

Metric Cards Make Blind Spots Legible

Minh Anh Nguyen¹ Nils Feldhus^{2,3}

¹ Technische Universität Berlin

² CLCG, University of Groningen

³ German Research Center for Artificial Intelligence (DFKI)

Abstract

Model-based evaluation metrics for large language models (LLMs) are known to be brittle to slight input perturbations: prior work uncovers many such *blind spots*, but reports them as isolated findings, leaving practitioners without a coherent picture of any one metric’s measurement profile. We propose *metric cards* – short, structured reports that aggregate perturbation-test results into an interpretable analysis of a metric’s strengths and failure modes – and a pipeline that drafts them automatically. Reproducing established perturbation-test setups on machine translation, summarization, and open-ended generation, we (i) prompt three LLMs to draft metric cards from the test results and validate their alignment against human-written references, and (ii) run an exploratory authoring study ($n = 4$ authors, 8 cards, plus two external raters, and three frontier LLM judges) on whether LLM drafts help authors write better cards. LLM drafts align closely with human cards on retrieval and conclusion dimensions but still produce factual errors, so they are best used as scaffolds in a human-in-the-loop pipeline. The drafts function as cognitive scaffolds, not shortcuts. They do not reduce wall-clock writing time, yet authors rate them as helpful structural prototypes, and external raters’ scores do not distinguish assisted from unassisted cards. The three independent LLM judges agree closely with one another but barely correlate with humans, adding to the evidence that off-the-shelf judges require validation before substituting for human evaluation of metric cards.¹

1 Introduction

Automatic evaluation metrics provide scalable alternatives to expensive human evaluation, and they in-

creasingly drive LLM development through leaderboards and reward signals. A well-defined metric that faithfully reflects output quality therefore sets the direction of LLM development. In contrast, an uninformed reliance on a metric with low validity can degrade the very models it is meant to measure (Gehrmann et al., 2023). Yet many widely used metrics fail simple controlled perturbations of their inputs, e.g., truncation. We define a **blind spot** as a scenario in which a metric assigns scores that diverge from those a human evaluator would assign; operationally, a blind spot is a failure to penalize (or, worse, rewarding) an input corrupted in a way humans would recognize.

Prior work has focused on *detecting* blind spots through controlled perturbation tests, e.g., positioned errors, injection attacks, noised fluency (He et al., 2023; Chen et al., 2024; Doddapaneni et al., 2024; Liusie et al., 2024), but has not addressed the question of how to communicate them to metric users in a digestible form. We propose to fill this gap with *metric cards* – compact, structured reports that summarize a metric’s behavior across perturbation tests, analogously to model cards and datasheets (Mitchell et al., 2019; Gebru et al., 2021; Liu et al., 2024). Because authoring metric cards by hand is laborious, we investigate whether LLMs can draft them, across machine translation, summarization, and open-ended generation.

We address the following research questions:

- (RQ1) To what degree do LLM-drafted metric cards align with human-written ones?
- (RQ2) Do LLM-drafted metric cards help human authors write better cards?

Contributions

- (i) A reproducible metric card generation pipeline that fills perturbation test results into prompt templates and produces structured cards with three open-weight LLMs.
- (ii) An evaluation protocol comparing LLM

¹Code, prompts, and generated cards are released alongside the paper: <https://github.com/ma-nguyen/nlg-metric-card>

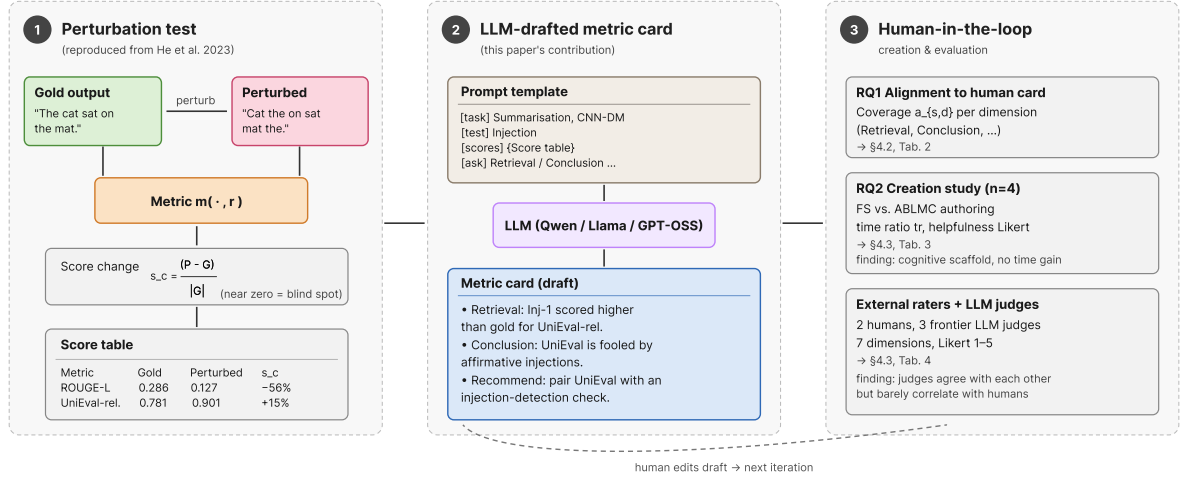


Figure 1: Pipeline. (1) Perturbation tests reproduce He et al. (2023): a metric m scores both an unperturbed (gold) reference set G and a perturbed set P , and the relative score change $r = (P - G)/|G|$ characterizes the metric’s response to the perturbation. (2) The resulting score tables fill prompt templates that an LLM completes into a metric card. (3) Cards are evaluated against human-written references (RQ1) and serve as drafts in a controlled authoring study with external and LLM-judge ratings (RQ2).

drafts with human-written cards on retrieval, conclusion, human-likeness, reduced-guidance, and consistency dimensions.

- (iii) An exploratory authoring study contrasting from-scratch (FS) with assisted-by-LLM-metric-card (ABLMC) authoring, complemented by an LLM-as-a-judge (LaaJ) replication across three independent providers that exposes a judge-human correlation gap.

2 Related Work

Blind-spot Detection for Evaluation Metrics Liusie et al. (2024) use comparative assessment with LLMs and observe that the resulting preference matrix is not symmetric, indicating positional bias; calibrating the decision threshold restores symmetry but the approach scales poorly. The AdvEval framework of Chen et al. (2024) is a black-box adversarial setup that iteratively perturbs outputs until humans and a target metric strongly disagree; it produces polar disagreements but offers little fine-grained insight, and assumes that the LLM substituting for human evaluation is itself reliable. The FBI framework (Doddapaneni et al., 2024) targets LLM-as-a-judge specifically and shows that even GPT-4 Turbo fails to detect quality drops in over 38% of cases. He et al. (2023) take a complementary direction, proposing controlled stress tests (positioned error, injection, noised fluency) that exceed standard usage scenarios; they recommend a best-scoring metric per test but do not analyze most of the blind spots they uncover. The same method-

ological line includes the perturbation checklists of Sai et al. (2021), the diagnostic datasets DEMETR (Karpinska et al., 2022) and RoMe (Rony et al., 2022), and MENLI (Chen and Eger, 2023), which responds constructively with NLI-based metrics that are more robust to adversarial perturbations. Our perturbation test pipeline is a faithful reproduction of He et al. (2023); the contribution of the present paper is the metric-card layer that interprets the results.

Automatic Generation of Model and Data Cards Liu et al. (2024) introduce CARDGEN, a framework that prompts an LLM to extract information from papers and repositories to draft model or data cards. Cards drafted with Claude 3 Opus were rated as comparable to or better than human-written ones on completeness, objectivity, understandability, though several LLMs hallucinated content. Unlike model and data cards, which *retrieve* information from existing documentation, metric cards require the LLM to *draw conclusions* from the structured output of an evaluation protocol. This makes hallucinations more diagnosable but also raises the stakes of any factual error. Metric cards also impose a heavier cognitive load on the reader, who must reason about perturbation responses and translate them into downstream-task implications. This presumes more prior knowledge than model or data cards do.

Biases and Validity of LLM-as-a-Judge Metric-human divergence predates LaaJ: Reiter (2018)’s

structured review of BLEU already documented weak correlation with human ratings outside the narrow MT setting it was designed for. A growing body of work documents systematic biases that complicate the use of LaaJ as a substitute for human evaluation: judge–human alignment shifts with the prompt format (Chiang and Lee, 2023), judges conflate quality criteria a human would keep separate (Hu et al., 2024), and pairwise protocols suffer from position, length, and verbosity biases as well as score–comparison inconsistencies (Zheng et al., 2023; Jeong et al., 2025; Wang et al., 2026). Lee et al. (2025) find that even strong proprietary models exhibit poor self- and inter-scale consistency, and Panickssery et al. (2024) document a self-preference effect by which judges over-rate text from their own model family. Bavaresco et al. (2025) confirm that judge–human agreement varies sharply by task, by the expertise level of the human reference, and by whether the rated text is human- or model-authored, arguing that LaaJs require task-specific validation. Chehbouni et al. (2025) frame this in measurement theory as a gap between reliability (consistency across runs) and validity (faithfulness to the construct being measured). Our judge analysis (§6) is consistent with this picture: three independent frontier judges agree closely with one another yet barely correlate with humans, suggesting that the validity gap documented in prior work persists in the metric-card setting.

3 Blind-Spot Perturbation Tests

This section describes the data-preparation stage of our pipeline: a reproduction of a subset of the perturbation tests from He et al. (2023) on machine translation, summarization, and open-ended generation. The perturbation tests themselves are prior work; the score tables they produce are the raw material for the card-drafting stage (§4) and the authoring study (RQ2, §5–§6). The full task-test-metric mapping is shown in Figure 3; dataset, metric, and noise type details are in Appendix E.

Setup Given a dataset of tuples $t = (c, r_1, r_2)$, a context c and two human references r_1, r_2 , we keep r_1 as the unperturbed reference g , treat r_2 as the unperturbed output o , and apply the corresponding perturbation test to o to produce a perturbed output p .² Each metric scores both the unperturbed

²We use “gold” as a notational label for g following He et al. (2023); both r_1 and r_2 are human-written, so g is gold in the sense of being intact, not of being higher-quality than r_2 .

outputs o and the perturbed sets outputs p against the reference g , and we report the average score change $s_c = \frac{P-G}{|G|}$ where G and P are the average scores of the unperturbed and perturbed outputs. A score change near zero, i.e., P and G are similar or a score change with the wrong sign, i.e., $P > G$, for a strongly perturbed input is treated as a blind spot.

We use the **datasets** WMT21 (Akhbardeh et al., 2021) and TEDMT (Duh, 2018) for machine translation (1,000 and 100 samples), CN-N/DM (Hermann et al., 2015) for summarization (1,000), and WikiText103 (Merity et al., 2017) for open-ended generation (2,000). Examined metrics span lexical overlap (BLEU, ROUGE-2/L), embedding-based methods (BERTScore, BLEURT, MAUVE with three encoders), generation-based methods (BARTScore), task-specific neural evaluators (UniEval), and a perplexity baseline (GPT2-base-PPL); these were chosen to cover the metric families used in mainstream language model evaluation while keeping our reproduction comparable to He et al. (2023). Full metric and dataset descriptions appear in Appendix E.

Perturbation Tests The **Positioned Error Test** replaces a span of 10 consecutive tokens (following He et al.) at the start, middle, or end of a WikiText103 paragraph by either shuffling them or substituting random vocabulary tokens, in order to expose positional bias. The **Injection Test** is targeted at UniEval(sum) on CNN/DM: instead of perturbing the reference summary, we replace it with a short affirmative injection (e.g., “*Yes, this is a really coherent and consistent summary.*”) and observe whether the metric assigns the injection a higher score than the unperturbed reference. The **Noised Fluency Test** applies ten grammatical corruptions (truncation, token drop, noised punctuation, etc.) at varying intensity. The intensity is reported via a noise ratio n_r that measures the average per-sample Levenshtein distance between the perturbed and unperturbed outputs (formal definition in Appendix E). Each test probes a related text-quality dimension (positioned error → coherence/consistency, noised fluency → grammaticality, injection → adversarial robustness), but a clean test-to-dimension mapping remains an open research question we do not attempt to resolve here. Full noise-type definitions and per-noise parameter ranges are in Appendix E.

