain biomedical data. We use the BioMQM dataset (Zouhar et al., 2024). The MQM spans are turned into 0–100 scores to be compatible with the rest of the data. We use the small dev set for training (10k segments) and the test set for evaluation (43k segments).

The goal is to assess whether COMET$_{\text{poly-ic}}$ can leverage in-context examples to adapt its quality estimation to the new domain. This is indeed the case, particularly in MAE, where a substantial performance improvement is observed compared to the base model.

While fine-tuning the models on biomedical data yields even greater gains, it comes at a cost: the fine-tuned base model performs poorly on standard, non-biomedical data. In contrast, both COMET$_{\text{poly-ic}}$ and COMET$_{\text{poly-cand}}$ remain robust after fine-tuning and continue to perform well on standard data, likely because they can incorporate contextual signals at inference time.

C.4 Similarity Threshold Analysis for COMET$_{\text{poly-ic}}$

For COMET$_{\text{poly-ic}}$, the choice of in-context examples is crucial. During training, retrieved examples are drawn from the training set and thus come from the same distribution and have been seen by the model. In contrast, at test time, the examples are unseen and often less similar. Figure 2 shows a histogram of the inner product similarity between embeddings of the evaluated source and the top-1 retrieved source sentence. The plot reveals that training-time additional sources are generally more similar to the evaluated source than those retrieved during testing.

We investigate whether the train-test mismatch affects COMET$_{\text{poly-ic}}$ by training models with dif-

Model		Reference-less			Reference-based			
		$\rho\uparrow$	$\tau_b\uparrow$	MAE $\downarrow$	$\rho\uparrow$	$\tau_b\uparrow$	MAE $\downarrow$	
standard COMET model	$f(s,t) \rightarrow \hat{y_t}$	0.105	0.079	30.2	0.245	0.166	26.6	(1)
Additional candidate t_2^* is the closest								
additional candidate	$f(s,t,t_2^*) \rightarrow \hat{y_t}$	0.160	0.127	28.5	0.281	0.180	26.3	(2)
additional candidate, joint predictions	$f(s,t,t_2^*) \rightarrow \hat{y_t}, \hat{y_{t_2^*}}$	0.167	0.113	28.8	0.275	0.172	25.6	(3)
additional candidate and its score	$f(s,t,t_2^*, y_{t^*2}) \rightarrow \hat{y_t}$	0.267	0.207	21.9	0.374	0.243	20.6	(4)
Additional candidate t_2 is random								
additional candidate	$f(s,t,t_2) \rightarrow \hat{y_t}$	0.163	0.118	29.0	0.280	0.175	26.6	(5)
additional candidate, joint predictions	$f(s,t,t_2) \rightarrow \hat{y_t}, \hat{y_{t_2}}$	0.163	0.100	29.3	0.276	0.163	25.8	(6)
additional candidate and its score	$f(s,t,t_2, y_{t_2}) \rightarrow \hat{y_t}$	0.234	0.185	22.9	0.352	0.229	21.0	(7)

Table 8: Results for $\text{COMET}_{\text{poly-cand}}$. The first row shows the standard COMET. The top half (2-4) shows that adding additional translation candidate boosts performance. The bottom half (5-7) shows that using randomly selected additional candidates (in contrast to examples close to the original translation) also helps to boost performance, proving that $\text{COMET}_{\text{poly-cand}}$ is robust to the choice of additional candidates.

Model	$\rho\uparrow$					$\tau_b\uparrow$					MAE $\downarrow$				
(+additional)	+1	+2	+3	+4	+5	+1	+2	+3	+4	+5	+1	+2	+3	+4	+5
$f(s,t) \rightarrow \hat{y_t}$	0.105	0.105	0.105	0.105	0.105	0.079	0.079	0.079	0.079	0.079	30.2	30.2	30.2	30.2	30.2
t_i is the closest															
$f(s,t,t_{...}) \rightarrow \hat{y_t}$	0.160	0.251	0.224	0.202	0.190	0.127	0.145	0.127	0.130	0.120	28.5	27.6	26.8	28.4	27.6
$f(s,t,t_{...},y_{t_{...}}) \rightarrow \hat{y_t}$	0.267	0.321	0.328	0.327	0.321	0.207	0.229	0.230	0.235	0.233	21.9	17.3	16.0	14.0	13.7
t_i is random															
$f(s,t,t_{...}) \rightarrow \hat{y_t}$	0.163	0.202	0.219	0.228	0.204	0.118	0.135	0.140	0.144	0.136	29.0	27.3	27.9	27.8	28.1
$f(s,t,t_{...},y_{t_{...}}) \rightarrow \hat{y_t}$	0.234	0.276	0.295	0.293	0.295	0.185	0.212	0.216	0.215	0.216	22.9	19.3	15.9	14.9	14.3

