WHICH REVISIONS ARE WORTH APPLYING? RISK-AWARE REPAIR DECISIONS FOR TRANSLATION SELF-CORRECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models can refine machine translations, but a proposed revision may also damage content that is already correct. This matters more as initial translations improve, *i.e.,* stronger translations contain fewer beneficial revisions and offer smaller attainable gains, motivating self-correction as a selective decision over whether and how to revise. We formulate translation repair as maximizing expected bounded improvement subject to a population degradation budget; the Lagrangian yields a net-value rule that compares each repair with retaining the current translation. We instantiate this rule in Reliable Correction via Action Selection (RCAS), which generates candidates along multiple repair directions, estimates their benefit and degradation relative to the current translation without test-time references, and applies the highest positive-net-value repair or retains the current translation. Across five WMT test sets, RCAS achieves the highest mean COMET among the evaluated repair pipelines on four datasets. Frozen-pool controls show that overriding its retain decisions substantially increases population regression, and a blinded audit of 250 executed edits finds that 23.2% repair a major error while only 2.0% introduce one. These results show that reliable repair depends as much on declining unjustified revisions as on generating good ones.

1 INTRODUCTION

Large language model (LLM) self-correction often follows a generate–critique–revise loop (Madaan et al., 2023), yet without external feedback it can turn correct answers into incorrect ones, as knowing *when to stop* requires ground truth (Huang et al., 2024). Translation post-editing instantiates this dilemma. Automatic post-editing (APE) treats the current translation as an input-specific baseline (Chatterjee et al., 2020), while LLM refiners propose replacements through critique, error guidance, or iteration (Raunak et al., 2023; Ki & Carpuat, 2024; Feng et al., 2025). Since a revision may either fix a mistranslation or damage a correct entity, relation, or qualification (Tan et al., 2026; Padmanabhan, 2025; Huang et al., 2025), reliable repair requires an acceptance decision rather than merely another generation step.

Figure 1 illustrates the decision landscape of translation repair across three levels. At the general level (Figure 1a), Huang et al. (2024) show that without external feedback, intrinsic self-correction can turn correct reasoning answers into incorrect ones, because knowing *when to stop* requires ground truth. At the repair level (Figure 1b), the same input admits revisions of very different scope and outcome, *i.e.,* a local fix may restore a missing entity, a structural rewrite may improve fluency, yet a retranslation can damage content that was already correct. At the population level (Figure 1c), stronger initial translations offer fewer beneficial revisions and smaller best available gains, while harmful revisions persist.

These observations pose three challenges for reliable repair: (*i*) many candidate pools contain no useful revision; (*ii*) multiple candidates require choosing *which* action to take as well as *whether* to intervene; and (*iii*) references supervise development but are unavailable at deployment. Because candidate generation bounds available improvement while the decision policy determines realized value and degradation, edit coverage alone cannot diagnose decision quality.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

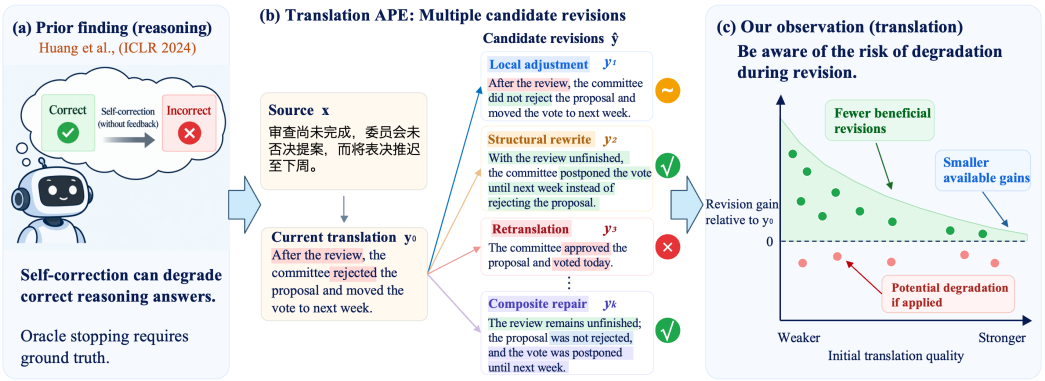


Figure 1: **A proposed revision requires an acceptance decision.** (a) Intrinsic self-correction can turn correct reasoning answers into incorrect ones without external feedback, oracle stopping requires ground truth (Huang et al., 2024). (b) An illustrative Chinese-to-English translation APE example contrasts a partial local fix, a structural rewrite, a harmful retranslation, and a composite repair. (c) In the studied candidate pools, stronger initial translations offer fewer beneficial revisions and smaller best available gains, while harmful revisions remain possible.

We formalize repair as a relative, multi-action decision over retaining the current translation or selecting a candidate. Opportunity bounds separate candidate supply from decision regret, while a risk–coverage frontier separates wrong-candidate selection from editing the wrong input. To balance value and risk, we maximize bounded expected benefit subject to a population expected-degradation budget. The Lagrangian assigns each repair a net value of expected benefit minus priced degradation, and since retaining the current translation has value zero, the highest positive-value action determines both whether and how to intervene.

Guided by this rule, *Reliable Correction via Action Selection* (RCAS) generates candidates along complementary repair directions, estimates each candidate's benefit and degradation relative to the current translation, and applies the highest positive-net-value repair or retains the current translation. References supervise development, but generation and selection require no references at test time. Across five WMT tests, RCAS improves initial translations and its retain decisions substantially reduce population regression, while a blinded audit confirms that executed repairs remove major errors far more often than they introduce them.

Our contributions are:

- **Empirical and theoretical analysis of repair opportunity and risk.** We show that stronger initial translations offer fewer and smaller beneficial revisions, while replacement still risks degradation. We bound a policy's gain by its candidate pool's opportunity, and decompose excess risk at a fixed edit rate into editing the wrong inputs and choosing the wrong repair.
- **A net-value decision rule from constrained optimization.** We formulate repair as maximizing bounded expected improvement subject to a population degradation budget. Its Lagrangian yields a per-candidate net-value rule, which we instantiate in RCAS to decide whether and how to repair without test-time references.
- **Evidence that selective repair is both safer and effective.** Across five WMT tests, frozen-pool controls, forced-edit comparisons, and blinded human adjudication demonstrate that RCAS recovers available repair value while limiting regression, and that its executed edits remove major errors far more often than they introduce them.

2 Repair Decisions under Scarce Opportunity

Problem setup. Given source x and current translation y^0, reference-free generation, validation, and deduplication produce a finite pool $\{y^a\}_{a=1}^{K}$. The state S contains these texts along with their

reference-free quality estimation (QE) (Blain et al., 2023) scores and edit evidence, and we write $K(S) = K$. The action set is $\mathcal{A}(S) = \{0, \dots, K(S)\}$, where action 0 retains y^0 and $a > 0$ outputs y^a. A policy $w(\cdot \mid s) \in \Delta(\mathcal{A}(s))$ uses only S.

For a retrospective utility Q, define the relative gain:

$$G_a = Q(x, y^a) - Q(x, y^0), \qquad G_0 = 0. \tag{1}$$

Reference-based Q reveals gains only during development and evaluation. The following subsections bound available improvement, characterize regression at fixed edit coverage, and derive the net-value rule implemented by RCAS, before examining its sensitivity to estimation error (proofs in Appendix B).

2.1 Relative Actions and Repair Opportunity

Let $\Gamma = \max_{a \in \mathcal{A}(S)} G_a \geq 0$ be the best available gain. For an initial-quality stratum $Z = z$ used only in retrospective analysis, define

$$\theta(z) = \Pr(\Gamma > 0 \mid Z = z), \qquad O(z) = \mathbb{E}[\Gamma \mid Z = z],$$
$$V_w(z) = \mathbb{E}\left[\sum_a w_a(S) G_a \mid Z = z\right], \qquad R_w(z) = \mathbb{E}\left[\Gamma - \sum_a w_a(S) G_a \mid Z = z\right]. \tag{2}$$

Here θ is opportunity frequency, O attainable gain, and R_w the value missed by the policy. We write $[t]_+ = \max\{t, 0\}$.

Proposition 1 (Opportunity and intervention limits)**.** *If positive gains are bounded above by a constant $M_+ > 0$, then for every policy and quality stratum,*

$$V_w(z) = O(z) - R_w(z) \leq O(z) \leq M_+ \theta(z). \tag{3}$$

Let $e_w(z) = \mathbb{E}[1 - w_0(S) \mid Z = z]$ be edit coverage. The mass of edits on inputs whose candidate pool contains no improving action satisfies

$$\mathbb{E}[(1 - w_0(S)) \mathbf{1}\{\Gamma = 0\} \mid Z = z] \geq [e_w(z) - \theta(z)]_+. \tag{4}$$

The identity separates available opportunity from policy regret. Coverage above $\theta(z)$ forces edits on some inputs with no improving candidate, but does not imply harm. See Appendix D.2 for a frozen-pool audit, Figure 3 for how opportunity changes with initial quality, and Appendix B.2 for a headroom bound.

2.2 Why Coverage and Candidate Choice Both Matter

The opportunity bound does not quantify harm. For tolerated decrease $m \geq 0$, define

$$h_a(s) = \Pr(G_a < -m \mid S = s), \qquad \underline{h}(s) = \min_{a > 0} h_a(s), \tag{5}$$

where $\underline{h}(s) = 1$ if no repair is available. For a policy w, write

$$C(w) = \mathbb{E}[1 - w_0(S)], \qquad H(w) = \mathbb{E}\left[\sum_{a>0} w_a(S) h_a(S)\right], \qquad \rho(w) = \frac{H(w)}{C(w)}. \tag{6}$$

Here C and H are population edit and regression mass, and ρ is regression risk among edits. Unlike stratum-level $e_w(z)$, $C(w) = \mathbb{E}[e_w(Z)]$ averages over the population.