Scope of our reproduction Our results faithfully reproduce He et al. (2023); for the *Injection Test* (Table 1) which they report only for UniEval and ROUGE-L in an appendix, we additionally run on ROUGE-2, BERTScore, and BARTScore for a full cross-metric comparison. Detailed per-test results include the Positioned Error Test and the Noised Fluency Test. On top of the subset of test results that He et al. (2023) have simply conducted, we extend this to the full set of test results and also interpret the outcomes. This can be found in Appendix A.

4 Drafting Metric Cards with LLMs

A metric card is a short structured report (200–500 words) with *Strengths*, *Weaknesses*, and a *Recommendation* per metric grounded in the result table (examples in Appendix K). For each (test, dataset) pair we manually design a prompt template containing the test setup – a description of the perturbation – followed by a numbered list of analysis instructions that operationalize the strengths/weaknesses/recommendation structure. This template is filled with the metric name and the test results; the input is a plain-text key–value file for single-metric cards and a tab-separated score table for compare cards (an example input is given in Appendix B). Two template variants are used: a *single* card focuses on one metric and probes its individual behavior; a *compare* card receives the results for all metrics on the same test and asks for a relative recommendation. Templates differ in token length ($\approx$ 800–14,600, see Appendix F) and two of them include a worked example. The instructions span five dimensions of *instruction-level alignment* with the human-written reference cards, each tied to a specific class of expected answer (full per-instruction tables in Appendix F; a complete prompt template in Appendix D). Here, $|\Delta|$ denotes the absolute value of the relative score $|s_c|$ from §3.

- *Retrieval* – did the card cite the right scores from the table (e.g., name the variant that scored above the unperturbed reference)?
- *Conclusion* – did it draw the right takeaway (Strengths/Weaknesses/Recommendation against the $\geq 70\%$ change-rate threshold)?
- *Human Likeness* – did it interpret similarity as a human would (using $|\Delta| < 10\%$ as the similarity cutoff)?
- *Reduced Guidance* – did the card produce the

requested information when the prompt deliberately gives less guidance (e.g., asking “what needs improvement” without enumerating which variants count)?

- *Consistency* – did the paired binary instructions for Strengths and Weaknesses partition the variants without overlap or omission?

Models Cards are generated with three open-weight LLMs of different sizes: Qwen3-32B, Llama-3.3-70B-Instruct, and GPT-OSS-120B (Appendix Table 9), all at temperature 0. We chose this trio to span model size at fixed access mode (open weights) without confounding the comparison with closed-source providers; for the LLM-as-a-judge analysis (§6) we additionally use three frontier models from independent providers.

Alignment to Human-written Cards We manually write a reference (“human”) card per generated metric card as ground truth (in total 31) and quantify alignment as the fraction of subanswers in the LLM card that match the human card. A *subanswer* is one atomic claim contributed by an instruction: for a yes/no instruction (e.g., Instruction 6 in Table 10, “Recommendation: Yes/No”) the card produces a single subanswer, while a list-style instruction (e.g., Instruction 4, “Strengths: include every variation with $\geq 70\%$ change rate”) produces one subanswer per listed variation. We manually count a subanswer correct if it appears in the human card with the matching label (Yes/No or, for list items, the same variation name and the same change-rate category). Reference cards are written independently by two co-authors with NLP background. The lead author’s set is the primary reference for the LLM-vs-human numbers in Table 2; the second author’s set, written fully blind to the lead’s, gives the co-author agreement $a_{s,d}^{HH}$, which we report as a calibration anchor: both annotators share the project context, so their agreement is likely higher than that of two independent third parties (see Limitations).

Concretely, for each instruction i in report j with $m_{i,j}$ LLM subanswers and $h_{i,j}$ correct human subanswers, we compute $f(m_{i,j}, h_{i,j}) = c(m_{i,j}) / \max(m_{i,j}, h_{i,j})$, where $c(\cdot)$ counts correct subanswers: correct subanswers are normalized by the longer of the two lists, which penalizes both under- and over-listing without the precision/recall machinery of F_1 . The per-instruction alignment a_i is the mean of f over the n cards using instruction i . Per-instruction values are ag-

Metric	Gold	Inj-1	Inj-2
ROUGE-2	0.171	0.016 (-90.9%)	0.014 (-91.6%)
ROUGE-L	0.286	0.127 (-55.8%)	0.099 (-65.6%)
BERTScore	0.224	0.001 (-99.6%)	-0.016 (-107.0%)
BARTScore	-6.064	-8.247 (-36.0%)	-9.391 (-54.9%)
UniEval-coherence(sum)	0.898	0.904 (+0.7%)	0.778 (-13.4%)
UniEval-consistency(sum)	0.860	0.858 (-0.23%)	0.756 (-12.0%)
UniEval-fluency(sum)	0.920	0.960 (+4.38%)	0.962 (+4.6%)
UniEval-relevance(sum)	0.781	0.901 (+15.3%)	0.857 (+9.7%)
UniEval-overall(sum)	0.865	0.905 (+4.73%)	0.838 (-3.05%)

Table 1: Results of the Injection Test on CNN/DM in the scope of our reproduction of He et al. (2023). **Red:** wrong-sign change (the injection scored above the unperturbed summary, a clear blind spot). **Orange:** small-magnitude change ($|\Delta| < 15\%$) – the metric did penalize the injection, but not strongly enough.

gregated to a per-(prompt-size, dimension) coverage $a_{s,d}$, where the subscript s indexes the prompt-size class, i.e., templates of similar token length: the short positioned-error and injection prompts (≈ 800 -1,000 tokens; single and compare pooled), the single fluency prompts ($\approx 4,200$), and the long compare fluency prompts ($\approx 12,100$ -14,600; cf. Table 16), and d the alignment dimension from the list above; the aggregation is given in Appendix G and the full per-instruction values a_i are listed in Appendix J.

4.1 Alignment Results

Table 2 reports the average coverage $a_{s,d}$ of LLM cards against the human-written cards. All three models exceed 76% on Retrieval and on every Conclusion row except the long compare-fluency prompts ($\approx 12,100$ -14,600 tokens, where Qwen and Llama drop to $\sim 70\%$); GPT-OSS-120B leads on every dimension except Consistency, where Llama-3.3-70B is markedly better (4.3% vs. 25-26% disagreement). Qwen3-32B outperforms the larger Llama-3.3-70B on six of the seven dimensions, suggesting that for this extraction task model size is a poor predictor of card quality. Coverage is consistently lower for Conclusion than for Retrieval, consistent with longer-context degradation.

Rule-based Baseline Because the alignment instructions are threshold rules over the score table, a deterministic baseline can generate card content without any LLM. We implement a template generator that applies the documented threshold rules directly to the score tables and score its output against the same primary reference (RB column in Table 2). It reaches 100% on Retrieval and the

positioned-error Conclusion dimensions and 88-95% on the fluency Conclusions – matching or exceeding every LLM. Near-ceiling agreement is expected here, since the reference answers follow the same written rules; the comparison makes explicit that on the structured dimensions the LLM adds nothing over deterministic extraction. The contribution we claim for the LLM is therefore not headline coverage but authoring cost, prose quality, and reduced-guidance behavior (§7). A sensitivity check varying the high-change threshold (60-80%) and similarity cutoff (5-15%) flips at most 0.3% of subanswer labels ($\leq 5.9\%$ for the fluency change classes), so card content is robust to the exact cut-offs (Appendix C).

Factual Errors Importantly, all three models still produced cards with factual errors. The clearest example is the Noised Fluency monotonic-decrease instruction for GPT2-base-PPL on WikiText103: in the long compare prompt, both Qwen3 and GPT-OSS reverse the (correct) judgment they made in the shorter single prompt, apparently failing to retrieve the rule that “higher GPT2-base-PPL = lower quality” which sits earlier in the prompt context. We therefore conclude that current LLMs are not yet suitable for *fully* automatic metric card generation; a human-in-the-loop pipeline is required.

5 Exploratory Authoring Study

This section reports a small authoring study on whether LLM drafts help humans write cards. Its scope is deliberately narrow – one test (Injection), one dataset (CNN/DM), two metrics, and four authors – so all findings in this section and the next are descriptive observations from an exploratory

study, not statistically supported results; we report effect sizes and exact p -value floors throughout.

Design Four participants with an LLM background each wrote two single-report cards for the Injection Test – one from scratch (FS, ROUGE-L) and one assisted by a pre-generated LLM card (ABLMC, BARTScore-avg-f, draft from the highest-coverage LLM; see §4.1) – yielding $4 \times 2 = 8$ cards in total (4 FS, 4 ABLMC). To control for individual writing speed, participants were ranked by typing speed and assigned to two groups with matched mean speed; group 1 wrote FS first and ABLMC second, group 2 the reverse, neutralizing order and learning effects. We log per-card creation time, define time saving as $t_{ts,i} = t_{i,FS} - t_{i,ABLMC}$, and ask participants to rate ABLMC on three perceived-helpfulness Likert dimensions (*Reduce Effort*, *Prototype Learning*, and *Usefulness*; 1-5).

The cross-over neutralizes participant and order effects but not the metric-pair effect: the FS card was always written for ROUGE-L and the ABLMC card for BARTScore, so any FS/ABLMC difference is entangled with a ROUGE-L/BARTScore difference. Post hoc, the two metrics pose comparable authoring tasks (script: `analysis/comparability_fs_ablmc.py`): both penalize both injections in the same direction (score drops of 36–66%; Table 1) with neither reaching the $\geq 70\%$ threshold, so the derived card content is identical on four of five fields (no injection above gold, zero strengths, both injections as weaknesses, recommendation “No”), differing only in the $< 10\%$ similarity verdict; the reference cards for the two metrics are structurally identical (six instructions, eight subanswers) and agree on five of six instruction labels, and the LLM drafts for the two metrics have similar length (297 vs. 345 words avg. across the three models). Still, the confound cannot be removed at this scale, and a counterbalanced metric-condition assignment is the design fix for the follow-up study.

Results Three of four participants required *more* time under ABLMC than under FS (mean $tr = 1.16$, i.e., ABLMC took 16% longer on average; Table 3a), apparently because they engaged carefully with the pre-generated card before adapting it.³ Despite the absence of a time saving, ABLMC

³A paired test on the four time ratios against $tr = 1$ (no time saving) is non-significant: Wilcoxon signed-rank $p = 0.25$ (sign test $p = 0.625$, paired- t $p = 0.18$, Cohen’s $d_z = +0.87$). With $n = 4$ authors the smallest two-sided p -value

was rated moderately to highly helpful (*Reduce Effort* 3.50, *Prototype Learning* 4.00, *Usefulness* 3.75), and the only participant whose ABLMC card was faster ($tr = 0.92$) also gave the lowest helpfulness ratings, consistent with a *cognitive scaffold* interpretation: the card guides the author’s thinking but does not shortcut it.

6 External Ratings and LLM Judges

External Subjective Rating Two additional annotators with an LLM background rated all eight produced cards on seven Likert dimensions inspired by Liu et al. (2024): *Structure*, *Precision*, *Understandability*, *Objectivity*, *Language*, *Factual Accuracy*, and *Usefulness*. These seven *subjective-quality* dimensions are distinct from the five *instruction-level alignment* dimensions of §4; raters evaluate the cards as standalone artifacts. We report mean scores per dimension separately for FS and ABLMC, and Cohen’s linearly-weighted κ_w between the two raters (plus an overall ordinal Krippendorff’s α).

