

Phrase at WMT26 Automated MT Evaluation Task

Alan Eckhardt, Ondřej Glembek, Craig Stewart*

Phrase a.s.

{first.last}@phrase.com

Abstract

We describe the Phrase submission to WMT26’s shared task on automated MT quality evaluation, covering all three of its tasks: segment-level error detection and span annotation (T1), segment-level quality score prediction (T2), and detection of error-free segments (T3). Treating independent LLMs as a panel of ESA (Error Span Annotation)-style error-span judges, we measure a ceiling: a perfect filter over the panel’s pooled spans would raise MPP-F (our span-matching F-score) by roughly two thirds over the strongest single judge. An exhaustive search for how to realize that headroom (diversity, rank-weighted aggregation, a per-judge reliability prior, and boundary refinement) never beats the strongest single judge on MPP-F (T1), because the judges’ shared recall is too low for any gold-free span selection to close the gap. The two tasks derived from the same spans behave differently: a static reliability prior wins the segment score (T2), and simple agreement wins the binary error-free decision (T3). Our primary submission, Phrase01, follows this split; a secondary submission, Phrase02, forces in a third judge (UFAL’s cat-v4) across all tasks as an explicit ablation, despite evidence it does not help. The lever for span-level detection is annotation coverage through generation, not aggregation; for the derived tasks, it is matching the aggregation mechanism to the metric.

1 Introduction

Ensembling large language model (LLM) judges is widely assumed to improve automated evaluation, on the intuition that independent judges cancel one another’s errors (Verga et al., 2024). We test that assumption thoroughly for span-level machine translation (MT) quality evaluation and find it largely does not hold under the shared task’s primary metric, for a reason that is specific and measurable.

*Authors are listed in alphabetical order and contributed equally.

We separate two questions usually conflated. First, how much could a panel (see Sec. 4.1 for definition) help if we combined it perfectly. Second, how much does it help with a realizable, gold-free combination. The gap between the two is the story: the ceiling is high, the realizable gain is zero or negative, and the difference is a recall wall.

The target adds a second reason to expect the wall. Human Error Span Annotation (ESA) has limited recall and limited inter-annotator span agreement (Kocmi et al., 2024). The gold each system is scored against is closer to what one annotator reacted to than to the full set of errors present, so part of the coverage the panel appears to miss is coverage the reference never had. Predicting error, under this metric, is partly predicting where a human reacts to it.

2 Related Work

Human evaluation of machine translation has converged on error-based protocols. The Multidimensional Quality Metrics (MQM) framework (Lommel et al., 2014) defines the severity and error-type taxonomy, and Freitag et al. (2021) established MQM as the standard for MT human evaluation by showing that expert error annotation reranks systems differently from crowd ratings. The shared task uses Error Span Annotation (Kocmi et al., 2024), which keeps MQM-style severity-marked spans but simplifies the protocol; the same work reports the limited inter-annotator recall and span agreement that our analysis relies on when it argues that the gold reflects one annotator’s reactions rather than the full set of errors.

Using LLMs to produce these annotations is now an established line. GEMBA showed a general-purpose LLM is a strong translation-quality evaluator (Kocmi and Federmann, 2023b), GEMBA-MQM extended this to detecting MQM error spans (Kocmi and Federmann, 2023a), and AutoMQM

prompts an LLM to identify and categorise errors, yielding spans that align with human annotation without a reference (Fernandes et al., 2023). Our judges follow this single-model span-annotation lineage. Closely related, concurrent work computes oracle upper bounds for a single model’s error span detection and cautions that such bounds are optimistic when measured against a single annotator’s gold (Lyu et al., 2025) (a caveat our own oracle analysis shares (Sec. 4.3)), though we bound a panel’s pooled spans rather than one model’s sampled candidates. Learned neural metrics are the competing baselines: COMET (Rei et al., 2020) and its fine-grained successor xCOMET, which adds error-span detection (Guerreiro et al., 2024), alongside regression metrics such as MetricX-24 (Juraska et al., 2024). The WMT metrics shared tasks track how these compare (Freitag et al., 2023, 2024), with the WMT24 results noting that LLM-based metrics increasingly rival purpose-trained ones.