Theorem 2 (Multi-action risk–coverage frontier)**.** *For attainable coverage $0 < c \leq \Pr(K(S) > 0)$, let $F_{\underline{h}}^{-1}$ denote the lower quantile function of $\underline{h}(S)$. The smallest achievable population regression mass is*

$$H^\star(c) = \inf_{w : C(w) = c} H(w) = \int_0^c F_{\underline{h}}^{-1}(u)\, du, \qquad \rho^\star(c) = \frac{H^\star(c)}{c}. \tag{7}$$

For any policy with $C(w) = c$, its excess risk decomposes exactly as

$$H(w) - H^\star(c) = \underbrace{\mathbb{E}\left[\sum_{a>0} w_a(S)\{h_a(S) - \underline{h}(S)\}\right]}_{\Delta_{\mathrm{act}}(w)} + \underbrace{\mathbb{E}[(1 - w_0(S))\underline{h}(S)] - H^\star(c)}_{\Delta_{\mathrm{rank}}(w;c)} \geq 0. \tag{8}$$

Δ_{act} measures risk from choosing a worse repair, while Δ_{rank} measures risk from editing the wrong inputs. The oracle conditional-risk floor $\rho^\star(c)$ cannot decrease with coverage. The frontier's decomposition motivates RCAS's two decisions: which repair to select for a given input, and which inputs to edit. The proof and matched-coverage results are in Appendices B.3 and D.6.

2.3 From a Population Risk Budget to an Action Rule

Minimizing regression alone may favor no edits, while maximizing improvement alone may accept damaging ones. We value benefit and degradation separately:

$$\beta(g) = \min\{[g]_+, B_{\max}\}, \qquad \ell_m(g) = \min\{[-g-m]_+, L_{\max}\}. \tag{9}$$

The caps bound extremes and m sets the tolerated decrease. Let $b_a(s) = \mathbb{E}[\beta(G_a) \mid S = s]$ and $\ell_a(s) = \mathbb{E}[\ell_m(G_a) \mid S = s]$. For a policy w, define

$$\mathcal{B}(w) = \mathbb{E}\left[\sum_a w_a(S) b_a(S)\right], \qquad \mathcal{L}(w) = \mathbb{E}\left[\sum_a w_a(S)\ell_a(S)\right]. \tag{10}$$

We maximize expected bounded benefit under a population degradation budget:

$$J(\kappa) = \sup_{w \in \mathcal{W}} \mathcal{B}(w) \quad \text{subject to} \quad \mathcal{L}(w) \le \kappa. \tag{11}$$

Here $\mathcal{W}$ contains policies over available actions, and $\kappa \ge 0$ is a shared expected-degradation budget, not a guarantee for individual edits.

For a degradation price $\lambda \ge 0$, the Lagrangian separates over inputs:

$$\mathcal{B}(w) - \lambda\{\mathcal{L}(w) - \kappa\} = \lambda\kappa + \mathbb{E}\left[\sum_a w_a(S) D_a^\lambda(S)\right], \qquad D_a^\lambda(s) = b_a(s) - \lambda\ell_a(s). \tag{12}$$

For fixed λ, the optimal action maximizes $D_a^\lambda(s)$:

$$\pi_\lambda(s) \in \arg\max_{a \in \mathcal{A}(s)} D_a^\lambda(s), \qquad g(\lambda) = \lambda\kappa + \mathbb{E}\left[\max_{a \in \mathcal{A}(S)} D_a^\lambda(S)\right]. \tag{13}$$

Since $D_0^\lambda(s) = 0$ and ties favor retaining y^0, the rule selects the highest-value repair only when its value is positive. Under finite-population duality or suitable population conditions,

$$J(\kappa) = \inf_{\lambda \ge 0} g(\lambda). \tag{14}$$

RCAS estimates b_a and ℓ_a from development data and applies this rule without test references (Section 3). Setting $\lambda = 1$ prices one unit of bounded degradation equally against one unit of bounded benefit; its fixed value selects a trade-off rather than certifying a test-population budget. Duality conditions, acceptance thresholds, and the signed-gain special case are in Appendix B.4.

2.4 Decision Stability and Its Empirical Scope

The rule also bounds the effect of consequence-estimation error. Let $\widehat{D}_a^\lambda = \widehat{b}_a - \lambda\widehat{\ell}_a$, $\widehat{a} \in \arg\max_a \widehat{D}_a^\lambda$, $a^\star \in \arg\max_a D_a^\lambda$, and $\varepsilon_\lambda(s) = \max_{a \in \mathcal{A}(s)} |\widehat{D}_a^\lambda(s) - D_a^\lambda(s)|$. For every input,

$$0 \le D_{a^\star}^\lambda(s) - D_{\widehat{a}}^\lambda(s) \le 2\varepsilon_\lambda(s). \tag{15}$$

On the same action set, the selected action loses at most $2\varepsilon_\lambda(s)$ in net value. A best-action margin above this bound preserves its selection, while near ties can change it. See Appendix B.5.

This bound motivates the relative predictor and consequence calibration in Section 3. Score alignment and transfer analyses appear in Appendices D.7 and D.4.

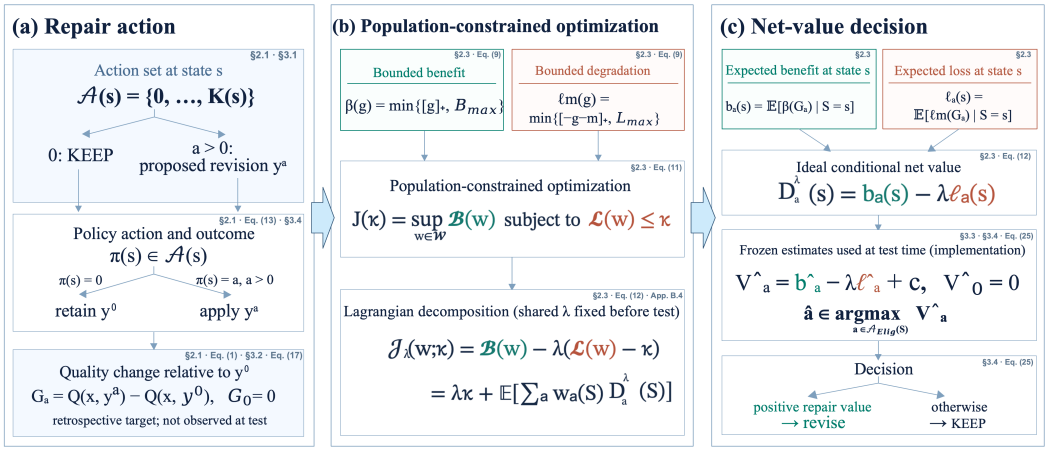


Figure 2: **RCAS decides whether and how to repair by weighing expected benefit against degradation.** (a) The KEEP action retains the current translation; other actions apply proposed revisions. Quality change is measured retrospectively relative to y^0. (b) Bounded benefit and degradation define a population optimization under a shared expected-degradation budget; its Lagrangian yields candidate net values. (c) RCAS estimates benefit and loss without test-time references and selects the highest positive-value repair, otherwise retaining the current translation. Section and equation references in the panels locate each step in the paper.

3 RCAS: Deciding Whether and How to Repair

Reliable Correction via Action Selection (RCAS) instantiates the net-value rule of Eq. (13) to decide whether and how to repair. It constructs an action set that includes the current translation (Section 3.1), predicts how each revision compares with it (Section 3.2), and uses held-out data to estimate the bounded benefit and degradation of each action (Section 3.3). These estimates determine whether to retain the current translation or apply a revision (Section 3.4).

RCAS uses reference-derived outcomes to train and calibrate these estimates during development. At test time, all fitted components are frozen, and selection relies only on the source, current translation, proposed revisions, and reference-free observations. Figure 2 traces the path from relative actions (Sections 2.1 and 3.1) through the population net-value rule (Section 2.3) to reference-free action selection (Section 3.4).

3.1 Constructing Candidate Actions

For source x and current translation y^0, RCAS constructs revisions $\{y^a\}_{a>0}$ and includes retaining y^0 as action $a = 0$. Each action selects a complete translation, not an instruction for a later edit. This input-dependent set is $\mathcal{A}(S)$ in Section 2.

Two evidence audits identify possible local errors and guide two local repairs but produce no translation candidates. The two local repairs, two generic rewrites of y^0, two retranslations from x, and one fusion call using earlier proposals yield up to seven proposals before validation. RCAS scores y^0 and the whole-sentence proposals with reference-free QE. If QE improvement or valid editing evidence supports composition, RCAS combines compatible clause blocks into at most three further proposals without another LLM call. Structural checks test well-formedness, not translation correctness.

After validation and deduplication, y^0 and the proposals form the prediction pool, and composed proposals also receive reference-free scores. The predictor uses non-rejected proposals even if the frozen action mask excludes them from final selection. Standalone local repairs inform fusion, composition, and prediction; their eligibility for final selection is determined by the action mask selected on the development policy subset and frozen before test evaluation. Workflow and eligibility details are in Appendix C.3.