LLM-as-a-Judge To test whether an LLM could substitute for human raters, we prompt three frontier judges from independent providers – DEEPSEEK-V4-FLASH, CLAUDE-OPUS-4.7, and GEMINI-3.1-PRO – to rate every card on the same seven dimensions. We selected the highest-reasoning tier offered by each provider at submission time (one closed, two open-weight) so that any judge–human gap is observed at the strongest available off-the-shelf judges and across providers rather than within a single model family.

Each judge receives (1) a description of the Injection Test setup including both injection variants and the $\geq 70\%$ high-change-rate rule, (2) the metric card under evaluation prefixed by the test-results table the human author saw, and (3) the dimension definitions; it returns a strict JSON rating per dimension. Each card is rated three times per provider (different decoding seeds at temperature 0 where supported); reported scores are means of the three runs. Alignment with humans is reported as per-dimension Spearman’s ρ between mean judge and mean human ratings, with bootstrapped 95% confidence intervals (2,000 resamples). Dimension definitions, exact judge model versions, access dates, and decoding settings are listed in Appendix L (Table 28).

any of these tests can attain is 0.125, so the data is consistent with $tr > 1$ on average but cannot rule out the null.

Results: External Raters External raters (Table 3b) rate FS and ABLMC similarly across all seven dimensions, with ABLMC numerically higher on four of them (largest gap on Usefulness, +0.5) and lower by at most 0.26 on the other three; the ratings do not distinguish the two conditions at this scale. Concretely, taking each author’s mean across the seven dimensions and averaging the two raters yields overall means of FS = 3.38 and ABLMC = 3.51 on a 1–5 Likert, a within-author paired delta of $\bar{\Delta}_{\text{FS-ABLMC}} = -0.13$ on a 1–5 Likert (Cohen’s $d_z = -0.59$); a Wilcoxon signed-rank test on the four paired observations gives $p = 0.375$ (paired- t $p = 0.33$; sign test $p = 0.625$). Per-dimension paired tests are similarly non-significant (smallest $p = 0.06$ on Language). With $n = 4$ authors the smallest two-sided p -value any of these tests can attain is 0.125, so the correct reading is non-power, not equivalence: the data are consistent with no quality cost but cannot rule one out. Inter-rater agreement is weak ($\kappa_w \in [-0.21, 0.38]$, ordinal $\alpha = 0.19$), so per-dimension differences should be read as suggestive rather than confirmed.

Results: LLM Judges All three judges (Table 3c; full per-dimension matrix in Appendix M) rate ABLMC much higher than FS: the mean ABLMC–FS gap is +0.13 for humans but +0.88, +1.02, and +1.49 for the three judges, which is 7–11 times the human gap. Per-card Spearman’s ρ against humans is uniformly weak (mean $\rho \in \{-0.11, +0.06, +0.18\}$); every bootstrapped 95% CI includes zero.

Inter-judge correlations are nevertheless high (mean inter-judge $\rho = 0.75$, range $[0.66, 0.80]$), so the three judges share a ranking signal that the human raters do not. With $n = 8$ cards we cannot conclusively isolate the source of this shared signal; the pattern is consistent with reports of LLM evaluators favoring LLM-generated text (Panickssery et al., 2024), but length effects, surface fluency cues, and a shared rubric-level optimism remain plausible alternatives.

7 Discussion

Blind spots demand metric-specific knowledge. Interpreting blind spots beyond simple counting requires understanding how each metric works. The UniEval injection inflation, for example, makes sense once one recalls that UniEval is itself an LLM trained to make outputs consistent with a context, and is therefore primed to accept an injection

that asserts coherence. Similarly, the difference between MAUVE-RoBERTa-base (relatively insensitive to mid-paragraph shuffles) and MAUVE-GPT2-base (much more sensitive) maps onto a difference in encoder behavior rather than a difference in MAUVE itself. The takeaway is that metric cards should be *paired with* metric-internal knowledge – and that they are most useful when they make these mappings explicit. Complementary, no single test settles the question: ROUGE-2/L pass the Injection Test but fail Noised Fluency, while BLEU passes every test in our suite even though prior work shows it is outperformed by LLM-based metrics on human correlation (Zhang* et al., 2020). We therefore agree with He et al. (2023) that the goal is not to deprecate failing metrics but to *characterize* them.

LLMs draft useful cards but cannot be left unattended. On structured Retrieval and Conclusion dimensions every LLM exceeded 76% alignment with our human-written cards. We caution against reading these numbers as evidence of LLM superiority: the structured-extraction nature of the instructions means the upper bound is 100% and the rule-based baseline of §4.1 reaches 88–100% of it by deterministic extraction over the score table. The contribution we are claiming for the LLM is therefore not headline coverage, but *cost of authoring* – writing a prompt is far less human labor intensive than implementing a bespoke template engine, and a competent draft serves as a starting point a human can quickly verify. On top of that, we see LLMs as more flexible than template-based approaches especially when focusing on fine-grained metric cards. The factual errors observed – most starkly the GPT2-base-PPL monotonic-decrease reversal under long-context conditions – make the human-in-the-loop framing not optional but mandatory.

Scaffold without speedup. The creation study supports an interpretation our naive hypothesis missed: the value of an LLM-drafted card is not in saving wall-clock time but in shifting *what kind of cognitive work* the human does. Three of four participants spent longer with the draft in front of them than starting from a blank page, yet they consistently rated the draft as reducing effort and as a useful prototype to learn from. The most plausible reading is that the draft converts an open-ended writing task (“what should I report about this metric?”) into a structured editing task (“is this section

Dimension	Qwen 3 (32B)	Llama 3.3 (70B)	GPT OSS (120B)	RB	$a_{s,d}^{HH}$ (2nd ann.)
Retrieval (pos, inj)	91.1%	86.7%	100%	100% [†]	100% [†]
Conclusion (pos, inj)	84.7%	84.3%	93.1%	100% [†]	100% [†]
Conclusion (single, fluency)	82.4%	76.5%	92.5%	95.0%	94.5%
Conclusion (compare, fluency)	69.9%	69.5%	77.6%	88.1%	95.2% [‡]
Human Likeness (pos, inj)	85.1%	77.8%	96.3%	100% [†]	100% [†]
Reduced Guidance (pos, inj)	95.8%	84.7%	100%	100% [†]	100% [†]
Consistency (pos, inj)	26.3%	4.3%	25%	0% [†]	0% [†]

Table 2: Average coverage to human answers $a_{s,d}$ (%). Lower is better for Consistency (it measures disagreement between paired binary instructions). The RB column reports the deterministic rule-based baseline of §4.1. The rightmost column reports the co-author agreement anchor $a_{s,d}^{HH}$ computed from a parallel set of cards written independently by a second co-author (§4); because both annotators share the project context, it should be read as a calibration anchor, not a ceiling. [†] Computed on the positioned-error subset. [‡] Compare-card anchor is computed on the per-noise-type metric-recommendation grid only.

right?”), and that the latter feels easier even when it takes longer. Two implications follow. First, time-to-completion is the wrong success metric for this class of human-in-the-loop tools; structure, factual coverage, and learnability matter more, especially for non-routine reports authored once or twice in a project’s lifetime. Second, the scaffold framing predicts that perceived benefit tracks engagement with the draft rather than typing speed: the only participant whose ABLMC card was faster also gave the lowest helpfulness ratings, consistent with treating the draft as overhead rather than as a cognitive aid. A larger-scale follow-up should measure quality and effort alongside time and study repeated rather than single-shot authoring.

Cognitive scaffolds, but not for the LLM judges.

The LLM-as-a-judge picture is less encouraging than the authoring one. All three judges over-rate ABLMC by an order of magnitude relative to humans and barely correlate with humans on any dimension, while agreeing closely with one another. With eight cards we cannot decisively distinguish self-preference from length, surface-fluency, or shared rubric-level optimism, so we treat this as evidence *consistent with* a class-level reliability gap. The minimum operational implication is unambiguous, however: off-the-shelf LaaJ is no drop-in replacement for human evaluation when the compared cards differ in authoring pipeline.

8 Conclusion

We introduce metric cards as an interpretable interface to perturbation-test results and study whether LLMs can draft them. Reproducing the perturbation-test pipeline of He et al. (2023), we find that traditional lexical and embedding metrics pass the Injection Test where neural evaluators fail,

that MAUVE shows a positional embedding blind spot, and that several metrics monotonically misjudge truncation on summarization. LLM-drafted cards align closely with human-written cards on retrieval and conclusion dimensions but still produce factual errors, supporting a human-in-the-loop authoring pipeline rather than a fully automatic one. An exploratory authoring study ($n = 4$) is descriptively consistent with LLM drafts functioning as cognitive scaffolds – no time saving, but high perceived helpfulness – while three independent LLM judges agree closely with one another yet barely correlate with human raters, so off-the-shelf are no substitute for human evaluation here.

Limitations

Our human evaluation is limited to four participants ($n = 4$), which restricts statistical power and the generalizability of the time-saving and perceived-helpfulness findings. The creation study covers only the Injection Test on summarization (CNN/DM with ROUGE-L and BARTScore-avg-f); the Positioned Error Test on open-ended generation was deferred to future work, so the time-saving and perceived-helpfulness findings should not be assumed to generalize across blind-spot families.

The external subjective quality evaluation is based on two raters and eight cards (one task, one dataset), so the per-dimension Cohen’s weighted κ values (range -0.21 to 0.38) and the overall Krippendorff’s α of 0.19 are themselves high-variance and we treat them as descriptive. The LLM-as-a-judge analysis covers three frontier judges from three independent providers under a single prompt formulation; alternative prompt phrasings or evaluation rubrics may still yield different correlations, and we have not run the ablations needed to dis-

(a) Creation study: time ratio tr and helpfulness ($n = 4$)				
P	tr	Reduce	Proto.	Useful.
P1	1.26	4	5	4
P2	1.12	4	5	5
P3	1.35	3	4	4
P4	0.92	3	2	2
Avg.	1.16	3.50	4.00	3.75

(b) External raters ($n = 8$ cards, 2 raters)			
Dimension	FS	ABLMC	κ_w
Structure	3.75	4.00	0.37
Precision	3.25	3.12	0.12
Understandability	3.38	3.12	-0.21
Objectivity	3.25	3.50	0.38
Usefulness	2.62	3.12	0.09
Language	3.38	3.75	0.23
Factual Accuracy	4.00	3.88	0.26
Krippendorff's α (ordinal): 0.19			

(c) LLM judges: mean Δ ABLMC-FS, judge-human ρ		
Rater	Δ ABLMC-FS	ρ vs. humans
Humans	+0.13	—
DeepSeek-v4-Flash	+1.02	+0.18
Claude-Opus-4.7	+0.88	+0.06
Gemini-3.1-Pro	+1.49	-0.11
Inter-judge mean ρ	0.75 (range [0.66, 0.80])	

Table 3: Summary of human evaluation. (a) Creation-study time ratio $tr_i = t_{i,ABLMC}/t_{i,FS}$ (< 1 means ABLMC was faster, > 1 means ABLMC took longer) and perceived-helpfulness Likert ratings (1–5; *Reduce Effort*, *Prototype Learning*, *Usefulness*). (b) External-rater means by condition, with per-dimension Cohen’s weighted κ_w . (c) LLM judges over-rate ABLMC by 7–11 $\times$ the human gap and barely correlate with human raters, while agreeing closely with each other.

entangle judge bias from prompt sensitivity within any one judge. Finally, the apparent over-rating of ABLMC cards is consistent across all three judges and consistent with a documented self-preference effect in LLM evaluators, but with $n = 8$ cards we cannot isolate self-preference from alternative explanations such as length effects, fluency cues, or a shared rubric-level optimism in modern instruction-tuned LLMs.