Our contribution concerns aggregating multiple judgments, where two strands are relevant. The first resamples a single stochastic judge: GEMBA-MQM V2 scores each segment ten times and combines the runs with a reciprocal-rank weighted average (RRWA) after outlier removal, motivated by run-to-run variance and a tendency to over-annotate (Junczys-Dowmunt, 2025); this generalises the self-consistency idea (Wang et al., 2023). We adopt its RRWA aggregation and test it on both self-consistency and heterogeneous panels.

The second strand uses panels of distinct models. Wang et al. (2024) show that a single LLM judge is biased and unstable. Motivated by this, Verga et al. (2024) propose a Panel of LLM Evaluators and report that a jury of smaller, diverse models aligns better with human judgment, reduces intra-model bias, and costs less than one large judge. That result is for scoring free-form generations. Our finding is that for span-level MT evaluation under a severity-weighted matching metric the panel advantage does not carry over on the primary localization metric, because the judges’ shared recall is low; the jury still helps the derived binary error-free classification, which is consistent with where panels are known to help.

3 Task and Data

We participate in all three tasks (Span annotation (T1), Segment score (T2), and error-free classifica-

tion (T3)) on the eight minimum-required language pairs: cs-de, cs-vi, en-ar, en-et, en-id, en-is, en-se, and zh-jia. For T1, systems produce ESA-style annotations over the MT hypothesis: character-offset error spans, each labelled minor or major, with the omissions field left blank (null). T1 submissions are scored with the official severity-weighted span-matching metric, match with partial overlap and partial credit (MPP) (Perrella et al., 2026): a one-to-one optimal matching between hypothesis and gold spans, credited by proportion of overlapping characters and reduced when predicted and gold severities disagree, pooled via micro-averaging, with severity weights 0.25 for minor and 1.0 for major. We report the resulting F-score as MPP-F throughout.

For development and all analysis we use two independent dev sets, both defined by UFAL (Charles University’s Institute of Formal and Applied Linguistics) (Hrabal and Bojar, 2026): Dev1 and their final version, Dev2. Dev1 has 2,032 segments across the 5 of 8 priority pairs that have gold (cs-de, cs-uk, en-ar, en-et, en-is)—cs-uk, not cs-vi; see Sec. 6 for why the two differ—and is the dev set used throughout Sec. 5. Dev2 is UFAL’s final dev set and has a narrower Gemini/Gemma quality gap than Dev1. It is drawn from the same 5 of 8 priority pairs (cs-de, cs-uk, en-ar, en-et, en-is), out of a 14-language-pair, 3,058-row snapshot before priority filtering. We use it in Sec. 6 both to evaluate the cat-v4 fusion (UFAL’s own system, Sec. 4.1) and, for consistency, to re-validate the core Gemini+Gemma strategy there. The other 3 priority pairs (zh-jia, en-id, en-se) are new for WMT26 and have no dev gold in either set, so every choice for them rests on a signal that transfers without gold. We report the pooled micro-average as primary, matching the official leaderboard’s own micro-averaging (Perrella et al., 2026), and the segment-weighted average as a secondary check.

4 Methodology

4.1 The Panel

By *panel* we mean a set (bag) of independent LLM judges; its exact membership varies by experiment.

LLM judges are prompted with an ESA-style instruction including the shared-task guidelines; each returns error spans with severities, parsed to a normalised character-span annotation. Exact prompt wording is proprietary and not reproduced here; at a high level, each judge is instructed to classify

every error’s severity as major or minor and return the exact substring of the translation each error span covers. The core panel is three independent commercial models: Gemini 3.5-flash, Claude Sonnet 4.6, and GPT-5.6-luna. We also evaluate local open-weight judges (Qwen3.5-9B and Gemma-4-12B, via vLLM) and an external MQM system, UFAL’s cat-v4 (Hrabal and Bojar, 2026): a two-stage MQM-style error detector built on the open-weight Gemma 4 31B instruction model, 4-bit activation-aware weight quantization (AWQ), using few-shot prompting and one-pass self-refinement to predict optional MQM categories from the source and candidate translation alone, without reference translations.