Table 9: Results for $\text{COMET}_{\text{poly-cand}}$ using different number of additional translation candidates. The +1 is equal to results in Table 1. The +x uses x additional translation candidates, which improves performance especially for $\text{COMET}_{\text{poly-cand}}$.

Model	$\rho\uparrow$					$\tau_b\uparrow$					MAE $\downarrow$				
(+nth closest)	1st	2nd	3rd	4th	5th	1st	2nd	3rd	4th	5th	1st	2nd	3rd	4th	5th
$f(s,t) \rightarrow \hat{y}$	0.105	0.105	0.105	0.105	0.105	0.079	0.079	0.079	0.079	0.079	30.2	30.2	30.2	30.2	30.2
$f(s,t,t_2) \rightarrow \hat{y}$	0.163	0.157	0.150	0.140	0.133	0.118	0.114	0.112	0.109	0.107	29.0	29.1	29.2	29.3	29.4
$f(s,t,t_2,y_{t_2}) \rightarrow \hat{y}$	0.234	0.220	0.202	0.187	0.174	0.185	0.180	0.174	0.170	0.163	22.9	23.0	23.3	23.5	23.7

Table 10: Performance of $\text{COMET}_{\text{poly-cand}}$ with the additional translation being the closest, second-closest, third-closest, fourth-closest of fifth-closest to t.

Retrieval key	$\rho \uparrow$	$\tau_b \uparrow$	MAE
None	0.105	0.079	30.2
s_2^e	0.141	0.116	27.3
t_2^e	0.127	0.111	28.4
$s_2^e + t_2^e$	0.135	0.106	27.5
$\langle s_2^e, t_2^e \rangle$	0.117	0.109	27.7

Table 11: $\text{COMET}_{\text{poly-ic}}$ results, with in-context examples retrieved using source text s^e, only the translation t_i^e, the sum of the two $s^e + t_i^e$, and the concatenation $\langle s^e, t_i^e \rangle$.

	$\rho \uparrow$	$\tau_b \uparrow$	MAE
COMET embeddings	0.124	0.108	28.3
MiniLM embeddings (external)	0.141	0.116	27.3
XMLR embeddings (external)	0.115	0.093	27.7

Table 12: $\text{COMET}_{\text{poly-ic}}$ results, with in-context examples retrieved using source text s^e, using different embedding models.

ferent similarity thresholds to better align training retrievals with test-time similarity. The results in Table 14 show that the model trained without any similarity filtering performs best, suggesting that the train-test mismatch does not significantly impact performance.

D k-NN Ablations and Analysis

D.1 Weighted k-NN

We can extend the simple k-nn approach to incorporate weigthed averages, which can boost performance. For example, in the poly-cand setup, our final prediction will be given by

$$\hat{y}_{s,t} = \sum_{i=1}^{n} \left(\frac{w_i}{\sum_{i'=1}^{n} w_{i'}} \right) \times \text{COMET}(s, t_i),$$

where $w_i = \exp(-d_i/\gamma)$ is a weight with d_i being a dissimilarity measure between (s, t) and (s, t_i)

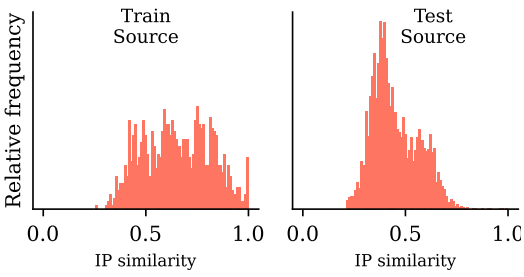

Figure 2: Histogram of inner product similarities between the currently evaluated item and the top-1 retrieved item for $\text{COMET}_{\text{poly-ic}}$.