3.2 PREDICTING RELATIVE OUTCOMES

Reference-derived supervision. During development, the predictor uses the source, current translation, proposed revisions, QE scores, and edit evidence (Section 3.1), while reference r_i supplies the outcome target. We define our reference-based composite utility, used as Q in Eq. (1), by combining COMET-22 (Rei et al., 2022) and chrF (Popović, 2015):

$$Q_\omega(x, y, r) = q_{\mathrm{COMET22}}(x, y, r) + \omega \, \frac{\mathrm{chrF}(y, r)}{100}, \tag{16}$$

where ω is determined from training-split gain scales and then frozen. Candidate a is supervised by its utility change relative to the current translation:

$$g_{ia} = Q_\omega(x_i, y_i^a, r_i) - Q_\omega(x_i, y_i^0, r_i), \qquad g_{i0} = 0. \tag{17}$$

This target measures the effect of replacing y_i^0 with y_i^a, rather than the candidate's absolute quality. A tolerance m gives the outcome label

$$c_{ia} = \begin{cases} \mathrm{imp}, & g_{ia} > m, \\ \mathrm{tie}, & |g_{ia}| \le m, \\ \mathrm{reg}, & g_{ia} < -m. \end{cases} \tag{18}$$

These are metric-defined labels, not human annotations of translation errors. The utility weight, tolerance, and consequence caps are fixed using development data as described in Appendix C.1.

Candidate-relative and set-aware representation. An edit script records insertions, deletions, and replacements between y^a and y^0, while the error certificates from Section 3.1 supply reference-free features. A shared text encoder reads the source, current translation, candidate, and edit script, producing e_a. For e_0, the candidate is y^0 and the script specifies no edit. The feature vector f_a includes QE, edit, and evidence indicators. The predictor compares each candidate with y^0 and uses context from the other proposals:

$$\begin{aligned} v_a &= [e_a;\, e_a - e_0;\, |e_a - e_0|;\, e_a \odot e_0;\, f_a], \\ p_a &= \mathrm{MLP}_r(v_a), \\ u_a &= \mathrm{MLP}_s([p_a;\, \bar{p};\, p_{\max};\, p_a - \bar{p}]), \\ (\widetilde{G}_a, \xi_a) &= (w_g^\top u_a,\, W_c u_a + \delta_a^{\mathrm{edit}} \mathbf{e}_{\mathrm{reg}}). \end{aligned} \tag{19}$$

Here $\bar{p}$ and $p_{\max}$ are the mean and elementwise maximum of p_a over non-rejected proposals for the input. The edit-span term δ_a^{edit} uses changed spans, and $\mathbf{e}_{\mathrm{reg}}$ selects the regression logit it adjusts. The outputs $\widetilde{G}_a$ and ξ_a predict signed gain and improvement/tie/regression logits. Section 3.3 corrects the gain and calibrates regression probability before estimating benefit and degradation. Encoder, edit-span, and feature details are in Appendix C.1.

Training objective. We train outcome classification, gain regression, within-pool ranking against other candidates and action 0, and harmful-tail detection:

$$\mathcal{L}_{\mathrm{train}} = \mathcal{L}_{\mathrm{cls}} + \eta_g \mathcal{L}_{\mathrm{gain}} + \eta_r \mathcal{L}_{\mathrm{rank}} + \eta_t \mathcal{L}_{\mathrm{tail}}. \tag{20}$$

For n training records, classification and gain regression use

$$\begin{aligned} \mathcal{L}_{\mathrm{cls}} &= \frac{1}{n} \sum_i \sum_{a>0} w_{ia}^c \, \mathrm{CE}_{\mathrm{bal}}(\xi_{ia}, c_{ia}), \\ \mathcal{L}_{\mathrm{gain}} &= \frac{1}{n} \sum_i \sum_{a>0} w_{ia}^g \, \mathrm{SL1}\left(\frac{\widetilde{G}_{ia}}{s_G}, \frac{g_{ia}}{s_G} \right). \end{aligned} \tag{21}$$

Here $\mathrm{CE}_{\mathrm{bal}}$ is class-balanced cross-entropy, SL1 is Smooth L1 loss, and s_G scales gain by its training-candidate standard deviation. The exposure weights w_{ia}^c and w_{ia}^g are normalized within each record so that records with more candidates do not dominate.

Let $\mathcal{P}_i$ contain within-record action pairs whose target gains differ by more than m, including comparisons with action 0. With $\sigma_{iab} = \mathrm{sign}(g_{ia} - g_{ib})$ and $\widetilde{G}_{i0} = 0$, the ranking loss is

$$\mathcal{L}_{\mathrm{rank}} = \frac{1}{n} \sum_i \frac{1}{|\mathcal{P}_i|} \sum_{(a,b) \in \mathcal{P}_i} \mathrm{softplus}\left(-\sigma_{iab} \frac{\widetilde{G}_{ia} - \widetilde{G}_{ib}}{s_G} \right). \tag{22}$$

This auxiliary loss ranks signed gains, while final selection uses calibrated benefit and degradation. For harmful candidates $\mathcal{H}_i = \{a > 0 : g_{ia} < -m\}$, the auxiliary tail term is

$$\mathcal{L}_{\text{tail}} = \frac{1}{n} \sum_i \frac{1}{|\mathcal{H}_i|} \sum_{a \in \mathcal{H}_i} [\tau - \xi_{ia,\text{reg}}]_+. \tag{23}$$

Here τ is a raw-logit margin. Empty pair or tail sets contribute zero, and the tail term does not calibrate degradation probability or enforce the population degradation constraint. Weighting and optimization details are in Appendix C.1.

3.3 Estimating Expected Benefit and Degradation

The predictor outputs signed gain $\widetilde{G}_a$ and outcome logits ξ_a, while Eq. (12) requires $b_a(s) = \mathbb{E}[\beta(G_a) \mid S = s]$ and $\ell_a(s) = \mathbb{E}[\ell_m(G_a) \mid S = s]$. In general, $\mathbb{E}[\beta(G_a) \mid S] \neq \beta(\mathbb{E}[G_a \mid S])$, and a similar nonlinearity applies to degradation. RCAS estimates both consequences from held-out outcomes rather than applying the bounds directly to $\widetilde{G}_a$.

On a calibration subset disjoint from predictor training, RCAS corrects $\widetilde{G}_a$ and calibrates $\mathrm{softmax}(\xi_a)_{\text{reg}}$, the probability of regression beyond m. Denote these outputs by $\widehat{g}_a$ and $\widehat{p}_a$. The same records provide bounded consequence targets:

$$B_{ia} = \beta(g_{ia}), \qquad L_{ia} = \ell_m(g_{ia}),$$

using Eq. (9). Both quantities transform the same observed gain, meaning that a small positive gain can still contribute to benefit even when its class label is tie.

The benefit table groups candidates by action category and corrected gain $\widehat{g}_a$, whereas the degradation table uses action category and calibrated regression probability $\widehat{p}_a$. Fusion and composition share the `FUSION` category, while the predictor retains their provenance. Each degradation group averages bounded loss magnitudes L_{ia}, not binary labels, so its estimate is not multiplied by $\widehat{p}_a$ again. To reduce variance, both estimates shrink toward their pooled group mean:

$$\widehat{T}_{d,k} = \frac{n_{d,k}\overline{T}_{d,k} + \alpha \overline{T}_k}{n_{d,k} + \alpha}, \qquad T \in \{B, L\}. \tag{24}$$

Here d is the action category, k a gain or probability bin, $n_{d,k}$ its candidate count, $\overline{T}_{d,k}$ its mean consequence, and $\overline{T}_k$ the pooled mean for that bin. The parameter α controls the shrinkage strength for small groups. At test time, frozen category–gain and category–probability lookups yield $\widehat{b}_a$ and $\widehat{\ell}_a$ without references or observed gains. Calibration and split details are in Appendix C.2.

3.4 Selecting Whether and How to Repair

Let $\mathcal{A}_{\text{elig}}(S) \subseteq \mathcal{A}(S)$ contain the retain action (action 0) and the repairs allowed by the frozen eligibility rules. RCAS applies the net-value rule with a development-selected offset c to this set:

$$\widehat{V}_a = \widehat{b}_a - \lambda \widehat{\ell}_a + c \quad (a > 0), \qquad \widehat{V}_0 = 0, \qquad \widehat{a} \in \arg\max_{a \in \mathcal{A}_{\text{elig}}(S)} \widehat{V}_a. \tag{25}$$

RCAS selects the highest-value repair only when its value is positive, meaning it exceeds the zero value of retaining the current translation. The selected action outputs $y^{\widehat{a}}$, and no further generation is needed.

For $c = 0$, the plug-in bound in Eq. (15) applies within $\mathcal{A}_{\text{elig}}(S)$. We fit calibration maps and choose the operating configuration on separate development subsets with no shared documents, then freeze both before test evaluation.

4 Experiments

Data and comparisons. Five WMT test sets (Kocmi et al., 2022; 2023; 2024) spanning three years cover both translation directions and multiple initial MT systems, with document-separated development and test partitions. Three sets support the initial analysis, while two later sets evaluate

frozen RCAS policies. We evaluate complete repair pipelines against adaptations of Ki–Carpuat and TEaR (Ki & Carpuat, 2024; Feng et al., 2025), and subsequently compare decision rules on the same RCAS candidate pools. These frozen-pool controls isolate the value of deciding *whether* to edit from *which* revision to apply. References supervise training, calibration, and operating choices during development and evaluate test outcomes, but remain unavailable to test-time generation and selection. Dataset, model, run, and comparator details are in Appendix E and Appendix E.1.

Evaluation metrics. Official human system scores index initial quality independently of candidate gains. COMET-22 serves as the primary reporting metric (Rei et al., 2022), while the selector is trained on the composite COMET–chrF utility Q_ω defined in Eq. (16). Both metrics are reported in points ($\times 100$), with Q_ω compared only within datasets. Automatic regression is defined as a raw COMET-22 decrease exceeding 0.01, and is reported both among edits and over the test population. Finally, a blinded human audit assesses major errors, with metric and uncertainty details provided in Appendix E. The experiments follow the theoretical chain: Q1 measures the repair opportunity supplied by the candidate pool (O, θ); Q2 assesses the value and regression realized by the decision policy; and Q3 examines major errors in the edits it executes.