4.2 Combination Strategies

We evaluated a wide family so the negative result does not rest on one design:

- **Per-segment routing:** for each segment, defer to whichever single judge is expected to be most reliable there, instead of one fixed judge for every segment. We only evaluated this as an oracle ceiling (Tab. 2, using gold to pick per segment); we did not pursue a realizable version, continuing instead with the character-level consensus approach below.
- **Character-level vote-depth consensus:** tally, per character, how many judges flag it as part of an error, and threshold on that count to decide which characters to keep.
- **Gemini-anchored doubly-corroborated top-up:** keep all of Gemini’s own spans and add only spans the other two judges both independently flag but Gemini missed.
- **Severity relabelling and escalation:** override a span’s severity using the panel’s aggregate judgment, e.g. escalating to major when enough judges agree it is severe.
- **Directed prompt-framing diversity:** generate additional judges by re-prompting under different framings (recall-leaning, human-reaction, criticality) rather than relying on model diversity alone.
- **Fusion of the external cat-v4 system:** add UFAL’s system into the panel’s pooled vote alongside the in-house judges.

- **Self-consistency resampling:** repeatedly sample a single judge at non-zero temperature and aggregate the resulting spans, treating run-to-run variance itself as a diversity source. This is also what covers undirected temperature diversity, as distinct from the directed framings above.
- **Reciprocal-rank weighted aggregation (RRWA):** weight each contributor’s spans (or each resample’s) by its rank among the panel on that segment, rather than voting all members equally; used both as a scalar score combiner and transplanted into span voting.

4.3 Oracles as Measurement Instruments

To bound the approach we compute oracles that use gold at selection time. These are not submittable systems; they measure how much of the score is in principle recoverable from the panel’s existing spans: an oracle span-subset (keep only pooled spans optimal matching pairs with gold), oracle per-segment routing, oracle severity assignment, and their join.

We compute this ceiling separately rather than simply tuning combination methods for the best dev MPP-F, because a dev-tuned score alone is ambiguous: it cannot distinguish a method that is near-optimal (little headroom exists) from one that is failing to realize headroom that is actually there. Dev tuning only ever selects the best of finitely many fixed, gold-free rules, applied uniformly at test time; it can reweight the panel’s own visible signals (severity, length, agreement) but cannot recover information (which specific spans are correct) that those signals do not carry. The oracle, by using gold per decision rather than in aggregate, measures what is recoverable if that missing information were available, giving a ceiling that no realizable rule can reach.

4.4 Experimental Protocol

Two ESA prompt variants are used across this paper: a baseline prompt, used for all dev-set validation and method search (Sec. 5), and a revised, directive-style prompt adopted for the real WMT26 submission after later prompt-iteration work found it transfers more reliably across judges (Sec. 6). Results under the two are not directly comparable, since T2’s best method changes between them.

All aggregation analysis runs on cached judge annotations with no further model calls; only the

Judge	T1	T2	T3
Gemini 3.5-flash	0.311	0.599	0.248
Claude Sonnet 4.6	0.273	0.535	0.223
GPT-5.6-luna	0.274	0.536	0.224
Gemma-4-12B (local)	0.228	0.220	0.061
Qwen3.5-9B (local)	0.205	0.264	0.071

Table 1: Single-judge dev results (pooled). Metrics for T1, T2 and T3 are MPP-F, Pearson, and Matthews correlation coefficient (MCC), respectively.

Oracle	MPP-F	Δ
Gemini baseline	0.311	—
oracle span-subset	0.518	+0.207 (+67%)
oracle routing	0.363	+0.052
oracle severity	0.362	+0.051
oracle joint	0.615	+0.304 (+100%)

Table 2: Oracle ceilings, segment-weighted MPP-F (the pooled micro-average was intractable on the high-cardinality oracle sets); Δ is vs. Gemini baseline. Oracle joint combines subset and severity.

prompt-framing and self-consistency judges required generation (via Gemini, Gemma-4-12B and UFAL’s system). Deltas between systems use a paired segment-level bootstrap (about 1,000 resamples, seed 42) over the 2,032 dev segments.

5 Results

Tab. 1 shows the results for the individual systems on Dev1 with Gemini being the strongest single judge, winning 4 of 5 dev pairs standalone.¹ The open-weight judges are meaningfully weaker on every metric, and no configuration that includes them beats the commercial panel; we report this as a negative for cost-free local judging.