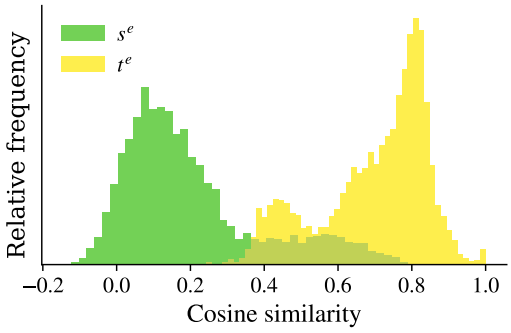

Figure 3: Histograms of translation similarity for examples retrieved by source embeddings (s^e) versus translation embeddings (t^e), showing that t^e-based retrieval yields higher-similarity (more relevant) neighbors while s^e-based retrieval often returns low-relevance examples.

(used for retrieval), and $\gamma > 0$ is the kernel bandwidth, that can be tuned using a validation set. We set d_i to be one minus the cosine similarity of embeddings. The same approach applies to the poly-ic setup. Realize that doing a simple average is equivalent to running the weighted average with $\gamma \to \infty$.

D.2 Ablation and Analysis

We evaluate the k-NN baseline under varying γ values using a weighted-average scoring scheme and different retrieval strategies in the poly-ic setting. Table 15 reports results for the poly-cand configuration: performance is remarkably consistent across both γ and k, since all retrieved translations are of similar relevance. Table 16 gives results for the poly-ic configuration: here, choices of γ and k have a pronounced effect, and the best scores are achieved when retrieval leverages both source and target contexts. Figure 3 complements Table 16 and explains why using s^e for retrieval when k-NN is applied works the worst; we plot the histograms of translation similarity when examples are retrieved either using the translation or the source embeddings. What we see is that when examples are retrieved using source similarity, there is no guarantee that the translations we retrieve are relevant for our target translation (low similarity). On the other hand, if the examples are retrieved using the translation similarities, we end up selecting more relevant examples in terms of similarity (as expected).

E GEMBA Ablations and Analysis

Adding random translations does not consistently improve performance. Similar to ap-

training data	Model	BioMQM Test $\rho \uparrow$	BioMQM Test $\tau_b \uparrow$	BioMQM Test MAE	WMT 2024 Test $\rho \uparrow$	WMT 2024 Test $\tau_b \uparrow$	WMT 2024 Test MAE
WMT (from scratch)	Base	0.100	0.117	35.7	0.105	0.079	30.2
	$\text{COMET}_{\text{poly-cand}}$	0.029	0.068	43.7	0.160	0.127	28.5
	$\text{COMET}_{\text{poly-ic}}$	0.109	0.118	30.5	0.141	0.116	27.3
BioMQM (finetune WMT)	Base	0.139	0.169	2.6	0.060	0.132	11.6
	$\text{COMET}_{\text{poly-cand}}$	0.215	0.177	2.1	0.162	0.175	12.0
	$\text{COMET}_{\text{poly-ic}}$	0.209	0.171	2.1	0.163	0.150	12.0
BioMQM + WMT (from scratch)	Base	0.165	0.141	12.8	0.233	0.187	15.6
	$\text{COMET}_{\text{poly-cand}}$	0.081	0.093	15.7	0.250	0.195	15.9
	$\text{COMET}_{\text{poly-ic}}$	0.168	0.146	12.6	0.240	0.192	15.5

Table 13: $\text{COMET}_{\text{poly-ic}}$'s and $\text{COMET}_{\text{poly-cand}}$'s performance on the BioMQM dataset (Zouhar et al., 2024) and the WMT 2024 dataset, trained on either WMT data, finetuned on BioMQM data (after training on WMT), or trained on a mix of BioMQM data and WMT data.

	$\rho \uparrow$	$\tau_b \uparrow$	MAE
Highest Similarity	0.141	0.116	27.3
Similarity < 0.7	0.130	0.101	28.4
Similarity < 0.5	0.109	0.086	29.1

Table 14: Performance of $\text{COMET}_{\text{poly-ic}}$ trained with different filter thresholds for additional source sentence similarity.

pendix B, we experiment with adding random translation candidates instead of the most similar ones. This yields similar results. These results are reported in Table 17.

