Confident but Wrong: A Constrained Decoding Diagnostic for Low-Resource Automatic Post-Editing

Isuru Wijesiri[1], Nisansa de Silva[2], Kavindu Warnakulasuriya[3]
Aloka Fernando[2], Surangika Ranathunga[4]
[1]WSO2, Sri Lanka [2]Dept. of CSE, University of Moratuwa, Sri Lanka
[3]National University of Singapore [4]Massey University, New Zealand
NisansaDdS@cse.mrt.ac.lk, alokaf@cse.mrt.ac.lk,
kavindu_warnakulasuriya@u.nus.edu, s.ranathunga@massey.ac.nz
Correspondence: imwijesiri@gmail.com

Abstract

Automatic Post-Editing (APE) for low-resource languages (LRLs) often fails to improve Machine Translation (MT), and the score alone cannot say why: whether more training would help, or whether the training data is too inconsistent to learn from. We introduce a black-box, inference-time diagnostic that tells these two cases apart without retraining or annotation. It varies an edit-distance penalty λ that drives the model from free editing towards copying the MT, and reads two signals: (1) the shape of the Translation Edit Rate (TER)-vs-λ curve, U-shaped if edits from the model reduce error and monotonically decreasing if none does; and (2) the ordering of constraint variants that trust model confidence to increasing degrees, which shows whether confidence tracks edit quality. Across decoder-only and encoder–decoder models on English–Sinhala, the diagnostic exposes two failure modes consistent with a heterogeneous post-edit signal as the underlying cause: **Binary Collapse**, where the model copies the MT or makes off-target edits, and **Confident Miscalibration**, where the confidence signals we test do not separate useful edits from unnecessary ones. The pattern holds on English–Marathi and English–Tamil, with the failure modes tracking the post-edit distribution rather than MT quality or language family. Beyond diagnosis, the curve shape prescribes a concrete next step for practitioners; in the favorable case, a static constraint yields a free inference-time accuracy gain. We release the first English–Sinhala (∼66k) and a new English–Tamil (∼39k) APE datasets with all code.

1 Introduction

Automatic Post-Editing (APE) corrects residual errors in Machine Translation (MT), so it should matter most where MT is weakest. As MT quality on high-resource pairs such as English–German (Chatterjee et al., 2018, 2020) has improved, APE research has shifted towards low-resource languages (LRLs); recent WMT shared tasks target low-resource pairs such as English–Marathi (Bhattacharyya et al., 2022, 2023; Zerva et al., 2024). Yet, low-resource APE has not delivered: in WMT 2023, no submitted system improved over the English–Marathi MT baseline (Bhattacharyya et al., 2023).

One possible source of that shortfall is the post-edit training signal: professional translators interleave error correction with stylistic refinement, mixing mandatory and discretionary changes into an inconsistent training signal (Chollampatt et al., 2020; Tebbifakhr et al., 2019). High-resource APE addresses this heterogeneity through synthetic-data scaling, curriculum pre-training, and word-level quality-estimation (QE) supervision (Deoghare and Bhattacharyya, 2022; Junczys-Dowmunt and Grundkiewicz, 2018). Resources for such techniques are absent for most LRLs (Ranasinghe et al., 2025; Deoghare et al., 2024). As we show (Section 6.1), even partial substitutes (fine-tuned QE, frontier-LLM probes, heterogeneity filtering) do not resolve this failure. Therefore, a practitioner whose fine-tuned checkpoint fails to beat MT cannot tell why: the model may carry a correction signal that more work could expose, or the data may not support APE at all. These call for opposite responses; the default one, building more infrastructure, assumes the first case even when the second may hold.

We introduce a black-box, inference-time diagnostic that tells these two cases apart, without further training or annotation. Where QE-guided APE uses external token-level annotations to choose which tokens to keep (Deoghare et al., 2025), we ask whether the model's own per-token confidence carries the same information. The diagnostic varies an edit-distance penalty λ that, as it grows, pushes the APE model from free editing to-

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 22838–22866
October 24-29, 2026 ©2026 Association for Computational Linguistics

wards copying the MT, and a single sweep across λ yields two signals. The first is the shape of the TER (Translation Edit Rate)-vs-λ curve: an edit is *net-positive* if it lowers TER against the reference, so the curve is U-shaped if such edits survived training and monotonically decreasing if none did. The second is the ordering of constraint variants that trust the model's confidence to increasing degrees (static, probability-scaled, entropy-scaled): if confidence identifies net-positive edits, the confidence-aware variants win; if it does not, the ordering reverses, and that reversal is itself the diagnostic. Together the two signals sort a checkpoint into one of four outcomes (Section 6.3): two let the practitioner deploy immediately, one of them with a static constraint that gives a free inference-time accuracy gain; the other two call for a more consistent training signal before retraining.

We instantiate the diagnostic on English–Sinhala APE across Gemma 3 1B and 4B (decoder-only) and NLLB-600M (encoder–decoder), and validate it two ways: a contrastive low-quality-MT condition where APE should succeed, and cross-lingual replication on English–Marathi (Bhattacharyya et al., 2022) and English–Tamil. The diagnostic exposes two failure modes, which we hypothesize share a common cause in the heterogeneous post-edit signal: **Binary Collapse**, where the model copies the MT or makes an off-target edit and loses nuanced, targeted editing; and **Confident Miscalibration**, where high token-level confidence does not separate unnecessary edits from useful ones. In summary, we contribute:

- A reusable, black-box, inference-time diagnostic built from two annotation-free signals, curve shape and constraint ordering (Section 4.4), with practical guidelines for acting on its output (Section 6.3).
- An empirical characterization of LRL APE failure under heterogeneous supervision, across three architectures and three language pairs (Section 5).
- The first English–Sinhala (∼66k) and a new English–Tamil (∼39k) 🤗 `APE datasets`, released with all `code`.

2 Related Work

APE under high- and low-resource conditions. High-resource APE is built on infrastructure that low-resource pairs lack: dual-encoder Transformers (Junczys-Dowmunt and Grundkiewicz, 2018), curriculum pre-training on millions of synthetic post-edits (Deoghare and Bhattacharyya, 2022), and word-level QE supervision (Deoghare et al., 2023), which together overcome the heterogeneity of professional post-edits.

The APE shared task moved from English–German (Chatterjee et al., 2018, 2020) to English–Marathi (Bhattacharyya et al., 2022, 2023) and, in 2024, to English–Hindi and English–Tamil within a quality-informed APE sub-task (Zerva et al., 2024). LLM-based APE has emerged in parallel, through few-shot prompting (Raunak et al., 2023) and error-guided prompting (Ki and Carpuat, 2024), but mostly on high-resource pairs; we probe its low-resource behavior in Appendix A. For most low-resource pairs, dedicated APE resources remain absent (Ranasinghe et al., 2025; Deoghare et al., 2024), and the failure modes of fine-tuned APE under the resulting heterogeneous post-edits remain unexamined.

Confidence as an edit-quality signal. The most directly related approach is Deoghare et al. (2025), who use external token-level QE annotations inside Grid Beam Search to preserve correct tokens during decoding; this requires word-level QE labels that are unavailable for most low-resource pairs, including English–Sinhala. Whether a model's own confidence can substitute for such external signals is unclear, and prior work on LLM uncertainty urges caution: networks grow overconfident after fine-tuning (Guo et al., 2017), token probabilities are poorly calibrated after RLHF (Kadavath et al., 2022), and training on model-generated data causes diversity collapse (Shumailov et al., 2024). Biasing APE towards the MT is itself an established modeling device: Huang et al. (2019) add a learned copy mechanism, and Lopes et al. (2019) apply a conservativeness penalty at decoding time, both to improve output quality. We use the same lever for a different purpose: swept from free editing to forced copying, the edit-distance penalty becomes a probe that reads whether edits are learnable and whether confidence tracks their quality. Constrained decoding more broadly has been explored for lexical and format constraints (Post and Vilar, 2018; Hokamp and Liu, 2017; Hu et al., 2019); repurposing it to compare confidence-trust variants as a calibration test is, to our knowledge, new.

3 Dataset

3.1 Dataset Creation

Our APE data comes from the EnSiTa English–Sinhala–Tamil dataset (Ranathunga et al., 2026), covering two under-resourced languages: in the categorization of Ranathunga and de Silva (2022), Sinhala is class 2 (low-resource) and Tamil class 3 (mid-resource), and Sinhala has the less mature NLP ecosystem (de Silva, 2026). The dataset was produced by professionally post-editing machine-translated or web-mined parallel corpora across several domains, from which we recover the (source, MT, post-edit) triplets that APE requires. EnSiTa itself releases only the final parallel corpus. We keep only triplets whose source is identical for the MT and the post-edit, discarding items whose source was itself edited during multi-round revision, which breaks the fixed-source APE setup (Appendix B.3). This gives the first ∼66k English–Sinhala APE triplets across four Google-Translate domains (D1: Literature, D2: News, D3: Wikipedia, D4: Maths and Health) and ∼39k English–Tamil triplets across five domains (Appendix B.4). Because EnSiTa post-editors were *permitted but not required* to make stylistic changes beyond minimal error correction, the resulting edits are heterogeneous, the property we analyze in Section 3.2.

We also construct D5, a contrastive low-quality-MT condition for this study: cleaned NLLB-corpus sources translated with NLLB-1.3B (NLLB Team et al., 2022), post-edited under the same protocol. For cross-lingual validation (Section 5.5) we use the WMT22 English–Marathi APE data (17,999 triplets; Bhattacharyya et al., 2022).

3.2 Edit Type Analysis: Heterogeneous, Not Consistent

Post-edits across our domains are **heterogeneous**: no single edit type dominates, and the mix varies by domain. Manual analysis of 90 randomly sampled (MT, PE) pairs across D1–D3 (30 per domain, stratified by TER bucket; Table 28 in Appendix D.1) shows the pattern: D1 (Literature) contains a mix of register shifts, loanword replacements, and structural reordering with no dominant category; D2 (News) is dominated by mechanical fixes (punctuation, spelling); D3 (Wikipedia) mixes style edits with fluency fixes such as transliteration of proper nouns.

We scale this analysis to 200 samples per domain using GPT-5.2 (OpenAI, 2025) with a simplified Multidimensional Quality Metrics (MQM-Core) taxonomy (Lommel et al., 2014; Freitag et al., 2021) (Section 4.3, Appendix D.2). On D1, 49.5% of post-edits are purely stylistic. Three native Sinhala speakers independently validate the classifier on 100 stratified D1 samples, each judging only the single primary category per sample, so inter-annotator agreement is measured by Fleiss' κ=0.70 (per-annotator breakdowns and confusion matrices in Appendix E). D1's post-edits are the most heterogeneous of our domains; D2 and D3 are progressively more consistent.

4 The Investigation

4.1 Task Formulation

We first fix notation for the APE task and the quantities the diagnostic reads. APE takes a source sentence $x^{\text{src}} = (x_1^{\text{src}}, \ldots, x_N^{\text{src}})$ and its machine translation $x^{\text{mt}} = (x_1^{\text{mt}}, \ldots, x_M^{\text{mt}})$, and produces a post-edit $\mathbf{y} = (y_1, \ldots, y_T)$. The post-edit is evaluated against a human reference $\mathbf{y}^\star$ using $\text{TER}(\mathbf{y}, \mathbf{y}^\star)$. At each step t, the fine-tuned APE model M produces a distribution $p_t(\cdot) = p(\cdot \mid x^{\text{src}}, x^{\text{mt}}, y_{<t})$ over vocabulary $\mathcal{V}$. Throughout, x_j^{mt} denotes the j-th MT token, and $\mathcal{W}_t \subseteq \{1, \ldots, M\}$ denotes the indices of a local alignment window of MT tokens centered on the current alignment pointer (defined in Section 4.4). Recall that an edit is *net-positive* if it reduces TER against the reference.

4.2 Models and Training

We first train each model on the APE triplets with standard supervised fine-tuning (*vanilla fine-tuning*), then apply constrained-decoding variants to the same checkpoints at inference time.

We evaluate the following model families: **Gemma 3 1B-IT and 4B-IT** (Gemma Team, 2025), instruction-tuned decoder-only LLMs with LoRA/QLoRA fine-tuning, and **NLLB-600M** (NLLB Team et al., 2022), a multilingual encoder–decoder NMT model, fine-tuned with LoRA. NLLB substitutes for the standard dual-encoder APE pipeline, which is infeasible for English–Sinhala. The available encoders (Dhananjaya et al., 2022) are not strong enough and large parallel corpora are absent (Ranasinghe et al., 2025).

We omit synthetic data, which would mask the heterogeneous nature of real post-edits. All models are trained with three independent random seeds (42, 52, 62); we report mean ± std through-

	D1 Lit.	D2 News	D3 Wiki	D4 Math/H	D5 NLLB
Train	5,728	13,305	17,471	5,754	10,332
Valid	716	1,663	2,184	719	1,291
Test	716	1,664	2,184	720	1,292
Total	7,160	16,632	21,839	7,193	12,915
MT TER	24.98	7.93	14.89	24.30	59.19
MT source		Google Translate			NLLB-1.3B

Table 1: Experimental splits D1–D5 (∼66k triplets). D1–D4 use Google Translate MT; D5 is a contrastive low-quality NLLB-1.3B condition. MT TER is test-set TER of unedited MT vs. PE. Full statistics in Appendix Table 12.

out. Full hyperparameters and the input formats for both families are in Appendix Table 27; evaluation metric details in Appendix F.

4.3 Binary Collapse: Copy or Off-Target Edit

On D1 (Literature), vanilla fine-tuning *degrades* TER across all three model families (Table 5): Gemma 1B by 0.87 points, 4B by 0.43, and NLLB-600M by 5.80. Within the decoder-only family, scale mitigates the degradation; the encoder–decoder NLLB-600M fails most severely.

The model's editing behavior has two modes. It copies the MT verbatim on most sentences (71.5% of the 200 classified D1 samples, every one of which requires an edit; Table 30). On the minority of sentences where it does edit, the edit rarely reaches the human correction (only 4.0% of outputs match PE). We term this **Binary Collapse**: fine-tuning collapses nuanced, targeted editing into copying or off-target edits. The collapse is one of edit *quality*, not magnitude (the edits are moderate, not wholesale rewrites), and it is specific to the inconsistent D1 signal: on D2, D3, and the contrastive D5 condition the model's edits are net-positive, whereas on D1 they are net-negative, leaving the fine-tuned model below the MT baseline (Appendix D.2 Table 34).

Independent confirmation comes from a Chain-of-Thought self-tagging experiment (Appendix G), in which the model is trained to predict its own edit level before post-editing. Its predicted distribution collapses the minor class from 20.9% in the gold labels to 0.1% (Table 40), with 90% of gold-minor samples tagged as `[major]`: the model has internalized a binary none-or-major view of editing. This self-assessment failure is also an early sign of miscalibration, the model staying confident in edit-level predictions that are wrong; we establish that miscalibration directly in Section 5.3. It is consistent with evidence that fine-tuned LLMs lose reliable self-knowledge under distribution shift (Kadavath et al., 2022).

Two other training-time interventions, Hidden State Anchoring (HSA) and Edit-Level Conditioning (ELC), also fail to reduce over-editing (Appendix Table 41). These failures led us to hypothesize **Confident Miscalibration**: the model's own confidence cannot tell good edits from bad.

4.4 Constrained Decoding as a Diagnostic

We test the Confident Miscalibration hypothesis with constrained decoding, an inference-time diagnostic that needs no retraining. It simply reads two signals, one per failure mode: the shape of the TER-vs-λ curve reads whether the model learned useful edits (Binary Collapse), and the ordering of the constraint variants reads whether the model's confidence can find those edits (Confident Miscalibration). The two modes share one suspected cause and one remedy (Section 6.1); we keep them apart because they prescribe different actions (Section 6.3).