Tab. 2 shows the oracle ceilings (Sec. 4.3): how much of the score is recoverable, in principle, from the panel’s existing spans if gold were available to select among them. A perfect filter over the panel’s spans would nearly double MPP-F. The span-subset oracle uses greedy rather than exact matching and is a lower bound.

5.1 The recall wall

Agreement among judges is a genuine, generalisable signal, but it cannot fix the problem, because the problem is not precision. Character-level gold

¹Gemini’s solo T1 score recurs at three different values later (0.311, 0.326, 0.334) because of two independent, unrelated causes: segment-weighted vs. true-pooled micro-averaging (Sec. 5.6), and the baseline vs. production prompt (Sec. 4.4). Each is explained where it first appears.

precision rises monotonically with judge vote depth on every dev pair: pooled, characters all three judges agree on are about 62% precise, against about 22% for a single judge’s own flagged characters. This is the only signal that survives leave-one-out validation across all five pairs; which specific judge dissents does not. Yet the union of everything the three judges ever flag touches only 30–40% of gold error characters. That is the wall: recall, not precision, bounds the score, and no reshuffling of spans the judges already proposed can add coverage none of them ever proposed.

5.2 Anatomy of the missed recall

Two questions locate where the wall comes from, both measured on Dev1 (1,538 segments carrying gold errors, 5,632 gold spans). First, the loss is not a segment-length effect: pooled character recall of the three-judge union is essentially flat across MT-length quartiles (0.48, 0.40, 0.39, 0.37 from shortest to longest) and the per-segment rank correlation between recall and length is negligible (Spearman -0.04). The shortest quartile is marginally easier, but long inputs do not drive the loss. Second, the uncovered recall is concentrated, not diffuse: 27.6% of gold spans are missed entirely by the union (Tab. 3), and the misses skew toward minor severity (33.8% vs. 20.5% for major) and short spans (34.7% for ≤ 10 characters vs. 9.5% for > 30), with a consistent per-language-pair spread (en-is 15.2% to cs-de 33.0%). The panel misses small, low-severity, localized errors far more than long or major ones, so the coverage gap is a systematic reluctance to flag minor local errors rather than an attention limit on long inputs.

5.3 Realizable policies do not convert the ceiling

Tab. 4 shows the scores for realizable policies, with paired bootstrap confidence intervals on the delta to Gemini (pooled micro-average). Every delta is at or below zero. The corroborated top-up, the mechanism designed to add recall with precision control, is the worst family: the regions the non-anchor judges agree on where Gemini is silent are not precision-safe, and under a harmonic metric the recall they add does not pay for the precision they cost. The directed prompt-framing result reproduces the same pattern: the recall framing ties the literal prompt as a standalone judge (winning a different 3 of 5 pairs, a genuine hedge), but combining framings by span union is significantly worse.

Stratum	Subset	% missed	mean cov.
Severity	major	20.5	0.71
	minor	33.8	0.57
Span length (chars)	≤10	34.7	0.65
	11–30	27.7	0.69
	>30	9.5	0.52
Lang. pair	en-is	15.2	0.74
	en-ar	26.7	0.46
	en-et	28.4	0.60
	cs-uk	32.1	0.64
	cs-de	33.0	0.61

Table 3: Gold spans on Dev1 (5,632 spans) entirely missed by the three-judge union, by severity, span length, and language pair; overall miss rate 27.6%. ‘mean cov.’ is the mean fraction of a span’s characters the union covers. Misses concentrate in minor, short spans.

Policy	MPP-F	Δ
Gemini (baseline)	0.326	—
recall framing (single)	0.326	+0.000 (n.s.)
severity escalation	0.319	−0.007 (n.s.)
framing union (max sev.)	0.317	−0.008 (worse)
vote-depth ≥ 2	0.307	−0.019 (n.s.)
corroborated top-up	0.295	−0.031 (worse)

Table 4: Realizable panel and framing policies, pooled micro-average MPP-F, with paired bootstrap significance; Δ is vs. Gemini baseline.

Taking the maximum severity across framings also inflates the major rate, because the criticality framing marks almost everything major.