Table 1: **Frozen WMT comparisons.** (a) Pipeline COMET-22 $\times 100$; bold marks the highest run mean. (b) Same-pool decisions: RCAS run means versus fixed comparators. Protocol: Appendix E.

(a) Repair-pipeline comparison

Method	WMT23 en–zh	WMT22 en–de	WMT24† en–zh	WMT24* en–de	WMT23* zh–en
RCAS (ours)	**86.342**	**86.126**	**85.687**	**81.156**	83.090
KiC-G (Ki & Carpuat, 2024)	85.111	85.712	84.975	80.725	82.763
KiC-X (Ki & Carpuat, 2024)	85.447	85.750	84.942	80.007	82.809
TEaR (Feng et al., 2025)	86.133	85.657	84.607	80.027	**83.141**

(b) Same-pool decision comparison

	Q_ω			COMET-22 $\times 100$			RCAS (%)	
Dataset	Current	QE-best	RCAS	Current	QE-best	RCAS	Edit	Reg./edit
WMT23 en–zh	104.840	109.149	110.857	81.946	86.766	86.342	72.7	18.9
WMT22 en–de	125.277	124.511	125.460	85.595	86.565	86.126	55.7	22.0
WMT24 en–zh	102.376	104.686	104.994	83.765	85.888	85.687	91.0	15.3
WMT24 en–de*	128.562	131.113	131.013	79.263	81.403	81.156	59.9	18.1
WMT23 zh–en*	116.157	119.584	119.720	81.067	83.298	83.090	85.6	17.4

(b) reports absolute scores; QE-best can retain the current translation, and dataset-specific Q_ω can exceed 100. Reg./edit: share of edits with raw COMET-22 drop > 0.01. †Score provenance: Appendix E.2; *preregistered RCAS tests. Pipeline adapters: Appendix E.1.

4.1 Q1: How Much Repair Opportunity Is Available?

Across initial MT systems, best-candidate gain falls by 2.53 COMET points per human-quality standard deviation (95% document-bootstrap interval $[-2.92, -2.13]$). Improving candidates become less frequent, while RCAS coverage has no detected slope: the opportunity itself contracts. Figure 3, subgroup checks, and negative best-gain slopes on the two later tests are in Appendix D.

4.2 Q2: What Value Does the Decision Achieve, and at What Risk?

Among complete pipelines (Table 1a), RCAS achieves the highest mean COMET on four tests, while TEaR leads on WMT23 zh–en. On fixed RCAS pools (Table 1b), RCAS improves both utilities over the current translation on every test. Although QE-best attains higher COMET, RCAS achieves higher mean composite utility on four tests, meaning neither rule strictly dominates the other across both metrics.

Missed value does not detectably increase as opportunity falls, with opportunity–regret analyses detailed in Appendix D.

Decision components. On frozen pools, Table 2 changes one component at a time. Retaining declined inputs improves both quality metrics and reduces population regression versus forced editing. At fixed acceptance, RCAS ranking lowers regression but gives up COMET relative to QE ranking. Benefit–degradation scoring matches calibrated signed gain on composite and regression metrics while reducing regression by 5.5 percentage points versus benefit-only scoring on zh–en, confirming the value of the degradation term (Appendix D.3). The top feasible repair's predicted net value is positively associated with realized bounded net value ($r = 0.60$ for en–zh and $r = 0.40$ for zh–en), with automatic regression concentrated in the lowest-score bins (Appendix D.7).

Table 2: **Decision ablations on frozen WMT23 pools.** Mean absolute outcomes across selector runs; each variant changes one component.

	WMT23 en–zh				WMT23 zh–en			
Variant	Q_ω	COMET	Edit	Reg.	Q_ω	COMET	Edit	Reg.
RCAS	110.857	86.342	72.7	13.8	119.720	83.090	85.6	14.9
Force edit	109.911	86.123	99.6	24.0	119.485	82.941	99.2	20.6
QE rank	110.047	86.725	72.7	16.4	119.813	83.284	85.6	17.1
Signed gain	110.875	86.419	71.9	13.2	119.902	83.230	86.3	14.5
Benefit only	110.763	86.466	72.5	14.1	119.204	82.908	86.1	20.4

COMET is COMET-22 $\times 100$; Q_ω is dataset-specific. Edit and Reg. are population percentages; Reg. means raw COMET-22 drop > 0.01. Force edit removes the retain action, QE rank changes candidate ranking at fixed acceptance, and the final two rows replace the net-value score. Control details: Appendix D.3.

Transfer to another candidate generator. With selector and calibration frozen on another generator's candidates, RCAS lowers population regression versus same-pool QE, while the composite difference remains within its bootstrap interval and QE retains higher COMET. Transfer and cost analyses are in Appendices D.4 and D.8.

4.3 Q3: Do Executed Repairs Reduce Major Errors?

Table 3: **Blinded audit of 250 RCAS edits (WMT23 en–zh).** Design-weighted rates represent 1,144 executed edits after adjudication, and brackets denote 95% document-bootstrap CIs. Net is the favorable minus the adverse rate in percentage points.

Endpoint	Favorable rate [95% CI]	Adverse rate [95% CI]	Net effect [95% CI] (pp)
Overall quality	Better: 52.5 $[45.3, 59.5]\%$	Worse: 6.4 $[3.6, 10.0]\%$	$+46.1$ $[37.2, 54.5]$
Major errors	Repaired: 23.2 $[18.2, 29.7]\%$	Introduced: 2.0 $[0.4, 4.1]\%$	$+21.2$ $[15.7, 28.1]$

The rate of major-error repair exceeds that of introduction by 21.2 points $[15.7, 28.1]$ (Table 3), reducing major-error presence by 16.5 percentage points $[12.2, 21.9]$ across the test population. Full audit details and sampling strata are provided in Appendices E and D.

5 Conclusion

This paper argues that reliable translation self-correction is fundamentally a decision problem, not a generation problem. As initial translations improve, the supply of beneficial revisions contracts, yet the risk of damaging correct content persists, making the decision layer the binding constraint on realized quality. Our net-value rule unifies *whether* to repair and *which* repair to apply into a single per-candidate comparison with retaining the current translation, operating without test-time references. The value of this selective discipline is evidenced not only by aggregate gains, but more importantly by the cost of overriding it: forcing declined edits substantially increases population regression, while executed edits remove major errors far more often than they introduce them. We believe this separation of candidate supply from decision quality extends beyond translation to any setting where generated revisions compete with an existing output.

AI USAGE STATEMENT

We used AI-based language tools solely for proofreading and polishing the manuscript, including improving grammar, clarity, and overall readability. All research ideas, methodology, experiments, analyses, and conclusions were developed and verified by the authors. The authors take full responsibility for the content of this paper.

ETHICS STATEMENT

Human Subjects and Ethics Approval: This study involves human evaluations. The study design and participant recruitment process were assessed in accordance with the policy of the authors' institution(s) and did not require formal IRB review under that policy, as the study presents minimal risk and collects anonymized data. Informed Consent and Privacy: All participants took part on a purely voluntary basis. Prior to the study, participants were fully informed of the research purpose, how their provided responses would be used, and their right to withdraw at any time without penalty. All collected data were strictly anonymized, ensuring no personally identifiable information (PII) was stored or released. Participants received non-monetary appreciation for their time.

REFERENCES

Frederic Blain, Chrysoula Zerva, Ricardo Rei, Nuno M. Guerreiro, Diptesh Kanojia, José G. C. de Souza, Beatriz Silva, Tânia Vaz, Yan Jingxuan, Fatemeh Azadi, Constantin Orasan, and André Martins. Findings of the WMT 2023 shared task on quality estimation. In Philipp Koehn, Barry Haddow, Tom Kocmi, and Christof Monz (eds.), *Proceedings of the Eighth Conference on Machine Translation*, pp. 629–653, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.wmt-1.52. URL https://aclanthology.org/2023.wmt-1.52/.

Rajen Chatterjee, Markus Freitag, Matteo Negri, and Marco Turchi. Findings of the WMT 2020 shared task on automatic post-editing. In Loïc Barrault, Ondřej Bojar, Fethi Bougares, Rajen Chatterjee, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Alexander Fraser, Yvette Graham, Paco Guzman, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, André Martins, Makoto Morishita, Christof Monz, Masaaki Nagata, Toshiaki Nakazawa, and Matteo Negri (eds.), *Proceedings of the Fifth Conference on Machine Translation*, pp. 646–659, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.wmt-1.75. URL https://aclanthology.org/2020.wmt-1.75/.

Pinzhen Chen, Zhicheng Guo, Barry Haddow, and Kenneth Heafield. Iterative translation refinement with large language models. In Carolina Scarton, Charlotte Prescott, Chris Bayliss, Chris Oakley, Joanna Wright, Stuart Wrigley, Xingyi Song, Edward Gow-Smith, Rachel Bawden, Víctor M Sánchez-Cartagena, Patrick Cadwell, Ekaterina Lapshinova-Koltunski, Vera Cabarrão, Konstantinos Chatzitheodorou, Mary Nurminen, Diptesh Kanojia, and Helena Moniz (eds.), *Proceedings of the 25th Annual Conference of the European Association for Machine Translation (Volume 1)*, pp. 181–190, Sheffield, UK, June 2024. European Association for Machine Translation (EAMT). URL https://aclanthology.org/2024.eamt-1.17/.