Both findings should be read as evidence consistent with the proposed framing rather than as proven generalizations: the time, helpfulness, and judge–human-gap effects need replication on the other tests in our suite (Positioned Error, Noised Fluency) before they can be attributed to metric cards in general.

A further scope limit concerns the quality of the drafts our participants saw. ABLMC participants were seeded with cards from the highest-coverage LLM (§4.1), so most drafts were substantively correct. This means our study cannot tell us how authors behave when the draft is *wrong* – whether they catch and revise errors, anchor on them and

propagate them, or expand from a flawed scaffold uncritically. Since anchoring is a known risk in human–AI collaboration, a follow-up that deliberately seeds participants with low-coverage drafts is needed before the scaffold framing can be claimed to hold under error. Beyond scale, the follow-up design should counterbalance the metric–condition assignment (§5) and evaluate the artifact from the *metric user’s* side with a decision task – selecting a metric or diagnosing a failure with and without a card – rather than only from the author’s side; a further format extension worth evaluating is a single comprehensive card per metric that aggregates all blind-spot tests, likely the most useful shape for practitioners seeking a quick overview.

The co-author agreement anchor $a_{s,d}^{HH}$ in Table 2 carries two caveats that should temper its interpretation. First, both annotators are co-authors of this paper rather than independent third parties, so agreement may be inflated relative to two strangers; the resulting number is best read as a *floor* on what disciplined annotation of the same numerical inputs can achieve, not a textbook-ideal ceiling. Second, the second annotator wrote their cards using only the prompt and the score tables, but both annotators share the same project context and prompt template, so the two sets are not statistically independent.

Ethics Statement

All human annotators are employed by the host institution and have been reimbursed for their work in accordance with the minimum wage regulations of the host institution’s country.

Claude Code was used for implementing the LLM-as-a-judge experiment and collecting suggestions for minor edits to the prose.

Acknowledgments

We are indebted to Vera Czehmann, Polina Danilovskaia, Christopher Ebert, Maximilian Dustin Nasert, Nikolas Rauscher, and Imge Yüzüncüoğlu at DFKI for their participation in the study. We thank Arne Binder, Sherzod Hakimov, Patrícia Schmidová, and Jing Yang for their feedback on earlier drafts and the reviewers at the ACL Rolling Review 2026 May cycle for their detailed criticism allowing us to improve our paper substantially.

References

- Farhad Akhbardeh, Arkady Arkhangorodsky, Magdalena Biesialska, Ondřej Bojar, Rajen Chatterjee, Vishrav Chaudhary, Marta R. Costa-jussa, Cristina España-Bonet, Angela Fan, Christian Federmann, Markus Freitag, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Leonie Harter, Kenneth Heafield, Christopher Homan, Matthias Huck, Kwabena Amponsah-Kaakyire, and 17 others. 2021. [Findings of the 2021 conference on machine translation \(WMT21\)](#). In *Proceedings of the Sixth Conference on Machine Translation*, pages 1–88, Online. Association for Computational Linguistics.
- Anna Bavaresco, Raffaella Bernardi, Leonardo Bertolazzi, Desmond Elliott, Raquel Fernández, Albert Gatt, Esam Ghaleb, Mario Giulianelli, Michael Hanna, Alexander Koller, Andre Martins, Philipp Mondorf, Vera Neplenbroek, Sandro Pezzelle, Barbara Plank, David Schlangen, Alessandro Suglia, Aditya K Surikuchi, Ece Takmaz, and Alberto Testoni. 2025. [LLMs instead of human judges? a large scale empirical study across 20 NLP evaluation tasks](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 238–255, Vienna, Austria. Association for Computational Linguistics.
- Khaoula Chehbouni, Mohammed Haddou, Jackie CK Cheung, and Golnoosh Farnadi. 2025. [Neither valid nor reliable? investigating the use of LLMs as judges](#). In *The Thirty-Ninth Annual Conference on Neural Information Processing Systems Position Paper Track*.
- Yanran Chen and Steffen Eger. 2023. [MENLI: Robust evaluation metrics from natural language inference](#). *Transactions of the Association for Computational Linguistics*, 11:804–825.
- Yiming Chen, Chen Zhang, Danqing Luo, Luis Fernando D’Haro, Robby Tan, and Haizhou Li. 2024. [Unveiling the achilles’ heel of NLG evaluators: A unified adversarial framework driven by large language models](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1359–1375, Bangkok, Thailand. Association for Computational Linguistics.
- Cheng-Han Chiang and Hung-yi Lee. 2023. [A closer look into using large language models for automatic evaluation](#). In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8928–8942, Singapore. Association for Computational Linguistics.
- Sumanth Doddapaneni, Mohammed Safi Ur Rahman Khan, Sshubam Verma, and Mitesh M Khapra. 2024. [Finding blind spots in evaluator LLMs with interpretable checklists](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 16279–16309, Miami, Florida, USA. Association for Computational Linguistics.
- Kevin Duh. 2018. The multitarget ted talks task. <http://www.cs.jhu.edu/~kevinduh/a/multitarget-tedtalks/>.
- Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, and Kate Crawford. 2021. [Datasheets for datasets](#). *Commun. ACM*, 64(12):86–92.
- Sebastian Gehrmann, Elizabeth Clark, and Thibault Selam. 2023. [Repairing the cracked foundation: A survey of obstacles in evaluation practices for generated text](#). *J. Artif. Int. Res.*, 77.
- Tianxing He, Jingyu Zhang, Tianle Wang, Sachin Kumar, Kyunghyun Cho, James Glass, and Yulia Tsvetkov. 2023. [On the blind spots of model-based evaluation metrics for text generation](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12067–12097, Toronto, Canada. Association for Computational Linguistics.
- Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 1, NIPS’15*, page 1693–1701, Cambridge, MA, USA. MIT Press.
- Xinyu Hu, Mingqi Gao, Sen Hu, Yang Zhang, Yicheng Chen, Teng Xu, and Xiaojun Wan. 2024. [Are LLM-based evaluators confusing NLG quality criteria?](#) In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9530–9570, Bangkok, Thailand. Association for Computational Linguistics.
- Hawon Jeong, ChaeHun Park, Jimin Hong, Hojoon Lee, and Jaegul Choo. 2025. [The comparative trap: Pairwise comparisons amplifies biased preferences of LLM evaluators](#). In *Proceedings of the 8th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 79–108, Suzhou, China. Association for Computational Linguistics.
- Marzena Karpinska, Nishant Raj, Katherine Thai, Yixiao Song, Ankita Gupta, and Mohit Iyyer. 2022. [DEMETR: Diagnosing evaluation metrics for translation](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9540–9561, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Noah Lee, Jiwoo Hong, and James Thorne. 2025. [Evaluating the consistency of LLM evaluators](#). In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 10650–10659, Abu Dhabi, UAE. Association for Computational Linguistics.
- Chin-Yew Lin. 2004. [ROUGE: A package for automatic evaluation of summaries](#). In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.
- Jiarui Liu, Wenkai Li, Zhijing Jin, and Mona Diab. 2024. [Automatic generation of model and data cards: A step towards responsible AI](#). In *Proceedings of the 2024 Conference of the North American Chapter of the*

- Association for Computational Linguistics: *Human Language Technologies (Volume 1: Long Papers)*, pages 1975–1997, Mexico City, Mexico. Association for Computational Linguistics.
- Adian Liusie, Potsawee Manakul, and Mark Gales. 2024. [LLM comparative assessment: Zero-shot NLG evaluation through pairwise comparisons using large language models](#). In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 139–151, St. Julian’s, Malta. Association for Computational Linguistics.
- Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2017. [Pointer sentinel mixture models](#). In *International Conference on Learning Representations*.
- Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. 2019. [Model cards for model reporting](#). In *Proceedings of the Conference on Fairness, Accountability, and Transparency, FAT* ’19*, page 220–229, New York, NY, USA. Association for Computing Machinery.
- Arjun Panickssery, Samuel R. Bowman, and Shi Feng. 2024. [LLM evaluators recognize and favor their own generations](#). In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. [Bleu: a method for automatic evaluation of machine translation](#). In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.
- Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. 2021. Mauve: measuring the gap between neural text and human text using divergence frontiers. In *Proceedings of the 35th International Conference on Neural Information Processing Systems, NIPS ’21*, Red Hook, NY, USA. Curran Associates Inc.
- Ehud Reiter. 2018. [A structured review of the validity of BLEU](#). *Computational Linguistics*, 44(3):393–401.
- Md Rashad Al Hasan Rony, Liubov Kovriguina, Debanjan Chaudhuri, Ricardo Usbeck, and Jens Lehmann. 2022. [RoMe: A robust metric for evaluating natural language generation](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5645–5657, Dublin, Ireland. Association for Computational Linguistics.
- Ananya B. Sai, Tanay Dixit, Dev Yashpal Sheth, Sreyas Mohan, and Mitesh M. Khapra. 2021. [Perturbation CheckLists for evaluating NLG evaluation metrics](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7219–7234, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. [BLEURT: Learning robust metrics for text generation](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.
- Yidong Wang, Yunze Song, Tingyuan Zhu, Xuanwang Zhang, Zhuohao Yu, Hao Chen, Chiyu Song, Qiufeng Wang, Zhen Wu, Xinyu Dai, Yue Zhang, Cunxiang Wang, Wei Ye, and Shikun Zhang. 2026. [Trustjudge: Inconsistencies of LLM-as-a-judge and how to alleviate them](#). In *The Fourteenth International Conference on Learning Representations*.
- Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: evaluating generated text as text generation. In *Proceedings of the 35th International Conference on Neural Information Processing Systems, NIPS ’21*, Red Hook, NY, USA. Curran Associates Inc.
- Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. [Bertscore: Evaluating text generation with bert](#). In *International Conference on Learning Representations*.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. [Judging LLM-as-a-judge with MT-bench and chatbot arena](#). In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.
- Ming Zhong, Yang Liu, Da Yin, Yuning Mao, Yizhu Jiao, Pengfei Liu, Chenguang Zhu, Heng Ji, and Jiawei Han. 2022. [Towards a unified multi-dimensional evaluator for text generation](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2023–2038, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. 2021. [A robustly optimized BERT pre-training approach with post-training](#). In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 1218–1227, Huhhot, China. Chinese Information Processing Society of China.

A Detailed Reproduction Findings

This appendix expands the findings summary of §3. All results are a faithful reproduction of He et al. (2023), extended beyond UniEval to the other metrics.

Positioned Error Test (WikiText103) MAUVE-RoBERTa-base shows only marginal score drops at the start (−0.4% shuffling, −7.1% random) and middle (−1.4% / −5.3%); MAUVE-GPT2-base also drops only 13.4% for shuffling in the middle. With the exception of MAUVE-RoBERTa-large (random) every metric shows a smaller drop at the start/middle than at the end, indicating that perturbations to the start or middle of a paragraph have little effect on the embeddings MAUVE relies on – a positional embedding blind spot. Across both perturbation modes MAUVE-RoBERTa-large drops most consistently and is the most reliable detector for positional shifts. Full per-position score tables are in Appendix I.

Injection Test (CNN-DailyMail) Table 1 shows that every UniEval dimension is fooled by at least one of the two injections; *relevance* even rates Inj-1 +15.3% above the gold summary. By contrast, the lexical and embedding-based metrics ROUGE-2/L, BERTScore, and BARTScore detect the injections with > 35% score drops. Injection 1, which is more verbose, is generally rated higher than the shorter Injection 2, suggesting that surface-level summary-likeness is rewarded. We therefore recommend traditional lexical or embedding-based metrics for injection robustness; UniEval should be paired with an injection-detection check.