5.4 External judge fusion

Fusing UFAL’s cat-v4 MQM system into the panel (on the 647-segment subset it covers) gives a small, consistent lift over the three-judge baseline (best about +0.007 pooled at a 3-of-4 threshold) but does not beat Gemini standalone, which tops every configuration on that subset (0.345). cat-v4 is precision-heavy and recall-light, so it is a poor two-judge partner for any single judge and helps only inside the fuller panel, in proportion to the base panel’s headroom. Results are directional, on a partial subset, and do not touch the three blind pairs.

5.5 Rank-weighted aggregation

Among the many combination levers tried, we also followed GEMBA-MQM V2 in resampling a single judge ten times and combining the runs with a reciprocal-rank weighted average after outlier rejection, and separately transplanted the same

rank-weighting into span voting over the heterogeneous panel. Neither went into the final submission (self-consistency resampling reduces one judge’s own noise but cannot inject capability it doesn’t have, and on T1 the transplanted version does not beat Gemini standalone (0.302 pooled on the three-judge panel)) but the rank-weighting idea itself is the seed that Sec. 5.6 builds on, by adding a per-judge reliability prior for the pair that made the final cut.

5.6 Judge-reliability priors, and an exhaustive search for one pair

Every aggregation method above ranks judges only by their own score on a given segment, with no notion that one judge is generally more reliable than another. We add an optional static per-judge *reliability prior* (each judge’s own standalone metric) that multiplies into the per-segment rank weight, for both the span-voting and scalar aggregation families, and use it to drive an exhaustive method search restricted to the pair that motivates the question: Gemini and Gemma-4-12B, the clearest known-quality gap in the panel (Tab. 1).

For T1 the search is genuinely exhaustive. Beyond union, intersection, and RRWA-weighted voting with and without the prior, we test a Gemini-anchored rescue family (keep 100% of Gemini’s spans, admit only Gemma’s exclusive spans past an increasingly strict severity-and-length filter), word-level voting, span-level one-to-one matching (via the same optimal matcher that scores MPP-F itself), a learned admission classifier (logistic regression over 9 span- and segment-level features, evaluated leave-one-language-pair-out), and boundary refinement, which blends Gemini’s and Gemma’s offsets for matched span pairs instead of an include/exclude decision. Every family tops out at or below Gemini alone (Tab. 5); the rescue filter makes the bound explicit, since tightening it can only ever discard admitted Gemma content, so the curve converges to the solo baseline bit-for-bit as the filter approaches “admit nothing.”

A precision/recall breakdown explains why: union’s precision collapses to Gemma’s own precision, since Gemma’s exclusive detections are mostly false positives, while intersection’s recall craters below Gemini’s own, since Gemini’s exclusive detections are mostly correct. Boundary refinement fails for the same underlying reason, and monotonically regardless of merge rule: Gemini’s boundaries are already the better estimate of the

T1 method family	Best MPP-F
Gemini (solo, reference)	0.326
Unweighted union / intersection	0.284
RRWA-weighted, with MPP-F prior	0.287
Gemini-anchored rescue filter	→ 0.326
Word-level voting	0.260
Span-level matching + rescue	0.312
Learned classifier (logistic regr.)	0.317
Boundary refinement (best: intersection)	0.322

Table 5: Every T1 combination family tried for Gemini+Gemma, true-pooled MPP-F on Dev1; solo Gemini here (0.326) differs from Tab. 1 only through true micro-averaging vs. per-pair averaging (footnote 1), not a different system. The rescue filter’s arrow denotes asymptotic convergence to the solo baseline as the filter tightens, never exceeding it.

true span, so blending in any of Gemma’s boundary information nudges away from gold rather than toward it.

The one informative byproduct is the learned classifier’s feature weights: each judge’s own segment-level confidence dominates the admission decision by a factor of 2–3 $\times$ over the span’s own severity or length. This is more useful as a diagnostic for a future iteration than as a route to a better classifier on its own.