Chao Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions on information theory*, 16(1):41–46, 1970.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Sourabh Deoghare, Diptesh Kanojia, and Pushpak Bhattacharyya. Giving the old a fresh spin: Quality estimation-assisted constrained decoding for automatic post-editing. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *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)*, pp. 914–925, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-190-2. doi: 10.18653/v1/2025.naacl-short.77. URL https://aclanthology.org/2025.naacl-short.77/.

Bryan Eikema and Wilker Aziz. Sampling-based approximations to minimum bayes risk decoding for neural machine translation, 2022. URL https://arxiv.org/abs/2108.04718.

Zhaopeng Feng, Ruizhe Chen, Yan Zhang, Zijie Meng, and Zuozhu Liu. Ladder: A model-agnostic framework boosting LLM-based machine translation to the next level. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 15377–15393, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.860. URL https://aclanthology.org/2024.emnlp-main.860/.

Zhaopeng Feng, Yan Zhang, Hao Li, Bei Wu, Jiayu Liao, Wenqiang Liu, Jun Lang, Yang Feng, Jian Wu, and Zuozhu Liu. TEaR: Improving LLM-based machine translation with systematic self-refinement. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 3922–3938, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-195-7. doi: 10.18653/v1/2025.findings-naacl.218. URL https://aclanthology.org/2025.findings-naacl.218/.

Patrick Fernandes, António Farinhas, Ricardo Rei, José G. C. de Souza, Perez Ogayo, Graham Neubig, and Andre Martins. Quality-aware decoding for neural machine translation. In Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz (eds.), *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1396–1412, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.naacl-main.100. URL https://aclanthology.org/2022.naacl-main.100/.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks, 2017. arXiv:1705.08500.

Zhibin Gou, Zhihong Shao, Yeyun Gong, yelong shen, Yujiu Yang, Nan Duan, and Weizhu Chen. Critic: Large language models can self-correct with tool-interactive critiquing. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 57734–57811, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/fef126561bbf9d4467dbb8d27334b8fe-Paper-Conference.pdf.

Chris Hokamp and Qun Liu. Lexically constrained decoding for sequence generation using grid beam search. In Regina Barzilay and Min-Yen Kan (eds.), *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1535–1546, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1141. URL https://aclanthology.org/P17-1141/.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 32808–32824, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/8b4add8b0aa8749d80a34ca5d941c355-Paper-Conference.pdf.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55, January 2025. ISSN 1558-2868. doi: 10.1145/3703155. URL http://dx.doi.org/10.1145/3703155.

Wenxiang Jiao, Wenxuan Wang, Jen tse Huang, Xing Wang, Shuming Shi, and Zhaopeng Tu. Is chatgpt a good translator? yes with gpt-4 as the engine, 2023. URL https://arxiv.org/abs/2301.08745.

Amita Kamath, Robin Jia, and Percy Liang. Selective question answering under domain shift. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 5684–5696, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.503. URL https://aclanthology.org/2020.acl-main.503/.

Ryo Kamoi, Yusen Zhang, Nan Zhang, Jiawei Han, and Rui Zhang. When can LLMs actually correct their own mistakes? a critical survey of self-correction of LLMs. *Transactions of the Association for Computational Linguistics*, 12:1417–1440, 2024. doi: 10.1162/tacl_a_00713. URL https://aclanthology.org/2024.tacl-1.78/.

Dayeon Ki and Marine Carpuat. Guiding large language models to post-edit machine translation with error annotations. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 4253–4273, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-naacl.265. URL https://aclanthology.org/2024.findings-naacl.265/.

Tom Kocmi, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Thamme Gowda, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Rebecca Knowles, Philipp Koehn, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Michal

Novák, Martin Popel, and Maja Popović. Findings of the 2022 conference on machine translation (WMT22). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pp. 1–45, 2022. doi: 10.18653/v1/2022.wmt-1.1. URL https://aclanthology.org/2022.wmt-1.1/.

Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Philipp Koehn, Benjamin Marie, Christof Monz, Makoto Morishita, Kenton Murray, Masaaki Nagata, Toshiaki Nakazawa, Martin Popel, Maja Popović, Mariya Shmatova, and Jun Suzuki. Findings of the 2023 conference on machine translation (WMT23): LLMs are here but not quite there yet. In *Proceedings of the Eighth Conference on Machine Translation*, pp. 1–42, 2023. doi: 10.18653/v1/2023.wmt-1.1. URL https://aclanthology.org/2023.wmt-1.1/.

Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Marzena Karpinska, Philipp Koehn, Benjamin Marie, Christof Monz, Kenton Murray, Masaaki Nagata, Martin Popel, Maja Popović, Mariya Shmatova, Steinthór Steingrímsson, and Vilém Zouhar. Findings of the WMT24 general machine translation shared task: The LLM era is here but MT is not solved yet. In *Proceedings of the Ninth Conference on Machine Translation*, pp. 1–46, 2024. doi: 10.18653/v1/2024.wmt-1.1. URL https://aclanthology.org/2024.wmt-1.1/.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 46534–46594. Curran Associates, Inc., 2023. doi: 10.52202/075280-2019. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/91edff07232fb1b55a505a9e9f6c0ff3-Paper-Conference.pdf.

Maciej Modrzejewski, Yash Bhaskar, and Chinmay Pateria. Predict and fix: A unified model for translation quality estimation and post-editing. In *Proceedings of the 17th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pp. 80–89. Association for Machine Translation in the Americas, 2026. URL https://aclanthology.org/2026.amta-research.5/.

Govardhan Padmanabhan. Can QE-informed (re)translation lead to error correction? In *Proceedings of the Tenth Conference on Machine Translation (Volume 2: Shared Task Papers)*, pp. 984–993, 2025. doi: 10.18653/v1/2025.wmt-1.73.

Maja Popović. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pp. 392–395, 2015. doi: 10.18653/v1/W15-3049.

Matt Post. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pp. 186–191, 2018. doi: 10.18653/v1/W18-6319.

Vikas Raunak, Amr Sharaf, Yiren Wang, Hany Awadalla, and Arul Menezes. Leveraging GPT-4 for automatic translation post-editing. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 12009–12024, 2023. doi: 10.18653/v1/2023.findings-emnlp.804.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2685–2702, 2020. doi: 10.18653/v1/2020.emnlp-main.213.

Ricardo Rei, José G. C. de Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André F. T. Martins. COMET-22: Unbabel-IST 2022 submission for the metrics shared task. In Philipp Koehn, Loïc Barrault, Ondřej Bojar, Fethi Bougares, Rajen Chatterjee, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Alexander Fraser, Markus Freitag, Yvette Graham, Roman Grundkiewicz, Paco Guzman, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Tom Kocmi, André Martins, Makoto Morishita, Christof

Monz, Masaaki Nagata, Toshiaki Nakazawa, Matteo Negri, Aurélie Névéol, Mariana Neves, Martin Popel, Marco Turchi, and Marcos Zampieri (eds.), *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pp. 578–585, Abu Dhabi, United Arab Emirates (Hybrid), December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.wmt-1.52. URL https://aclanthology.org/2022.wmt-1.52/.

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning, 2023. URL https://arxiv.org/abs/2303.11366.

Shaomu Tan, Dawei Zhu, Ke Tran, Michael Denkowski, Sony Trenous, Leonardo F. R. Ribeiro, Bill Byrne, and Felix Hieber. What does LLM refinement actually improve? a systematic study on document-level literary translation. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5929–5957, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.268. URL https://aclanthology.org/2026.acl-long.268/.

Ji Hun Wang and Siyu Wu. Diagnose, then repair: A two-stage MQM-guided post-editing framework for domain-specific machine translation. In Yunyao Li, Georg Rehm, and Mei Tu (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, pp. 1683–1698, San Diego, California, USA, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-394-4. doi: 10.18653/v1/2026.acl-industry.115. URL https://aclanthology.org/2026.acl-industry.115/.

Ke Wang, Jiayi Wang, Niyu Ge, Yangbin Shi, Yu Zhao, and Kai Fan. Computer assisted translation with neural quality estimation and automatic post-editing. In Trevor Cohn, Yulan He, and Yang Liu (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 2175–2186, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.197. URL https://aclanthology.org/2020.findings-emnlp.197/.

Qingni Wang, Yue Fan, and Xin Wang. Safer: Risk-constrained sample-then-filter in large language models. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 53723–53747, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/582e9771ac8527cb6390e5e9444a0fee-Paper-Conference.pdf.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models, 2023. URL https://arxiv.org/abs/2203.11171.

A Related Work

LLM self-correction and revision reliability. LLM self-correction typically obtains feedback on an initial output and uses it to generate a revision. Feedback may come from the model itself (Madaan et al., 2023; Shinn et al., 2023) or from external tools (Gou et al., 2024), and its reliability is central to whether revision helps (Kamoi et al., 2024). Without reliable external feedback, intrinsic self-correction can reduce reasoning accuracy by changing correct answers; oracle analyses show that avoiding inappropriate revisions can recover value (Huang et al., 2024). The oracle, however, does not provide a deployable acceptance rule. Our work addresses the corresponding problem in translation: given a usable current translation and completed revision proposals, decide which replacement, if any, is justified without a test-time reference. This is a decision downstream of, and complementary to, feedback and revision generation.

