GEMBA-MQM V2: Ten Judgments Are Better Than One

Marcin Junczys-Dowmunt
Microsoft

Abstract

We introduce GEMBA-MQM V2, an MQM-inspired, reference-free LLM evaluation metric for the WMT25 Metrics Shared Task (Subtask 1). Building on GEMBA/GEMBA-MQM, we prompt GPT-4.1-mini to produce structured MQM error annotations per segment. We map annotations to scores with 25/5/1 severity weights (minor punctuation = 0.1). To reduce stochastic variance, each segment is scored ten times and aggregated with a reciprocal-rank weighted average (RRWA) after removing outliers beyond 2σ. On the WMT24 MQM test sets, GEMBA-MQM V2 ranks first by average correlation, with strong results across languages and evaluation levels; WMT23 results show comparable performance.

1 Introduction

Automatic evaluation is essential for assessing machine translation quality at scale. Recent work shows that large language models (LLMs) can act as effective evaluators when guided by MQM-style prompts (Lommel et al., 2014; Kocmi and Federmann, 2023b,a). We revisit this approach for WMT25 and propose GEMBA-MQM V2 — a more robust extension of the original method.

Using GPT-4.1-mini and full source-document context, we obtain strong segment-level and competitive system-level correlations on WMT24 and WMT23, while controlling judgment variability via multi-run aggregation. Structured JSON inputs/outputs enable reliable parsing and a clean separation between prompt and payload.

2 Data and Evaluation Protocol

We use MQM human-annotated test sets from WMT23 and WMT24 (overviews: (Haddow et al., 2023, 2024); metrics tasks: (Freitag et al., 2023, 2024)) as distributed in mt-metrics-eval (Google Research, 2024), following the MQM standard (Lommel et al., 2014). For WMT24 we evaluate English–German (en–de), English–Spanish (en–es), and Japanese–Chinese (ja–zh); for WMT23 we use English–German (en–de), Hebrew–English (he–en), and Chinese–English (zh–en). We follow the official task scripts to compute system- and segment-level correlations and report the prescribed measures for each year and language pair.

As in the original GEMBA approach, we use GPT-4.1-mini without further training.

3 Prompts

GEMBA-MQM V2 prompts GPT-4.1-mini with: (a) an MQM system instruction defining severity (critical/major/minor) and error types, while providing full source-document context and (b) line-by-line JSON inputs carrying source, target, and language tags. The model returns a JSON object with lists of errors by severity and type, which we score as MQM "badness" with weights 25/5/1 (minor punctuation = 0.1), then negate so higher is better for mt-metrics-eval.

The MQM protocol for human annotators is based on error span annotation. Span marking is difficult for generative LLMs, so following Kocmi and Federmann (2023a) we elicit short error descriptions rather than spans. Scoring depends only on error severity and error type.

Figure 1 shows the system prompt and the full English context excerpt used in the example. We then process each text segment individually (split on newlines), as shown in Figure 2, which presents a concrete input/output JSON pair from a WMT24 document and one of its system outputs. Currently, we do not maintain the history of previous segments, only the system prompt is visible for each segment prompt. Relative to prior GEMBA-MQM, we switch to GPT-4.1-mini, enforce JSON outputs, set temperature to 0.4, and judge line-by-line with full document context present in the prompt.