T2 and T3 diverge from this pattern (Tab. 6). For T2, a purely static prior-weighted scalar mean, with no per-segment adaptivity at all, is the strongest method found (0.622, versus 0.574 for the best prior-free combination, and versus 0.600 for adding per-segment RRWA ranking on top of the same prior, which costs a little rather than helping). For T3 the prior adds nothing: plain unweighted intersection, unchanged from the earlier sweep, remains the best method (0.252). A follow-up search confirms both numbers sit close to a real optimum for this pair rather than an artifact of the first pass: an interpolation between the per-segment rank weight and the static prior is fully monotonic toward the pure-prior endpoint for T2 (no interior sweet spot), and plateaus below intersection’s score for T3 regardless of which metric’s prior is substituted in.

The asymmetry across tasks is the header result: whether a reliability prior helps depends on the metric. A harmonic, span-level metric (T1) is bound by shared recall, so weighting a weaker judge more or less carefully cannot outweigh the precision it costs; a continuous score (T2) benefits from a static reliability weighting with no per-segment component at all; a binary decision (T3) is already well served by simple agreement, and gains

Metric	Best method	Score	Δ
T1	Gemini alone	0.326	—
T2	Scalar prior-weighted mean	0.622	+0.023
T3	Unweighted intersection	0.252	+0.004

Table 6: Best method found per metric for Gemini+Gemma after the exhaustive priors search, pooled Dev1; Δ is vs. Gemini alone. Metrics for T1, T2 and T3 are MPP-F, Pearson, and MCC, respectively. Whether a static reliability prior helps depends on the metric, not on the judge pair.

nothing further from knowing which judge is generally stronger. This strategy was re-derived and replicated under the actual production prompt for the real submission (see Sec. 6) and the T2 finding sharpens further on Dev2.

5.7 Secondary tasks behave differently

The derived tasks do not share T1’s ceiling. Agreement- and outlier-based methods improve the binary error-free classification (T3): the reaction and criticality framings lift T3 MCC by 0.08–0.11 over the literal prompt, and RRWA gives the best T3 of any merge we tried. For the segment-score task (T2), deriving the score from predicted spans is markedly better than a direct LLM score (about 0.64 versus 0.46 Pearson), a methodological point worth flagging given the task’s opt-in direct-scoring design. This localization-versus-classification split is the mechanism behind the whole result: agreement helps a coarse binary decision but cannot help a harmonic localization metric whose recall is already capped.

5.8 Analysis

The ceiling and the realizable result are only apparently in tension. The oracle is a precision instrument: it holds recall at the union’s level and removes false positives using gold. Any gold-free method must instead choose which spans to keep from the panel’s own signals, and the only signal that generalises, agreement, trades recall for precision. Since recall binds, the trade cannot win, and every diversity source we tried (undirected sampling, directed prompt framing, an external MQM system, principled rank weighting) reproduces it. The one mechanism that buys recall, the corroborated top-up, loses because the regions the panel adds are noisier than the gold implies.

This points effort elsewhere. Aggregation cannot exceed the union’s coverage, and that coverage is a

Task	Method	Dev1	Dev2
T1	Gemini solo	0.334	0.289
T2	RRWA, prior	0.642 (+0.005)	0.442 (+0.039)
T3	Intersection	0.306 (+0.064)	0.233 (+0.069)

Table 7: Final replicated strategy (Phrase01), pooled scores on two independent dev sets under the production prompt; deltas are vs. the corresponding single-judge or no-prior baseline on that dev set.

minority of gold characters, so the lever that moves the score is raising a single strong judge’s coverage through the prompt, not combining judges. It also connects to the target: with human inter-annotator recall around 66% (Kocmi et al., 2024), some of the missing coverage is the reference’s own, so part of the wall is a property of the metric rather than of the judges.

6 Submitted System

The strategy of Sec. 5.6 was derived under the baseline prompt (Sec. 4.4). Re-deriving it under the revised, directive-style prompt actually used for the real test-set run changes T2’s winner: the plain prior-weighted mean loses its edge, and a multiplicative RRWA-with-prior combination takes over instead. The revised strategy was then replicated on Dev2: UFAL’s final dev set, used here for consistency with the cat-v4 fusion analysis below, which evaluates UFAL’s own system on their own final set. Dev2 also has a narrower Gemini/Gemma quality gap than Dev1, confirming the strategy holds and that the T2/T3 gains are larger there, consistent with the intuition that a reliability prior matters more the closer two judges are in quality.