Translation post-editing and refinement. Automatic post-editing starts from an existing machine translation and seeks to improve it rather than translate from scratch (Chatterjee et al., 2020). LLM-based systems perform direct post-editing (Raunak et al., 2023; Jiao et al., 2023), refine translations iteratively (Chen et al., 2024; Feng et al., 2025), or guide revision with model-agnostic procedures and explicit error diagnoses (Feng et al., 2024; Ki & Carpuat, 2024; Wang & Wu, 2026). Recent analyses show that their gains depend on the initial translation and that rewriting can trade adequacy for fluency or introduce unnecessary errors (Tan et al., 2026; Padmanabhan, 2025). Several systems already incorporate selective behavior: quality estimation can route inputs to post-editing (Wang et al., 2020), TEaR can retain translations judged error-free (Feng et al., 2025), and unified or constrained systems combine quality prediction with editing (Deoghare et al., 2025; Modrzejewski et al., 2026; Hokamp & Liu, 2017). These methods establish the value of selective refinement, with their decision mechanisms designed as part of a particular editing pipeline. We isolate the decision after a heterogeneous repair pool has been constructed. This permits all repair directions to be compared in one relative action space and, crucially, separates a pool that contains no useful revision from a policy that fails to select one. The distinction is necessary in the regime studied here: stronger initial translations provide fewer and smaller gains, while replacement can still damage content that was already correct.

Candidate selection and selective prediction. Candidate-selection methods seek the highest-utility output from a set of hypotheses. In machine translation, quality-aware and minimum Bayes risk decoding estimate candidate utility and use it to guide selection (Fernandes et al., 2022; Eikema & Aziz, 2022). Selective prediction studies a complementary problem: a system may abstain when confidence is insufficient, trading coverage against error risk (Chow, 1970; Geifman & El-Yaniv, 2017; Kamath et al., 2020). Recent work extends risk-aware selection to LLM sample-then-filter pipelines Cobbe et al. (2021); Wang et al. (2023), using calibrated population constraints to govern which generations are returned Wang et al. (2026). These lines provide general mechanisms for ranking, abstention, and risk control. Translation repair introduces a particular decision structure: the current translation is an input-specific baseline, and each input offers several possible replacement actions. KEEP-inclusive quality estimation is therefore a strong direct baseline, but ranks the current translation and its revisions by predicted absolute quality. Our formulation instead evaluates the expected improvement and degradation of each action relative to the current translation and optimizes population improvement under a degradation budget.

B Formal details and proofs

This appendix makes explicit the information structure, regularity conditions, and optimization arguments underlying Section 2. The oracle quantities below are used to characterize opportunity and evaluate decisions; they are not available to RCAS at deployment.

B.1 Probability space, actions, and policies

Let $(\Omega, \mathcal{F}, P)$ carry the deployment state S, an initial-quality stratum Z, and the retrospective action outcomes. The state S contains exactly the information available at test time: the source, current translation, proposed revisions, and reference-free observations. The reference used to evaluate an outcome is excluded from S.

For state s, the finite action set $\mathcal{A}(s) = \{0, 1, \ldots, K(s)\}$ contains KEEP ($a = 0$) and the available repair candidates ($a > 0$). A randomized policy is a measurable Markov kernel $w(\cdot \mid s) \in \Delta(\mathcal{A}(s))$. We write $w_a(s) = 0$ for $a \notin \mathcal{A}(s)$, so $\sum_{a \geq 0} w_a(s) = 1$. A deterministic policy π is the special case in which $w_a(s) = \mathbf{1}\{a = \pi(s)\}$. The retrospective gain G_a is defined as in Eq. (1), with $G_0 = 0$.

We assume throughout that the conditional action values used below are measurable and that the relevant expectations exist. Bounded benefit and degradation follow directly from Eq. (9); the finite empirical problems used in the experiments therefore require no additional regularity. These assumptions do not posit that opportunity decreases with initial quality. That relationship remains an empirical property of the studied candidate pools.

B.2 CANDIDATE SUPPLY, DECISION REGRET, AND INTERVENTION LIMITS

For a realized pool, let $\Gamma = \max_{a \in \mathcal{A}(S)} G_a \geq 0$. Conditional on $Z = z$, recall

$$\theta(z) = \Pr(\Gamma > 0 \mid Z = z), \qquad O(z) = \mathbb{E}[\Gamma \mid Z = z],$$
$$V_w(z) = \mathbb{E}\left[\sum_a w_a(S)\, G_a \;\middle|\; Z = z\right], \qquad R_w(z) = \mathbb{E}\left[\Gamma - \sum_a w_a(S)\, G_a \;\middle|\; Z = z\right]. \tag{26}$$

Lemma 3 (Supply–decision decomposition)**.** *For every policy w and stratum z,*

$$V_w(z) = O(z) - R_w(z), \qquad R_w(z) \geq 0. \tag{27}$$

If positive gains are bounded by M_+, then $V_w(z) \leq O(z) \leq M_+ \theta(z)$. The first inequality is an equality exactly when the policy places mass only on actions attaining Γ, almost surely conditional on $Z = z$.

Proof. Linearity of conditional expectation gives

$$\begin{aligned} O(z) - R_w(z) &= \mathbb{E}\left[\Gamma - \Big(\Gamma - \sum_a w_a(S) G_a\Big) \;\middle|\; Z = z\right] \\ &= \mathbb{E}\left[\sum_a w_a(S) G_a \;\middle|\; Z = z\right] = V_w(z). \end{aligned} \tag{28}$$

Because a convex combination of action gains cannot exceed their maximum, $\sum_a w_a(S) G_a \leq \Gamma$ pointwise. Hence $R_w(z) \geq 0$, with equality if and only if all selected mass is supported on maximizers of G_a. Finally,

$$O(z) = \mathbb{E}[\Gamma\, \mathbf{1}\{\Gamma > 0\} \mid Z = z] \leq M_+ \Pr(\Gamma > 0 \mid Z = z) = M_+\, \theta(z). \tag{29}$$

□

Lemma 4 (Excess-coverage certificate)**.** *Let $z_w(s) = 1 - w_0(s)$ be the probability of editing state s, and let $e_w(z) = \mathbb{E}[z_w(S) \mid Z = z]$. The unsupported edit mass*

$$U_w(z) = \mathbb{E}[\, z_w(S)\, \mathbf{1}\{\Gamma = 0\} \mid Z = z\,] \tag{30}$$

satisfies

$$U_w(z) \geq [\, e_w(z) - \theta(z)\,]_+. \tag{31}$$

Proof. With $\mathcal{P} = \{\Gamma > 0\}$ and $0 \leq z_w \leq 1$,

$$\begin{aligned} U_w(z) &= e_w(z) - \mathbb{E}[\, z_w(S)\, \mathbf{1}\{\mathcal{P}\} \mid Z = z\,] \\ &\geq e_w(z) - \Pr(\mathcal{P} \mid Z = z) = e_w(z) - \theta(z). \end{aligned} \tag{32}$$

Taking the positive part proves Eq. (31). At the level of set mass, the bound is tight whenever edit allocation can exhaust the positive-opportunity set before editing its complement. A deployable policy need not attain this oracle allocation because Γ is not observed at test time. Also, $\Gamma = 0$ means only that the realized pool offers no positive-gain action; an unsupported edit can be neutral or harmful. □

Candidate expansion and headroom. Suppose two pools for the same input satisfy $\mathcal{A}(s) \subseteq \mathcal{A}'(s)$. Their oracle opportunities obey

$$\Gamma'(s) \geq \Gamma(s), \qquad \theta'(z) \geq \theta(z), \qquad O'(z) \geq O(z). \tag{33}$$

Thus adding candidates cannot reduce oracle supply. It can nevertheless reduce realized value under an imperfect selector, because Eq. (27) contains the separate decision-regret term R_w. Conversely, if $Q \leq Q_{\max}$, then

$$\Gamma \leq Q_{\max} - Q(x, y^0), \qquad O(z) \leq \mathbb{E}[\, Q_{\max} - Q(x, y^0) \mid Z = z \,]. \tag{34}$$

This is a ceiling, not a theorem that a given generator's realized opportunity must contract; the latter is what the empirical opportunity analysis tests.

B.3 The multi-action risk–coverage frontier

Let $\mathcal{A}_+(s) = \mathcal{A}(s) \setminus \{0\}$ and $q(s) = \mathbf{1}\{\mathcal{A}_+(s) \neq \emptyset\}$. When $q(s) = 1$, define

$$h_a(s) = \Pr(G_a < -m \mid S = s), \qquad \underline{h}(s) = \min_{a \in \mathcal{A}_+(s)} h_a(s); \tag{35}$$

when $q(s) = 0$, set $\underline{h}(s) = 1$. The maximum attainable coverage is $c_{\max} = \mathbb{E}[q(S)]$. For any policy, $z_w(s) = \sum_{a>0} w_a(s)$ satisfies $0 \leq z_w(s) \leq q(s)$.

Lemma 5 (Within-instance reduction)**.** *For every state and policy,*

$$\sum_{a>0} w_a(s)\, h_a(s) = z_w(s)\, \underline{h}(s) + \sum_{a>0} w_a(s)\, \{h_a(s) - \underline{h}(s)\}. \tag{36}$$

The second term is nonnegative and vanishes if and only if edit mass is supported on minimum-risk repair actions.