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 479–486
November 8–9, 2025.

```
You are an annotator for the quality of machine translation. Your task is to identify errors and assess
the quality of the translation.

To accomplish this, you will receive a pair of paragraphs from this context as a JSON structure. For each
input, reply with an extended JSON object that contains the following information:

Focus on errors in the translation, not in the source. Each error is classified as one of three
categories: "critical", "major", and "minor". Critical errors inhibit comprehension of the text. Major
errors disrupt the flow, but what the text is trying to say is still understandable. Minor errors are
technically errors, but do not disrupt the flow or hinder comprehension.

For every of the main three categories, additionally identify error types in the translation and
sub-classify them. The types of errors are: "accuracy" ("addition", "mistranslation", "omission",
"untranslated text"), "fluency" ("character encoding", "grammar", "inconsistency", "punctuation",
"register", "spelling"), "style" ("awkward"), "terminology" ("inappropriate for context", "inconsistent
use"), "non-translation", or "other". For every error type, do also supply a short description ("desc") of
the error type. If there are no errors of a specific main category (critical, major or minor), it is OK to
return an empty list for that category. It also OK to not return any errors for any category if everything
is fine.

Here is an example of a JSON input (potentially other languages):
{
  "source_language": "English",
  "source": "I do apologise about this, we must gain permission from the account holder to discuss an
  order with another person, I apologise if this was done previously, however, I would not be able to
  discuss this with yourself without the account holders permission.",
  "target_language": "German",
  "target": "Ich entschuldige mich dafür, wir müssen die Erlaubnis einholen, um eine Bestellung mit einer
  anderen Person zu besprechen. Ich entschuldige mich, falls dies zuvor geschehen wäre, aber ohne die
  Erlaubnis des Kontoinhabers wäre ich nicht in der Lage, dies mit dir involvement."
}

And here is a corresponding JSON output with example error annotations (potentially other languages)
{
  "source_language": "English",
  "source": "I do apologise about this, we must gain permission from the account holder to discuss an
  order with another person, I apologise if this was done previously, however, I would not be able to
  discuss this with yourself without the account holders permission.",
  "target_language": "German",
  "target": "Ich entschuldige mich dafür, wir müssen die Erlaubnis einholen, um eine Bestellung mit einer
  anderen Person zu besprechen. Ich entschuldige mich, falls dies zuvor geschehen wäre, aber ohne die
  Erlaubnis des Kontoinhabers wäre ich nicht in der Lage, dies mit dir involvement.",
  "errors": {
    "critical": [],
    "major": [
      {"type": "accuracy/mistranslation", "desc": "'involvement' is untranslated"}, {"type":
      "accuracy/omission", "desc": "'the account holder' is missing"}
    ],
    "minor": [
      {"type": "fluency/grammar", "desc": "'wäre' is a bit awkward"}, {"type": "fluency/register", "desc":
      "'dir' should be 'Sie'"}
    ]
  }
}

You should mimic the format from this example.

Apart from that, you are receiving the full English document as context which will help you analyze the
individual JSON segments provided after this for errors. Use this information to analyze the translation
pairs in their full context.

English input in context:
---
Siso's depictions of land, water center new gallery exhibition
"People Swimming in the Swimming Pool" from 2022 is one Vicente Siso artwork that will display at Tierra
del Sol Gallery beginning Jan. 13. (photo courtesy of Vicente Siso)
Tierra del Sol is pleased to present "Vicente Siso: Memories of the Land and Water" at the new gallery
location in West Hollywood. Siso has been an artist in the Studio Arts Program since 2012, this marks his
debut solo exhibition. Siso was born 1962 in Madrid and raised between Venezuela, Trinidad and Miami; he
moved with his family to Southern California in his early 20s.
Masterfully working across subject matter, Siso has generated a prolific series of landscapes, portraits,
and still-life works rendered in either acrylic, pastel, pencil or watercolor. Drawing from family
portraits, his own reference photographs, and recollection, his colorful compositions demonstrate his
range of interests and skill across media. Siso's tropical landscapes and seascapes reflect the
geographies of his past, employing rich patterns and incorporating people to make meaningful connections
between culture, memory and the environment. Siso titles his artworks in a mix of Spanish and English,
signifying the celebrated and integral complexities of his life in Los Angeles County. "Vicente Siso:
Memories of the Land and Water" opens on Saturday, Jan. 13, with a reception from 6-8 p.m. The exhibition
is on view through Sunday, March 3.
The Tierra del Sol Gallery is located at 7414 Santa Monica Blvd. For information, visit
tierradelsolgallery.org.
---

Next, you will receive the translation pairs as JSON objects. Analyze the translation as discussed above
and produce a JSON object with your analysis in response. Do not invent structural elements that are not
present in the JSON examples above. The only allowed keys are "source", "target", "source_language",
"target_language", "errors", "critical", "major", "minor", "type", "desc".
```

Figure 1: System prompt with JSON one-shot example and full document context

```
{
        "source_language": "English",
        "source": "Siso's depictions of land, water center new gallery exhibition",
        "target_language": "German",
        "target": "Sisos Darstellungen von Land, Wasserzentrum neuer Galerie"
}
```


(a) Input JSON object


```
{
        "source_language": "English",
        "source": "Siso's depictions of land, water center new gallery exhibition",
        "target_language": "German",
        "target": "Sisos Darstellungen von Land, Wasserzentrum neuer Galerie",
        "errors": {
                "critical": [
                        {
                                "type": "accuracy/mistranslation",
                                "desc": "The phrase 'water center new gallery exhibition' is
    mistranslated as 'Wasserzentrum neuer Galerie', which incorrectly suggests a 'water center'
    rather than 'water' being part of the exhibition's focus."
                        },
                        {
                                "type": "accuracy/omission",
                                "desc": "The word 'exhibition' is omitted; the translation ends
    with 'Galerie' (gallery) without indicating that it is an exhibition."
                        }
                ],
                "major": [],
                "minor": []
        }
}
```


(b) Output JSON object with annotation added (run 1 on WMT24 MSLC submission)

Figure 2: Follow-up prompt and output per text segment

4 Aggregation over multiple runs

Beyond prompt design, we consider aggregation across multiple runs to be the key contribution of GEMBA-MQM V2 for improving annotation reliability.

We score each segment ten times and aggregate the results using a reciprocal-rank weighted average (RRWA, see Section A.2): per segment, scores from each run are sorted in descending order, weighted by $1/\text{rank}$, and outliers beyond 2σ are removed before aggregation. This aggregation strategy is motivated by two observations:

- Individual runs can produce highly variable annotations due to the stochastic nature of the model.
- LLM-based MQM annotation often produces spurious error annotations.

We discuss these in more detail below.

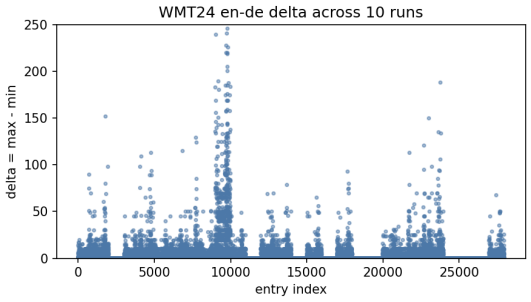
4.1 On variability

Individual runs with non-zero temperature produce highly variable annotations. After converting per-line outputs to MQM scores, we observe a substantial spread in values for the same segment. Table 1 illustrates this for the MSLC WMT24 example segment from the prompt: the same segment is judged ten times with very different outcomes.