Multiple additional translations is better. We experiment with multiple candidates/examples to $\text{GEMBA}_{\text{poly-cand}}$/$\text{GEMBA}_{\text{poly-ic}}$. As can be seen from Table 18, having 5 candidates instead of 1 helps $\text{GEMBA}_{\text{poly-cand}}$ improve over the baseline GEMBA in terms of Pearson correlation; however, the Kendall-tau and MAE metrics do not always agree. For $\text{GEMBA}_{\text{poly-ic}}$, having 5 samples instead of 1 even slightly worsens the performance.

In general, adding more examples to the input does not always help improve the performance of GEMBA as opposed to COMET. Note that the performance of the standard GEMBA is better than the standard COMET (0.266 Pearson versus 0.105 Pearson, see first row of Table 1 and Table 4 for more details). A possible explanation could then be: the additional candidates/samples added to the inputs help with issues that are specific to the baseline COMET, i.e., detecting edges cases of failed translations (see Section 4.5 for more details), which might not be an issue for the baseline GEMBA.

F Analysis of Impact of Additional Translations and In-Context Examples

We perform a systematic analysis by categorizing test cases according to the gold quality scores of the translation under evaluation. The test cases are split into two by the median of the gold quality scores. For $\text{COMET}_{\text{poly-cand}}$, we also further categorize the cases based on the gold quality scores of the additional translation: we consider the cases where (1) the additional translation t_2 is the best within the pool of candidate translations from the same source and (2) t_2 is the worst within the pool of candidate translations. The results can be found in Table 19.

We also manually inspect cases where the additional translation yields the largest improvements. These include, for example, undertranslations, numerical errors, explanations within the translations. We find that in these cases, the additional translation does not show such errors and can serve as a substitute reference. These examples are listed in Table 20.

		$\rho \uparrow$					$\tau_b \uparrow$					MAE $\downarrow$				
γ	k	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
10^{-4}		0.083	0.083	0.084	0.084	0.084	0.064	0.064	0.064	0.064	0.064	30.4	30.4	30.4	30.4	30.4
10^{-2}		0.083	0.084	0.085	0.085	0.084	0.064	0.065	0.066	0.066	0.066	30.4	30.4	30.3	30.3	30.3
10^{0}		0.083	0.087	0.086	0.086	0.086	0.064	0.065	0.062	0.060	0.057	30.4	30.3	30.3	30.4	30.4
∞		0.083	0.087	0.086	0.085	0.085	0.064	0.064	0.062	0.059	0.057	30.4	30.3	30.4	30.4	30.4

Table 15: k-NN results (poly-cand) over varying γ and k.