The first signal is the curve shape. Raising λ from 0 (unconstrained) toward large values forces the model to copy, meaning that at large λ, TER equals the MT baseline. Read against that baseline, the TER-vs-λ curve reveals whether the model learned any net-positive edits in training: a U-shaped curve dips below it at some intermediate λ, where partial editing beats copying, so it learned some. A monotonically decreasing curve never falls below the baseline, reaching it only as editing is suppressed, so every model edit is, on average, harmful.

All variants penalize divergence from the MT output using *window-based alignment*: a pointer tracks position in the MT sequence, and a window $\mathcal{W}_t$ of $2k{+}1$ MT positions centered on the current pointer defines the set of MT tokens allowed at step t. The window distance is binary: $d(y_t, \mathcal{W}_t) = 0$ if $y_t = x_j^{\text{mt}}$ for some $j \in \mathcal{W}_t$, else 1.

Static (λ_{static}). A uniform penalty $\lambda \geq 0$ is applied at every step, **ignoring model confidence**:

$$\hat{p}(y_t) \propto p_t(y_t) \cdot \exp\big(-\lambda \cdot d(y_t, \mathcal{W}_t)\big) \quad (1)$$

Probability-scaled (PMT; λ_{pmt}). The penalty is scaled by how much probability mass the model puts on MT-window tokens:

$$\lambda_t = \lambda \cdot P_{\text{copy}}^{\beta} \quad (2)$$

where $P_{\text{copy}} = \max_{j \in \mathcal{W}_t} p_t(x_j^{\text{mt}})$ is the maximum probability the model assigns to any MT-window token, and $\beta > 0$ controls sharpness. When the model already prefers an MT-window token (P_{copy} large), the constraint is reinforced; when it does not (P_{copy} small), the constraint relaxes.

Entropy-scaled (λ_{entropy}). The penalty is scaled by how uncertain the model is at step t:

$$\lambda_t = \lambda \cdot \left(\frac{H(p_t)}{H_{\text{max}}} \right)^{\gamma} \tag{3}$$

where $H(p_t) = -\sum_v p_t(v) \log p_t(v)$ is the entropy of the output distribution at step t, $H_{\text{max}} = \log |\mathcal{V}|$ is the maximum possible entropy, and $\gamma > 0$ controls how sharply the constraint switches off as the model becomes confident.

The three variants form a hierarchy that trusts the model's confidence to increasing degrees: static ignores it, probability-scaled relaxes the constraint where the model prefers to diverge from the MT, and entropy-scaled relaxes wherever the model is confident. Their ordering is the second signal. If the model's confidence correlates with which of its edits are net-positive, the confidence-aware variants should win; if it does not, the ordering reverses to Static > Probability > Entropy, the signature of Confident Miscalibration, confirmed in Section 5.2. A fourth variant, sequence-scaled ($\lambda_{\text{seq-NLL}}$), replaces the per-token signal with the running confidence over the whole generated prefix (Appendix I.3); it checks that the ordering is not an artifact of token-level signals. Search ranges and the validation-selected values for λ, γ, β, and the window size k are reported in Table 43; Appendix I.4 shows the diagnostic is stable across k.

5 Results and Analysis

This section follows the diagnostic's two signals. Sections 5.1 and 5.2 read the curve shape and the constraint ordering on D1, Section 5.3 confirms both at the token level, and Sections 5.4 to 5.6 replicate the pattern across architectures, MT quality, and language pairs.

5.1 No Net-Positive Edits on D1

Figure 1 (left) plots TER as a function of λ for the static constraint on D1 (seed 42; full sweep in Appendix Table 16). We observe a **monotonically decreasing** curve from TER 25.84 (λ=0) toward 24.97 ($\lambda \geq 2.5$), with no dip below the MT baseline at any λ. This indicates that every deviation from MT is, on average, harmful: the model has not learned a net-positive editing pattern, and copying the MT outright is the best it can do. The curve shape is not an artifact of TER: recomputing the sweep under chrF++ and BLEU preserves the monotonic-versus-U-shaped distinction on D1, D5, and English–Marathi (Appendix F.1).

Method	Logic	TER ↓	ΔTER
MT Baseline	–	24.98	–
Vanilla FT	–	25.85±0.02	−0.87
λ_{entropy}	Trust if confident	25.72±0.09	−0.74
λ_{pmt}	Trust if hates MT	25.37±0.25	−0.39
$\lambda_{\text{seq-NLL}}$	Trust if seq. confident	25.32±0.21	−0.34
λ_{static}	**Don't trust model**	**24.98±0.08**	**0.00**

Table 2: Constrained decoding on D1 (Gemma 3 1B, mean ± std across 3 seeds). The **Static > Probability > Entropy** ordering is the signature of Confident Miscalibration (zero-shot: Appendix A, Table 9). The sequence-level variant patterns with the token-level ones, so the ordering is not an artifact of token-level signals.

5.2 Confident Miscalibration: Confidence Does Not Identify Net-Positive Edits

Binary Collapse (Section 4.3) tells us *what* the model does wrong. The constraint ordering tells us *why*. Table 2 gives the central result: the static constraint, which ignores the model's confidence, beats every confidence-aware variant, so per-token confidence does not separate net-positive edits from unnecessary ones.

The ordering confirms Confident Miscalibration. The static constraint (λ=2.5) recovers the MT baseline exactly, where the λ curve also flattened (Section 5.1). The probability variant lands in the middle, since P_{copy} carries some information. The entropy variant is worst: its TER stays flat across all base λ (Table 16). The sequence-level variant, which sets the penalty from the model's confidence over the whole generated prefix rather than a single token, lands with the token-level variants (TER 25.32±0.21). No confidence signal we test, token-level or sequence-level, reaches the confidence-free static baseline, so the failure lies in the confidence itself, not in how we integrate it.

5.3 Token-Level Confidence: Direct Evidence for Both Failures

We measure model confidence directly on correct vs. incorrect editing decisions. We classify each of the 20,784 tokens generated by the vanilla 1B

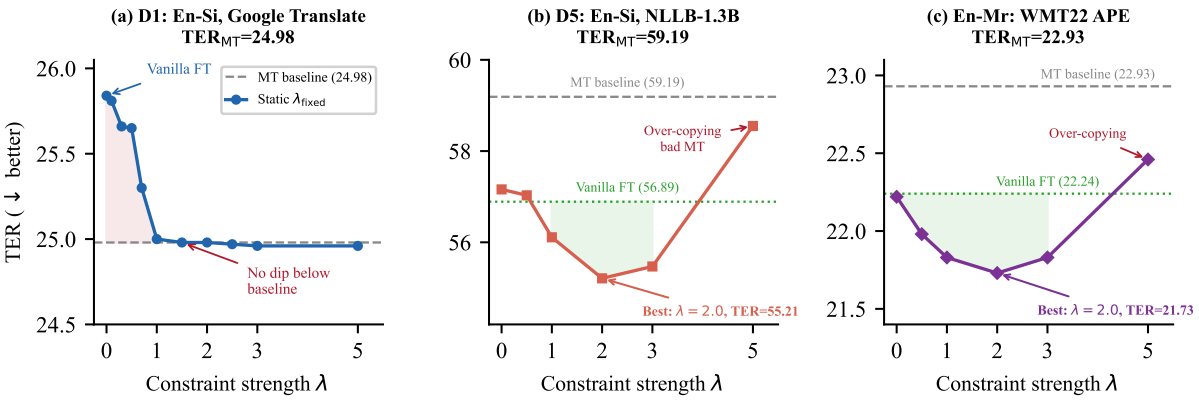

Figure 1: TER vs. λ curves (Gemma 3 1B, seed 42). **Left:** D1 (TER 24.98): monotonic. **Center:** D5 (TER 59.19): U-shaped, min λ=2.0. **Right:** En-Mr (WMT22, TER 22.93): U-shaped, min λ=2.0. En-Mr's MT quality matches D1 yet its curve is U-shaped: the pattern is consistent with a post-edit-distribution effect, not an MT-quality effect.

Token Category	%	Mean p	Mean H
Correct copy	64.5	0.95	0.30
Missed edit	29.0	0.95	0.33
Unnecessary edit	5.2	0.80	0.91
Correct edit	1.2	0.81	0.84

Table 3: Token-level confidence for Gemma 3 1B on D1 (vanilla decoding). p: mean token probability; H: mean output entropy.

Token Category	D1	D2	D5
Correct copy	64.5 (.95)	87.0 (.99)	29.4 (.90)
Missed edit	29.0 (.95)	7.6 (.97)	36.9 (.86)
Unnecessary edit	5.2 (.80)	3.7 (.90)	28.4 (.77)
Correct edit	1.2 (.81)	1.7 (.95)	5.3 (.84)

Table 4: The Table 3 breakdown repeated on D2 (clean MT) and the U-shaped D5 (Gemma 3 1B, seed 42). Each cell gives the token share in % (mean p in parentheses).

model (seed 42, D1 test, window k=2) by whether it matches the MT and the *gold PE* (the human reference post-edit $\mathbf{y}^\star$). This gives four categories: *correct copy* (matches both), *correct edit* (differs from MT, matches gold PE), *unnecessary edit* (differs from MT, but MT was correct), and *missed edit* (copies MT, but gold PE differs). Table 3 presents the results.

At the token level, both failure modes carry a confidence signature. Binary Collapse: the model copies 93.5% of tokens, and *missed edits* (p=0.95, H=0.33) are confidence-indistinguishable from *correct copies* (p=0.95, H=0.30), so it cannot tell which tokens need editing. Confident Miscalibration: when the model does edit, *unnecessary edits* (p=0.80, H=0.91) and *correct edits* (p=0.81, H=0.84) are equally confident, so it cannot tell which edits are useful. The overlap is starkest at the high-confidence end: 59.4% of unnecessary edits and 54.8% of correct edits both fall above p>0.9, as do 84.8% of missed edits (full distribution in Appendix C Figure 2).

The breakdown is not specific to the failing D1 setting. Repeating it on D2 (clean MT) and on the U-shaped D5 (Table 4) shows what changes as the post-edits become more consistent, and what does not. The share of correct edits rises (1.2% to 1.7% to 5.3%): the model does make useful edits when the data supports them, consistent with the D1 data contributing to the failure. The confidence overlap, however, persists everywhere: in every setting 47% to 77% of unnecessary edits still carry probability above 0.9, and unnecessary edits outnumber correct ones by two to five times. Even when APE works, the model's own confidence is not a reliable gate.

The entropy variant defers to these high-confidence predictions, so it fails. Neither structured prompting nor recalibration resolves the failure. GPT-5.2 with explicit Stage-1 gating reaches only TER 36.75 (gating precision 0.75, 2 points above an always-edit baseline; Appendix A.3). Temperature scaling at T^*=2.0 worsens unconstrained generation while preserving the ordering (Appendix C.3).

5.4 The Same Failure Across Architectures, Scale, and Domains

Table 5 compares all three model families with constrained decoding on D1.

Scale mitigates but does not fix. Gemma 4B halves the degradation (−0.43 vs. −0.87 for 1B), and is the only model to *beat* the MT baseline with static constraint (ΔTER= + 0.15). Its λ curve is weakly U-shaped (Figure 4 in Ap-

Model	Method	TER ↓	BLEU ↑	ΔTER
Gemma 3 1B (decoder-only, LoRA):				
	Base (zero-shot)	190.31	8.88	−165.33
	Vanilla FT	25.85	67.16	−0.87
	λ_{entropy}	25.72	67.23	−0.74
	λ_{pmt}	25.37	67.45	−0.39
	λ_{static}	**24.98**	**67.38**	**0.00**
Gemma 3 4B (decoder-only, QLoRA):				
	Base (zero-shot)	93.81	24.32	−68.83
	Vanilla FT	25.41	67.37	−0.43
	λ_{entropy}	25.05	67.53	−0.07
	λ_{pmt}	24.90	67.76	+0.08
	λ_{static}	**24.83**	**67.52**	**+0.15**
NLLB-600M (encoder–decoder, LoRA):				
	Vanilla FT	30.78	61.98	−5.80
	λ_{entropy}	31.28	61.57	−6.30
	λ_{pmt}	29.97	62.58	−4.98
	λ_{static}	**29.59**	**62.63**	**−4.61**

Table 5: Cross-model constrained decoding on D1. **All three architectures show the same ordering: Static > Probability > Entropy.**

	TER ↓	BLEU ↑	ΔTER
D1: High-quality MT (Google Translate):			
MT Baseline	24.98	67.38	–
Vanilla FT	25.85	67.16	−0.87
λ_{static}	24.98	67.38	0.00
D5: Low-quality MT (NLLB-1.3B), Gemma 1B:			
MT Baseline	59.19	30.54	–
Vanilla FT	56.89	31.60	+2.30
λ_{static}	**55.21**	**33.12**	**+3.98**
λ_{pmt}	56.58	32.20	+2.61
λ_{entropy}	56.99	31.88	+2.20
D5: Low-quality MT (NLLB-1.3B), NLLB-600M:			
Vanilla FT	57.14	32.01	+2.05

Table 6: D1 vs. D5 (seed 42). On D5 (genuine error correction) both Gemma 1B and NLLB-600M improve over MT; constrained decoding further improves Gemma 1B, and Static > PMT > Entropy still holds.

pendix C.7), unlike 1B's monotonic curve. Yet the Static > PMT > Entropy ordering still holds, with entropy TER essentially flat (25.05–25.08). Confidence remains uninformative even with 4× more parameters: scale mitigates Binary Collapse but not Confident Miscalibration.

Encoder-decoder failure. NLLB-600M degrades by −5.80 TER, far worse than Gemma 1B (−0.87) and 4B (−0.43). Even aggressive static constraint only reaches TER 29.59 (λ=5.0), still 4.61 above MT. A plateau at $\lambda \geq 10$ confirms constraint saturation (Figure 4): the over-editing is too severe for inference-time correction. However, this failure is task-specific: the same model succeeds on D5 (ΔTER= + 2.05; Section 5.5), showing that the architecture can succeed under a contrasting condition.

Domain robustness. Vanilla fine-tuning degrades TER on D1 and D4 but not D2 (News) or D3 (Wiki; Appendix Table 26). Qualitative examples of recurring failure modes (unnecessary edits on correct MT, content corruption, hallucinated corrections) are in Appendix D.3.

5.5 Contrastive Validation

A valid diagnostic must also detect success. D5 is such an example. Its MT is far worse than D1's (TER 59.19 vs. 24.98), so post-editors mostly fix real errors instead of making optional style changes. That gives D5 a consistent, learnable signal, unlike D1. The same models that fail on D1 succeed on D5 (Table 6): vanilla fine-tuning *improves* TER by +2.30 for Gemma 1B and +2.05 for NLLB-600M, reversing their −0.87 and −5.80 on D1. So the D1 failure comes from the post-edit distribution, not the models or fine-tuning.

The Gemma 1B λ curve for D5 is **U-shaped** (Figure 1, center): TER reaches a minimum of 55.21 at λ=2.0, well below both vanilla FT and the MT baseline, then rises as the constraint forces copying of low-quality MT. NLLB-600M has its minimum at $\lambda \approx 0$ (Appendix Figure 4): on D5, its TER is minimized without a constraint.

5.6 Cross-Lingual Replication

So far, the D1 failure and the D5 success both come from English–Sinhala. We replicate the diagnostic on two more language pairs: English–Marathi (Indo-Aryan, like Sinhala) and English–Tamil (Dravidian). In both, the two diagnostic signals, curve shape and constraint ordering, behave just as they do on English–Sinhala.

On WMT22 **English–Marathi** APE (17,999 triplets; Bhattacharyya et al., 2022), MT quality matches D1 (TER 22.93 vs. 24.98). Yet vanilla fine-tuning *improves* TER by +0.69 (Table 7). The λ curve is U-shaped (Figure 1, right), and Static > PMT > Entropy holds.