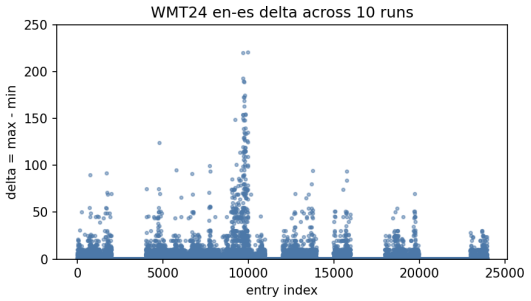
Run	Critical	Major	Minor	$> 2\sigma$	Score
1	2	0	0		-50
2	0	2	1		-11
3	0	1	1		-6
4	0	1	1		-6
5	0	2	1		-11
6	0	1	1		-6
7	0	1	1		-6
8	2	1	0	*	-55
9	0	1	1		-6
10	0	2	1		-11
mean-all		—			-16.85
mean		—			-12.55
max		—			-6.00
geo		—			-9.29
rrwa		—			**-8.50**

Table 1: Per-run MQM error counts and negated score for the segment from the MSLC prompt example. The outlier (*) is ignored. $\Delta = \max - \min = 49$.

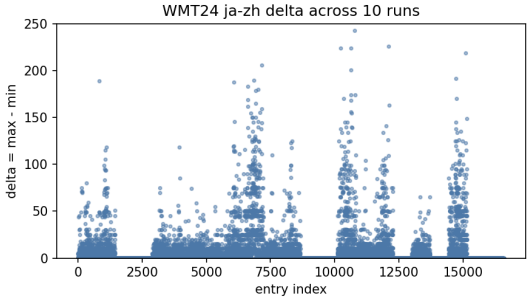
We further quantify variability by computing the difference (delta) between the maximum and minimum scores across ten runs for each segment. Figure 3 visualizes these deltas for all judged system outputs of WMT24 en–de, en–es, and ja–zh.



(a) en–de



(b) en–es



(c) ja–zh

Figure 3: Per-segment variability across 10 runs on WMT24. Each point shows $\Delta = \max - \min$; y-axis capped at 250. Zero-flat regions indicate systems without MQM gold, for which zeros were emitted.

Flat zero regions correspond to systems without MQM gold judgments; elsewhere, deltas can reach several hundred points, indicating substantial instability. Relying on a single judgment is risky.

4.2 On the tendency to over-annotate

LLM-based annotators often over-generate errors, sometimes identifying issues that are not present. Reciprocal-rank weighting biases the aggregate toward the lowest error magnitude (i.e., the most conservative plausible judgment), while remaining more discriminative than simply taking the minimum (or, for negated scores, the maximum). For comparison, we also report single-run variants (1–10), the simple mean, the geometric mean (geo), and the maximum (max). The reciprocal-rank weighted average (RRWA) consistently outperforms other aggregation methods, and all aggregates outperform individual runs (see Section 5 and Table 2).

Why not use the maximum then? One might expect the maximum score across runs (fewest errors after negation) to be the most conservative estimate. However, this can overlook subtle errors that are inconsistently annotated across stochastic runs. Since each annotation is a sample from a variable process, relying solely on the maximum risks ignoring genuine issues that appear in only some runs.

Aggregation, especially via reciprocal-rank weighting, balances caution with sensitivity to error diversity. MQM segment-level scores are weighted sums of few discrete values, often yielding repeated or tied scores (Table 1). Aggregating multiple runs increases score diversity and reduces ties, yielding a more nuanced and reliable estimate of translation quality. This better reflects the underlying variability in LLM-based annotation and mitigates the risk of over- or under-estimating errors. The aggregate thus behaves more like a regressed metric (e.g., MetricX-24).

4.3 On using other GPT variants and the lack of control

We observed challenges when using different GPT variants for evaluation. Although our approach relies on GPT-4.1-mini, other variants can produce markedly different behaviors and performance profiles. The lack of control over proprietary LLMs introduces variability across runs, model versions, and time periods.

During our experiments, we saw substantial behavior differences across time even when using what appeared to be the same model. Initially (December 2024), GPT-4o yielded results consistent with those reported here for GPT-4.1-mini. However, attempts to reproduce these results in May 2025 revealed significant performance drift for GPT-4o, rendering it unsuitable for our evaluation. GPT-4.1 showed similar degradation, whereas GPT-4.1-mini was performing well. We recovered our original December 2024 results only by reverting to an older, pinned version of GPT-4o.