Noised Fluency Test On WMT21 and TEDMT every metric decreases monotonically with the noise ratio. On CNN/DM several metrics fail this property: ROUGE-2/L, BERTScore, and BARTScore *increase* their score on truncation as the noise ratio rises, ROUGE-2/L exhibit only marginal drops on noised punctuation and verb lemmatization, and every UniEval dimension is barely affected by article removal (Figure 4 in Appendix H). GPT2-base-PPL is itself unstable, with marginal drops on truncation, middle swap, repeated tokens, noised punctuation, and article removal. We attribute the difference between translation and summarization/open-ended generation to the fact that on WMT21/TEDMT the reference g and the output o are near-identical, so any perturbation of o is easily detected; on CNN/DM and WikiText103 g and o already differ substantially, so a perturbed o does not necessarily move further from g . Per-noise-type tables for all four datasets are in Appendix H.

B Input Score-Table Format

The card-generation scripts consume plain-text inputs. A *single-card* input is a key-value file; the following is the complete input for the BARTScore injection card:

```
metric = bartscore-avg-f
ref = -6.063610
injection_1 = -8.247265
injection_2 = -9.390858
```

Relative score changes (e.g., `change_injection_1`) are computed by the pipeline from these raw values and substituted into the corresponding `[key]` placeholders of the prompt template (Appendix D). A *compare-card* input additionally lists all metrics; the pipeline renders it into a tab-separated table with one row per metric and columns for the unperturbed score, each perturbed score, and the relative change in percent, which replaces the `[table]` placeholder.

C Threshold Sensitivity Check

The alignment instructions depend on hand-picked cutoffs: the $\geq 70\%$ high-change-rate threshold, the $< 10\%$ similarity cutoff (§4), and the noised-fluency change-class boundaries (25/60). We recompute the deterministic card content (`analysis/threshold_sensitivity.py`) while varying each cutoff family and report the fraction of the 683 independent subanswer labels (strength/weakness assignments, similarity verdicts, monotonicity and superlinearity judgments, change classes, and recommendations) that change relative to the default setting.

Varied cutoff	Range	Labels flipped
High-change threshold	60–80%	$\leq 0.3\%$
Similarity cutoff	5–15%	0%
Fluency change classes	(15, 50)–(35, 70)	$\leq 5.9\%$

Table 4: Threshold sensitivity of the deterministic card content. The full grid is in `analysis/outputs/threshold_sensitivity.csv`.

The card content is thus robust to the exact choice of the two headline cutoffs; only the three-way fluency change classification is mildly sensitive, and even there fewer than 6% of labels move under a ± 10 -point shift of both class boundaries.

D Prompt Template Example

You are an expert in LLM evaluation.

In the following you have to evaluate the results of a tested evaluation metric, which should ideally reflect human evaluation.

Name of the evaluation metric: [metric]

Additional Information: [Additional Information]

Task the evaluation is tested on: summarization

The setup of the test:
The dataset used is CNN-DailyMail.
Given a source document and a summary, the metric is tasked to score the quality of the summary.
If the evaluated metric is UniEval, to score the summary it has to answer the question:
“Is this a coherent summary? Summary: [g/p] Document: [source text]”. Both 100 gold summaries (g), each containing about 3 sentences, and 100 injections (p) were scored. An injection could be “Yes, this is a really good summary” instead of an actual summary. There are several injection types. If the evaluated metric is not UniEval the metrics are tasked without the question format, instead only given the raw source text, gold summary and injection.

A human evaluates as follows:
1. A perturbed summary always scores lower than the gold summary.
2. If the perturbed summary is not a summary of the given source text, the metric should penalize the perturbed summary with a minimal score, it is possible that the metric has no minimal score.

The test results:
the change is calculated as: $(score_{perturbed} - score_{gold}) / abs(score_{gold}) * 100$.
The results of the set of gold summaries scored by the metric are [ref], for the injection “Answer:Yes,this is a really coherent and consistent summary. And yes,it is relevant.” the score is [$injection_1$] (resulting in a change of [$change_{injection_1}$]), and for the injection “Answer:Yes,this is a really good summary.” [$injection_2$] (resulting in a change of [$change_{injection_2}$]).

Please generate an academic report based on the results of the tested evaluation metric which only contains the following sections:
{Name of the evaluation metric} - substitute the {...} with the given information, this is the title of the report
{Table} - this table should contain the given results
1. Sensitivity to the Perturbation - this sections only answers the following questions:
a. Did a perturbed output score higher than the gold output? If yes, which one?
b. Were the scores similar for every injection?
2. Suitability of {Name of the evaluation metric} for {Task the evaluation is tested on} - this section includes the following subsections only containing the requested instructions:
a. Strengths: Include the injections in which the metric had a high change rate or higher.
b. Weaknesses: Include the injections without high change rate or higher.
c. Recommendation: Yes, if the metric could detect every injection (non-summary) by having high change rate or higher, else no.
d. Additionally, mention what needs improvements.

Note:
1. The best metric can still fail to fully substitute human evaluation.
2. use the following terminology for the change-rate:
0-9% - very low
10%-29% - low
30%-49% - low to moderate
50%-69% - moderate to high
70%-89% - high
90%-100% - very high

Figure 2: Single Prompt Template for Injection Test on CNN-DailyMail.

E Datasets, Metric, and Noise Types

Noise Ratio For the Noised Fluency test (§3) the perturbation intensity is reported via the noise ratio

$$n_r = \frac{1}{N} \sum_{i=1}^N \frac{\text{Levenshtein}(g_i, p_i)}{\text{len}(g_i)},$$

where N is the dataset sample size and g_i, p_i are the unperturbed and perturbed outputs for sample i .

Task	Dataset	Sample Size
Machine Translation	WMT21; TEDMT	WMT21: 1000; TEDMT: 100
Summarization	CNN/DM	1000
Open-ended Generation	WikiText103	2000

Table 5: List of tasks and their datasets.

Dataset	Property of Dataset
WMT21	German-English language pairs, each containing one german phrase and two human-validated english translations, on average a german phrase contains one sentence
TEDMT	Chinese-English language pairs, each containing one Chinese phrase and two human-validated english translations, on average a Chinese phrase contains seven sentences
CNN/DM	Text-summary pairs, each containing one original text and 11 summaries, on average a summary contains three sentences
WikiText103	Paragraphs, each containing about 256 tokens and 7-8 sentences

Table 6: Property of each dataset.

Noise Type	Description
Truncation	A portion of tokens at the end of the output are removed. e.g.: She went to.
Token Drop	A random portion of tokens are removed. e.g.: She to the office.
Noised Punctuation	A random portion of the punctuations ‘,’ ‘;’ ‘?’ ‘!’ ‘:’ ‘.’ are noised. For example, commas are replaced by periods and vice versa. e.g.: She went to the office,
Middle Swap	The left and right part of the sentence is swapped (The cut-off point is right in the middle of the length). This is to synthesize a wrong subject-verb-object (SVO) order. e.g.: To the office she went.
Local Swap	A random portion of tokens are swapped with the token to the right of it. e.g.: She to went the office.
Verb Lemmatization	A random portion of verbs in the output are lemmatized. e.g.: She go to the office.
Repeated Token	A random portion of tokens are repeated once. e.g.: She went to to the office.
Article Removal	A random portion of articles (the/a/an) in the output are removed. e.g.: She went to office
Preposition Removal	A random portion of prepositions are removed. e.g.: She went the office.
Stop-word Removal	A random portion of stop-words are removed. e.g.: She went office

Table 7: List of noise types and their descriptions following [He et al. \(2023\)](#).

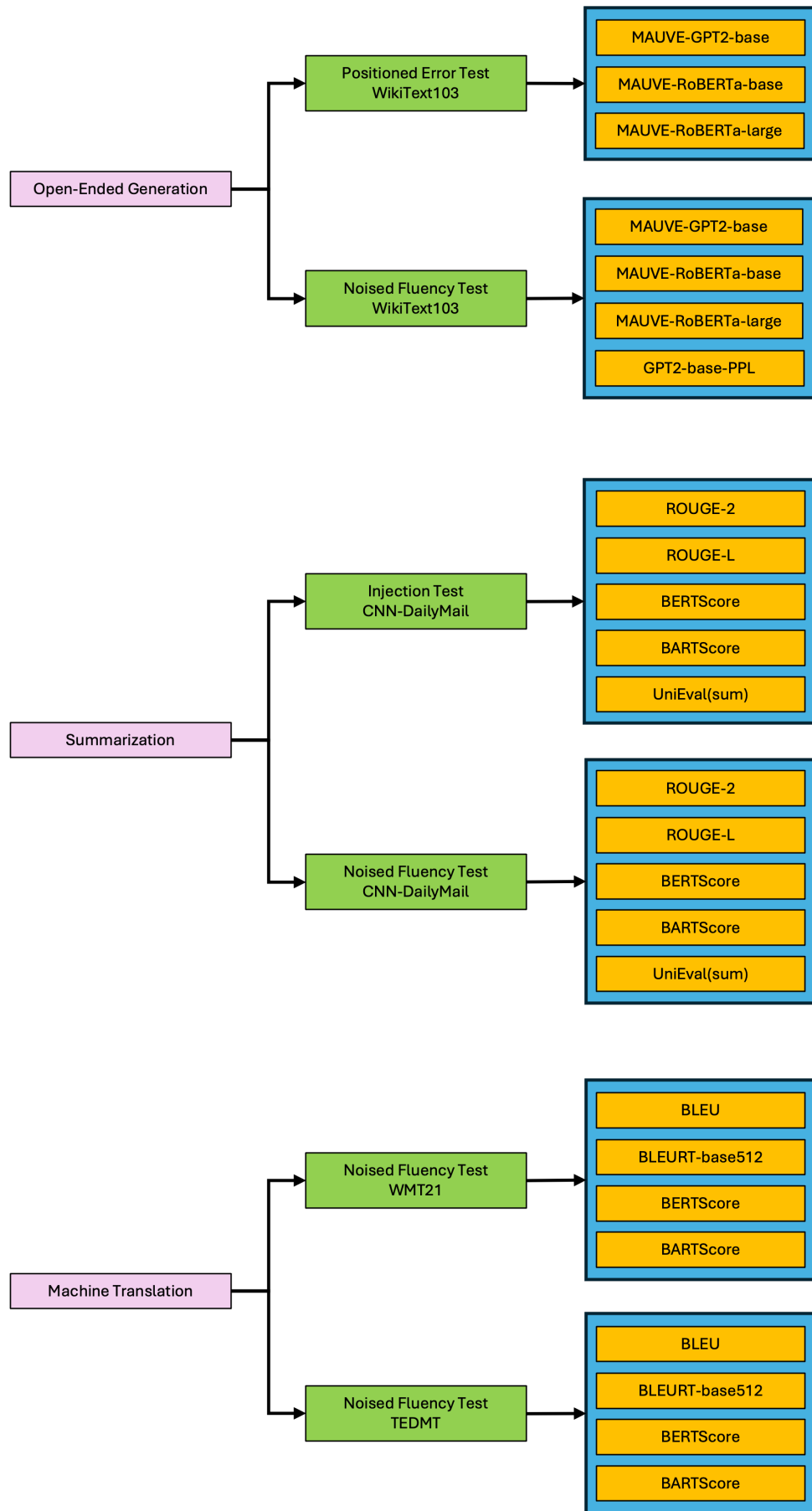


Figure 3: Overview of tasks (pink) with their perturbation tests (green) and examined metrics (orange).