English–Tamil contributes a new News corpus (10,000 triplets, Google Translate MT, TER_{MT}=20.57). Vanilla fine-tuning degrades TER by −1.21 (Table 8), and the λ curve is monotonic (Figure 3a), mirroring D1. Tamil News shares D1's heterogeneous edit profile (44/20/36 none/minor/major; Appendix Table 13). A within-language contrastive condition using NLLB-1.3B MT (TER_{MT}=47.54) on the same sources produces a U-shaped curve (Figure 3b,

System	TER ↓	BLEU ↑	ΔTER
MT Baseline	22.93	67.01	–
Vanilla FT	22.24	69.81	+0.69
λ_{entropy}	22.06	69.95	+0.87
λ_{pmt}	21.83	69.96	+1.10
λ_{static}	**21.73**	**69.72**	**+1.20**

Table 7: English–Marathi validation (Gemma 3 1B, seed 42, WMT22 APE dev set; test refs unavailable). **Static** > **PMT** > **Entropy** holds cross-lingually. Full sweeps in Appendix Table 19.

System	TER ↓	BLEU ↑	ΔTER
MT Baseline	20.57	73.17	–
Vanilla FT	21.77	71.39	−1.21
λ_{static}	20.68	73.00	−0.11

Table 8: English–Tamil validation (News, Gemma 3 1B, seed 42, static constraint λ=5.0). The monotonic λ curve approaches but never crosses the MT baseline. Full sweep in Appendix Table 20; per-domain in Table 22.

minimum at λ=3.0), mirroring the D1/D5 pattern.

6 Discussion

6.1 What Drives Both Failures: Inconsistency, Not Edit Type

Section 5.4 showed the failure is domain-specific, which points at the data, not the model. Our hypothesis, which the evidence below supports but does not prove, is that the heterogeneous post-edits (Section 3.2) are the source, and that the problem is not their variety but their *inconsistency*: the same kind of error gets fixed in conflicting ways, so no stable rule connects a correction to its context. Under this hypothesis, the model cannot learn a coherent editing strategy, so it falls back to copying or off-target rewriting (Binary Collapse); and because no consistent signal links a correction to its quality, the model's confidence cannot tell a useful edit from a harmful one (Confident Miscalibration). The two modes are thus symptoms of one suspected cause, sharing one remedy; we keep them apart only because they are read from different signals and prescribe different actions (Section 6.3). External QE signals fail the same way (Appendix J). The evidence here is correlational, but three independent lines converge on it.

D3 is a useful contrast. If edit type drove the failure, two domains with the same edit-type mix would behave alike. D3 has almost the same mix as D1: identical 41.0% Style, and 48.0% vs 49.5% subjective overall (Table 33). Yet on the full test set, D3's edits are net-positive while D1's are net-negative (per-edit ΔTER −1.1 vs +6.3; Appendix Table 34). With the mix held fixed, the results are consistent with a more learnable correction signal. Annotation policy, MT quality, and language family are also less plausible explanations (Sections 5.5, 5.6): D2's mechanical edits are learnable under the same policy, English–Marathi matches D1's MT quality yet edits net-positively, and English–Tamil reproduces the failure though it is Dravidian, not Indo-Aryan.

Gradient analysis. Inspecting the model's gradients supports the same account (Appendix K): two independent instruments, one white-box and one black-box, are consistent with inconsistent post-edits as the driver. First, the gradient has no stable target: on D1, Style is the largest edit group yet produces the *smallest* mean gradient (norm 1.21), because its subtypes pull in opposite directions and cancel, and its per-sample loss is the most variable of any group (coefficient of variation 1.11). Second, the model cannot tell the edit types apart: on D1, their gradients overlap (max pairwise cosine 0.275), while on D5, where the corrections are genuine, they are near-orthogonal (0.051). Third, the same signal sharpens when its surroundings are consistent: Fluency's mean gradient norm rises from 1.46 on D1 to 3.60 on D5, as morphological and punctuation corrections diluted by heterogeneous neighbors on D1 emerge as the clearest signal on D5. The geometry tracks the λ curve: both collapse on D1 and separate on D5.

If the hypothesis is right and inconsistency rather than any single edit type drives the failure, removing one edit type should not repair it. Retraining Gemma 3 1B on D1 with all Style and Retranslation samples removed confirms this. The TER-vs-λ curve stays monotonically decreasing, never crossing the MT baseline, and shows no advantage over a size-matched random subsample (Appendix D.2 Table 48). The Accuracy and Fluency edits that remain are themselves inconsistent, matching the cross-domain pattern (Table 33) and the gradient analysis above.

6.2 Implications for Quality Estimation and Confidence

External QE signals do not reliably identify segments needing edits: fine-tuned XLM-R (Conneau et al., 2020) reaches only AUROC 0.745 on D1, no better when trained on test data (Ap-

pendix J, Table 45). Pretrained QE also fails: LaBSE (Feng et al., 2022) similarity is below chance and COMET-QE (Rei et al., 2022) near it (AUROC 0.45 and 0.58, Table 45). QE-guided APE pipelines should therefore check post-edit consistency before investing in QE infrastructure (Deoghare et al., 2025).

The model's own confidence is no more reliable, at least across the signals we test (token-level probability and entropy, and a sequence-level one): it stays certain even when wrong. A GPT-5.2 probe across zero-shot, few-shot, RAG, and two-stage conditions fails the same way (Appendix A), pointing to a task-level rather than capacity-level failure. Our validated claims are confined to APE, across three architectures and three language pairs. Because the diagnostic is black-box and not tied to APE, it may extend to other editing tasks with similarly inconsistent supervision, but that extension is untested and remains future work.

6.3 From Diagnosis to Action

A single λ sweep does more than diagnose: in the favorable case, it also improves accuracy. It is also cheap: each λ value costs one decoding pass over the evaluation set, and the constraint logic itself adds negligible overhead (<5% wall-clock; Appendix I). The shape of the curve and the location of its minimum sort each fine-tuned checkpoint we evaluate into one of four outcomes (Appendix I.7), each with a concrete next step:

1. **U-shaped, dipping below the MT baseline.** Net-positive edits survived training. Deploy the static constraint at the optimal λ for an inference-time accuracy gain, with no retraining or annotation (as shown robustly on D5 and English–Marathi, Tables 6 and 7; marginally on Gemma 4B at D1).

2. **Minimum at λ=0.** The model already edits net-positively without any constraint. Deploy the vanilla checkpoint unchanged (as for D2).

3. **Monotonic, reaching the MT baseline.** No net-positive edits survived, and copying is optimal. Copy the MT for now, and make the post-edits more consistent before training again (as for D1).

4. **Monotonic, plateauing above the MT baseline.** Over-editing is too severe to undo at inference. Fix the data or MT quality before training again (NLLB-600M on D1).

Whenever a constraint helps, deploy the *static* variant: the Static > PMT > Entropy ordering (Section 5.2) shows that trusting the model's confidence only costs accuracy (Appendix I.6).

Outcomes 3 and 4 cannot be repaired at inference; the fix has to come from the data. The suspected cause is inconsistent post-edits (Section 6.1), so the corrections must be made more consistent before retraining. One concrete route is to separate mandatory error correction from discretionary stylistic refinement at annotation time. Filtering after the fact does not work in one pass: dropping stylistic edits wholesale leaves the rest just as inconsistent (Appendix L). A promising future direction is therefore an iterative loop that filters the post-edits, retrains, and re-evaluates, repeating until the model becomes trainable or a retry budget runs out; our diagnostic supplies the stopping rule such a loop needs, since each round can simply check whether the λ curve has turned U-shaped. Deeper fixes such as MT customization, task decomposition, or edit-type-conditioned generation are future work. Our first attempt at the last, edit-level tagging at training time, fails (Appendix G).

7 Conclusion

When APE fails to improve low-resource MT, the score alone cannot inform whether the model needs more training or the training data is too inconsistent to learn from. Our constrained-decoding diagnostic answers this without retraining or annotation: we sweep a single knob, an edit-distance penalty λ that drives the model from free editing (at λ=0) toward simply copying the MT output (at large λ), and read off whether the model learned net-positive edits and whether its confidence tracks edit quality. On English–Sinhala it exposes **Binary Collapse** and **Confident Miscalibration**, and both replicate on English–Marathi and English–Tamil. The evidence converges on our hypothesis that the heterogeneous post-edit distribution shapes both failure modes; model capacity, MT quality, and language family explain them less. Where genuine corrections survive training, a static constraint turns the same sweep into a free inference-time gain. Where they do not, the diagnostic reframes the failure as a data problem: test whether the post-edits form a learnable signal before investing in more data or compute. We release the first English–Sinhala and a new English–Tamil 🤗 `APE datasets` with all `code`.

Limitations

We identify six limitations. *First, language coverage*: our findings are established on English–Sinhala and validated cross-lingually on English–Marathi (Indo-Aryan) and English–Tamil (Dravidian); additional language families (e.g., Turkish, Swahili) and the canonical high-resource APE setting (English–German) remain future work. *Second, TER and the paraphrase concern*: the binary edit-distance penalty (d=0 if the token appears in the MT window, 1 otherwise) penalizes a valid paraphrase as heavily as an error, and TER itself does not credit semantically equivalent rewrites. The token-level confidence finding (Table 3) is independent of TER, however: it directly measures the probability and entropy distributions of the model's edit decisions and shows that the model is no less confident on unnecessary edits than on correct ones ($p\approx0.80$ vs. $p\approx0.81$). So the central calibration result is robust to the paraphrase concern, even though TER itself is imperfect. *Third, semantic and human evaluation*: we report TER, BLEU, and chrF++ throughout, and the diagnostic's curve-shape distinction is verified under all three (Appendix F.1), but human adequacy/fluency judgements and semantic-similarity QE on the predicted post-edits remain future work. *Fourth, calibration measures*: our analysis covers token-level probability and entropy plus a sequence-level signal ($\lambda_{\text{seq-NLL}}$, Table 2), which agrees with the token-level ones; ensemble-based agreement measures remain untested and may provide better-calibrated trust signals. *Fifth, dataset scale*: our datasets ($\sim$66k English–Sinhala, $\sim$39k English–Tamil) are the largest for these language pairs but remain modest by WMT standards. *Sixth, dataset-artifact framing*: our findings characterize APE under the *inconsistent* post-edits of low-budget translation pipelines (Section 6.1); a reader might mistake this inconsistency for a removable data artifact rather than an intrinsic property of the APE task. But the conflicting corrections are often individually defensible (different post-editors make different yet legitimate discretionary choices), which is why naive filtering does not remove the inconsistency (Appendix L). Reducing it would require annotation protocols that separate mandatory error correction from discretionary refinement (Section 6.3), at some cost to ecological validity. The diagnostic remains useful under either framing, as a test of whether a given dataset supports learnable APE.

Ethics Statement

The English–Sinhala and English–Tamil data underlying our APE experiments were produced as part of the EnSiTa dataset-creation effort (Ranathunga et al., 2026), which documents the translator recruitment, compensation, and quality-control protocol and confirms that the data contains no personally identifiable information; post-editors there were compensated at rates that meet or exceed standard local professional-translation wages. Our work adds only automated cleaning to derive APE triplets (Appendix B.3) and involves no new human data collection. We note that APE systems, including ours, may propagate or amplify biases present in the underlying MT system; users should exercise caution when deploying APE for sensitive content. The human annotators who contributed to this study were compensated for their work.

Licenses and Use of Artifacts

We use Gemma 3 1B and 4B (Gemma Team, 2025) under the Gemma Terms of Use, NLLB-600M and NLLB-1.3B (NLLB Team et al., 2022) under CC-BY-NC 4.0, and the WMT22 English–Marathi APE data (Bhattacharyya et al., 2022) under its shared-task license; all such use is for non-commercial research on automatic post-editing, consistent with each artifact's stated intended use. We release our English–Sinhala ($\sim$66k) and English–Tamil ($\sim$39k) APE datasets under CC-BY-SA 4.0 for research and academic use in machine translation, post-editing, and related NLP tasks; this is compatible with the conditions under which the underlying publicly available source texts and MT outputs were accessed. The D5 and English–Tamil NLLB-MT contrastive subsets contain NLLB-1.3B outputs, and downstream commercial redistribution of those subsets should additionally observe NLLB's CC-BY-NC 4.0 terms. Our code is available on GitHub, and the released datasets are available on Hugging Face (gated only to deter automated scraping, with access granted promptly to any researcher on request), under the terms above.

Acknowledgments

This research was supported by a Google Diversity and Inclusion grant received by Surangika Ranathunga and Nisansa de Silva.

Use of AI Assistants

We used a frontier general-purpose LLM assistant for iterative feedback on the paper (argument structure, clarity, and framing), for LaTeX editing, and for developing experiment scripts; a second frontier LLM was used for code generation. GPT-5.2 (OpenAI, 2025) was used for the automated edit-type classification described in Section 4.3 and Appendix D.2, and for the in-context-learning probes in Appendix A. No AI system was used to generate experimental results, statistical analyses, or scientific claims, except where an AI system was itself the object of study. All AI-assisted text and code were reviewed, validated, and where necessary rewritten by the authors, who take full responsibility for the content.

References

Pushpak Bhattacharyya, Rajen Chatterjee, Markus Freitag, Diptesh Kanojia, Matteo Negri, and Marco Turchi. 2022. Findings of the WMT 2022 shared task on automatic post-editing. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 109–117, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Pushpak Bhattacharyya, Rajen Chatterjee, Markus Freitag, Diptesh Kanojia, Matteo Negri, and Marco Turchi. 2023. Findings of the WMT 2023 shared task on automatic post-editing. In *Proceedings of the Eighth Conference on Machine Translation*, pages 672–681, Singapore. Association for Computational Linguistics.

Rajen Chatterjee, Markus Freitag, Matteo Negri, and Marco Turchi. 2020. Findings of the WMT 2020 shared task on automatic post-editing. In *Proceedings of the Fifth Conference on Machine Translation*, pages 646–659, Online. Association for Computational Linguistics.

Rajen Chatterjee, Matteo Negri, Raphael Rubino, and Marco Turchi. 2018. Findings of the WMT 2018 shared task on automatic post-editing. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 710–725, Belgium, Brussels. Association for Computational Linguistics.

Shamil Chollampatt, Raymond Hendy Susanto, Liling Tan, and Ewa Szymanska. 2020. Can automatic post-editing improve NMT? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2736–2746, Online. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Nisansa de Silva. 2026. Survey on Publicly Available Sinhala Natural Language Processing Tools and Research. *arXiv preprint arXiv:1906.02358v27*.

Sourabh Deoghare and Pushpak Bhattacharyya. 2022. IIT Bombay's WMT22 automatic post-editing shared task submission. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 682–688, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Sourabh Deoghare, Diptesh Kanojia, and Pushpak Bhattacharyya. 2024. Together we can: Multilingual automatic post-editing for low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10800–10812, Miami, Florida, USA. Association for Computational Linguistics.

Sourabh Deoghare, Diptesh Kanojia, and Pushpak Bhattacharyya. 2025. Giving the old a fresh spin: Quality estimation-assisted constrained decoding for automatic post-editing. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 914–925, Albuquerque, New Mexico. Association for Computational Linguistics.

Sourabh Deoghare, Diptesh Kanojia, Fred Blain, Tharindu Ranasinghe, and Pushpak Bhattacharyya. 2023. Quality estimation-assisted automatic post-editing. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1686–1698, Singapore. Association for Computational Linguistics.

Vinura Dhananjaya, Piyumal Demotte, Surangika Ranathunga, and Sanath Jayasena. 2022. BERTifying Sinhala - a comprehensive analysis of pre-trained language models for Sinhala text classification. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 7377–7385, Marseille, France. European Language Resources Association.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. 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.

Gemma Team. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1321–1330.

Chris Hokamp and Qun Liu. 2017. Lexically constrained decoding for sequence generation using grid beam search. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1535–1546, Vancouver, Canada. Association for Computational Linguistics.

J. Edward Hu, Huda Khayrallah, Ryan Culkin, Patrick Xia, Tongfei Chen, Matt Post, and Benjamin Van Durme. 2019. Improved lexically constrained decoding for translation and monolingual rewriting. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 839–850, Minneapolis, Minnesota. Association for Computational Linguistics.

Xuancheng Huang, Yang Liu, Huanbo Luan, Jingfang Xu, and Maosong Sun. 2019. Learning to copy for automatic post-editing. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6122–6132, Hong Kong, China. Association for Computational Linguistics.