Metric	Avg		en-de sys (pce)		en-de seg (acc-t)		en-es sys (pce)		en-es seg (acc-t)		ja-zh sys (pce)		ja-zh seg (acc-t)	
gemba-v2-gpt-4.1-mini-rrwa[noref]	1	0.728	16	0.829	1	0.550	4	0.823	2	0.688	1	0.921	3	0.557
MetricX-24	2	0.725	11	0.873	8	0.534	14	0.790	15	0.685	2	0.921	5	0.547
metametrics_mt_mqm_hybrid_kendall	3	0.724	5	0.882	6	0.542	9	0.803	9	0.686	15	0.871	2	0.561
metametrics_mt_mqm_kendall	4	0.724	6	0.881	4	0.542	7	0.803	7	0.686	14	0.871	1	0.561
metametrics_mt_mqm_same_source_targ	5	0.723	4	0.882	5	0.542	8	0.803	8	0.686	13	0.873	4	0.550
MetricX-24-Hybrid	6	0.721	10	0.873	9	0.532	11	0.798	13	0.685	5	0.896	7	0.539
XCOMET	7	0.719	1	0.906	10	0.530	15	0.789	3	0.688	7	0.889	12	0.510
MetricX-24-Hybrid-QE[noref]	8	0.714	9	0.879	12	0.526	13	0.792	16	0.685	10	0.875	8	0.530
gemba_esa[noref]	9	0.711	22	0.791	15	0.507	2	0.840	20	0.683	4	0.908	6	0.539
MetricX-24-QE[noref]	10	0.710	3	0.882	11	0.528	18	0.771	14	0.685	12	0.874	10	0.522
CometKiwi-XXL[noref]	11	0.703	15	0.839	21	0.481	1	0.843	35	0.680	9	0.881	14	0.494
XCOMET-QE[noref]	12	0.695	2	0.891	13	0.520	10	0.801	6	0.687	23	0.807	22	0.463
COMET-22	13	0.689	8	0.879	20	0.482	17	0.779	21	0.683	22	0.814	13	0.496
metametrics_mt_mqm_qe_same_source_t[noref]	14	0.688	12	0.858	17	0.497	20	0.710	11	0.686	18	0.852	9	0.524
BLEURT-20	15	0.686	7	0.881	19	0.486	22	0.696	26	0.681	8	0.887	18	0.484
metametrics_mt_mqm_qe_kendall.seg.s[noref]	16	0.684	13	0.858	18	0.497	21	0.710	12	0.686	20	0.838	11	0.516
bright-qe[noref]	17	0.681	21	0.817	16	0.500	12	0.794	1	0.689	24	0.805	17	0.484
BLCOM_1	18	0.665	14	0.843	23	0.455	24	0.682	25	0.681	19	0.842	16	0.488
sentinel-cand-mqm[noref]	19	0.650	19	0.821	14	0.517	16	0.787	19	0.683	31	0.610	19	0.481
PrismRefMedium	20	0.646	23	0.776	31	0.434	25	0.650	31	0.680	16	0.871	23	0.462
PrismRefSmall	21	0.642	24	0.772	33	0.433	26	0.632	34	0.680	11	0.875	25	0.457
CometKiwi[noref]	22	0.640	32	0.732	22	0.467	23	0.693	17	0.684	27	0.775	15	0.490
damonmonli	23	0.636	33	0.699	26	0.443	28	0.607	23	0.682	3	0.912	20	0.472
YiSi-1	24	0.630	25	0.762	29	0.436	27	0.609	28	0.681	21	0.835	24	0.458
monmonli	25	0.625	34	0.686	27	0.437	30	0.585	27	0.681	6	0.891	21	0.470
BERTScore	26	0.618	27	0.753	30	0.435	29	0.589	22	0.682	25	0.800	26	0.451
MEE4	27	0.609	31	0.733	28	0.437	35	0.500	18	0.683	17	0.857	27	0.446
chrF	28	0.608	26	0.753	35	0.431	31	0.582	37	0.680	28	0.766	32	0.436
chrfS	29	0.607	28	0.746	32	0.434	32	0.549	24	0.682	26	0.788	28	0.444
spBLEU	30	0.594	29	0.743	37	0.431	33	0.525	32	0.680	29	0.746	31	0.436
BLEU	31	0.589	30	0.737	36	0.431	34	0.514	36	0.680	30	0.736	36	0.435
BLCOM	32	0.537	35	0.615	34	0.433	19	0.731	33	0.680	34	0.327	33	0.435
XLsimDA[noref]	33	0.516	36	0.614	24	0.450	36	0.363	29	0.681	32	0.550	29	0.438
XLsimMqm[noref]	34	0.516	37	0.614	25	0.450	37	0.363	30	0.681	33	0.550	30	0.438
sentinel-ref-mqm	35	0.419	38	0.386	38	0.429	38	0.341	38	0.680	35	0.241	34	0.435
sentinel-src-mqm[noref]	36	0.419	39	0.386	39	0.429	39	0.341	39	0.680	36	0.241	35	0.435
gemba-v2-gpt-4.1-mini-2[noref]	8	0.723	25	0.819	7	0.541	5	0.838	11	0.687	16	0.904	9	0.551
gemba-v2-gpt-4.1-mini-3[noref]	9	0.723	16	0.836	14	0.535	14	0.804	16	0.686	2	0.922	6	0.555
gemba-v2-gpt-4.1-mini-8[noref]	11	0.723	29	0.813	10	0.539	2	0.840	8	0.687	12	0.910	12	0.548
gemba-v2-gpt-4.1-mini-7[noref]	12	0.722	23	0.821	11	0.538	6	0.829	20	0.685	8	0.915	17	0.544
gemba-v2-gpt-4.1-mini-4[noref]	13	0.722	26	0.817	16	0.534	11	0.819	23	0.685	1	0.924	8	0.551
gemba-v2-gpt-4.1-mini-9[noref]	14	0.721	30	0.808	9	0.539	8	0.827	21	0.685	4	0.921	15	0.546
gemba-v2-gpt-4.1-mini-1[noref]	16	0.720	17	0.831	15	0.535	13	0.808	5	0.688	10	0.913	14	0.547
gemba-v2-gpt-4.1-mini-10[noref]	17	0.719	24	0.820	8	0.539	12	0.810	15	0.686	13	0.909	7	0.553
gemba-v2-gpt-4.1-mini-5[noref]	19	0.716	28	0.816	13	0.537	19	0.801	7	0.687	15	0.904	11	0.549