6.1 Phrase01 (primary)

Phrase01 applies Tab. 7’s strategy to the real, gold-free test data across all eight priority pairs (Sec. 3). The join key across judges is the composite (segment id, system name), since the real test-set records carry no usable single per-row identifier, unlike the dev sets. The T2 prior is pooled across both dev sets: {gemini: 0.5647, gemma: 0.4270}.

A pre-existing bug surfaced only when cross-checking against the organisers’ official priority-pair list: our own priority-pair constant still listed cs-uk (Czech-Ukrainian), which the organisers had replaced with cs-vi (Czech-Vietnamese) in the high-priority set on 2026-07-22. The bug did not affect the real judge data, which already covered the correct pair, but it does mean neither dev set above has

System	MPP-F	T2 Pearson	T3 MCC
Gemini	0.289	0.403	0.164
Gemma	0.223	0.353	0.149
UFAL cat-v4	0.186	0.343	0.206

Table 8: Standalone scores on Dev2. The external system is weakest on T1/T2 but strongest on T3. Standalone quality again does not imply a combination benefit (Sec. 5.6).

data for that specific pair, under either prompt; the replication evidence in Tab. 7 is accordingly strong for seven of the eight submitted pairs but rests on the transfer argument alone for the eighth.

6.2 Phrase02 (secondary): forcing in a third, external judge

Before finalising Phrase01, we tested whether adding a third, structurally different system—UFAL’s cat-v4 (Sec. 4.1), not run through our own pipeline—to the Gemini+Gemma pair could do better, using the same exhaustive-search methodology as Sec. 5.6, on Dev2 only. Tab. 8 reports the result: the external system is competitive standalone on the derived binary task, but adding it to Phrase01’s pair does not beat Phrase01 on any of the three tasks, under either of the two aggregation mechanisms it tried best.

We nonetheless submit a second variant, Phrase02, that forces all three systems into every task, using the best three-way method found in that search per task—a genuinely different aggregation mechanism from Phrase01’s for T2 and T3, not simply Phrase01 plus a third vote: unweighted span union for T1, a static prior-weighted scalar mean for T2, and a per-segment prior-weighted RRWA over each system’s own binary prediction for T3. The external system does not cover every record Gemini and Gemma do; wherever it is missing, Phrase02 falls back to Phrase01’s two-judge result for that record rather than dropping it, so every segment still receives a label. Phrase02 is explicitly an ablation, submitted despite the negative result above, to give the shared task a second, honestly-labelled data point on whether a third judge helps once more of the panel-composition space is forced open. Both Phrase01 and Phrase02 were uploaded before the deadline.

cat-v4 is itself a WMT26 shared-task submission from UFAL, described more fully in their own system paper (Hrabal and Bojar, 2026).

7 Conclusion

For span-level MT quality evaluation under a severity-weighted matching metric, a panel of independent LLM judges has a large ceiling but almost none of it is realizable without gold, because the judges’ shared recall is low and the only generalisable panel signal improves precision alone. We reach this conclusion from every diversity source and combination lever tried, including an exhaustive, reliability-prior-aware search restricted to the specific pair used in the final submission. A single strong judge remains the better system for span-level detection (T1). The two tasks derived from the same spans tell a different story: a static, per-judge reliability prior is the best method found for the segment score (T2), and simple agreement remains the best method for the binary error-free decision (T3). Aggregation helps exactly where the metric rewards it, and not where it doesn’t. Our submitted system follows this split (Phrase01), and we additionally submit an explicit ablation that forces a third judge into every task (Phrase02) despite evidence it does not help, as a second data point for the shared task. The productive direction for span-level detection is annotation coverage through generation; for the derived tasks, it is matching the aggregation mechanism to the metric rather than defaulting to one combination rule everywhere.

This paper measures the recall wall and its anatomy (Sec. 5.2); turning that diagnosis into higher coverage is future work. Because the misses concentrate in small, low-severity, localized spans rather than long inputs, the productive interventions are on the generation side rather than aggregation: highlighting candidate spans in the prompt, evaluating spans at a finer granularity than the whole segment, and seeding recall with non-LLM candidate detectors (e.g. word alignment) before LLM adjudication, each aimed at the coverage ceiling this paper identifies as the binding constraint.