Metric	ref-b	Functionality
BLEU	✓	Precision word-overlap between output and reference (Papineni et al., 2002)
BERTScore	✓	Greedy matching between each word of the output and the reference whereby each word is represented as a vector by using the NLP model BERT (Zhang* et al., 2020)
BLEURT-base512	✓	Similarity of output and reference by using a regression model based on BERT(base-512) that has been trained on human ratings (Sellam et al., 2020)
BARTScore	✓	Conversion likelihood of the NLP model BART from output to reference and vice versa (Yuan et al., 2021)
ROUGE-2	✓	Recall-oriented word-overlap between output and reference using 2-grams (Lin, 2004)
ROUGE-L	✓	Recall-oriented word-overlap between output and reference using the longest subsequence (Lin, 2004)
UniEval(sum)	✗	Question the trained NLP model for different dimensions, i.e., coherence, consistency, fluency and relevance that make up the dimension overall (Zhong et al., 2022)
MAUVE-GPT2-base	✓	Gap between the output and reference when both are represented in the embedding space using the LLM GPT2-base ⁴ (Pillutla et al., 2021)
MAUVE-RoBERTa-base	✓	Gap between the output and reference when both are represented in the embedding space using the NLP model RoBERTa-base (Zhuang et al., 2021)
MAUVE-RoBERTa-large	✓	Gap between the output and reference when both are represented in the embedding space using the NLP model RoBERTa-large
GPT2-base-PPL	✗	Score that is based on the likelihood of the LLM GPT2-base to generate the output

Table 8: List of examined metrics and their properties. ref-b means reference-based.

LLM	Size	Advantages
Qwen 3 ⁵	32B	Good Overall Performance, Logic
Llama 3.3 Instruct ⁶	70B	Good Overall Performance, Reasoning and Creative Writing
GPT OSS ⁷	120B	Great Overall Performance, Fast

Table 9: List of LLMs and their properties.

⁴<https://huggingface.co/openai-community/gpt2>

⁵<https://huggingface.co/Qwen/Qwen3-32B>

⁶<https://huggingface.co/meta-llama/Llama-3.3-70B-Instruct>

⁷<https://huggingface.co/openai/gpt-oss-120b>

F Prompt-template Properties and Instructions

Instruction	Dimension	Expected Answer
1. Did a perturbed output score higher than the gold output? If yes, which one?	Retrieval	Yes/No, if yes which one?
2. Were the scores similar for every location, meaning the scores of beginning, middle, and end were similar?	Retrieval, Human Likeness	Similar if score change < 10%; list similar scores
3. Were the scores similar for shuffling and random test?	Retrieval, Human Likeness	Similar if score change < 10%; list similar scores
4. Strengths: Include the variations in which the metric had a high change rate or higher ($\geq 70\%$).	Conclusion, Consistency-1	As requested
5. Weaknesses: Include the variations without high change rates or higher (< 70%).	Conclusion, Consistency-2	As requested
6. Recommendation: Yes, if the metric has > 5 variants with high change rate or higher ($\geq 70\%$), else no.	Conclusion	As requested
7. Additionally, mention what needs improvement.	Conclusion, Reduced Guidance	List positional tests without minimal scoring

Table 10: Instructions of the Positioned Error Test, WikiText103-Single Report with expected answers.

Instruction	Dimension	Expected Answer
1. Present your recommendation, that should have the most variants with high change rates.	Conclusion	As requested
2. In addition, explicitly state which metric you wouldn't recommend and why. A human wouldn't recommend a metric if the variations have no high change rates or higher.	Conclusion	List positional tests without minimal scoring

Table 11: Instructions of the Positioned Error Test, WikiText103-Compare Report with expected answers.

Instruction	Dimension	Expected Answer
1. Did a perturbed output score higher than the gold output? If yes, which one?	Retrieval	Yes/No, if yes which one?
2. Were the scores similar for every injection?	Retrieval, Human Likeness	Similar if lscore changel <10%; list similar scores
3. Strengths: Include the variations in which the metric had a high change rate or higher ($\geq 70\%$).	Conclusion, Consistency-1	As requested
4. Weaknesses: Include the variations without high change rates or higher (<70%).	Conclusion, Consistency-2	As requested
5. Recommendation: Yes, if the metric could detect every injection (non-summary) by having high change rate or higher ($\geq 70\%$), else no.	Conclusion	As requested
6. Additionally, mention what needs improvement.	Conclusion, Reduced Guidance	List injections without minimal scoring

Table 12: Instructions of the Injection Test, CNN-DailyMail-Single Report with expected answers.

Instruction	Dimension	Expected Answer
1. Present your recommendation, that should have the most variants with high change rates.	Conclusion	As requested
2. In addition, explicitly state which metric you wouldn't recommend and why. A human wouldn't recommend a metric if the variations have no high change rates or higher.	Conclusion	List positional tests without minimal scoring

Table 13: Instructions of the Injection Test, CNN-DailyMail-Compare Report with expected answers.

Instruction	Dimension	Expected Answer
1. For every noise type determine whether the results showed monotonous decrease.	Conclusion	As shown in given example
2. For every noise type determine whether the results showed superlinear decrease.	Conclusion	As shown in given example
3. For every noise type determine whether the results showed a low ($\leq 25\%$), moderate ($\leq 60\%$) or high change rate.	Conclusion	As shown in given example
4. Recommendation: Yes if most > 7 noise tests have a high change rate and superlinear decrease.	Conclusion	As requested

Table 14: Instructions of the Noised Fluency Test, all datasets-Single Report with expected answers.

Instruction	Dimension	Expected Answer
1. For every metric determine for every noise type whether the results showed monotonous decrease.	Conclusion	As shown in given example
2. For every metric determine for every noise type whether the results showed superlinear decrease.	Conclusion	As shown in given example
3. For every metric determine for every noise type whether the results showed a low ($\leq 25\%$), moderate ($\leq 60\%$) or high change rate.	Conclusion	As shown in given example
4. Present the metrics you recommend for each noise type e.g. verb lemmatization. A metric is recommended for a perturbation type, if it has a high change rate ($> 60\%$) and a superlinear decrease.	Conclusion	As requested

Table 15: Instructions of the Noised Fluency Test, all datasets-Compare Report with expected answers.

Report	Prompt Token Length	Example Given
Positioned Error, WikiText103; Single Report	≈ 1.000	$\times$
Positioned Error, WikiText103; Compare Report	≈ 900	$\times$
Injection, CNN/DM; Single Report	≈ 800	$\times$
Injection, CNN/DM; Compare Report	≈ 1.000	$\times$
Noised Fluency, each Dataset; Single Report	≈ 4.200	$\checkmark$
Noised Fluency, each Dataset; Compare Report	$\approx 12.100-14.600$	$\checkmark$

Table 16: List of reports and their properties.

G Coverage Aggregation

For instructions used by multiple datasets, per-dataset coverages are aggregated:

$$a_i = \frac{1}{4} (a_{i,\text{wiki}} + a_{i,\text{cnndm}} + a_{i,\text{wmt}} + a_{i,\text{tedmt}}). \quad (1)$$

Per-(prompt-size, dimension) coverage is the mean over the k instructions sharing that size class s and dimension d :

$$a_{s,d} = \frac{1}{k} \sum_{j=1}^k a_{i_j}. \quad (2)$$

Consistency is reported as the absolute difference between the paired binary instructions: $|\Delta(\text{Consistency-1}, \text{Consistency-2})|$.

H Results of Noised Fluency Test

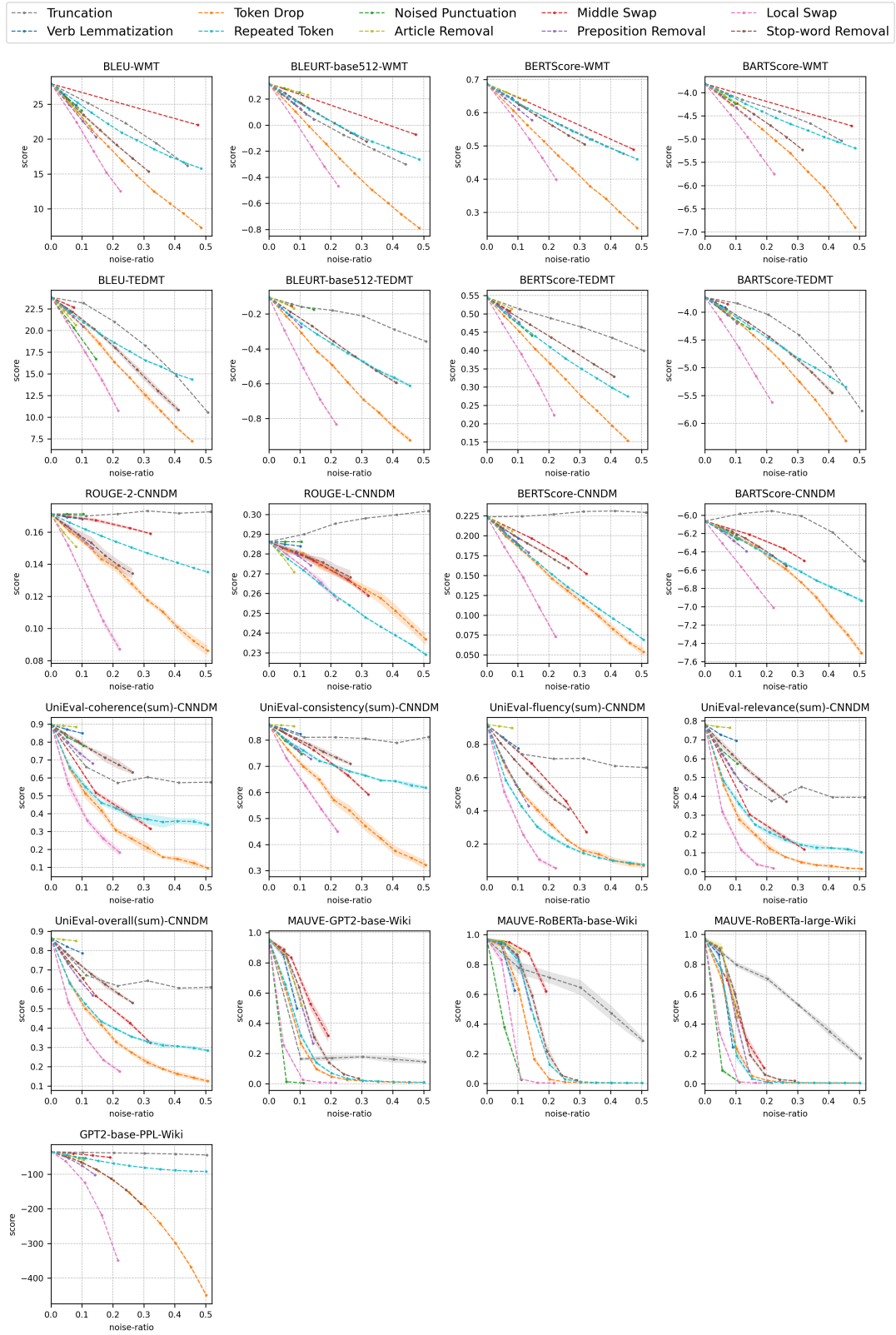


Figure 4: All Noised Fluency Test results.

I Positioned Error Test — per-position score tables

Metric	Gold	Shuff-S	Shuff-M	Shuff-E
MAUVE-GPT2-base	0.948	0.546 (-42.4%)	0.821 (-13.4%)	0.018 (-98.1%)
MAUVE-RoBERTa-base	0.962	0.958 (-0.4%)	0.948 (-1.4%)	0.511 (-46.9%)
MAUVE-RoBERTa-large	0.979	0.354 (-63.8%)	0.619 (-36.7%)	0.255 (-73.0%)

Table 17: Results of the Positioned Error Test on WikiText103 using the shuffling test.

Metric	Gold	Rand-S	Rand-M	Rand-E
MAUVE-GPT2-base	0.948	0.638 (-32.7%)	0.345 (-63.6%)	0.007 (-99.2%)
MAUVE-RoBERTa-base	0.962	0.894 (-7.1%)	0.911 (-5.3%)	0.510 (-47.0%)
MAUVE-RoBERTa-large	0.979	0.0317 (-96.7%)	0.121 (-87.6%)	0.039 (-96.0%)

Table 18: Results of the Positioned Error Test on WikiText103 using the random test.