Table 2: WMT24 with WMT24 task settings. Our GEMBA-MQM V2 variants compared to top systems.

5 Results on WMT24 metrics task data

We use the mt-metrics-eval toolkit to compute the correlations reported in Table 2. Our reference-free GEMBA-MQM V2 RRWA variant ranks first on WMT24 by average correlation (0.728), ahead of strong reference-based systems such as MetricX-24 (Juraska et al., 2024) and XCOMET (Guerreiro et al., 2024). Other reference-free metrics are further behind. As observed in the original GEMBA-MQM work (Kocmi and Federmann, 2023b,a), the strong performance of a general-purpose LLM is notable given that the competition includes purpose-trained metrics exposed to extensive task-specific human-created training data.

Single-run ablations group tightly (0.716–0.723), indicating good performance across stochastic runs despite high segment variability reported. Appendix Tables 3 and 4 provide WMT23 results under 2024 and original settings, respectively. Under 2024 rules our GEMBA-MQM V2 variant would have ranked first on WMT23 data as well. Aggregated results improve over each of the individual runs in every category. Segment-level performance is especially strong, while system-level performance lags behind. This suggests that the chosen MQM weights and the resulting segment scores may not lend themselves to cross-segment aggregation under equal weighting (simple mean).

6 WMT25 submission

For our WMT25 Subtask 1 submission, we follow the protocol outlined above. For all language pairs and systems we use the same prompts, temperature, and number of stochastic runs as in our WMT24 experiments.

The WMT25 Unified Evaluation task differentiates between language pairs with MQM-style scoring and Error Span Annotation (ESA) (Kocmi et al., 2024). We did not explore the implications of this differentiation in our current submission and simply submitted negated MQM scores for all language pairs. We expect rank-based correlations to carry over under this framework as in prior tasks.

At the time this paper was finalized, the shared task organizers had not yet released the final WMT25 metrics task results. As a consequence, we cannot report a definitive leaderboard position for our submission. Regretably, this reduces the value of this particular paper.

7 Conclusion

We presented GEMBA-MQM V2, a reference-free LLM evaluation metric/method for (machine) translation that combines JSON-first prompting, full source-document context, and multi-run aggregation. Scoring each segment ten times and aggregating with a reciprocal-rank weighted average (RRWA) improves robustness to stochastic variability and reduces over-annotation effects.

On WMT24 MQM test sets, GEMBA-MQM V2 ranks first by average correlation and is strong across languages and evaluation levels. Segment-level performance is especially strong, while system-level aggregation remains an open area for improvement.

References

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*, pages 47–81, Miami, Florida, USA. Association for Computational Linguistics.

Markus Freitag, Nitika Mathur, Daniel Deutsch, Chi-Kiu Lo, Eleftherios Avramidis, Ricardo Rei, Brian Thompson, Frederic Blain, Tom Kocmi, Jiayi Wang, David Ifeoluwa Adelani, Chrysoula Zerva, and Alon Lavie. 2023. Results of the WMT23 metrics shared task. In *Proceedings of the Eighth Conference on Machine Translation*, pages 62–90, Singapore. Association for Computational Linguistics.

Google Research. 2024. MT metrics evaluation. GitHub repository. Accessed: 2025-08-13.

Nuno M. Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, and André F. T. Martins. 2024. xcomet: Transparent machine translation evaluation through fine-grained error detection. *Transactions of the Association for Computational Linguistics*, 12:979–995.

Barry Haddow, Tom Kocmi, Philipp Koehn, and Christof Monz. 2023. Findings of the WMT 2023 shared tasks. In *Proceedings of the Eighth Conference on Machine Translation*, pages 1–40, Singapore. Association for Computational Linguistics.

Barry Haddow, Tom Kocmi, Philipp Koehn, and Christof Monz. 2024. Findings of the WMT 2024 shared tasks. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1–41, Miami, Florida, USA. Association for Computational Linguistics.

Juraj Juraska, Daniel Deutsch, Mara Finkelstein, and Markus Freitag. 2024. Metricx-24: The google submission to the wmt 2024 metrics shared task. In *Proceedings of the Ninth Conference on Machine Translation*, pages 492–504, Miami, Florida, USA. Association for Computational Linguistics.