Limitations

Every quantitative result in this paper is measured on development sets: no configuration was chosen or validated on a held-out test split, and official test scores arrive only after submission, so the conclusions should be read as dev-set findings pending that confirmation. Dev covers 5 of 8 priority pairs, so the three blind pairs rest entirely on the transfer argument; one of the eight submitted pairs (Sec. 6) has never been directly covered by any dev-set ex-

periment in this line of work, under either prompt, for reasons unrelated to the transfer argument itself. The oracle span-subset uses greedy matching and is a lower bound. External-fusion results in Sec. 5.4 are on a 647-segment subset the external system covers, are directional rather than held out; the later three-way search behind Phrase02 (Sec. 6) is scoped to one dev set only and was not cross-checked against the first. We test specific commercial judges plus two open-weight models and one external system, and trained no component; results may not transfer to other judges. Reproducibility is limited in two ways. The commercial judges are closed, versioned models, and the prompts used for the real run are production assets we do not release, so the exact system cannot be reconstructed from this paper alone. More substantively, the per-task recommendations are prompt-contingent: the best T2 method changes from a plain prior-weighted scalar mean under the baseline prompt to a multiplicative RRWA-with-prior under the production prompt (Sec. 6), so the specific winning aggregator should be read as conditional on the prompt, whereas the qualitative split we claim as robust (T1 bound by shared recall, T2 aided by a static reliability prior, T3 by simple agreement) does not depend on it. Configuration was selected on gold that carries the same annotation-recall limits we invoke in the analysis.

Acknowledgment

This work has received funding from the Project OP JAK Mezišektorová spolupráce Nr. CZ.02.01.01/00/23_020/0008518 named “Jazyková, umělá inteligence a jazykové a řečové technologie: od výzkumu k aplikacím.”

The authors used a combination of Claude models (Opus 4.7, 4.8 and Sonnet 5) to assist with implementing code for the experiments and with drafting and reviewing portions of this paper. The research questions, experimental design, analysis, and conclusions are the authors’ own. All AI-generated code was reviewed and tested by the authors, and all text was reviewed and edited by them; the authors take full responsibility for the content.

References

Patrick Fernandes, Daniel Deutsch, Mara Finkelstein, Parker Riley, André F. T. Martins, Graham Neubig, Ankush Garg, Jonathan Clark, Markus Freitag, and

- Orhan Firat. 2023. The devil is in the errors: Leveraging large language models for fine-grained machine translation evaluation. In *Proceedings of the Eighth Conference on Machine Translation*, pages 1066–1083, Singapore. Association for Computational Linguistics.
- 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*, 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.
- 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.
- Miroslav Hrabal and Ondřej Bojar. 2026. CUNI at WMT26 automated MT evaluation task.
- Marcin Junczys-Dowmunt. 2025. GEMBA-MQM V2: Ten judgments are better than one. In *Proceedings of the Tenth Conference on Machine Translation*, pages 926–933. 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).
- Boxuan Lyu, Haiyue Song, Hidetaka Kamigaito, Chenchen Ding, Hideki Tanaka, Masao Utiyama, Kotaro Funakoshi, and Manabu Okumura. 2025. Minimum Bayes risk decoding for error span detection in reference-free automatic machine translation evaluation. *arXiv preprint arXiv:2512.07540*.
- Stefano Perrella, Eric Morales Agostinho, and Hugo Zaragoza. 2026. Span-level machine translation meta-evaluation. *arXiv preprint arXiv:2603.19921*.
- 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)*, pages 2685–2702, Online. Association for Computational Linguistics.
- Pat Verga, Sebastian Hofstätter, Sophia Althammer, Yixuan Su, Aleksandra Piktus, Arkady Arkhangorodsky, Minjie Xu, Naomi White, and Patrick Lewis. 2024. Replacing judges with juries: Evaluating LLM generations with a panel of diverse models. *arXiv preprint arXiv:2404.18796*.
- Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and Zhifang Sui. 2024. Large language models are not fair evaluators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9440–9450. Association for Computational Linguistics.
- Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. [Self-consistency improves chain of thought reasoning in language models](#). In *The Eleventh International Conference on Learning Representations*.