Marcin Junczys-Dowmunt and Roman Grundkiewicz. 2018. MS-UEdin submission to the WMT2018 APE shared task: Dual-source transformer for automatic post-editing. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 822–826, Belgium, Brussels. Association for Computational Linguistics.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, and 17 others. 2022. Language models (mostly) know what they know. *Preprint*, arXiv:2207.05221.

Dayeon Ki and Marine Carpuat. 2024. Guiding large language models to post-edit machine translation with error annotations. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 4253–4273, Mexico City, Mexico. Association for Computational Linguistics.

Arle Lommel, Hans Uszkoreit, and Aljoscha Burchardt. 2014. Multidimensional quality metrics (MQM): A framework for declaring and describing translation quality metrics. *Tradumàtica tecnologies de la traducció*, pages 455–463.

António V. Lopes, M. Amin Farajian, Gonçalo M. Correia, Jonay Trénous, and André F. T. Martins. 2019. Unbabel's submission to the WMT2019 APE shared task: BERT-based encoder-decoder for automatic post-editing. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 118–123, Florence, Italy. Association for Computational Linguistics.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, and 20 others. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

OpenAI. 2025. Introducing GPT-5.2. `https://openai.com/index/introducing-gpt-5-2/`. Accessed: 2026-08-29.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Matt Post and David Vilar. 2018. Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1314–1324, New Orleans, Louisiana. Association for Computational Linguistics.

Tharindu Ranasinghe, Hansi Hettiarachchi, Nadeesha Chathurangi Naradde Vidana Pathirana, Damith Premasiri, Lasitha Uyangodage, Isuri Nanomi Arachchige, Alistair Plum, Paul Rayson, and Ruslan Mitkov. 2025. Sinhala encoder-only language models and evaluation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8623–8636, Vienna, Austria. Association for Computational Linguistics.

Surangika Ranathunga and Nisansa de Silva. 2022. Some languages are more equal than others: Probing deeper into the linguistic disparity in the NLP world. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 823–848, Online only. Association for Computational Linguistics.

Surangika Ranathunga, Nisansa de Silva, Aloka Fernando, Kavindu Warnakulasuriya, Isuru Wijesiri, Menan Velayuthan, Charitha Rathnayaka, Thivaharan Varatharajan, Sajeevi Silva, Piumi Kandanaarachchi, and Uthayasanker Thayasivam. 2026. EnSiTa: A trilingual multi-domain parallel dataset and benchmark for domain-specific machine translation. Preprint.

Vikas Raunak, Amr Sharaf, Yiren Wang, Hany Awadalla, and Arul Menezes. 2023. Leveraging GPT-4 for automatic translation post-editing. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12009–12024, Singapore. Association for Computational Linguistics.

Ricardo Rei, Marcos Treviso, Nuno M. Guerreiro, Chrysoula Zerva, Ana C Farinha, Christine Maroti, José G. C. de Souza, Taisiya Glushkova, Duarte Alves, Luisa Coheur, Alon Lavie, and André F. T. Martins. 2022. CometKiwi: IST-unbabel 2022 submission for the quality estimation shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 634–645, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. 2024. AI models collapse when trained on recursively generated data. *Nature*, 631(8022):755–759.

Amirhossein Tebbifakhr, Matteo Negri, and Marco Turchi. 2019. Effort-aware neural automatic post-editing. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 139–144, Florence, Italy. Association for Computational Linguistics.

Chrysoula Zerva, Frederic Blain, José G. C. De Souza, Diptesh Kanojia, Sourabh Deoghare, Nuno M. Guerreiro, Giuseppe Attanasio, Ricardo Rei, Constantin Orasan, Matteo Negri, Marco Turchi, Rajen Chatterjee, Pushpak Bhattacharyya, Markus Freitag, and André Martins. 2024. Findings of the quality estimation shared task at WMT 2024: Are LLMs closing the gap in QE? In *Proceedings of the Ninth Conference on Machine Translation*, pages 82–109, Miami, Florida, USA. Association for Computational Linguistics.

A In-Context Learning Results

A.1 In-Context Learning Probe

To test whether model capacity resolves the failure, we evaluate GPT-5.2 (OpenAI, 2025) on the same 200 stratified D1 samples used for edit-type classification (Section 3.2) as an upper-bound probe under three prompting strategies: zero-shot, fixed few-shot (5 examples explicitly showing "copy" behavior), and retrieval-augmented (RAG, k=5 via TF-IDF). All strategies degrade substantially over the MT baseline (TER 34.93–37.92 vs. MT 28.82; Table 9). RAG performs *worse* than both fixed few-shot and zero-shot, possibly because its retrieved examples reflect heterogeneous editing patterns. This probe points to task-level rather than capacity-level failure: even a frontier LLM with carefully chosen examples cannot distinguish necessary from unnecessary edits under this training signal.

A.2 Detailed Results

All experiments use temperature 0.0 and `max_completion_tokens`=512.

Strategy	TER↓	BLEU↑	chrF++↑
MT Baseline	28.82	63.24	77.96
Zero-shot	37.51	54.90	72.84
Fixed few-shot	34.93	57.61	74.44
RAG (k=5)	37.92	54.47	72.06
Two-stage (gate+edit)	36.75	55.28	73.47

Table 9: GPT-5.2 APE results on 200 stratified D1 test samples (same samples as edit-type analysis, Section 3.2). All prompting strategies degrade substantially over the MT baseline. RAG performs *worse* than both fixed few-shot and zero-shot; the two-stage decomposition (Section A.3) is intermediate and shows the same failure mode for structured prompting.

Fixed few-shot performs best (TER 34.93) by explicitly showing "copy" behavior, but still degrades by 6.12 TER points over the MT baseline. Zero-shot degrades further (TER 37.51).

The most revealing result is that RAG performs *worst* of all three strategies (TER 37.92), even below zero-shot. Retrieved examples faithfully reflect the training distribution, where 69.3% of post-edits involve non-trivial changes (Section 3.2), teaching the model the post-editor's heterogeneous editing patterns and amplifying over-editing. More context about the task appears to worsen performance, consistent with the training signal itself being inconsistent.

Base model zero-shot. To confirm that fine-tuning is necessary (rather than prompt engineering being sufficient with a local model), we evaluate the base Gemma 3 1B-it and 4B-it with the same zero-shot APE prompt used for GPT-5.2 (Section A.1) on the full D1 test set (n=716). The 1B base model achieves TER 190.31 and the 4B achieves TER 93.81, compared to the MT baseline of 24.98 (Table 5): both generate fluent but largely unrelated Sinhala text, unable to perform APE without task-specific training. Fine-tuning re-

duces these to TER 25.85 (1B) and 25.41 (4B), confirming that the models *do* learn the task; the problem is not capacity but *what* they learn (Binary Collapse).

A.3 Two-Stage Decomposition (Gate + Edit)

A structured-prompting strategy more sophisticated than zero-shot, few-shot, or RAG is to decompose APE into two independent decisions: *(Stage 1)* given (source, MT), decide whether the MT needs editing at all; *(Stage 2)* if Stage 1 says yes, generate a post-edit. We run both stages with GPT-5.2 at temperature 0 on the same 200 stratified D1 samples. Stage 1 is asked to output a single token `EDIT` or `KEEP`; Stage 2 uses the same prompt as the zero-shot APE setting (Section A.1). When Stage 1 outputs `KEEP`, we emit the MT unchanged.

Metric	Value	Baseline
TER ↓	36.75	MT 28.82
Gate precision	0.75	"always edit" 0.73
Gate recall	0.99	–
Gate F1	0.85	–
Edits called	192/200 (96%)	–

Table 10: Two-stage decomposition with GPT-5.2 on 200 stratified D1 test samples. Confusion matrix: TP=144, FP=48, TN=6, FN=2. Gold edit rate (positive class proportion) is 0.73; the "always edit" baseline therefore achieves gate precision 0.73 by definition. GPT-5.2's gate barely improves on this (0.75), and triggers an edit on 96% of inputs, so the pipeline collapses to vanilla zero-shot APE for almost all samples.

The diagnostically important number is the gate precision (Table 10): 0.75, only 2 points above the trivial "always edit" baseline (gold edit rate 0.73). GPT-5.2 says `EDIT` on 192/200 samples (recall 0.99, precision 0.75, F1 0.85), so the pipeline collapses to vanilla zero-shot APE on 96% of inputs. The resulting TER (36.75) is intermediate between zero-shot (37.51) and fixed few-shot (34.93), and still 7.93 TER points above the MT baseline. This gate failure is the same one XLM-R showed in Appendix J (AUROC 0.745, gating precision below 55% at the 90th-percentile threshold): a second, independent indication that, for the tested models and prompting strategies, the (source, MT) feature space does not provide a reliable edit-need signal.

Stage 1 system prompt.

```
You are a gating system for
English-to-Sinhala automatic
post-editing. You will receive an
English source sentence and its
Sinhala machine translation (MT).
Decide whether the MT requires any
post-editing.

Output exactly one token:

  • KEEP if the MT is acceptable as-is
    (no genuine errors).
  • EDIT if the MT contains a
    genuine error (wrong meaning,
    mistranslation, missing content,
    ungrammatical form).

Do not output anything else.
```

A.4 Few-Shot Examples and RAG Retrieval

The fixed few-shot mode prepends 5 static examples (Table 11) as alternating user/assistant turns: 2 "copy" cases (TER=0), 2 minor edits, and 1 major edit. For RAG, k=5 training examples are retrieved per test sample using TF-IDF cosine similarity over English source sentences, formatted identically but with dynamically selected examples (TF-IDF fit on full D1 training set, n=5,728).

A.5 System Prompts

Zero-shot and fixed few-shot. Both modes use the same system prompt:

```
You are an Automatic Post-Editing
system for English-to-Sinhala machine
translation. You will receive an
English source sentence and its
Sinhala machine translation (MT).
Your task:

1. If the MT is correct, output it
EXACTLY as-is. Do not change anything.

2. If the MT has a genuine translation
error (wrong meaning, mistranslation,
missing content), fix ONLY that error.

3. Do NOT change style, register, word
choice, loanwords, or word order when
the meaning is already correct. These
are stylistic preferences, not errors.

4. Make the MINIMUM number of changes
needed. Fewer changes is better.

Output ONLY the post-edited Sinhala
text, nothing else.
```

RAG mode. The RAG system prompt instructs the model to follow the post-editor's style from retrieved examples:

```
You are an Automatic Post-Editing
system for English-to-Sinhala machine
translation. You will receive
an English source sentence and
its Sinhala machine translation
(MT), along with examples of how a
professional post-editor corrected
similar sentences.
```

#	User (Source + MT)	Assistant (Post-Edit)	Type
1	Source: When he saw who his visitor was his eyes brightened. MT: තම අමුත්තා කවුදැයි දුටු විට ඔහුගේ දෑස් දීප්තිමත් විය.	තම අමුත්තා කවුදැයි දුටු විට ඔහුගේ දෑස් දීප්තිමත් විය.	Copy
2	Source: They went to the football field and started their game. MT: ඔවුන් පාපන්දු පිටියට ගොස් ඔවුන්ගේ ක්‍රීඩාව ආරම්භ කළහ.	ඔවුන් පාපන්දු පිටියට ගොස් ඔවුන්ගේ ක්‍රීඩාව ආරම්භ කළහ.	Copy
3	Source: Monte Cristo bowed, and went to Madame de Villefort. MT: මොන්ටේ ක්‍රිස්ටෝ හිස නමා මැඩම් ඩි විල්ෆෝර්ට් වෙත ගියේය.	මොන්ටේ ක්‍රිස්ටෝ හිස නමා, මැඩම් ඩි විල්ෆෝර්ට් වෙත ගියේය.	Minor
4	Source: "I do not know her name; but it is she, sir, it is she!" MT: "මම ඇගේ නම දන්නේ නැහැ; නමුත් ඒ ඇයයි, සර්, මේ ඇයයි!"	"මම ඇගේ නම දන්නේ නැහැ; නමුත් ඒ ඇයයි, මහත්මයා, ඒ මේ ඇයයි!"	Minor
5	Source: He melted in the hot sun. MT: ඔහු උණුසුම් හිරු තුළ දිය වී ගියේය.	ඔහු උණුසුම් හිරු එළියේ දිය වී ගියේය.	Major

Table 11: Fixed few-shot examples for GPT-5.2. Examples 1–2 show "copy" behavior (TER=0), 3–4 show minor edits, and 5 shows a major edit. All from D1 training set.

```
Follow the post-editor's style shown
in the examples:

- If the examples show the post-editor
kept the MT unchanged, you should
also keep the MT unchanged for similar
cases.

- If the examples show small targeted
fixes, make only similar small fixes.

- Match the level of editing shown in
the examples.

Output ONLY the post-edited Sinhala
text, nothing else.
```

A.6 User Message Format

For all modes, each test sample is formatted as:

```
Source: {source}
MT: {mt}
```

B Dataset Creation Details

All English–Sinhala and English–Tamil data come from the EnSiTa corpus (Ranathunga et al., 2026).

B.1 D1–D4 (Google Translate)

The EnSiTa dataset-creation effort produced the D1–D4 source sentences (D1: literary texts, D2: news, D3: Wikipedia, D4: mathematics/health textbooks), Google Translate MT outputs, and professional Sinhala post-edits (Ranathunga et al., 2026). It documents source selection, translator recruitment, compensation, and quality control. We obtain the resulting triplets and construct APE splits through the cleaning pipeline below.

B.2 D5 (From NLLB Corpus, NLLB MT)

D5 is a controlled contrastive condition we construct for this study. English sentences from the NLLB parallel corpus (NLLB Team et al., 2022) were cleaned and translated into Sinhala with NLLB-1.3B (distilled) to obtain a deliberately lower-quality MT, then post-edited under the same protocol as D1–D4.

All English–Sinhala datasets follow an 80/10/10 train/validation/test split.

B.3 Preprocessing: From Released Dataset to Experimental Splits

The EnSiTa creation process yields, for each sentence, the source, an initial MT, and the professional post-edit; we extract these as (source, MT, PE) triplets for D1–D4 and the English–Tamil domains. We discard any item whose source text was itself revised during EnSiTa's editing: some items passed through several rounds of back-and-forth revision that also changed the source, violating the fixed-source assumption of APE (the source must be identical for the MT and the post-edit). The retained triplets, together with the D5 triplets we construct separately (Table 12), are cleaned into the experimental splits D1–D5 (Table 1) by the following pipeline, applied identically to all domains:

1. **Remove empty/null rows**: drop any triplet where SRC, MT, or PE is missing or empty.

2. **Deduplication**: remove exact (SRC, MT, PE) duplicates, keeping the first occurrence.

Domain	Triplets	Mean TER	Edit Dist.
Literature	7,209	25.7	30/21/49
News	16,632	9.9	51/33/16
Wikipedia	21,839	14.9	25/45/30
Maths	6,487	23.3	29/26/45
Health	1,000	42.1	13/18/69
NLLB Corpus	12,915	61.0	2/5/93
Total	66,082	–	–

Table 12: English–Sinhala APE dataset release statistics. Mean TER is the corpus-level mean TER(MT, PE) over all triplets before cleaning. Edit Dist. shows none/minor/major percentages (TER thresholds: 0, 20). Literature, News, Wikipedia, Maths, and Health use Google Translate MT; NLLB Corpus uses NLLB-1.3B MT. Experimental splits (D1–D5) derived from these domains after cleaning are in Table 1.

3. **Length filtering**: remove triplets where any field has fewer than 5 or more than 2,000 characters.

4. **PE/MT length ratio filtering**: remove triplets where len(PE)/len(MT) $\notin [0.5, 1.5]$, excluding extreme rewrites or truncations that would distort TER-based evaluation.

5. **Shuffle and split**: shuffle with a fixed seed (42), then split 80/10/10 into train/validation/test.

This pipeline removes fewer than 1% of triplets for most domains. The released datasets include all triplets before cleaning, enabling researchers to apply their own preprocessing.