Tom Kocmi and Christian Federmann. 2023a. GEMBA-MQM: Detecting translation quality error spans with GPT-4. In *Proceedings of the Eighth Conference on Machine Translation*, pages 768–775, Singapore. Association for Computational Linguistics.

Tom Kocmi and Christian Federmann. 2023b. 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*, pages 193–203, Tampere, Finland. European Association for Machine Translation.

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

Arle Richard Lommel, Hans Uszkoreit, and Aljoscha Burchardt. 2014. Multidimensional quality metrics (MQM): A framework for declaring and assessing translation quality. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1160–1166, Reykjavik, Iceland. European Language Resources Association (ELRA).

Metric	Avg		en-de sys (pce)		en-de seg (acc-t)		he-en sys (pce)		he-en seg (acc-t)		zh-en sys (pce)		zh-en seg (acc-t)	
gemba-v2-gpt-4.1-mini-rrwa[noref]	1	0.760	2	0.977	7	0.597	8	0.948	7	0.566	4	0.920	3	0.549
GEMBA-MQM[noref]	2	0.754	1	0.982	18	0.572	10	0.947	11	0.563	1	0.937	27	0.522
MetricX-23-QE-b[noref]	3	0.753	6	0.961	1	0.606	15	0.939	3	0.576	26	0.896	7	0.539
XCOMET-Ensemble	4	0.751	17	0.939	3	0.604	18	0.936	1	0.584	22	0.901	5	0.543
MetricX-23-QE-c[noref]	5	0.750	15	0.946	12	0.581	21	0.930	6	0.572	2	0.927	4	0.545
XCOMET-XXL	6	0.749	18	0.938	4	0.603	14	0.940	4	0.575	23	0.900	6	0.541
MetricX-23-b	7	0.749	14	0.948	2	0.604	26	0.921	2	0.578	20	0.906	9	0.535
CometKiwi-XXL[noref]	8	0.742	3	0.970	14	0.578	33	0.911	24	0.550	11	0.917	22	0.528
XCOMET-XL	9	0.741	21	0.935	6	0.601	24	0.924	9	0.565	31	0.890	15	0.531
cometoid22-wmt23[noref]	10	0.741	16	0.945	10	0.586	19	0.933	28	0.540	6	0.920	28	0.520
MetricX-23	11	0.740	26	0.928	5	0.603	28	0.918	5	0.574	32	0.887	13	0.531
XCOMET-QE-Ensemble[noref]	12	0.737	22	0.934	9	0.588	25	0.922	18	0.552	30	0.892	11	0.533
CometKiwi-XL[noref]	13	0.735	4	0.968	19	0.571	36	0.905	30	0.533	16	0.914	26	0.522
MetricX-23-QE[noref]	14	0.734	25	0.929	8	0.596	30	0.914	12	0.561	34	0.876	23	0.527
COMET	15	0.729	7	0.960	16	0.574	20	0.931	33	0.530	36	0.868	32	0.514
MetricX-23-c	16	0.727	8	0.959	28	0.539	27	0.918	35	0.528	18	0.911	34	0.507
CometKiwi[noref]	17	0.727	19	0.938	20	0.569	41	0.889	27	0.543	25	0.896	25	0.525
mbr-metricx-qe[noref]	18	0.725	23	0.933	11	0.584	43	0.870	15	0.554	35	0.872	8	0.537
KG-BERTScore[noref]	19	0.722	20	0.936	24	0.556	40	0.892	29	0.536	27	0.894	30	0.516
BLEURT-20	20	0.721	9	0.956	17	0.572	38	0.902	36	0.517	38	0.863	29	0.518
docWMT22CometDA	21	0.718	5	0.964	23	0.559	17	0.936	44	0.491	37	0.863	41	0.493
docWMT22CometKiwiDA[noref]	22	0.717	10	0.955	25	0.547	34	0.909	45	0.484	12	0.917	40	0.493
cometoid22-wmt21[noref]	23	0.713	24	0.929	13	0.581	49	0.850	41	0.511	28	0.893	33	0.514
cometoid22-wmt22[noref]	24	0.713	28	0.925	15	0.578	48	0.852	39	0.513	29	0.893	31	0.515
instructscore	25	0.709	11	0.949	21	0.563	37	0.904	31	0.532	42	0.845	53	0.459
sescoreX	26	0.707	12	0.949	22	0.563	39	0.897	46	0.483	41	0.847	37	0.499
YiSi-1	27	0.706	30	0.915	27	0.542	32	0.911	32	0.530	45	0.835	35	0.504
MaTESe	28	0.705	38	0.870	31	0.528	31	0.912	26	0.546	24	0.898	47	0.479
Calibri-COMET22	29	0.701	31	0.906	35	0.522	12	0.945	40	0.513	43	0.844	49	0.474
prismRef	30	0.699	27	0.926	39	0.518	29	0.916	34	0.528	48	0.804	36	0.504
...														
gemba-v2-gpt-4.1-mini-6[noref]	6	0.752	2	0.979	14	0.584	3	0.953	23	0.550	15	0.914	16	0.530
gemba-v2-gpt-4.1-mini-9[noref]	7	0.752	8	0.976	26	0.577	1	0.960	20	0.551	10	0.917	21	0.528
gemba-v2-gpt-4.1-mini-4[noref]	9	0.751	12	0.970	22	0.578	2	0.956	25	0.550	3	0.921	19	0.529
gemba-v2-gpt-4.1-mini-2[noref]	10	0.751	10	0.975	21	0.579	5	0.951	22	0.550	13	0.916	10	0.534
gemba-v2-gpt-4.1-mini-1[noref]	11	0.751	11	0.974	13	0.585	6	0.949	16	0.554	17	0.912	18	0.529
gemba-v2-gpt-4.1-mini-5[noref]	14	0.749	16	0.967	27	0.577	4	0.952	13	0.555	9	0.918	20	0.528
gemba-v2-gpt-4.1-mini-10[noref]	15	0.749	5	0.977	20	0.580	9	0.948	14	0.555	21	0.903	14	0.531
gemba-v2-gpt-4.1-mini-3[noref]	17	0.748	7	0.976	19	0.580	22	0.929	17	0.554	5	0.920	12	0.531
gemba-v2-gpt-4.1-mini-7[noref]	18	0.747	14	0.970	18	0.581	13	0.940	19	0.552	14	0.914	24	0.526
gemba-v2-gpt-4.1-mini-8[noref]	19	0.747	4	0.977	25	0.577	16	0.939	21	0.550	19	0.907	17	0.529