Proof. Equation (36) follows by adding and subtracting $\sum_{a>0} w_a(s)\, \underline{h}(s) = z_w(s)\, \underline{h}(s)$. Each residual is nonnegative by the definition of $\underline{h}(s)$. □

Lemma 5 reduces multi-action risk minimization to allocating edit probability $z(s)$ across instances. For a target coverage $c \in (0, c_{\max}]$, consider

$$\inf_{0 \leq z \leq q} \big\{ \mathbb{E}[z(S)\, \underline{h}(S)] : \mathbb{E}[z(S)] = c \big\}. \tag{37}$$

Lemma 6 (Optimal instance allocation)**.** *Let $t_c = F_{\underline{h}}^{-1}(c)$. An optimizer of Eq.* (37) *edits all eligible instances with $\underline{h} < t_c$, edits none with $\underline{h} > t_c$, and assigns just enough mass on the boundary $\underline{h} = t_c$ to reach coverage c. Its objective value is*

$$\int_0^c F_{\underline{h}}^{-1}(u)\, du. \tag{38}$$

Proof. Let $z_c^\star$ have the threshold form above. Boundary randomization exists because randomized policies allow any mass in $[0, 1]$ on the level set. For any other feasible z, $\mathbb{E}[z - z_c^\star] = 0$, and pointwise

$$(z(s) - z_c^\star(s))(\underline{h}(s) - t_c) \geq 0 : \tag{39}$$

below t_c, $z_c^\star = 1$ and $z - z_c^\star \leq 0$; above t_c, $z_c^\star = 0$ and $z - z_c^\star \geq 0$; on the boundary the product is zero. Therefore

$$\mathbb{E}[z(S)\, \underline{h}(S)] - \mathbb{E}[z_c^\star(S)\, \underline{h}(S)] = \mathbb{E}[(z - z_c^\star)(\underline{h} - t_c)] \geq 0. \tag{40}$$

The generalized inverse representation of a lower-level-set allocation yields Eq. (38). If ineligible and eligible instances tie at $\underline{h} = 1$, boundary mass is assigned only where $q = 1$; the objective is unchanged. □

Proof of Theorem 2. Applying Lemma 5 and then Lemma 6 gives

$$H(w) \geq \mathbb{E}[z_w(S)\,\underline{h}(S)] \geq \int_0^c F_{\underline{h}}^{-1}(u)\,du = H^\star(c). \tag{41}$$

Both bounds are simultaneously attained by selecting a minimum-risk action on the instances chosen by the lower-level-set allocation. For an arbitrary policy with $C(w) = c$, adding and subtracting $\mathbb{E}[z_w(S)\,\underline{h}(S)]$ yields the exact gap

$$\begin{aligned} H(w) - H^\star(c) = {} & \underbrace{\mathbb{E}\left[\sum_{a>0} w_a(S)\{h_a(S) - \underline{h}(S)\}\right]}_{\Delta_{\mathrm{act}}(w)} \\ & + \underbrace{\mathbb{E}[z_w(S)\,\underline{h}(S)] - \int_0^c F_{\underline{h}}^{-1}(u)\,du}_{\Delta_{\mathrm{rank}}(w;\,c)}. \end{aligned} \tag{42}$$

The two terms are nonnegative by Lemmas 5 and 6. Hence equality requires both correct within-instance action choice and correct across-instance edit allocation, modulo action ties and boundary randomization. This proves Eq. (8). □

Shape of the frontier. Because $F_{\underline{h}}^{-1}$ is nondecreasing, $H^\star(c) = \int_0^c F_{\underline{h}}^{-1}(u)\,du$ is nondecreasing and convex, with

$$\frac{d}{dc} H^\star(c) = F_{\underline{h}}^{-1}(c) \quad \text{for almost every } c. \tag{43}$$

Thus the marginal regression mass of expanding coverage is the risk at the new boundary. Its running average

$$\rho^\star(c) = \frac{1}{c}\int_0^c F_{\underline{h}}^{-1}(u)\,du \tag{44}$$

is nondecreasing, because the average of a nondecreasing function over $[0, c]$ cannot exceed its right-boundary value. Expanding edit coverage cannot lower the oracle conditional-risk floor.

Finite-population form. For n evaluation instances, let n_+ be the number with an available repair and sort their minimum conditional risks as $h_{(1)} \leq \cdots \leq h_{(n_+)}$. For $0 < c \leq n_+/n$, put $k = \lfloor nc \rfloor$ and $\alpha = nc - k$. The empirical frontier is

$$H_n^\star(c) = \frac{1}{n}\Big\{\sum_{i=1}^{k} h_{(i)} + \alpha\, h_{(k+1)}\Big\}, \tag{45}$$

with the final term omitted when $\alpha = 0$. This is the population theorem's sorting construction, with fractional mass corresponding to randomization on the boundary instance. It uses conditional risks, not realized binary test outcomes; sorting the latter with references would only be a retrospective oracle diagnostic.

B.4 Population improvement–degradation optimization

Let $b_a(s)$ and $\ell_a(s)$ be the bounded conditional benefit and degradation in Section 2. The attainable improvement–degradation set is

$$\mathcal{V} = \{(\mathcal{L}(w),\, \mathcal{B}(w)) : w \in \mathcal{W}\}. \tag{46}$$

Lemma 7 (Convex attainable set)**.** *$\mathcal{V}$ is convex. On a finite empirical population it is a compact polytope.*

Proof. For policies w, w' and $\alpha \in [0, 1]$, the pointwise mixture $w_\alpha = \alpha w + (1-\alpha)w'$ is a valid policy and

$$(\mathcal{L}(w_\alpha),\, \mathcal{B}(w_\alpha)) = \alpha(\mathcal{L}(w),\, \mathcal{B}(w)) + (1-\alpha)(\mathcal{L}(w'),\, \mathcal{B}(w')). \tag{47}$$

For finitely many instances and actions, the policy space is a product of simplices, hence compact; its linear image $\mathcal{V}$ is a compact polytope. □

Finite-population primal and dual. For an empirical population $s_1, \dots, s_n$, write $b_{ia} = b_a(s_i)$ and $\ell_{ia} = \ell_a(s_i)$. The constrained problem is the linear program

$$\begin{aligned} \max_{w_{ia} \geq 0} \quad & \frac{1}{n} \sum_{i=1}^{n} \sum_{a \in \mathcal{A}(s_i)} b_{ia}\, w_{ia} \\ \text{subject to} \quad & \frac{1}{n} \sum_{i=1}^{n} \sum_{a \in \mathcal{A}(s_i)} \ell_{ia}\, w_{ia} \leq \kappa, \qquad \sum_{a \in \mathcal{A}(s_i)} w_{ia} = 1 \quad (1 \leq i \leq n). \end{aligned} \tag{48}$$

Introducing multiplier $\lambda \geq 0$ for the degradation budget and dual variables ν_i for the simplex constraints gives

$$\begin{aligned} \min_{\lambda \geq 0,\, \nu_1, \dots, \nu_n} \quad & \lambda\kappa + \frac{1}{n} \sum_{i=1}^{n} \nu_i \\ \text{subject to} \quad & \nu_i \geq b_{ia} - \lambda\, \ell_{ia} \quad \text{for every } i, a. \end{aligned} \tag{49}$$

At the dual optimum, each ν_i is the tightest feasible upper bound,

$$\nu_i = \max_{a \in \mathcal{A}(s_i)} \{b_{ia} - \lambda\, \ell_{ia}\}. \tag{50}$$

Eliminating ν_i recovers the empirical version of $g(\lambda)$ in Eq. (13). Linear-program strong duality gives

$$J_n(\kappa) = \min_{\lambda \geq 0} \Big\{ \lambda\kappa + \frac{1}{n} \sum_{i=1}^{n} \max_{a \in \mathcal{A}(s_i)} D_a^{\lambda}(s_i) \Big\}. \tag{51}$$

Population duality and pointwise separation. For the population problem, the Lagrangian is

$$\begin{aligned} \mathcal{J}_\lambda(w;\, \kappa) &= \mathcal{B}(w) - \lambda\{\mathcal{L}(w) - \kappa\} \\ &= \lambda\kappa + \mathbb{E}\left[\sum_a w_a(S)\, D_a^{\lambda}(S) \right]. \end{aligned} \tag{52}$$

For fixed λ, measurable pointwise maximization gives

$$\sup_{w \in \mathcal{W}} \mathcal{J}_\lambda(w;\, \kappa) = \lambda\kappa + \mathbb{E}\left[\max_{a \in \mathcal{A}(S)} D_a^{\lambda}(S) \right] = g(\lambda). \tag{53}$$

Strong duality holds for the finite program above. In the population setting, it follows under standard measurable-selection and closedness conditions plus a constraint qualification. In particular, when $\kappa > 0$, KEEP is strictly feasible because its deterministic policy w_0 satisfies $\mathcal{L}(w_0) = 0 < \kappa$, providing Slater feasibility. Under these conditions, $J(\kappa) = \inf_{\lambda \geq 0} g(\lambda)$.

Let $(w^\star, \lambda^\star)$ be primal–dual optimal. The optimality and complementary-slackness conditions are

$$w_a^\star(s) > 0 \implies a \in \arg\max_{j \in \mathcal{A}(s)} D_j^{\lambda^\star}(s), \tag{54}$$

$$\lambda^\star\{\mathcal{L}(w^\star) - \kappa\} = 0, \qquad \mathcal{L}(w^\star) \leq \kappa, \qquad \lambda^\star \geq 0. \tag{55}$$

The first condition is precisely the candidate-level net-value rule. The second states that a positive risk price occurs only when the expected-degradation budget is active.

Budget value and the meaning of λ. The value function $J(\kappa)$ is nondecreasing because a larger budget expands the feasible policy set. It is concave because mixing policies feasible at κ_1 and κ_2 yields a policy feasible at the corresponding convex combination of budgets. If $\lambda^\star$ is dual-optimal at κ, then for every feasible κ',

$$J(\kappa') \leq \lambda^\star\kappa' + \mathbb{E}\Big[\max_a D_a^{\lambda^\star}(S)\Big] = J(\kappa) + \lambda^\star(\kappa' - \kappa). \tag{56}$$

Thus $\lambda^\star$ is a supergradient of J: it is the marginal price assigned to relaxing the expected-degradation budget. Conversely, fixing λ directly selects a supported point of $\mathcal{V}$ and induces a degradation level; it does not by itself certify an externally specified κ.