B.4 English–Tamil APE Dataset

Alongside the English–Sinhala data, we derive an English–Tamil APE dataset spanning five domains (Table 13) from the English–Tamil part of the same EnSiTa effort (Ranathunga et al., 2026): ∼39k (source, Google Translate MT, post-edit) triplets built with the same cleaning pipeline as D1–D4. Constrained decoding experiments use the News subset (10,000 triplets); per-domain vanilla fine-tuning results are in Table 22. The remaining domains are released to support future APE research on Dravidian languages. Tamil, a Dravidian language, is typologically distinct from the Indo-Aryan languages in our main experiments (Sinhala, Marathi).

C Full Results Tables

C.1 Per-Seed Breakdown

Table 14 gives the per-seed TER, BLEU, and chrF++ behind the three-seed means reported in the main text.

Domain	Triplets	Mean TER	Edit Dist.
Wiki	19,864	12.7	41/37/23
News	10,000	20.2	44/20/36
Literature	3,501	26.8	42/13/45
Maths	3,423	23.6	25/30/44
Health	1,870	7.3	69/17/15
Total	38,658	–	–

Table 13: English–Tamil APE dataset statistics. Edit Dist. shows the percentage of none/minor/major edits (TER thresholds: 0, 20). All domains use Google Translate MT. Experiments in this paper use only News; remaining domains are released for future research.

C.2 Token-Level Confidence Distribution

Figure 2 shows the full per-token probability–entropy distribution summarized in Table 3.

C.3 Temperature Scaling Results

Table 15 reports the effect of temperature scaling, which worsens unconstrained generation while preserving the constraint ordering. Recalibrating confidence therefore does not fix the failure. Only the static constraint, which ignores model confidence, stays near the MT baseline.

C.4 Lambda Sensitivity (Full Results)

Tables 16, 17, and 18 report the full λ sweeps for D1 and D5 (Gemma and NLLB-600M); Tables 19 and 20 cover English–Marathi and English–Tamil News; Tables 23–25 cover D2–D4.

C.5 Domain Robustness

Vanilla fine-tuning degrades TER on D1 (Literature, -0.87) and D4 (Maths/Health, -0.68), the two domains with higher MT TER (∼24–25). On D2 (News, $+0.02$) and D3 (Wiki, $+0.01$), where MT quality is already high (TER 8 and 15), it barely changes TER. The λ curves, however, reveal distinctions that these aggregate scores hide (Figure 5): D4's U-shaped curve beats the MT baseline at λ=2.0 (ΔTER= $+$ 0.21), whereas D1's monotonic curve only recovers to the MT level. D3 likewise dips below the MT baseline (ΔTER= $+0.45$, Table 24). D2 behaves differently: vanilla fine-tuning already improves marginally over MT, and any constraint worsens TER (Table 23), so the minimum sits at λ=0. This is consistent with net-positive editing where the edits are mostly mechanical (punctuation, spelling). Together, these curves support the hypothesis that Literature's style-transfer post-edits contribute to the failure.

Model	Method	Seed 42			Seed 52			Seed 62		
		TER	BLEU	chrF++	TER	BLEU	chrF++	TER	BLEU	chrF++
Gemma 1B	Vanilla	25.84	67.34	79.58	25.84	67.06	79.65	25.88	67.07	79.49
	HSA	25.26	67.42	79.78	25.89	67.20	79.58	25.60	67.24	79.62
	λ_{entropy}	25.81	67.30	79.55	25.65	67.24	79.71	25.69	67.14	79.54
	λ_{pmt}	25.65	67.34	79.65	25.18	67.59	79.88	25.29	67.43	79.72
	λ_{static}	24.97	67.31	79.86	25.07	67.43	79.86	24.91	67.39	79.86
Gemma 4B	Vanilla	25.15	67.43	79.88	25.36	67.46	79.84	25.72†	67.21†	79.61†
	λ_{entropy}	25.05	67.53	79.95	–	–	–	–	–	–
	λ_{pmt}	24.90	67.76	79.97	–	–	–	–	–	–
	λ_{static}	24.83	67.52	79.95	–	–	–	–	–	–
NLLB	Vanilla	31.10	61.77	77.30	30.44	62.25	77.54	30.80	61.92	77.29
	λ_{entropy}	31.28	61.57	77.09	–	–	–	–	–	–
	λ_{pmt}	29.97	62.58	77.68	–	–	–	–	–	–
	λ_{static}	29.59	62.63	77.61	–	–	–	–	–	–

Table 14: Full per-seed results on D1 (Literature). Most constrained decoding methods improve over vanilla fine-tuning, with static constraint consistently achieving the best results; entropy is the exception, trailing vanilla on NLLB. †Gemma 4B seed 62 diverged during QLoRA training; seed 72 used instead.

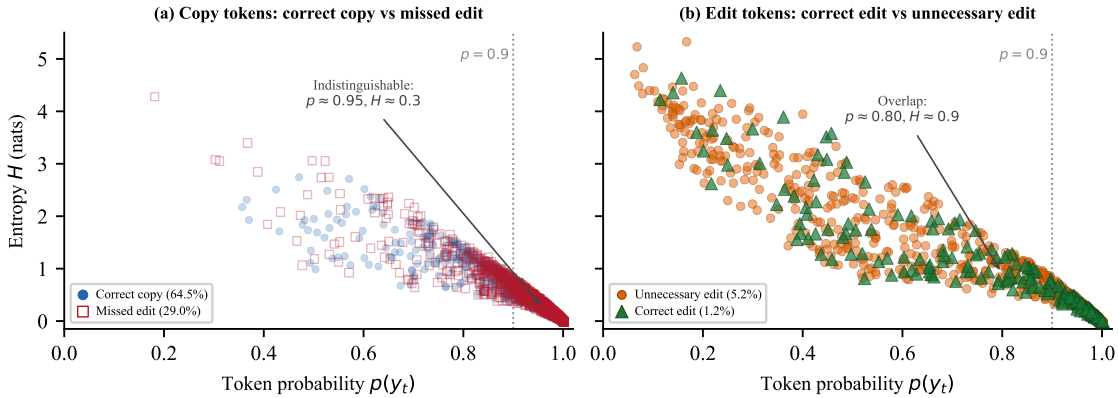

Figure 2: Token-level confidence vs. edit correctness for Gemma 3 1B on D1. 59.4% of unnecessary edits have $p > 0.9$, indistinguishable from the 54.8% of correct edits at $p > 0.9$: per-token probability does not separate net-positive edits from unnecessary ones. The 84.8% of missed edits at $p > 0.9$ shows confidence is equally uninformative about whether to edit at all.

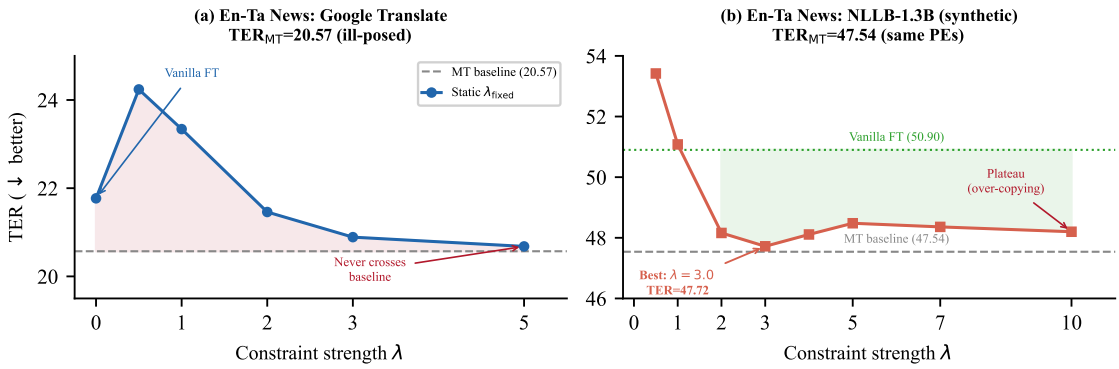

Figure 3: English–Tamil News λ curves: same post-edits, different MT. **(a)** Google Translate MT (TER_{MT}=20.57): monotonic, never crosses MT baseline. **(b)** NLLB-1.3B MT (TER_{MT}=47.54): U-shaped with minimum at λ=3.0 (TER = 47.72). The within-language contrast is consistent with a task-difficulty effect rather than a language-specific effect. Corresponding sweep values in Tables 20 and 21.

C.6 Seed Stability

Signal-agnostic constrained methods (static, entropy) are 3–4× more stable than HSA ($\sigma \leq 0.09$ vs. 0.32; Table 14), as expected for inference-time interventions that add no training stochasticity. PMT shows higher variance (σ=0.25) because its constraint strength depends on token-level probabilities that vary across checkpoints.

Method	TER (T=1)	TER (T=2)
MT Baseline	24.98	24.98
Vanilla FT (λ=0)	25.84	27.63
λ_{entropy} (best)	25.81	27.40
λ_{pmt} (best)	25.65	25.66
λ_{static} (best)	**24.97**	**25.08**

Table 15: Effect of temperature scaling (T^*=2.0, tuned on val NLL) on D1, seed 42. Calibration *worsens* unconstrained generation (25.84→27.63) and prevents the static constraint from recovering the MT baseline. The ordering Static > PMT > Entropy is preserved.

λ	Static	Entropy	PMT
0.0	25.84	–	–
0.1	25.81	–	–
0.3	25.66	–	–
0.5	25.65	25.81	25.79
1.0	25.00	25.81	25.69
2.0	24.98	25.81	25.65
2.5	**24.97**	–	–
3.0	24.96	25.81	25.76

Table 16: TER vs. λ sweep (D1, Gemma 3 1B, seed 42). Static converges to MT-level quality at $\lambda \geq 1.0$. Entropy is flat ($\sim$25.81 at all λ): the constraint barely activates. PMT shows a real curve with minimum at λ=2.0, but never approaches the MT baseline.

λ	TER ↓	Δ vs MT
MT Baseline	59.19	–
0.0	57.16	+2.03
0.1	56.99	+2.20
0.3	57.12	+2.07
0.5	57.03	+2.16
0.7	56.75	+2.44
1.0	56.11	+3.08
1.5	55.98	+3.21
2.0	**55.21**	**+3.98**
2.5	56.05	+3.14
3.0	55.47	+3.72
5.0	58.55	+0.64

Table 17: TER vs. λ sweep (D5, Gemma 3 1B, seed 42, static constraint). Unlike D1's monotonic curve, D5 shows a U-shape: TER dips below both vanilla FT (56.89) and MT baseline at λ=2.0, then rises at λ=5.0 as the constraint forces copying of low-quality MT.

C.7 Cross-Model λ Curves

Figure 4 extends the λ-curve analysis from Gemma 3 1B (Figure 1) to the 4B scale and to NLLB-600M on both D1 and D5, showing the curves for additional architectures and task conditions.

D Dataset Examples

D.1 Edit Type Taxonomy

Table 29 lists representative D1 examples of each edit type.

λ	TER ↓	Δ vs MT
MT Baseline	59.19	–
−5.0	77.37	−18.18
−3.0	71.24	−12.05
−2.0	66.05	−6.87
−1.0	61.10	−1.91
−0.5	58.87	+0.31
0.0	**57.14**	**+2.05**
1.0	57.34	+1.85
1.5	57.34	+1.85
2.0	57.71	+1.47
3.0	59.94	−0.76
6.0	64.59	−5.41
14.0	71.34	−12.15

Table 18: TER vs. λ sweep (D5, NLLB-600M, seed 42, static constraint). The curve is U-shaped with its minimum at λ=0 (vanilla FT). Negative λ penalizes MT tokens (encouraging hallucination); positive λ forces MT copying. Both directions degrade performance, with the negative direction steeper, consistent with low-quality MT structure providing useful scaffolding. Values saturate at $|\lambda| \geq 12$.

λ	λ_{static}	λ_{pmt}	λ_{entropy}
MT Baseline		22.93	
Vanilla FT		22.24	
0.5	21.98	22.25	22.21
1.0	21.83	22.19	22.15
2.0	**21.73**	22.05	22.10
3.0	21.83	21.90	22.09
5.0	22.46	21.83	22.06

Table 19: TER vs. λ sweep (English–Marathi, Gemma 3 1B, seed 42). All three constraint methods shown. The static constraint produces a U-shaped curve with best TER 21.73 at λ=2.0. The ordering Static > PMT > Entropy holds at every λ value tested. Entropy TER is nearly flat across λ (same pattern as D1 and D5), confirming that the entropy-scaled constraint barely activates.

λ	λ_{static} TER
MT Baseline	20.57
Vanilla FT	21.77
0.5	24.24
1.0	23.34
2.0	21.46
3.0	20.89
5.0	20.68

Table 20: TER vs. λ sweep (English–Tamil News, Gemma 3 1B, seed 42, static constraint). The curve is monotonic, approaching but never crossing the MT baseline (20.57). Plotted in Figure 3(a).

D.2 LLM-Based Edit-Type Classification

To scale edit-type analysis beyond manual annotation, we classify 200 stratified samples per domain using GPT-5.2 with structured output. We adopt a simplified subset of the Multidimensional Qual-

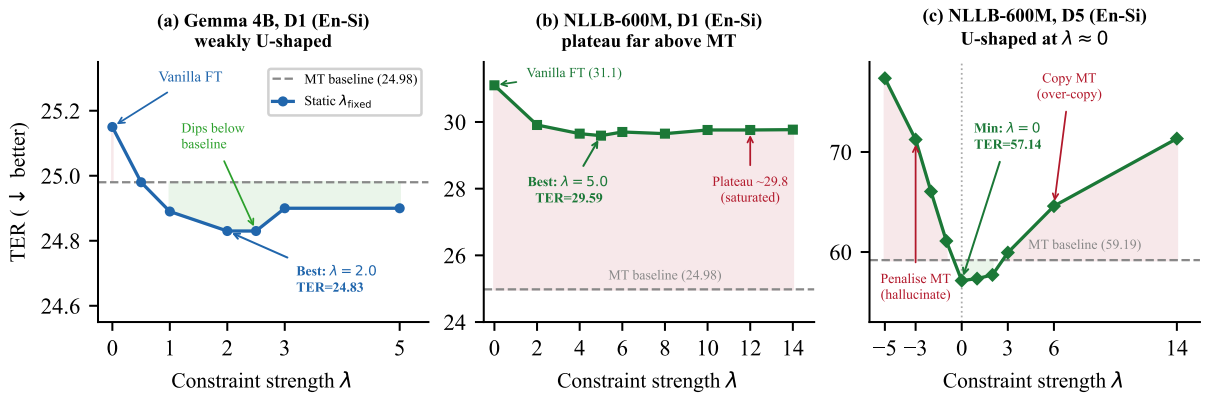

Figure 4: TER vs. λ curves across model architectures. **Left:** Gemma 4B on D1 (seed 42): weakly U-shaped, the only model to dip below the MT baseline (ΔTER$= +0.15$ at λ=2.0). Compare with the monotonic 1B curve in Figure 1(a). **Center:** NLLB-600M on D1 (seed 42): decreasing until λ=5, then plateauing at TER$\approx$29.8 (λ=15 and λ=20 give identical results, confirming constraint saturation). Best TER 29.59 at λ=5.0, still 4.61 above MT; the encoder–decoder's over-editing is too severe for an inference-time constraint to undo. **Right:** NLLB-600M on D5 (seed 42): U-shaped, centered at $\lambda\approx$0; neither constraint direction improves performance. Negative λ penalizes MT-window tokens; the steeper left arm is consistent with MT structure providing useful scaffolding. The D1/D5 contrast is consistent with task differences rather than architecture alone.