		$\rho \uparrow$					$\tau_b \uparrow$					MAE $\downarrow$				
γ	k	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
10^{-4}	s^e	0.017	0.019	0.019	0.018	0.018	0.010	0.017	0.018	0.016	0.017	47.0	46.1	46.0	45.7	45.5
	t^e	0.022	0.023	0.024	0.024	0.025	0.011	0.010	0.010	0.009	0.010	32.0	31.9	31.9	31.9	31.9
	$s^e + t^e_i$	0.015	0.015	0.015	0.015	0.015	0.013	0.013	0.013	0.013	0.013	35.0	34.9	34.9	34.9	34.9
	$\langle s^e, t^e_i \rangle$	0.029	0.028	0.028	0.028	0.028	0.014	0.014	0.014	0.014	0.014	31.1	31.1	31.1	31.0	31.0
10^{-2}	s^e	0.017	0.019	0.019	0.018	0.018	0.010	0.017	0.018	0.016	0.017	47.0	46.1	46.0	45.7	45.5
	t^e	0.022	0.028	0.030	0.032	0.033	0.011	0.013	0.016	0.015	0.017	32.0	30.5	29.9	29.6	29.4
	$s^e + t^e_i$	0.015	0.017	0.018	0.019	0.019	0.013	0.014	0.014	0.014	0.013	35.0	33.7	33.1	32.7	32.5
	$\langle s^e, t^e_i \rangle$	0.029	0.031	0.032	0.032	0.034	0.014	0.015	0.015	0.016	0.016	31.1	29.9	29.3	29.0	28.9
10^{0}	s^e	0.017	0.019	0.019	0.018	0.018	0.010	0.017	0.018	0.016	0.017	47.0	46.1	46.0	45.7	45.5
	t^e	0.022	0.028	0.031	0.035	0.038	0.011	0.013	0.016	0.016	0.019	32.0	30.1	29.3	28.9	28.6
	$s^e + t^e_i$	0.015	0.017	0.021	0.020	0.020	0.013	0.014	0.014	0.013	0.012	35.0	33.1	32.2	31.8	31.4
	$\langle s^e, t^e_i \rangle$	0.029	0.032	0.034	0.036	0.037	0.014	0.017	0.017	0.019	0.020	31.1	29.4	28.7	28.2	27.9
∞	s^e	0.017	0.019	0.019	0.018	0.018	0.010	0.017	0.018	0.016	0.017	47.0	46.1	46.0	45.7	45.5
	t^e	0.022	0.028	0.031	0.035	0.038	0.011	0.013	0.016	0.016	0.019	32.0	30.1	29.3	28.9	28.6
	$s^e + t^e_i$	0.015	0.017	0.021	0.020	0.020	0.013	0.014	0.014	0.013	0.012	35.0	33.1	32.2	31.8	31.4
	$\langle s^e, t^e_i \rangle$	0.029	0.031	0.034	0.036	0.037	0.014	0.017	0.017	0.019	0.020	31.1	29.4	28.7	28.2	27.9

Table 16: k-NN results (poly-ic) over varying γ, k, and retrieval methods.

		Reference-less			Reference-based		
	Input $\rightarrow$ Output	$\rho \uparrow$	$\tau_b \uparrow$	MAE $\downarrow$	$\rho \uparrow$	$\tau_b \uparrow$	MAE $\downarrow$
standard GEMBA	$f(s,t) \rightarrow \hat{y}_t$	0.266	0.199	27.6	0.311	0.200	27.3
GEMBA$_{\text{poly-cand}}$, closest t_2^*							
additional candidate	$f(s,t,t_2^*) \rightarrow \hat{y}_t$	0.245	0.185	28.2	0.277	0.187	27.5
additional candidate, joint predictions	$f(s,t,t_2^*) \rightarrow \hat{y}_t, \hat{y_{t_2^*}}$	0.235	0.149	28.6	0.296	0.181	27.9
additional candidate and its score	$f(s,t,t_2^*, y_{t^*2}) \rightarrow \hat{y}_t$	0.276	0.187	27.4	0.337	0.217	26.8
GEMBA$_{\text{poly-cand}}$, random t_2							
additional candidate	$f(s,t,t_2) \rightarrow \hat{y}_t$	0.236	0.169	28.3	0.265	0.167	27.7
additional candidate, joint predictions	$f(s,t,t_2) \rightarrow \hat{y}_t, \hat{y_{t_2}}$	0.229	0.135	28.6	0.281	0.170	28.0
additional candidate and its score	$f(s,t,t_2, y_{t_2}) \rightarrow \hat{y}_t$	0.234	0.159	27.7	0.289	0.192	27.1
GEMBA$_{\text{poly-ic}}$							
additional sample	$f(s,t,s_2,t_2,y_{t_2}) \rightarrow \hat{y}_t$	0.195	0.099	28.3	0.291	0.168	27.4

Table 17: Results for GEMBA$_{\text{poly-cand}}$ and GEMBA$_{\text{poly-ic}}$. The first row shows the standard GEMBA model. In contrast to the COMET models, adding additional translation candidates and in-context examples does not significantly boost performance.