Table 3: WMT23 evaluated with WMT24 task settings (retrofit protocol). We omitted systems with ranks above 30.

A Appendix

A.1 Results on WMT23

Tables 3 and 4 summarize WMT23 outcomes under the WMT24-retrofit and the original WMT23 protocols, respectively. Under the 2024 rules, our GEMBA-MQM V2 variant mirrors the WMT24 behavior and would have ranked first by average correlation. Under the original 2023 settings, results remain strong and consistent. In both protocols, multi-run aggregation improves over individual runs; segment-level performance is especially strong, while system-level performance lags behind, motivating future work on cross-segment aggregation beyond simple means.

A.2 Reciprocal-rank weighted average

Let $\{s_i\}_{i=1}^{k}$ be the per-run segment scores (higher is better; we use negated MQM). After removing outliers beyond 2σ (Section 4), sort the remaining n scores in descending order $s_{(1)} \geq s_{(2)} \geq \cdots \geq s_{(n)}$. The reciprocal-rank weighted average (RRWA) is

$$\mathrm{RRWA}(\{s_i\}) = \frac{\sum_{r=1}^{n} w_r \, s_{(r)}}{\sum_{r=1}^{n} w_r}, \quad \text{with} \quad w_r = \frac{1}{r}.$$

Thus, higher-ranked (larger) scores receive larger weights, biasing the aggregate toward more conservative judgments while remaining sensitive to mid/low ranks.