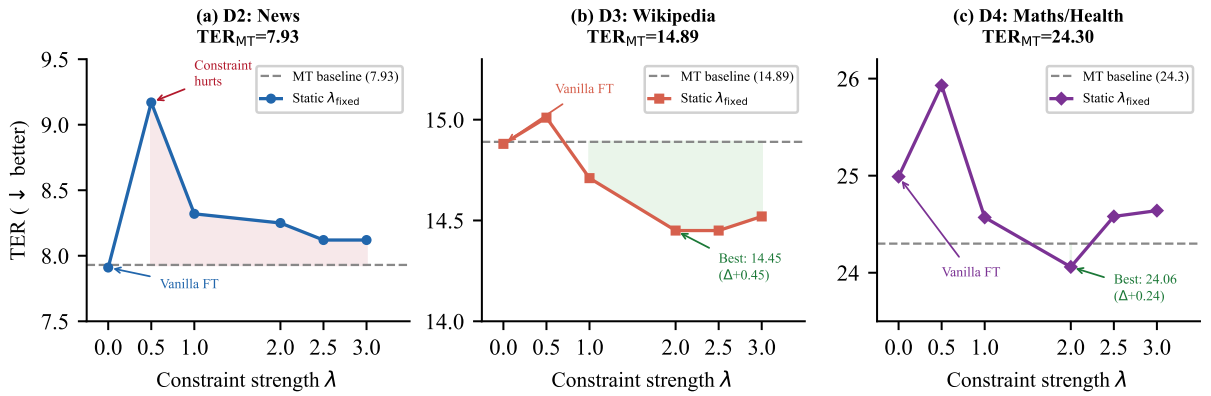

Figure 5: TER vs. λ curves for D2–D4 (Gemma 3 1B, seed 42, static constraint). **Left:** D2 (News, TER_{MT}=7.93): vanilla FT marginally improves over MT; the constraint monotonically worsens TER, confirming the model's edits are already net-positive. **Center:** D3 (Wikipedia, TER_{MT}=14.89): vanilla FT shows negligible change (ΔTER$= +0.01$), but the static constraint reveals genuine correctable errors, reaching TER$=14.45$ at λ=2.0 (ΔTER$= +0.45$); mild U-shape with uptick at λ=3.0. **Right:** D4 (Maths/Health, TER_{MT}=24.30): vanilla FT degrades TER (-0.68), but unlike D1 the curve is U-shaped with best TER$=24.09$ at λ=2.0 (ΔTER$=+0.21$), revealing genuine correctable errors despite similar MT quality.

ity Metrics (MQM) framework (Lommel et al., 2014), following the MQM-Core error typology used in WMT human evaluation campaigns (Freitag et al., 2021). For each sample, we produce word-level diffs and assign one primary edit from 14 categories. Thirteen belong to four MQM groups: **Accuracy** (mistranslation, polysemy, omission, addition, untranslated), **Fluency** (morphological, agreement, punctuation), **Style** (loanword nativization, register shift, word order, lexical preference), and **Retranslation** (complete rewrite). The residual `no_meaningful_diff` class covers exact or near-identical copies. We classify three views per domain: human edits (MT→PE), model actions (MT→predicted), and remaining gap (predicted→PE), enabling direct comparison of human corrections and model actions.

D1 (Literature): Per-type breakdown. Table 31 presents the full per-type classification for D1. Human edits are dominated by stylistic changes: lexical preference (16.5%), register shift (13.0%), word order (7.0%), and loanword nativization (4.5%). Among genuine errors, mistranslation is the most common (22.0%). The model's actions are strikingly narrow: 71.5% exact copies, 20.0% punctuation changes, and 5.0% mistranslation fixes. No retranslations, no register

λ	TER ↓	Δ vs MT
MT Baseline	47.54	–
Vanilla FT	50.90	−3.36
0.5	53.42	−5.87
1.0	51.08	−3.54
2.0	48.16	−0.62
3.0	**47.72**	**−0.18**
4.0	48.11	−0.57
5.0	48.48	−0.94
7.0	48.36	−0.82
10.0	48.20	−0.66

Table 21: TER vs. λ sweep (English–Tamil News with NLLB-1.3B MT, Gemma 3 1B, seed 42, static constraint). Same post-edits as Table 20 but with synthetic NLLB MT. The curve is U-shaped with minimum at λ=3.0, then plateaus at ∼48.2 for $\lambda \geq 7$. Plotted in Figure 3(b).

Domain	MT TER	FT TER	ΔTER	Triplets
Wiki	12.67	12.95	−0.28	23,880
News	20.57	21.77	−1.21	10,000
Combined†	19.07	18.96	+0.11	8,794

Table 22: English–Tamil per-domain vanilla fine-tuning results (Gemma 3 1B, seed 42). MT source is Google Translate for all domains. Constrained decoding experiments are conducted only on the News domain (Section 5.5). †Literature (3,501), Maths (3,423), and Health (1,870) are combined into a single split because individually each is too small for stable fine-tuning; Literature alone failed to converge.

λ	TER ↓	Δ vs MT
MT Baseline	7.93	–
0.0	7.91	+0.02
0.5	9.17	−1.24
1.0	8.32	−0.39
2.0	8.25	−0.33
2.5	8.12	−0.19
3.0	8.12	−0.20

Table 23: TER vs. λ sweep (D2 News, Gemma 3 1B, seed 42, static constraint). Vanilla FT already marginally improves over MT; the constraint monotonically worsens TER, confirming the model already makes net-positive edits and no inference-time correction is needed.

shifts, no word order changes.

Cross-tabulation. Table 32 shows what the model does for each human edit category. When humans make style edits, the model either copies (56/82) or makes unrelated fluency changes (19/82), almost never producing actual style edits (1/82). When humans fix accuracy errors, the model copies 48/59 and attempts only 11, with 0 matching PE exactly.

λ	TER ↓	Δ vs MT
MT Baseline	14.89	–
0.0	14.88	+0.01
0.5	15.01	−0.12
1.0	14.71	+0.19
2.0	14.45	+0.45
2.5	14.45	+0.44
3.0	14.52	+0.38

Table 24: TER vs. λ sweep (D3 Wikipedia, Gemma 3 1B, seed 42, static constraint). The constraint beats the MT baseline at $\lambda \geq 1.0$, with best TER 14.45 at λ=2.0 (ΔTER$= +0.45$). Mild U-shape with uptick at λ=3.0.

λ	TER ↓	Δ vs MT
MT Baseline	24.30	–
0.0	24.99	−0.68
0.5	25.93	−1.63
1.0	24.57	−0.27
2.0	24.09	+0.21
2.5	24.58	−0.27
3.0	24.64	−0.34

Table 25: TER vs. λ sweep (D4 Maths/Health, Gemma 3 1B, seed 42, static constraint). Despite vanilla FT degrading TER by 0.68, the U-shaped curve reveals genuine correctable errors, with best TER 24.09 at λ=2.0 (ΔTER$= +0.21$, beats MT baseline).

Dataset	System	TER ↓	BLEU ↑	ΔTER
D2 (News)	MT Baseline	7.93	89.25	–
	Vanilla FT	7.91	89.56	+0.02
	λ_{static}	8.12	89.39	−0.19
D3 (Wiki)	MT Baseline	14.89	79.11	–
	Vanilla FT	14.88	78.95	+0.01
	λ_{static}	14.45	79.20	+0.45
D4 (Math/H)	MT Baseline	24.30	71.18	–
	Vanilla FT	24.99	69.28	−0.68
	λ_{static}	24.09	70.43	+0.21
D1 (Lit.)	MT Baseline	24.98	67.38	–
	Vanilla FT	25.85	67.16	−0.87
	λ_{static}	24.98	67.38	0.00

Table 26: Domain robustness (Gemma 3 1B, seed 42). Vanilla FT degrades TER on D1 (−0.87) and D4 (−0.68), but the static constraint reveals different underlying patterns: D4 shows a U-shaped curve that beats MT (ΔTER$= +0.21$), while D1 only recovers to MT level. D3's constraint similarly reveals correctable errors (ΔTER$= +0.45$). D2 is the opposite case: vanilla FT already marginally improves, and any constraint worsens TER (−0.19), so no tested constraint improves this result. Only on D1 (Literature) do no net-positive edits survive.

Cross-domain comparison. Table 33 summarizes the edit-type profile and model behavior across D1–D3, which share the same MT system (Google Translate) and task structure. The pattern is consistent: higher proportions of subjective edits correlate with lower model success

Parameter	Gemma 1B	Gemma 4B	NLLB
Learning rate	3e-5	3e-5	2e-4
Scheduler	Linear	Linear	Linear
Warmup ratio	0.10	0.10	0.10
Batch (eff.)	64	16×4	48×1
Max epochs	10	10	8
Eval steps	50	50	100
Early stop.	4 (δ=.002)	4 (δ=.002)	2 evals
Weight decay	0.01	0.01	0.01
LoRA r/α	32/32	32/32	32/64
LoRA dropout	0.00	0.00	0.05
Quantization	None	4-bit NF4	None
Max seq. len.	2048	2048	512/256
Time (A100)	∼0.5h	∼0.5h	∼0.5h

Table 27: Training hyperparameters and compute. Batch size is per-device × gradient accumulation = effective. Decoder-only Gemma models are fine-tuned to generate the post-edit from a `Source: {src} \n MT: {mt}` prompt; NLLB-600M takes `src </s> mt` as encoder input.

Edit Type	Example (MT → PE)
Stylistic (MT semantically correct):	
Loanword → native	ස්පයිඩර් → මකුළුවා (*"Spider" → native term*)
Register shift	දනිමු → දන්නවා (*literary → spoken "know"*; shifting between diglossic forms (de Silva, 2026))
Word order (SVO→SOV)	Clause reordering to Sinhala-preferred order
Retranslation	Complete rewrite, same meaning
Correction (actual MT error):	
Morphological	නිදිමත → නිදිමතින් (*missing instrumental -ින්*)
Polysemy error	ණන්කාරයා → කවුන්ට්වරයා (*"creditor" → "count" (title)*)

Table 28: Edit type taxonomy from manual analysis of 90 (MT, PE) pairs across D1–D3 (30 per domain). Edits span both stylistic and corrective categories with no single type dominating, particularly in D1 (Literature).

rates and higher copy rates. The net edit effect (Table 34) is consistent with signal *inconsistency*, rather than composition, contributing to the outcome: D3 matches D1's subjective proportion yet its edits stay net-positive, while only D1's editing is net-negative.

D2 (News) is the cleanest domain: 68.0% of human edits are fluency corrections (mostly punctuation, 58.5% of all edits), only 24.5% are subjective, and the model achieves the highest success rate (24.0%, 48/200). D3 (Wikipedia) resembles D1 in edit profile (48.0% subjective) but with more fluency edits (33.0% vs 20.5%), yielding a moderate success rate (7.5%). On D1 (Literature), 49.5% of edits are subjective, the copy rate peaks at 71.5%, and only 4.0% of model outputs match PE.

Editing is net-negative only on D1. The copy and success rates above are measured on the 200 stratified samples (edit-needed rows only). Table 34 reports the complementary picture on the *full* test sets, including the decisive signal: the net effect of the model's edits. Over the sentences the model chooses to edit, we compute the mean change in TER relative to leaving the MT unchanged. Editing is net-negative only on D1 (+6.3); on D2, D3, and the low-quality-MT contrast D5 the model's edits are net-positive, even though D3's edit-type composition matches D1's (Table 33). Test-time edit behavior thus tracks the consistency of the post-edit signal, paralleling the vanilla-FT degradation pattern (D1 alone degrades) and the constrained-decoding curves (Section 4.4).

NLLB-600M comparison. Applying the same analysis to NLLB-600M predictions on D1 reveals a different failure mode. The encoder–decoder copies less (48.0% vs Gemma's 71.5%), editing more aggressively, yet achieves an even lower success rate (2.5% vs 4.0%). Copy rates are uniformly lower across all categories (accuracy 52.6%, style 50.6%, fluency 34.1%), confirming that NLLB over-edits indiscriminately rather than selectively. The model's extra edits land mostly in fluency (punctuation: 61/104 non-copy actions), mirroring Gemma's narrow repertoire at higher volume. This is consistent with NLLB's catastrophic TER degradation (−5.80; Section 5.2): the over-editing is too severe for inference-time correction.

Classification prompt. Each batch of 10 samples is sent to GPT-5.2 with structured output (Pydantic schema) at temperature 0. The system prompt instructs the model to act as a Sinhala linguistics expert and provides the full taxonomy with Sinhala-specific examples. The user message supplies the English source, word-level diff, and TER for each sample. The full system prompt is shown below.

```
You are a Sinhala linguistics
expert classifying edit types in
English-to-Sinhala translation
post-editing.
```

Domain	Source (English)	Key Edit Type	TER
D1 (Lit.)	Spider replied, “Yes, I’ll take you on an adventure...”	Loanword: ස්පයිඩර්→මකුළුවා (“Spider” → native term) + verb formality	27.3
D1 (Lit.)	“I didn’t think,” she sobbed.	Mistranslation: කෑගැසුවාය (“screamed”) → වැළපුණා ය (“sobbed”)	33.3
D1 (Lit.)	Thank you, Mrs. Goose.	Loanword: මිසිස් ගුස්→පාත්ත මහත්මිය (English title → native)	66.7
D1 (Lit.)	“Ahmed?” drowsily.	Morphological: නිදිමත→නිදිමතින් (missing instrumental case -ින්)	50.0

Table 29: Representative examples from D1, illustrating the heterogeneity of edit types. The examples include loanword replacement, mistranslation correction, and morphological fixes.

MQM Category	Human	Copied	Fixed	Copy%
Accuracy	59	48	0	81.4
Fluency	41	26	7	63.4
Style	82	56	1	68.3
Retranslation	17	12	0	70.6
Total	200	143	8	71.5

Table 30: Edit-type decomposition of Binary Collapse on D1 (Gemma 3 1B, seed 42, 200 samples). **Human**: number of samples where humans made this type of edit. **Copied**: model copied MT exactly. **Fixed**: model output matches PE. Binary Collapse is not selective: the model copies at high rates across all categories. Of 8 successful corrections, 7 are punctuation. (Summarized in Section 4.3.)

Edit Type	Human	Model	Gap
Accuracy			
mistranslation	44	10	45
addition	6	–	6
untranslated	4	2	3
polysemy_sense	3	–	1
omission	2	1	2
Fluency			
punctuation	29	40	25
morphological	12	–	15
Style			
lexical_preference	33	1	39
register_shift	26	1	24
word_order	14	–	12
loanword_nativ.	9	2	3
retranslation	17	–	16
no_meaningful_diff	1	143	9

Table 31: Per-type edit classification for D1 (200 samples, Gemma 3 1B, seed 42). **Human**: MT→PE edits. **Model**: MT→predicted actions. **Gap**: predicted→PE remaining differences. The model’s repertoire is limited to punctuation (40) and a few mistranslation fixes (10); all other edit types are effectively abandoned.

```
MQM-Core Edit Type Taxonomy (adapted
for MT post-editing):

Category 1 --- ACCURACY errors
(genuine MT errors):
mistranslation, polysemy_sense,
omission, addition, untranslated

Category 2 --- FLUENCY errors
(semantically correct, grammatically
ill-formed):
morphological, agreement, punctuation

Category 3 --- STYLE/REGISTER
preferences (correct and well-formed):
loanword_nativization, register_shift,
word_order, lexical_preference

Category 4 --- RETRANSLATION:
retranslation (complete rewrite,
semantically adequate)

Other: no_meaningful_diff (for [EXACT
COPY] or near-identical)

For each sample: (1) classify PRIMARY
edit type, (2) list secondary types,
(3) set mqm_category, (4) write 5--10
word note.
```

	Model Action Category				
Human	**None**	**Acc.**	**Flu.**	**Sty.**	**Ret.**
Accuracy	48	5	4	2	–
Fluency	26	–	15	–	–
Style	56	6	19	1	–
Retranslation	12	2	2	1	–

Table 32: Cross-tabulation of human edit category vs. model action category on D1 (200 samples). Rows: what humans edited. Columns: what the model did. The “None” column (exact copies) dominates every row. When the model does act on style edits, it typically makes fluency (punctuation) changes instead.