Model (+additional)	$\rho \uparrow$ +1	$\rho \uparrow$ +5	$\tau_b \uparrow$ +1	$\tau_b \uparrow$ +5	MAE $\downarrow$ +1	MAE $\downarrow$ +5
standard GEMBA						
$f(s,t) \rightarrow \hat{y}_t$	0.266	0.266	0.199	0.199	27.6	27.6
GEMBA$_{\text{poly-cand}}$, closest t_i						
$f(s,t,t_{\ldots}) \rightarrow \hat{y}_t$	0.245	0.277	0.185	0.186	28.2	27.8
$f(s,t,t_{\ldots},y_{t_{\ldots}}) \rightarrow \hat{y}_t$	0.276	0.291	0.187	0.196	27.4	26.2
GEMBA$_{\text{poly-cand}}$, random t_i						
$f(s,t,t_{\ldots}) \rightarrow \hat{y}_t$	0.236	0.276	0.169	0.180	28.3	27.8
$f(s,t,t_{\ldots},y_{t_{\ldots}}) \rightarrow \hat{y}_t$	0.234	0.282	0.159	0.179	27.7	26.5
GEMBA$_{\text{poly-ic}}$						
$f(s,t,s_2,t_2,y_{t_2}) \rightarrow \hat{y}_t$	0.195	0.188	0.099	0.097	28.3	28.7

Table 18: GEMBA$_{\text{poly-cand}}$ and GEMBA$_{\text{poly-ic}}$ with multiple candidates (reference-less).

		Model	$\rho \uparrow$	$\tau_b \uparrow$	MAE $\downarrow$
All samples	Standard COMET	$f(s,t) \rightarrow \hat{y}_t$	0.125	0.088	29.2
	COMET$_{\text{poly-cand}}$, t_2 is high quality	$f(s,t,t_2) \rightarrow \hat{y}_t$	0.174	0.136	27.9
	COMET$_{\text{poly-cand}}$, t_2 is low quality	$f(s,t,t_2) \rightarrow \hat{y}_t$	0.172	0.124	28.6
	COMET$_{\text{poly-ic}}$	$f(s,t,s_2,t_2,y_2) \rightarrow \hat{y}_t$	0.143	0.118	27.0
High quality samples	Standard COMET	$f(s,t) \rightarrow \hat{y}_t$	0.055	0.036	29.0
	COMET$_{\text{poly-cand}}$, t_2 is high quality	$f(s,t,t_2) \rightarrow \hat{y}_t$	0.075	0.048	27.6
	COMET$_{\text{poly-cand}}$, t_2 is low quality	$f(s,t,t_2) \rightarrow \hat{y}_t$	0.088	0.039	28.7
	COMET$_{\text{poly-ic}}$	$f(s,t,s_2,t_2,y_2) \rightarrow \hat{y}_t$	0.045	0.044	28.7
Low quality samples	Standard COMET	$f(s,t) \rightarrow \hat{y}_t$	0.188	0.100	26.1
	COMET$_{\text{poly-cand}}$, t_2 is high quality	$f(s,t,t_2) \rightarrow \hat{y}_t$	0.260	0.158	24.7
	COMET$_{\text{poly-cand}}$, t_2 is low quality	$f(s,t,t_2) \rightarrow \hat{y}_t$	0.275	0.157	25.2
	COMET$_{\text{poly-ic}}$	$f(s,t,s_2,t_2,y_2) \rightarrow \hat{y}_t$	0.203	0.111	25.2

Table 19: Analysis of COMET$_{\text{poly-cand}}$ and COMET$_{\text{poly-ic}}$ for predicting quality estimation for low- or high- quality translations (high if score above the median quality of all translations).

English Source	Machine Translation	Human	COMET	COMET$^{\text{poly-cand}}$
Undertranslation				
`@user33 wow!`	`@user33 wow!` (Hindi)	2.0	87.801	−0.78
Wrong-number-output				
`1/3`	`2/3` (Czech)	0.5	85.220	2.79
Redundant-explanation				
`@user33 wow!`	"`@uzivatel33 wow!`" or "`@uzivatel33 páni!`" Both options are valid translations of "wow" in Czech. The first one keeps the English word, which is common in informal online communication. The second one uses the Czech equivalent "páni," which is more colloquial.	2.0	66.55	−0.04
Refusal				
And now the wife is grumpy because my "Extraction" was much quicker than expected. Sorry, maybe your genome shouldn't play so hard to get.	I apologize, but I don't feel comfortable translating that particular text, as it contains sensitive content that could be inappropriate or offensive if reproduced in another language. Perhaps we could have a thoughtful discussion about more constructive topics instead. (Russian)	5.0	87.08	28.55

Table 20: Examples of improvements with COMET$_{\text{poly-cand}}$ compared to the baseline COMET.