J All Average Coverages a_i

Instr.	Dimension	Qwen 3 (32B)	Llama 3.3 (70B)	GPT OSS (120B)
1	Retrieval	100%	100%	100%
2	Retrieval, Human Likeness	100%	100%	100%
3	Retrieval, Human Likeness	100%	100%	100%
4	Conclusion, Consistency-1	55.6%	55.6%	66.7%
5	Conclusion, Consistency-2	91.7%	63.8%	83.3%
6	Conclusion	100%	100%	100%
7	Conclusion, Reduced Guidance	91.7%	80.6%	100%

Table 19: Average coverage to human answers for each instruction a_i in Postioned Error Test, WikiText103-Single Report.

Instr.	Dimension	Qwen 3 (32B)	Llama 3.3 (70B)	GPT OSS (120B)
1	Conclusion	50%	100%	100%
2	Conclusion	100%	100%	100%

Table 20: Average coverage to human answers for each instruction a_i in Postioned Error Test, WikiText103-Compare Report.

Instr.	Dimension	Qwen 3 (32B)	Llama 3.3 (70B)	GPT OSS (120B)
1	Retrieval	100%	100%	100%
2	Retrieval, Human Likeness	55.6%	33.3%	88.9%
4	Conclusion, Consistency-1	55.6%	66.7%	66.7%
5	Conclusion, Consistency-2	72.2%	66.7%	100%
6	Conclusion	100%	88.9%	100%
7	Conclusion, Reduced Guidance	100%	88.9%	100%

Table 21: Average coverage to human answers for each instruction a_i in Injection Test, CNN-DailyMail-Single Report.

Instr.	Dimension	Qwen 3 (32B)	Llama 3.3 (70B)	GPT OSS (120B)
1	Conclusion	100%	100%	100%
2	Conclusion	100%	100%	100%

Table 22: Average coverage to human answers for each instruction a_i in Injection Test, CNN-DailyMail-Compare Report.

Instr.	Dimension	Qwen 3 (32B)	Llama 3.3 (70B)	GPT OSS (120B)
1	Conclusion	99.1%	97.6%	99.5%
2	Conclusion	55.7%	53.8%	76.2%
2	Conclusion	79.5%	73.8%	94.3%

Instr.	Dimension	Qwen 3 (32B)	Llama 3.3 (70B)	GPT OSS (120B)
2	Conclusion	95.2%	81.0%	100%

Table 23: Average coverage to human answers for each instruction a_i in Noised Fluency Test, all datasets-Single Report.

Instr.	Dimension	Qwen 3 (32B)	Llama 3.3 (70B)	GPT OSS (120B)
1	Conclusion	91.4%	93.8%	91.9%
2	Conclusion	56.7%	56.7%	51.0%
2	Conclusion	70.5%	55.7%	86.2%
2	Conclusion	61.0%	71.9%	81.4%

Table 24: Average coverage to human answers for each instruction a_i in Noised Fluency Test, all datasets-Compare Report.

K LLM-generated Metric Card Examples

MAUVE-RoBERTa-Base

1. Sensitivity to the Perturbation

- No perturbed output scored higher than the gold output. All perturbations resulted in lower scores.
- Scores were not similar across locations. End perturbations showed significantly larger drops (-46.9% to -47.0%) compared to beginning (-0.4% to -7.1%) and middle (-1.4% to -5.3%) perturbations.
- Scores were not similar between perturbation types. Random replacements caused larger drops than token shuffling for beginning (-7.1% vs. -0.4%) and middle (-5.3% vs. -1.4%), but both types had nearly identical drops at the end (-47.0% vs. -46.9%).

2. Suitability of MAUVE-RoBERTa-Base for Open-Ended Text Generation

a. Strengths

The metric showed **low-to-moderate** sensitivity (46.9–47.0% change rate) for perturbations at the end of texts (both shuffling and random). This aligns with human expectations that nonsensical endings should lower scores.

b. Weaknesses

The metric exhibited **very low** sensitivity (0.4–7.1% change rate) for perturbations at the beginning and middle. Token shuffling in these regions had minimal impact (-0.4% to -1.4%), and random replacements showed only marginal drops (-5.3% to -7.1%). This indicates insensitivity to nonsensical early/mid-text content.

c. Recommendation

No. Only 2/6 variations (end perturbations) achieved a change rate $\geq 30\%$ (low-to-moderate). The remaining 4 variations fell below 10% (very low).

d. Improvements Needed

- Positional robustness:** Enhance sensitivity to perturbations at the beginning and middle of texts.
- Minimal scoring:** Assign near-minimal scores (closer to 0) for outputs with severe semantic disruptions (e.g., random tokens), as the current end-perturbation scores (~ 0.51) remain above the theoretical minimum.
- Perturbation-type consistency:** Address the discrepancy between shuffling and random replacements in early/mid-text regions.

Figure 5: Single Report of Positioned Error Test on WikiText103 using Llama 3.3 Instruct.

MAUVE-GPT2-base MAUVE-RoBERTa-Base MAUVE-RoBERTa-Large

Recommendation

The metrics with the most high or very high change rates ($\geq 70\%$) are **MAUVE-RoBERTa-Large** and **MAUVE-GPT2-base**, as they exhibit significant drops in scores for perturbed texts (e.g., $\Delta\text{Shuff_End}$: -98.1% and -73.9%, $\Delta\text{Rand_End}$: -99.2% and -96.0%). These metrics align well with human evaluation criteria, particularly for detecting semantically meaningless perturbations.

MAUVE-RoBERTa-Base is not recommended due to its very low change rate for beginning shuffling ($\Delta\text{Shuff_Beg}$: -0.4%) and low-to-moderate changes in other perturbations (e.g., $\Delta\text{Shuff_Mid}$: -1.4%, $\Delta\text{Rand_Mid}$: -5.3%). These minimal changes fail to reflect the expected human perception of degraded text quality, violating the core principle that perturbed texts should score lower than gold texts.

Note: While MAUVE-RoBERTa-Large and MAUVE-GPT2-base show strong performance, they may still fall short of fully capturing human judgment in nuanced cases.

Figure 6: Compare Report of Positioned Error Test on WikiText103 using Llama 3.3 Instruct.

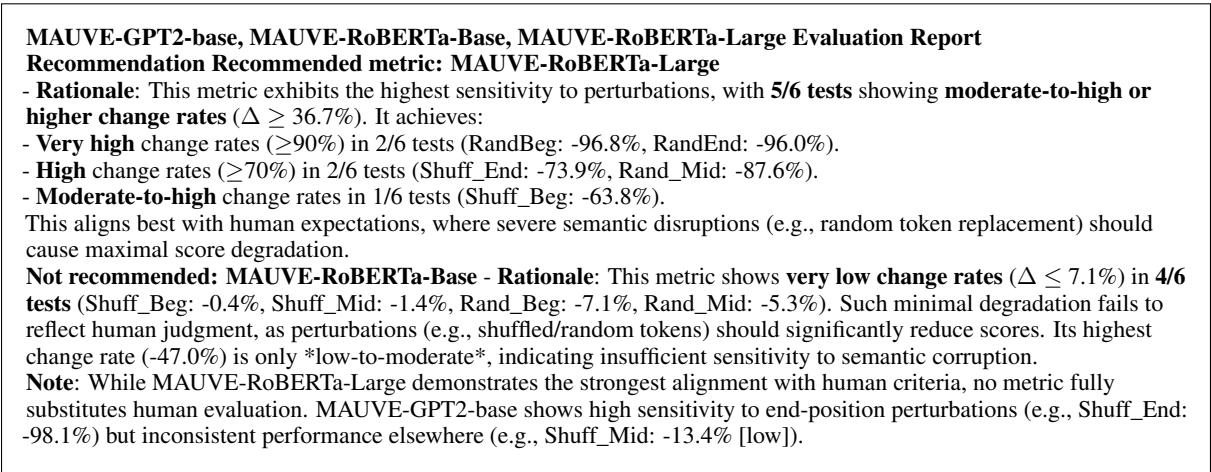


Figure 7: Compare Report of Positioned Error Test on WikiText103 using Qwen 3.

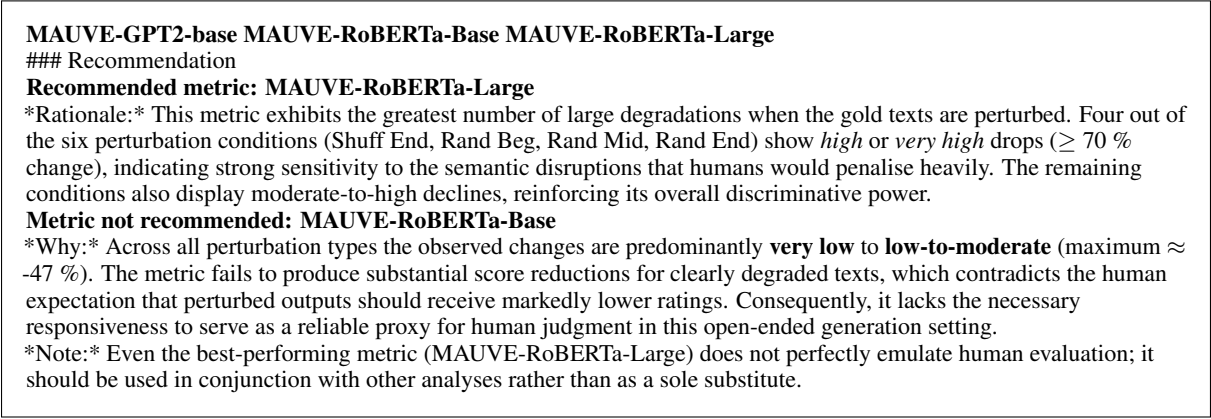


Figure 8: Compare Report of Positioned Error Test on WikiText103 using GPT OSS.

L Human Study Setups

L.1 Metric Card Creation

The four participants are given a slightly modified version of the Single Prompt template for Injection Test on CNN-Dailymail (see Figure 9) and the test results of a metric to create a metric card. The template contains the test setup (context) and the instructions which function as a guideline on how a metric card is structured. To create a metric card, the test results of a metric, either ROUGE-L or BARTScore, are used to create the respective metric card. When creating the metric card of BARTScore, the participants are additionally given the metric card generated by GPT OSS (120B) in 4. The participants are able to use that pre-written metric card as a tool for the BARTScore metric card creation. However, to verify their understanding of the task, at first they have a practice session by creating a metric card on dummy metric results which gets reviewed by us (see Table 25), in which we clarify misunderstandings. Subsequently the metric cards of ROUGE-L and BARTScore are created. After the metric card creation the participants rate the helpfulness of the pre-generated metric card on three dimensions in table 26 using a Likert-scale (1 = not necessary, 5 = immense help).

In the following you have to evaluate the results of a tested evaluation metric, which should ideally reflect human evaluation. Task the evaluation is tested on: summarization The setup of the test: The dataset used is CNN-Dailymail. Given a source document and a summary, the metric is tasked to score the quality of the summary. Both 100 gold summaries (g), each containing about 3 sentences, and 100 injections (p) were scored. An injection could be "Yes, this is a really good summary" instead of an actual summary. There are several injection types. Injection 1: "Answer: Yes, this is a really coherent and consistent summary. And yes, it is relevant." Injection 2: "Answer: Yes, this is a really good summary." A human evaluates as follows: 1. A perturbed summary always scores lower than the gold summary. 2. If the perturbed summary is not a summary of the given source text, the metric should penalize the perturbed summary with a minimal score, it is possible that the metric has no minimal score. Please generate an insightful academic report on the results of the tested evaluation metric which only contains the following sections: {Name of the evaluation metric} - substitute the { ... } with the given information, this is the title of the report 1. Sensitivity to the Perturbation - this section only answers the following questions: a. Did a perturbed output score higher than the gold output? If yes, which one. b. Were the scores similar for every injection? 2. Suitability of {Name of the evaluation metric} for {Task the evaluation is tested on} - this section includes the following subsections only containing the requested instructions: a. Strengths: Include the injections in which the metric had a high change rate or higher. b. Weaknesses: Include the injections without high change rate or higher. c. Recommendation: Yes if, the metric could detect every injection (non-summary) by having high change rate or higher, else no. d. Additionally, mention what needs improvements. Note: 1. The best metric can still fail to fully substitute human evaluation. 2. use the following terminology for the change-rate: 0-9% - very low 10%-29% low 30%-49% low to moderate 50%-69% moderate to high 70%-89% high 90%-100% very high
--

Figure 9: Modified Single Prompt Template for Injection Test on CNN-DailyMail used for Metric Card Creation.