	D2 *News*	**D3** *Wiki*	**D1** *Lit.*
Human edit distribution (%)			
Accuracy	7.5	19.0	29.5
Fluency	68.0	33.0	20.5
Style	19.0	41.0	41.0
Retranslation	5.5	7.0	8.5
Model behavior (%)			
Copy rate	40.5	51.5	71.5
Success rate	24.0	7.5	4.0
Subjective %	24.5	48.0	49.5

Table 33: Cross-domain edit-type profiles and model behavior (200 samples per domain, Gemma 3 1B, seed 42). Subjective % = style + retranslation. Domains ordered by increasing subjectivity. Success rate = model output exactly matches PE.

Domain	n	Edit rate	Edits ≠PE	per-edit ΔTER
D1 Literature	716	23.6	88.8	+6.3
D2 News	1664	37.7	72.6	−0.9
D3 Wikipedia	2184	41.6	80.6	−1.1
D5 (low-qual. MT)	1292	79.8	98.4	−3.2

Table 34: Test-time edit behavior on the *full* test sets (Gemma 3 1B, seed 42). **Edit rate**: % of sentences where the output differs from the MT. **Edits≠PE**: of those, % not matching the post-edit. **per-edit ΔTER**: mean [TER(output,PE) − TER(MT,PE)] over edited sentences; positive means editing moves the output *away* from the post-edit. Editing is net-negative only on D1, the domain with the most inconsistent post-edit signal; D1, D2, D3 share the same MT (Google Translate), D5 uses NLLB and is shown as a contrast. TER computed with Asian-aware tokenization.

Each sample in the user message follows the format: `id`, `SRC` (English source), `diff` (word-level changes marked with `(- deleted -)` and `(+ inserted +)`), and `TER`. The structured output schema enforces a list of classifications, each with `primary_type` (enum), `secondary_types` (list of enums), `mqm_category` (string), and `brief_note` (string).

D.3 Qualitative Model Output Examples

Table 35 presents three qualitative examples from D1, contrasting the vanilla fine-tuned model with the static constraint.

Example 2 is the prototypical case: the reference PE changes only a pronoun form and a verb ending (both stylistic), yet the vanilla model, rather than replicating these minor fixes, corrupts a content word ("cobra") into a non-word. Example 3 shows a rarer case where both PE and the model identify the same genuine error (an untransliterated proper noun), but the model hallucinates the wrong transliteration. In both cases, the entropy-based constraint would *not* intervene (the model is confident in its edits), which is precisely why it underperforms the static constraint.

E MQM Classification Validation

E.1 Annotator Instructions

Three native Sinhala speakers with professional or near-professional bilingual competence in English and Sinhala were the annotators. Annotators were compensated at fair market rates and completed the task independently with no inter-annotator discussion.

The following instructions were provided in both English and Sinhala:

```
You will be shown an English sentence,
its Sinhala machine translation (MT),
the human post-edited version, and a
word-level diff showing what changed
between MT and post-edit. An automated
system has assigned a category
label to the edit. Your task is to
judge whether this label correctly
describes the PRIMARY change made by
the post-editor.

The four categories are:

  • ACCURACY: The MT contained a
    genuine error (wrong meaning,
    mistranslation, omission,
    addition, untranslated text).
    The post-editor fixed this error.
  • FLUENCY: The MT was semantically
    correct but grammatically
    ill-formed (wrong morphological
    form, agreement error,
    punctuation). The post-editor
    fixed the grammatical issue.
  • STYLE/REGISTER: The MT was
    semantically correct and
    grammatically well-formed, but
    the post-editor changed it for
    stylistic reasons (replaced a
    loanword with a native term,
    changed register from formal
    to colloquial, reordered words,
    substituted a synonym).
  • RETRANSLATION: The post-editor
    completely rewrote the sentence
    while preserving the meaning.

For each sample answer: (1) Is the
assigned label correct? YES / NO /
PARTIALLY. (2) If NO or PARTIALLY,
select the correct category.
```

E.2 Calibration Examples

Before the main task, annotators completed five calibration samples with correct answers revealed. Two examples are shown in Table 36 to illustrate the ACCURACY/STYLE distinction, which is the most frequently confounded boundary.

E.3 Sample Selection

We drew 100 samples from the GPT-5.2 classified D1 set, stratified by MQM category: 25 each for Accuracy, Fluency, and Style; all 17 available Retranslation samples; and 8 additional samples drawn proportionally from the larger categories to reach 100. Samples within each category were selected randomly (seed 42). The same 100 samples were evaluated by all three annotators.

E.4 Results

Table 37 reports per-category agreement and Fleiss' κ.

Example 1: Unnecessary Edit (MT = PE, $\text{TER}_{\text{ref}} = 0$)	
Source	"I must buy you."
MT Output	"මම ඔබව මිලදී ගත යුතුයි."
Human PE	"මම ඔබව මිලදී ගත යුතුයි." (identical to MT)
Vanilla FT	"මම ඔබව මිලදී ගන්නවා." (changed ගත යුතුයි "must buy" → ගන්නවා "will buy": unnecessary modality shift)
Constrained	"මම ඔබව මිලදී ගත යුතුයි." (correctly preserved MT, TER = 0)
MT is correct. The model changes obligation to simple future at high confidence, altering the meaning.	
Example 2: Stylistic Over-Edit (PE makes 2 small changes; model corrupts a content word)	
Source	Cut off a cobra's head and it could only wriggle until sunset.
MT Output	නාගයෙකුගේ හිස කපා දමන්න, එය ඉර බැස යන තුරු පමණක් දඟලන්නට විය.
Human PE	නාගයෙකුගේ හිස කපා දමන්න, ඌට ඉර බැස යන තුරු පමණක් දඟලන්නට පුළුවන. (*Only 2 changes: pronoun* එය→ඌට *and verb form* විය→පුළුවන)
Vanilla FT	නැගියෙකුගේ හිස කපා දමන්න, එය ඉර බැස යන තුරු පමණක් දඟලන්නට විය. (corrupts නාගයෙකුගේ "cobra's" → නැගියෙකුගේ, a non-word)
Constrained	නාගයෙකුගේ හිස කපා දමන්න, එය ඉර බැස යන තුරු පමණක් දඟලන්නට විය. (preserves MT)
PE makes minor stylistic adjustments. The model ignores PE's changes, instead corrupting "cobra" into a non-word.	
Example 3: Correct Intent, Wrong Execution (MT leaves proper noun in Latin script)	
Source	That weekend, she went to Ville Belle with her father.
MT Output	එම සති අන්තයේ ඇය තම පියා සමඟ Ville Belle වෙත ගියාය. (Ville Belle left untransliterated)
Human PE	එම සති අන්තයේ, ඇය තම පියා සමඟ විල් බෙල් වෙත ගියාය. (correctly transliterates to විල් බෙල්)
Vanilla FT	එම සති අන්තයේ, ඇය තම පියා සමඟ විල් බල්ෙ වෙත ගියාය. (attempts transliteration but produces විල් බල්ෙ, an incorrect rendering)
Constrained	එම සති අන්තයේ ඇය තම පියා සමඟ Ville Belle වෙත ගියාය. (preserves MT)
Both PE and vanilla identify the same issue (untransliterated name), but the model hallucinates the wrong transliteration.	

Table 35: Qualitative examples from D1 (all text is real model output from Gemma 3 1B, seed 42; constrained = λ_{static} at λ=2.0). Example 1: the model edits perfect MT. Example 2: the model ignores PE's minor stylistic fixes and instead corrupts a content word. Example 3: the model attempts the same correction as PE but hallucinates the wrong transliteration. Constrained decoding preserves the MT output in all three cases.

Field	Content	Label
Source	"I didn't think," she sobbed.	
MT	"මම හිතුවේ නැහැ," ඇය කෑගැසුවාය.	
PE	"මම හිතුවේ නැහැ," ඇය වැළපුණා ය.	
Diff	කෑගැසුවාය (screamed) → වැළපුණා ය (sobbed)	**ACCURACY**
Source	Spider replied, "Yes..."	
MT	ස්පයිඩර් පිළිතුරු දුන්නේ...	
PE	මකුළුවා පිළිතුරු දුන්නේ...	
Diff	ස්පයිඩර් (loanword) → මකුළුවා (native)	**STYLE**

Table 36: Calibration examples for the ACCURACY/STYLE boundary. The first is ACCURACY ("screamed" is a mistranslation of "sobbed"). The second is STYLE (both are valid Sinhala renderings; the native term is a stylistic preference).

E.5 Confusion Matrix

Table 38 shows the confusion matrix between GPT-5.2 labels and consensus human labels.

F Evaluation Metrics

All TER scores use SacreTER (Post, 2018) with default tokenization (no normalization); BLEU uses SacreBLEU with `intl` tokenization; chrF++ uses β=2 with word bigrams (`word_order`=2). The

Category	GPT (%)	Corrected (%)	Agree (%)	κ
Accuracy	26	29	73	0.64
Fluency	26	27	77	0.72
Style	31	28	65	0.62
Retranslation	17	16	82	0.92
Overall	100	100	73	0.70

Table 37: GPT-5.2 classification validation. **GPT**: original automated distribution. **Corrected**: distribution after human corrections. **Agree**: proportion confirmed by all three annotators. κ: per-category binary Fleiss' κ across the three annotators (overall: multiclass Fleiss' κ).

GPT label →	Acc.	Flu.	Sty.	Ret.
Human: Accuracy	26	0	3	0
Human: Fluency	0	26	1	0
Human: Style	0	0	27	1
Human: Retranslation	0	0	0	16

Table 38: Confusion matrix between GPT-5.2 labels (columns) and consensus human labels (rows) on 100 validation samples. Diagonal entries indicate agreement; off-diagonal entries indicate systematic confusions. GPT overclassifies Style: 3 cases relabeled as Accuracy, 1 as Fluency.

MT baseline is evaluated under identical settings. Zero-shot TER values for base LLMs are high be-

cause uninstructed models generate fluent but task-unrelated Sinhala text (Appendix A).

F.1 Metric Robustness of the Diagnostic

The diagnostic reads the shape of the constraint curve, not absolute scores, so it should not depend on the metric that traces the curve. We recomputed the static-constraint sweep under chrF++ and BLEU for D1, D5, and English–Marathi (Gemma 3 1B, seed 42). Table 39 reports the best improvement over the MT baseline under each metric. The U-shaped versus monotonic distinction holds under every metric: on D1 the gains are within noise on all three, while on D5 and English–Marathi the curve peaks clearly above the MT baseline on each. The calibration finding (Table 3) is measured on the probability and entropy values directly, so it is independent of the sequence metric.

Dataset	ΔTER	ΔchrF++	ΔBLEU
D1 (monotonic)	+0.03	+0.07	+0.05
D5 (U-shaped)	+3.98	+0.97	+2.58
En-Mr (U-shaped)	+1.20	+0.90	+3.03

Table 39: Best improvement over the MT baseline along the static λ sweep, per metric (Gemma 3 1B, seed 42). All deltas are oriented so that positive is better. The monotonic/U-shaped distinction survives under chrF++ and BLEU.

G Failed Interventions

Before arriving at constrained decoding as a diagnostic, we explored three training-time interventions aimed at reducing the over-editing behavior documented in Section 4.3. Each targets a different hypothesis about the failure mechanism: **Hidden State Anchoring (HSA)** regularizes the model's internal representation of the MT span to prevent drift during fine-tuning; **Edit-Level Conditioning (ELC)** provides explicit edit-level tags (none/minor/major) as input to guide the model's editing intensity; and **Chain-of-Thought (CoT) Self-Tagging** trains the model to predict its own edit level before generating the post-edit, testing whether the model can learn to self-diagnose. All three failed to reduce over-editing, and in some cases worsened TER substantially (Table 41). In these experiments, the failures show that the tested internal signals did not provide reliable regularization targets (HSA), conditioning signals (ELC), or self-predicted tags (CoT), motivating the inference-time, confidence-agnostic constraints in Section 4.4.

Hidden State Anchoring (HSA). HSA adds an auxiliary regularization loss during training that penalizes drift in the model's hidden-state representation of the MT span. At each training step: (1) identify the MT token span in the input, (2) extract hidden states from the final Transformer layer and apply layer normalization, (3) mean-pool the MT span hidden states into a single vector $\mathbf{h}_{\text{mt}}$, and (4) add a weighted MSE penalty:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{APE}} + \alpha \cdot \|\mathbf{h}_{\text{mt}} - \mathbf{a}\|^2 \quad (4)$$

where $\mathbf{a}$ is the anchor target. We ablated three anchor strategies (Section H): **zero-scalar** (α=0.1), **zero-vector** (α=0.001), and **MT-vector** (α=0.001). The simplest variant (zero-scalar) performed best, but with high seed variance (σ=0.26).

Edit-Level Conditioning (ELC). The training prompt augments the input with a gold edit-level tag:

```
Source: {source}
MT: {mt}
Edit level: {none | minor | major}
```

Tags are derived from TER(MT, PE): TER=0 → `none`, TER<20 → `minor`, TER≥20 → `major`. At inference, gold tags are unavailable (they require the reference), so a single fixed tag must be applied to all test samples. "ELC (oracle)" uses per-sample gold tags (a cheating upper bound) but actually *degrades* performance (TER 27.29 vs. baseline 24.98), primarily due to noise on the `none` class where the model still hallucinates edits. The fixed `minor` tag (TER 25.88) performs comparably to vanilla fine-tuning, confirming that a single tag cannot capture the heterogeneous edit requirements. This is consistent with the QE results in Appendix J, where even fine-tuned XLM-R achieves only AUROC 0.745 for edit-level detection.

Chain-of-Thought (CoT) Self-Tagging. Instead of providing the edit level as input, the model is trained to *predict* it before generating the post-edit. The input contains only source and MT:

```
Source: {source}
MT: {mt}
```

and the target is the edit-level tag followed by the post-edited text:

```
[none | minor | major]
{post-edited text}
```

Table 40 shows that the model's predicted tag distribution diverges dramatically from the gold distribution. The confusion matrix confirms: 90% of gold-minor samples (135/150) are tagged as `[major]`, and 66% of gold-none samples (137/208) are also tagged as `[major]`. The resulting TER (29.55) is the worst of all methods, with the heaviest degradation on the none class (ΔTER$= -7.09$).

Tag	Gold (%)	Predicted (%)	Δ
`none`	29.1	17.7	−11.4
`minor`	20.9	0.1	−20.8
`major`	50.0	82.1	+32.1
Tag accuracy		53.9%	

Table 40: CoT self-tagging: gold vs. predicted edit-level distribution on D1 test set. The minor class is virtually eliminated (−20.8pp), absorbed almost entirely by the major class (+32.1pp).

G.1 Summary of Failed Interventions

Method	TER ↓	ΔTER	σ	Note
MT Baseline	24.98	–	–	–
Vanilla FT	25.84	−0.86	0.02	–
HSA (mean)	25.58	−0.60	0.26	High σ
ELC (oracle)	27.29	−2.31	–	Noise on none class
ELC (minor)	25.88	−0.90	–	≈ vanilla
CoT	29.55	−4.57	–	82% predict `[major]`

Table 41: Failed interventions on D1 (Literature), Gemma 3 1B. HSA is seed-dependent (σ=0.26); ELC degrades even with oracle tags; CoT collapses to predicting `[major]` for 82% of samples.

H HSA Ablation

Table 42 compares the three HSA anchor strategies on D1 (Gemma 3 1B, seed 42). All variants use the final Transformer layer with layer normalization; α was tuned per variant: α=0.1 for zero-scalar and α=0.001 for both zero-vector and MT-vector.

Anchor	Target	α	TER↓	Δ
Vanilla FT	–	–	25.84	–
Zero-scalar	$\bar{h}_{\text{mt}} \to 0$	0.1	25.26	+0.58
MT-vector	$\mathbf{h} \to \mathbf{h}^{\text{frz}}$	0.001	25.73	+0.11
Zero-vector	$\mathbf{h} \to \mathbf{0}$	0.001	25.69	+0.15

Table 42: HSA anchor ablation on D1 (seed 42). Zero-scalar performs best but is modest (+0.58 TER) and highly seed-dependent (σ=0.26).