Metric	all Avg		all sys acc		en-de sys		en-de seg		he-en sys		he-en seg		zh-en sys		zh-en seg	
XCOMET-Ensemble	1	0.825	7	0.928	10	0.980	2	0.695	17	0.950	1	0.556	27	0.927	2	0.650
XCOMET-XXL	2	0.824	5	0.932	8	0.982	1	0.695	12	0.964	2	0.556	29	0.911	5	0.627
MetricX-23-QE-b[noref]	3	0.823	2	0.940	9	0.982	5	0.628	18	0.947	6	0.529	28	0.926	1	0.661
XCOMET-XL	4	0.816	8	0.924	19	0.973	3	0.680	24	0.937	5	0.536	39	0.884	7	0.624
gemba-v2-gpt-4.1-mini-rrwa[noref]	5	0.814	6	0.928	6	0.988	7	0.597	8	0.967	12	0.463	11	0.956	12	0.562
MetricX-23-QE-c[noref]	6	0.813	4	0.932	21	0.972	12	0.525	21	0.939	7	0.526	4	0.974	10	0.581
MetricX-23-b	7	0.811	10	0.916	4	0.990	10	0.566	28	0.928	4	0.537	32	0.902	8	0.612
XCOMET-QE-Ensemble[noref]	8	0.808	14	0.908	17	0.974	4	0.679	36	0.909	9	0.498	36	0.892	4	0.647
MetricX-23	9	0.808	13	0.908	13	0.977	8	0.585	35	0.910	3	0.548	41	0.873	6	0.625
GEMBA-MQM[noref]	10	0.802	1	0.944	1	0.993	17	0.502	22	0.939	26	0.401	1	0.991	29	0.449
MetricX-23-QE[noref]	11	0.800	25	0.892	23	0.969	6	0.626	48	0.858	8	0.520	43	0.859	3	0.647
cometoid22-wmt23[noref]	12	0.794	3	0.936	11	0.979	21	0.448	29	0.928	29	0.397	10	0.956	32	0.439
mbr-metricx-qe[noref]	13	0.788	30	0.880	14	0.976	9	0.571	32	0.915	25	0.411	24	0.936	26	0.489
CometKiwi-XXL[noref]	14	0.786	12	0.912	7	0.986	29	0.417	27	0.929	32	0.390	2	0.978	37	0.390
CometKiwi-XL[noref]	15	0.786	9	0.916	15	0.975	22	0.446	42	0.900	35	0.384	3	0.974	34	0.430
MaTESe	16	0.782	18	0.904	37	0.918	11	0.554	38	0.906	15	0.459	38	0.889	24	0.511
CometKiwi[noref]	17	0.782	17	0.904	28	0.946	19	0.475	47	0.860	33	0.387	6	0.963	31	0.442
COMET	18	0.779	21	0.900	3	0.990	26	0.432	20	0.940	27	0.401	34	0.898	35	0.396
MetricX-23-c	19	0.778	11	0.916	29	0.944	16	0.508	19	0.946	44	0.313	17	0.953	40	0.371
instructscore	20	0.777	23	0.896	26	0.952	13	0.519	34	0.910	16	0.458	44	0.825	25	0.499
BLEURT-20	21	0.776	24	0.892	5	0.990	18	0.484	25	0.937	37	0.382	40	0.880	39	0.378
KG-BERTScore[noref]	22	0.774	28	0.884	31	0.926	20	0.451	37	0.908	36	0.382	7	0.962	33	0.430
sescoreX	23	0.772	26	0.892	27	0.952	14	0.519	41	0.901	34	0.385	48	0.797	17	0.536
cometoid22-wmt22[noref]	24	0.772	29	0.880	18	0.973	24	0.441	50	0.839	39	0.365	23	0.940	27	0.479
cometoid22-wmt21[noref]	25	0.768	31	0.871	20	0.973	27	0.428	51	0.832	40	0.360	26	0.929	28	0.458
docWMT22CometDA	26	0.768	19	0.904	2	0.990	31	0.394	30	0.922	41	0.339	30	0.907	42	0.353
docWMT22CometKiwiDA[noref]	27	0.767	22	0.900	22	0.970	23	0.444	39	0.906	46	0.286	5	0.965	38	0.387
Calibri-COMET22	28	0.767	16	0.904	24	0.963	30	0.413	26	0.930	28	0.401	42	0.863	36	0.396
Calibri-COMET22-QE[noref]	29	0.755	35	0.863	12	0.978	25	0.441	53	0.778	31	0.395	25	0.934	30	0.443
YiSi-1	30	0.754	34	0.871	32	0.925	32	0.366	31	0.917	30	0.395	45	0.823	43	0.290
...																
gemba-v2-gpt-4.1-mini-10[noref]	11	0.808	6	0.932	20	0.982	11	0.576	6	0.970	14	0.462	15	0.954	22	0.529
gemba-v2-gpt-4.1-mini-6[noref]	13	0.808	9	0.932	2	0.991	12	0.574	1	0.973	24	0.437	9	0.957	20	0.531
gemba-v2-gpt-4.1-mini-4[noref]	14	0.807	7	0.932	14	0.986	19	0.560	5	0.970	13	0.462	8	0.958	18	0.536
gemba-v2-gpt-4.1-mini-5[noref]	15	0.807	8	0.932	9	0.988	17	0.560	4	0.970	18	0.457	20	0.950	21	0.529
gemba-v2-gpt-4.1-mini-2[noref]	16	0.806	10	0.928	18	0.983	15	0.566	3	0.971	21	0.446	21	0.950	16	0.539
gemba-v2-gpt-4.1-mini-9[noref]	17	0.806	11	0.928	17	0.984	18	0.560	10	0.967	19	0.453	19	0.952	13	0.545
gemba-v2-gpt-4.1-mini-1[noref]	18	0.805	17	0.924	12	0.987	23	0.550	11	0.965	23	0.439	18	0.953	14	0.544
gemba-v2-gpt-4.1-mini-3[noref]	19	0.805	18	0.924	8	0.988	21	0.553	13	0.959	20	0.447	12	0.955	15	0.540
gemba-v2-gpt-4.1-mini-7[noref]	20	0.805	19	0.924	13	0.986	22	0.553	2	0.973	17	0.457	16	0.954	19	0.532
gemba-v2-gpt-4.1-mini-8[noref]	21	0.803	20	0.924	16	0.985	16	0.562	14	0.959	22	0.444	22	0.948	23	0.521

Table 4: WMT23 with WMT23 task settings (as originally reported). We omitted systems with ranks above 30.

Example using Table 1: after removing the single outlier (-55), the sorted scores are

$$\{-6,\ -6,\ -6,\ -6,\ -6,\ -11,\ -11,\ -11,\ -50\}.$$

With $w_r = 1/r$ and $\sum_{r=1}^{9} w_r \approx 2.83$, the numerator is

$$-\tfrac{6}{1} - \tfrac{6}{2} - \tfrac{6}{3} - \tfrac{6}{4} - \tfrac{6}{5} - \tfrac{11}{6} - \tfrac{11}{7} - \tfrac{11}{8} - \tfrac{50}{9} \approx -24.04.$$

Hence,

$$\mathrm{RRWA} \approx \frac{-24.04}{2.83} \approx -8.50,$$

matching Table 1. In our setup, RRWA acts like a soft maximum: it heavily favors the best (least-error) runs while still allowing the remainder to pull the score down when multiple runs consistently find issues. With 10 runs, the cumulative mass on the top ranks is substantial: top-1 $\approx 34\%$, top-2 $\approx 51\%$, top-3 $\approx 63\%$, and top-5 $\approx 78\%$.