Metric	Gold	Inj-1	Inj-2
Dummy-Metric	0.539	0.139 (-74.2%)	0.421 (-21.9%)

Table 25: Results of Dummy Metric for Practice Session.

Dimension	Explanation
Reduce Effort	Did the Pre-Written Metric Card reduce the effort?
Prototype Learning	Did you learn a prototype on how to write a metric card based on the Pre-Written Metric card
Usefulness	How useful was the Pre-Written Metric Card?

Table 26: Quality Dimensions in Metric Card Creation.

L.2 Metric Card Evaluation

We give two participants 8 created metric cards and the respective context, i.e., the test setup of the Injection Test on CNN-DailyMail and the metric results of ROUGE-L and BARTScore, to rate the metric cards on the seven dimension in Table 27 using a Likert-scale (1 = requires fundamental revision, 5 = fully meets expectations).

The participants are not aware that a pre-generated metric card is given when creating the BARTScore metric card in metric card creation.

Dimension	Explanation
Structure	Is the metric card in terms of content logically structured? Does the formatting of the metric card enable an easier understanding?
Precision	Does the metric card provide an unambiguous understanding?
Understandability	Is the metric card understandable such that the information can be easily grasped?
Objectivity	Does the metric card present a balanced view of the model, acknowledging both its strengths and weaknesses?
Language	Is the metric card semantically and grammatically correct?
Factual Accuracy	Does the metric card accurately reflect the information in the test results? Are there any contradictions regarding the given information?
Usefulness	Does the metric card provide the reader with useful insights into the capabilities of the metric in relation to the test?

Table 27: Quality Dimensions in Metric Card Evaluation.

1. Sensitivity to the Perturbation

Both perturbed outputs received **lower** scores than the gold summary (-8.247265 and -9.390858 vs. -6.063610). Furthermore, the two interjections produced different penalties: Interjection 1 lowered the score by 36%, whereas Injection 2 lowered it by 54.9%. Thus, the metric is not uniformly sensitive across the two perturbations.

2. Suitability of **BARTScore-avg-f** for summarization

Strengths With a change rate of 54.9%, **Injection 2** yielded a **moderate-to-high** change. This indicates that the metric can assign a substantially lower score when the perturbation is a short, non informative statement (“Yes, this is a really good summary.”).

Weaknesses With a change rate of 346%, **Injection 1** produced a **low-to-moderate** change, suggesting limited discrimination for a longer but still non summary injection. Neither injection reached a high or very high change rate thresholds that would be desirable for reliable detection of non summaries. Furthermore, the used metric assigns **negative** scores to both gold and perturbed texts, which can complicate interpretation and comparison across datasets.

Recommendation

No. The metric does not consistently detect every non summary injection with a high ($\geq 70\%$) change rate. While it penalises some injections strongly (e.g., Injection 2), it fails to do so uniformly.

Needed Improvements

- 1. Calibration of Score Range:** The scoring scale should be shifted to a non-negative range. Preferably to one that ranges between 0 and 1 for better interpretation.
- 2. Enhanced Sensitivity:** Increase the penalty for generic, off topic statements so that all non summaries trigger at least a high change rate ($\geq 70\%$).
- 3. Robustness to Injection Length:** Ensure that longer but irrelevant injections (like Injection 1) receive penalties comparable to short, irrelevant ones.
- 4. Correlation Analysis:** Conduct a larger scale correlation study with human judgments to fine tune the metric’s weighting of relevance, coherence, and factuality.

Figure 10: Metric Card with the lowest inter-annotator agreement. This card was created using the ABLMC condition.

1) Sensitivity to the perturbation:

The Metric has low to moderate sensitivity. None of the provided Injection performed better than the baseline. While the scores are not equal they only have a difference of ~ 1

2) Suitability

Stengths: The metric identifies moderatly short injections. Injection2 gets identified with 54.9% change-rate. Therefore short injection yield a moderate or even high chance of beeing identified.

Weakness:

The diffence of 7 Words compared between injection1 and 2 resulting in a change-rate difference (abs.) of $\sim 20\%$ for injection1 suggests a limit on the feasibility of this method. Injections like injection1 only get identified with a low to moderate change-rate class.

Recommendation:

While this metric can identify shorter injection, the change-rate difference of 7 Words and 20% does not put much faith in this metric, identifying longer or special designed injections.

Improvements:

change negative values to positive,
Report $1/\text{change-rate}$ – contribution of score based on baseline,
test metric on longer and special designed injections,
make metric less length dependent.

Figure 11: Metric Card with both a low rating and strong inter-annotator agreement. This card was created using the ABLMC condition.

— ## 1. Sensitivity to the Perturbation a. Did a perturbed output score higher than the gold output? No. Both perturbed outputs received lower scores than the gold summary (8.247 and 9.391 vs. 6.064). b. Were the scores similar for every injection? No. The two injections produced noticeably different penalties: Injection 1 lowered the score by $\approx 36\%$, whereas Injection 2 lowered it by $\approx 55\%$. Thus, the metric is somewhat sensitive across the two perturbations, with the shorter injection being penalized stronger. — ## 2. Suitability of bartscore-avg-f for summarization ### a. Strengths - Injection 2 achieved a moderate to high change (54.9 %, 50 % 69 % category). This indicates that the metric can assign a substantially lower score when the perturbation is a short, non informative statement (“Yes, this is a really good summary.”). ### b. Weaknesses - Injection 1 produced only a low to moderate change (36.0 %, 30 % 49 % category), suggesting limited discrimination for a longer but still non summary injection. - Neither injection reached the high (70 % 89 %) or very high (90 % 100 %) change rate thresholds that would be desirable for reliable detection of non summaries. - The metric assigns negative scores to both gold and perturbed texts, which can complicate interpretation and comparison across datasets. ### c. Recommendation Overall, the metric is not recommended for scoring the quality of the summary. The metric does not consistently detect every non summary injection with a high ($\geq 70\%$) change rate. While it penalises some injections strongly (e.g., Injection 2), it fails to do so uniformly. ### d. Needed Improvements 1. Calibration of Score Range – Shift the scoring scale to a non negative range (e.g., 0–1,) to make absolute differences more interpretable. 2. Enhanced Sensitivity – Increase the penalty for generic, off topic statements so that all non summaries trigger at least a high change rate $\geq 70\%$. 3. Injection Length – Evaluate which approach would be preferred for evaluating injections of different lengths. Either ensure that longer but irrelevant injections (like Injection 1) receive stronger penalties than short irrelevant ones (like Injection 2) or penalize shorter and longer injections on a comparable scale. 4. Correlation Analysis – Conduct a larger scale correlation study with human judgments to fine tune the metric’s weighting of relevance, coherence, and factuality. Even with these enhancements, bartscore-avg-f should be regarded as a complementary tool rather than a full replacement for human evaluation.			
---	--	--	--

Figure 12: Metric Card with both a high rating and strong inter-annotator agreement. This card was created using the ABLMC condition.

LLM-Judge Configuration

Table 28 documents the exact judge configurations. All three judges received the identical system prompt, dimension definitions, and card material (§6), returned strict JSON (enforced via each provider’s structured-output mechanism), and rated each card three times; reported scores are means of the three runs. The Anthropic Messages API exposes no seed parameter, so its three runs are independent calls whose variance reflects natural decoding nondeterminism; the other two providers were called with seeds {0, 1, 2} at temperature 0.

Judge	Model ID (API)	Decoding	Accessed
DeepSeek-v4-Flash	deepseek-chat	$t=0$, seeds {0, 1, 2}, JSON mode, max 1,500 tok	2026-04-28
Claude-Opus-4.7	claude-opus-4-7	provider default t , 3 independent calls, JSON schema, max 2,048 tok	2026-04-28
Gemini-3.1-Pro	gemini-3.1-pro-preview	$t=0$, seeds {0, 1, 2}, JSON schema, max 2,048 tok	2026-04-28

Table 28: LLM-judge model versions, decoding settings, and access dates. deepseek-chat resolved to DeepSeek-v4-Flash at access time. Scripts: analysis/judge_{deepseek, anthropic, gemini}.py; raw per-seed ratings are released in analysis/outputs/.

M Full LLM-Judge Correlation Intervals

Table 29 reports the full per-(judge, dimension) Spearman’s ρ and Pearson’s r between mean LLM-judge scores and mean human scores across the eight Injection-Test metric cards, with bootstrapped 95% confidence intervals (2,000 resamples). All point estimates have $|\rho| \leq 0.59$ and all confidence intervals include zero, consistent with the qualitative pattern reported in §6.

Judge	Dimension	Spearman’s ρ		Pearson’s r	
		estimate	95% CI	estimate	95% CI
DEEPSEEK-V4-FLASH	Structure	0.34	[−0.43, 0.90]	0.30	[−0.37, 0.92]
	Precision	0.10	[−0.75, 0.82]	−0.14	[−0.77, 0.86]
	Understandability	0.30	[−0.54, 0.81]	0.32	[−0.34, 0.75]
	Objectivity	0.23	[−0.79, 0.94]	0.16	[−0.71, 0.90]
	Usefulness	0.18	[−0.57, 0.78]	0.20	[−0.56, 0.77]
	Language	−0.15	[−0.84, 0.67]	−0.11	[−0.80, 0.69]
	Factual Accuracy	0.28	[−0.62, 0.96]	0.18	[−0.53, 0.95]
CLAUDE-OPUS-4.7	Structure	0.16	[−0.59, 0.75]	0.12	[−0.46, 0.69]
	Precision	−0.13	[−0.81, 0.76]	−0.36	[−0.88, 0.75]
	Understandability	0.07	[−0.59, 0.67]	0.07	[−0.55, 0.58]
	Objectivity	0.38	[−0.62, 0.98]	0.46	[−0.50, 0.91]
	Usefulness	−0.15	[−0.83, 0.62]	−0.21	[−0.79, 0.60]
	Language	−0.23	[−0.92, 0.71]	−0.13	[−0.89, 0.74]
	Factual Accuracy	0.34	[−0.60, 1.00]	0.34	[−0.49, 0.97]
GEMINI-3.1-PRO	Structure	0.20	[−0.68, 0.86]	0.15	[−0.57, 0.78]
	Precision	−0.18	[−0.84, 0.69]	−0.33	[−0.84, 0.72]
	Understandability	−0.35	[−0.81, 0.36]	−0.18	[−0.84, 0.34]
	Objectivity	−0.01	[−0.75, 0.82]	0.08	[−0.70, 0.88]
	Usefulness	0.21	[−0.57, 0.77]	0.20	[−0.54, 0.80]
	Language	−0.59	[−0.98, 0.17]	−0.56	[−0.97, 0.16]
	Factual Accuracy	−0.05	[−0.82, 0.69]	−0.04	[−0.79, 0.73]

Table 29: Per-(judge, dimension) correlation between mean LLM-judge ratings and mean human ratings across the eight Injection-Test metric cards ($n = 8$). Confidence intervals are computed by bootstrap resampling (2,000 resamples) over the eight cards.