Zero-scalar penalizes the global mean of all pooled MT span features toward zero, a coarse regularizer that prevents the model from drifting the MT representation space too far from initialization. **MT-vector** anchors each sample's pooled MT hidden state to the corresponding representation from the frozen base model (precomputed before fine-tuning). **Zero-vector** anchors each sample directly to the zero vector via per-sample MSE. Paradoxically, the simplest variant (zero-scalar) outperforms the more targeted alternatives, suggesting that the fine-grained per-sample anchoring adds noise in the low-resource setting.

I Constrained Decoding Details

I.1 Algorithm

Algorithm 1 presents the pseudocode for constrained decoding. At each step, the method computes a (possibly dynamic) λ_t, then subtracts it from all token logits and adds it back for tokens in the MT alignment window $\mathcal{W}_t$. A pointer tracks alignment through the MT sequence and advances when the generated token matches a window token. When no window token matches (e.g., under reordering or insertion), the pointer does not advance and the window effectively widens until alignment is re-established. This makes the scheme robust to local reordering, at the cost of occasionally under-penalizing divergent continuations.

Algorithm 1: Constrained Decoding

Input: Model M, MT tokens $\mathbf{x}^{\text{mt}}$, type, λ, window k, γ or β
Output: Post-edited tokens $\mathbf{y}$

Initialize MT pointer ptr $\leftarrow 0$
for $t = 1, 2, \ldots$ **do**
  $\mathbf{l}_t \leftarrow M.\text{logits}(\mathbf{y}_{<t})$
  $p_t \leftarrow \text{softmax}(\mathbf{l}_t)$
  $\mathcal{W}_t \leftarrow \{\, j : |\, j - \text{ptr} \,| \leq k \,\}$
  $\lambda_t \leftarrow$ GetLambda(p_t, $\mathcal{W}_t$, type, λ)
  $\mathbf{l}_t[:] \mathrel{-}= \lambda_t$ *// penalize all tokens*
  $\mathbf{l}_t[\{x_j^{\text{mt}} : j \in \mathcal{W}_t\}] \mathrel{+}= \lambda_t$ *// restore MT-window tokens*
  $y_t \leftarrow \text{decode}(\mathbf{l}_t)$
  Update ptr based on y_t and $\mathcal{W}_t$
end for

GetLambda(p_t, $\mathcal{W}_t$, type, λ):
  if type = static: **return** λ
  if type = entropy: **return** $\lambda \cdot (H(p_t)/H_{\text{max}})^{\gamma}$
  if type = prob: **return** $\lambda \cdot (\max_{j \in \mathcal{W}_t} p_t(x_j^{\text{mt}}))^{\beta}$

I.2 Hyperparameter Search

I.3 The Sequence-Level Variant

The sequence-scaled variant ($\lambda_{\text{seq-NLL}}$, Table 2) replaces the per-token confidence signal with one computed over the whole generated prefix. At step

Parameter	Search Range	Optimal
λ (all)	0.1, 0.3, 0.5, 1.0, 2.0, 2.5, 3.0	2.5
γ (entropy)	0.5, 1.0, 2.0, 3.0	2.0
β (probability)	0.5, 1.0, 2.0	1.0
k (window)	1, 2, 3	2

Table 43: Hyperparameter search ranges and optimal values (tuned on D1 validation set, Gemma 3 1B).

t, let c_t be the running geometric-mean token probability of the tokens generated so far, i.e. the exponentiated length-normalized log-likelihood of the prefix. The base penalty is scaled by the sequence-level *uncertainty* $(1 - c_t)$:

$$\lambda_t = \lambda \cdot (1 - c_t)^{\gamma},\ c_t = \exp\big(\tfrac{1}{t-1} \textstyle\sum_{s<t} \log p_s(y_s)\big), \tag{5}$$

with $\gamma{=}1$, so the constraint relaxes when the model is confident about the sequence as a whole and tightens when it is not, mirroring the entropy variant at the sequence level. Because the fine-tuned model is overconfident even at the sequence level ($c_t \approx 0.94$ throughout, as most tokens are high-probability copies), the scaling factor $(1 - c_t)$ is small, and the base λ grid is widened accordingly to $\{0, 2, 5, 10, 20, 40, 60\}$ so that the effective penalty can reach the static optimum. On D1 (Gemma 3 1B, 3 seeds), its best-λ TER is 25.32±0.21: between the two token-level variants and, like them, short of the confidence-free static constraint (24.98±0.08). The Static > confidence-aware ordering therefore does not depend on reading confidence one token at a time.

I.4 Sensitivity to the Window Size and Penalty Parameters

The diagnostic's conclusions could in principle depend on the window size k and the penalty parameters γ and β. They do not. λ itself is swept rather than tuned, so no single value produces the curve shape, and the static constraint that carries the shape signal has no γ or β; those parameters only affect the two dynamic variants.

For k, we reran the static sweep on D1 at $k{=}1$ and $k{=}3$ (Gemma 3 1B, seed 42). The curve stays monotonically decreasing at every k, and its lowest point recovers the MT baseline (TER 24.98): best-λ TER is 24.98, 24.96, and 25.09 for $k{=}1$, 2, 3. The window size changes how sharply the penalty acts, not the direction of the trend.

For γ, the entropy variant's weakness is structural rather than a tuning choice: the fine-tuned model's normalized entropy is far below its maximum ($H/H_{\text{max}} \approx 0.02$ to 0.07, Table 3), so the entropy penalty barely activates for any $\gamma \geq 1$.

I.5 Computational Overhead

Constrained decoding adds negligible computational overhead (<5% wall-clock time on average, Gemma 3 1B, single A100 GPU). The constraint logic (window lookup and logit subtraction) is trivially cheap; runtime variation is dominated by output length rather than the constraint computation itself.

I.6 Why Static Wins Everywhere

On D1 (heterogeneous edits across a high-quality MT baseline), the pattern is consistent with an inconsistent training signal that mixes orthographic, stylistic, semantic, and structural corrections that vary unpredictably by sentence. The model nonetheless edits at high confidence whether or not the edit helps (Binary Collapse). Dynamic methods relax the constraint where the model is confident, but on D1 that confidence is systematically misplaced.

On D5 (genuine error correction), the explanation is subtler. The model *does* learn useful edits (the λ curve is U-shaped), yet the constraint ordering still holds: Static > PMT > Entropy. Here, static may win because useful corrections coexist with residual over-editing, and token-level confidence does not reliably distinguish the two. The edit-distance penalty does punish over-editing—every deviation from the MT window is penalized—but static applies it uniformly, whereas the dynamic variants waive it exactly where the model is confident, and confident tokens mix genuine corrections with confident over-edits in roughly equal measure. Static therefore suppresses residual over-edits regardless of a confidence signal that cannot separate the two. The D5 result indicates that miscalibration need not be specific to tasks that yield no net-positive edits. In our settings, it persists even when genuine corrections exist, alongside residual confidence-incoherent edits after fine-tuning on small heterogeneous corpora.

I.7 Practical Guidelines

The diagnostic is a single λ sweep on a held-out validation set with references. Run the static constraint over a range of λ (Table 43, extended upward until TER flattens), record validation TER at each value, and read off two things: the shape of the TER-vs-λ curve and the location of its minimum. The sweep costs one inference pass per λ

value and adds negligible overhead (Appendix I.5). Shape and minimum location place a fine-tuned checkpoint into one of four outcomes, each with a concrete next step (Table 44); the second signal, the constraint ordering, then fixes which variant to deploy (Section 6.3).

J Quality Estimation Details

LaBSE semantic similarity. Using LaBSE (Feng et al., 2022), we compute source–MT cosine similarity per test sample (n=716). Similarities are near-identical across edit categories (0.881/0.877/0.865 for none/minor/major), the distributions overlapping almost completely; AUROCs for none vs. minor (0.45) and none vs. major (0.42) fall below chance. On D1, cross-lingual semantic similarity thus cannot distinguish sentences that need post-editing from those that do not.

COMET-QE. We evaluate a state-of-the-art reference-free QE model, `Unbabel/wmt22-cometkiwi-da` (Rei et al., 2022), on D1 (n=716). Per-category scores are nearly flat (0.868/0.870/0.843 for none/minor/major): AUROC 0.58 (near chance), Spearman $r=-0.18$ with TER, and gating precision plateaus at 71% even at 95% recall, barely above the edit-rate prior (71%). Despite training on millions of human quality judgments, COMET-QE cannot separate segments needing post-editing from those that do not, in this setting.

Fine-tuned QE formulations. We fine-tune `xlm-roberta-large` (Conneau et al., 2020) on D1 (n=5,728) to predict TER from (source, MT) pairs, in three formulations: (i) *regression* on log(TER+1) (MSE loss), AUROC 0.745; (ii) *3-class* (none/minor/major) with inverse-frequency weights, AUROC 0.703, macro-F1 0.43; (iii) *binary* (edit-needed vs. not) with 50/50 oversampling, AUROC 0.732. The binary classifier gates at only 44% precision (default) and 53% at the 90th-percentile threshold, unsuitable for reliable gating. Multi-dataset training (adding D2, D3) degraded D1 performance. Table 45 summarizes all six approaches.

K Gradient Signal Analysis

To probe the mechanistic basis of Binary Collapse, we measure gradient signals from the fine-tuned Gemma 3 1B checkpoint (seed 42) on classified D1 and D5 samples, grouped by MQM category. Per sample we run one forward/backward pass (autocast off, LoRA parameters in float32), flatten the LoRA gradient into a single vector (d=26,091,520 across 364 LoRA tensors), and accumulate weighted group mean gradients, then take pairwise cosine similarities between them. Per-sample gradients are clipped at norm 30.0; those exceeding 1e6 are skipped (0 on both datasets). The analysis uses one checkpoint and seed; replication across seeds and checkpoints is future work.

K.1 D1 vs. D5: Gradient Norms

Table 46 reports mean gradient norms and within-group loss variance by MQM category on D1 and D5. Three patterns hold on both. First, Style has the lowest mean gradient norm despite the largest sample count (D1: n=82, norm 1.21; D5: n=119, norm 1.53), consistent with within-group cancellation: its four subtypes (lexical preference, register shift, word order, loanword nativization) have meaningfully different per-sample norms (8.1–10.3 on D1) and pull in different directions. Second, Style gives the highest within-group loss variance on D1 (coefficient of variation CV=1.11 vs. 0.88 for Accuracy, 0.86 for Fluency) and is the only main group with CV>1.0: per-sample surprise within Style exceeds its mean, the signature of an inconsistent signal. Third, Fluency shows the largest D1→D5 norm increase (1.46→3.60, $\Delta = +2.14$): on genuinely bad MT (D5) morphological and punctuation fixes are the clearest signal, whereas on D1 they are diluted by surrounding heterogeneous edits.

K.2 D1 vs. D5: Cosine Similarity Matrices

Table 47 gives pairwise cosine similarities between group mean gradients on D1 and D5, and the contrast is sharp. On D1 the matrix is far from orthogonal (Style–Accuracy and maximum off-diagonal both 0.275), consistent with Style and Accuracy gradients aligning under a shared copy prior despite different semantic content. On D5 every off-diagonal collapses toward zero (maximum 0.051; Style–Accuracy 0.027, a 10× reduction) as each edit type pulls in its own direction. Fluency stays near-orthogonal on both (D1: ≤0.019; D5: ≤0.029): punctuation fixes are partially learned on D1 (Table 30), well-separated on D5. This mirrors the λ-curve contrast (Figure 1): monotonic on D1, U-shaped on D5.

Curve signature	What it means	Recommended action	Example
U-shaped: minimum at $\lambda>0$, dipping below MT	Net-positive edits survived training	Deploy the static constraint at that λ as a free inference-time gain (§6.3)	D5; English–Marathi; Gemma 4B on D1 (marginal)
Minimum at $\lambda=0$	The model is already net-positive without any constraint	Deploy the vanilla fine-tuned checkpoint directly	D2 (News)
Monotonic, reaching the MT baseline	No net-positive edits survived; copying is optimal	Copy MT for now; make the post-edits more consistent before training again	D1 (Literature)
Monotonic, plateauing above the MT baseline	Over-editing is too severe to undo at inference	Address data or MT quality before training again	NLLB-600M on D1

Table 44: The four outcomes read off a single λ sweep, with the next step each implies. The first two checkpoints are deployable as is; the last two call for upstream work on the training signal (§6.3).

QE Approach	Best AUROC	Note
LaBSE (pretrained)	0.45	Below chance
COMET-QE (pretrained)	0.58	Near chance
XLM-R regression	0.745	Best overall
XLM-R 3-class	0.703	$F1_{macro}=0.43$
XLM-R binary	0.732	Precision=0.44
XLM-R (train on test)	0.707	Cannot memorize

Table 45: QE approaches on D1. Fine-tuned models plateau at AUROC ∼0.73–0.75, insufficient for reliable gating.

	D1 (heterogeneous)		D5 (genuine errors)	
MQM Category	n	Mean grad norm	n	Mean grad norm
Accuracy	58	1.58	32	2.68
Fluency	41	1.46	19	3.60
Style	82	1.21	119	1.53
Retranslation	17	2.04	29	2.97

Table 46: Mean gradient norms (norm of group mean LoRA gradient vector) by MQM category on D1 (heterogeneous edits) and D5 (genuine error correction). Loss CV (within-group coefficient of variation of per-sample loss) on D1: Accuracy 0.88, Fluency 0.86, Style 1.11, Retranslation 1.64.

	D1 (heterogeneous)				D5 (genuine errors)			
	Acc.	Flu.	Sty.	Ret.	Acc.	Flu.	Sty.	Ret.
Accuracy	1.00	−0.01	0.28	0.14	1.00	−0.02	0.03	0.03
Fluency	−0.01	1.00	0.02	0.00	−0.02	1.00	0.03	0.00
Style	0.28	0.02	1.00	0.14	0.03	0.03	1.00	0.05
Retranslation	0.14	0.00	0.14	1.00	0.03	0.00	0.05	1.00

Table 47: Pairwise cosine similarity between group mean LoRA gradients on D1 (left) and D5 (right), Gemma 3 1B, seed 42. On D1 the matrix is far from orthogonal; on D5 all off-diagonal entries approach zero (see text).

L Data-Level Mitigation: Edit-Type Filtering

To test whether the failure is driven by edit-type composition rather than inconsistency, we retrained Gemma 3 1B on D1 under two matched-size conditions. **Filtered** removes samples GPT-5.2 classified as Style or Retranslation (Section 3.2), leaving n=3,818 (∼67% of D1's 5,728); **Random control** is a size-matched random subsample (seed 42), isolating filtering from set size. Both reuse the main D1 hyperparameters (Table 27, seed 42) and the original test set (n=716).

λ	Filtered	Random ctrl.
MT Base.	24.98	24.98
0.0	26.74	26.37
0.5	26.52	25.45
1.0	26.46	25.15
2.0	26.06	**24.97**
3.0	26.06	**24.97**
5.0	26.09	25.04
7.0	26.08	25.04
10.0	**25.00**	25.02

Table 48: TER vs. λ sweep on D1 (Gemma 3 1B, seed 42 only, static constraint; cross-seed replication pending) for the two matched-size retraining conditions of Appendix L: **Filtered** and **Random control**. $\lambda=0.0 \equiv$ vanilla fine-tuning; **MT Base.** is the shared MT baseline. Best (lowest) TER per condition in **bold**.

Two findings emerge. First, the TER-vs-λ curve (Table 48) stays monotonically decreasing on filtered D1 (versus the U-shaped D5 curve in Figure 1, center): no λ crosses the MT baseline, so no net-positive edits survive. Second, filtering gives no advantage over the random control, whose TER stays about a point lower; where the control marginally reaches the baseline (best ΔTER$= +0.01$ at λ=2), the filtered set never does. Removing stylistic edits therefore does not improve outcomes: the residual Accuracy and Fluency edits have high within-group variance (Fluency CV 0.86, Accuracy CV 0.88), matching the cross-domain pattern (Table 33) and the gradient analysis (Appendix K). Filtering thus joins Appendix G's failed interventions: none resets the diagnostic on D1.