KEEP and nested intervention sets. Because $b_0(s) = \ell_0(s) = D_0^\lambda(s) = 0$, KEEP is selected under KEEP-preferred tie breaking exactly when

$$M_\lambda(s) := \max_{a>0} D_a^\lambda(s) \le 0. \tag{57}$$

Here and below, $M_\lambda(s) = -\infty$ when no repair action is available. Moreover, if $\lambda_2 \ge \lambda_1$, then $D_a^{\lambda_2}(s) \le D_a^{\lambda_1}(s)$ for every action because $\ell_a(s) \ge 0$. Hence the exact-value intervention sets are nested:

$$\{s : M_{\lambda_2}(s) > 0\} \subseteq \{s : M_{\lambda_1}(s) > 0\}. \tag{58}$$

Increasing the degradation price can change which repair is preferred, but cannot turn an exact-value KEEP decision into an intervention.

Action-dependent acceptance threshold. For a two-outcome repair, let p be its conditional probability of improvement and let $u_+, u_- \ge 0$ be its benefit and degradation consequences, with $u_+ + \lambda u_- > 0$. Then $b_a = pu_+$ and $\ell_a = (1-p)u_-$. The repair is preferred to KEEP exactly when

$$pu_+ - \lambda(1-p)u_- > 0 \iff p > \frac{\lambda u_-}{u_+ + \lambda u_-}. \tag{59}$$

The threshold varies with the action's relative benefit and degradation consequences. A universal confidence cutoff represents the population objective only when the resulting threshold is constant across actions. When $u_+ > 0$ and $0 < \varrho < 1$, with improvement prior ϱ and evidence likelihood ratio Λ, posterior odds equal prior odds times Λ, so Eq. (59) is equivalent to

$$\Lambda > \frac{\lambda u_-}{u_+} \cdot \frac{1-\varrho}{\varrho}. \tag{60}$$

When $m = 0$, the caps are removed, and $\lambda = 1$,

$$b_a(s) - \ell_a(s) = \mathbb{E}[(G_a)_+ - (-G_a)_+ \mid S = s] = \mathbb{E}[G_a \mid S = s]. \tag{61}$$

Expected signed gain is therefore a special case of the same decision family. The factorized benefit–degradation estimator permits asymmetric consequence control; the derivation does not claim that it must be statistically easier to estimate.

B.5 STABILITY OF THE PLUG-IN RCAS DECISION

RCAS replaces $b_a(s)$ and $\ell_a(s)$ with calibrated estimates $\widehat{b}_a(s)$ and $\widehat{\ell}_a(s)$. The following deterministic result isolates how estimation error affects the selected action for a fixed λ; it is not an assumption that these errors are uniformly small.

Define

$$\widehat{D}_a^\lambda(s) = \widehat{b}_a(s) - \lambda\,\widehat{\ell}_a(s), \qquad \varepsilon_\lambda(s) = \max_{a \in \mathcal{A}(s)} \left|\widehat{D}_a^\lambda(s) - D_a^\lambda(s)\right|, \tag{62}$$

and let $a^\star(s) \in \arg\max_a D_a^\lambda(s)$ and $\widehat{a}(s) \in \arg\max_a \widehat{D}_a^\lambda(s)$.

Lemma 8 (Plug-in action regret)**.** *For every state,*

$$0 \le D_{a^\star(s)}^\lambda(s) - D_{\widehat{a}(s)}^\lambda(s) \le 2\,\varepsilon_\lambda(s). \tag{63}$$

Consequently, the population net-value regret of the plug-in policy is at most $2\,\mathbb{E}[\varepsilon_\lambda(S)]$.

Proof. Optimality of $\widehat{a}$ under the estimated scores gives $\widehat{D}_{a^\star}^\lambda - \widehat{D}_{\widehat{a}}^\lambda \le 0$. Adding and subtracting the estimated scores yields

$$\begin{aligned} D_{a^\star}^\lambda - D_{\widehat{a}}^\lambda &= (D_{a^\star}^\lambda - \widehat{D}_{a^\star}^\lambda) + (\widehat{D}_{a^\star}^\lambda - \widehat{D}_{\widehat{a}}^\lambda) + (\widehat{D}_{\widehat{a}}^\lambda - D_{\widehat{a}}^\lambda) \\ &\le 2\,\varepsilon_\lambda(s). \end{aligned} \tag{64}$$

The lower bound follows from the definition of $a^\star$; taking expectations gives the population statement. □

If the oracle action has a true net-value margin exceeding $2\varepsilon_\lambda(s)$ over every alternative, then for every $a \ne a^\star$,

$$\widehat{D}_{a^\star}^\lambda(s) - \widehat{D}_a^\lambda(s) \ge D_{a^\star}^\lambda(s) - D_a^\lambda(s) - 2\varepsilon_\lambda(s) > 0. \tag{65}$$

Hence $\widehat{a}(s) = a^\star(s)$, as stated in Section 2.

Decision-margin implication. Assume KEEP is represented exactly, so $\widehat{D}_0^\lambda = D_0^\lambda = 0$, and recall the true repair margin $M_\lambda(s)$ from Eq. (57). Then

$$M_\lambda(s) > 2\varepsilon_\lambda(s) \implies \widehat{a}(s) > 0 \text{ and } \widehat{D}^\lambda_{\widehat{a}(s)}(s) > 0, \tag{66}$$

$$M_\lambda(s) < -\varepsilon_\lambda(s) \implies \widehat{a}(s) = 0. \tag{67}$$

The first implication follows because the best repair remains positive after score perturbation and Eq. (63) keeps the selected action's true value above zero. For the second, every estimated repair value is negative, so KEEP is uniquely preferred. Decisions can change only near the zero-value boundary or when estimation error is large relative to the margin. This clarifies the role of relative outcome estimation and calibration in RCAS without turning the population objective into a per-edit reliability guarantee.

C Additional RCAS implementation details

C.1 Outcome supervision and training objective

Reference-derived targets. For record i and non-`KEEP` action a, the supervision g_{ia} and the utility Q_ω are defined in Section 3 (Eqs. (16)–(17)); sentence-level chrF uses SacreBLEU's default `CHRF` configuration, and outcome labels follow Eq. (18). Writing ΔC and ΔF for the candidate-relative COMET-22 and chrF/100 differences, the utility weight is

$$\omega = \frac{\mathrm{sd}_{\mathrm{train}}(\Delta C)}{\mathrm{sd}_{\mathrm{train}}(\Delta F)}, \tag{68}$$

computed over non-rejected training candidates with available scores, including the zero differences of `KEEP`, and set to $\omega = 0$ if the denominator is zero; the value is frozen. The tolerance and consequence caps are mapped from their COMET-scale configuration to composite-utility units by

$$\rho = \frac{\mathrm{sd}_{\mathrm{tune}}(g)}{\mathrm{sd}_{\mathrm{tune}}(\Delta C)}, \qquad m = 0.01\,\rho, \qquad B_{\max} = L_{\max} = 0.10\,\rho, \tag{69}$$

where the scale ratio uses non-rejected, non-`KEEP` candidates of the development tune pool; mapped values are rounded to six decimal places and frozen. Thus ω is training-derived, whereas m, $B_{\max}$, and $L_{\max}$ are development-scale mappings of the COMET-scale configuration.

Text encoding and edit-span branch. The candidate encoder reads four fields: source, current translation, candidate, and deterministic edit script. The `KEEP` representation uses the current translation in the candidate field and a `[NO_EDIT]` script. Valid-token outputs of the shared XLM-R-large encoder are mean-pooled to obtain each text representation. The edit script is extracted by token-level `SequenceMatcher`, retaining insertion, deletion, and replacement operations and omitting unchanged spans. Each edit unit is separately encoded as four fields: source, old span, new span, and operation label; a missing old or new span is represented by `<EMPTY>`. A linear head maps the mean-pooled representation of edit unit j to a scalar t_{aj}, and the edit contribution is

$$\delta_a^{\mathrm{edit}} = \begin{cases} \log \sum_{j=1}^{J_a} \exp(t_{aj}) - \log J_a, & J_a > 0, \\ 0, & J_a = 0, \end{cases} \tag{70}$$

added to the degradation-class logit in Eq. (19), not to the signed-gain prediction. All packed inputs are limited to 384 tokens including special tokens; the remaining budget is shared across fields, with unused capacity from shorter fields redistributed to longer fields, and each field is prefix-truncated to its assigned budget. The relative and set-level projections use hidden dimension 256, GELU activations, and dropout 0.1; output heads are affine maps whose biases are omitted from the main-text notation.

Reference-free features. The feature vector f_a contains 27 scalar inputs in four groups:

- **QE:** initial and candidate QE, QE difference from `KEEP`, QE gap to the best candidate, QE rank, top-QE margin, and MBR consensus.

- **Edit and length:** edit rate, candidate-to-initial length ratio, length-change ratio, duplication with `KEEP`, source, initial, and candidate lengths, and the source-to-target length ratio.
- **Composition:** fusion and composition indicators, composition block count, edited-block count, edit fraction, mean consensus, and mean family support.
- **Risk and pool:** best pool-level QE improvement, valid-evidence edit indicator, unique-candidate ratio, and critical-risk and constraint-violation summaries.

These inputs are standardized using training-set means and standard deviations, replacing standard deviations no greater than 10^{-6} by one; the transformation is frozen thereafter. Number, entity, and polarity checks enter through the aggregate risk and constraint features rather than as separate input dimensions.

Pointwise objectives and weighting. The estimator produces a signed-gain prediction $\widetilde{G}_{ia}$ and logits ξ_{ia} over $\{\mathrm{imp}, \mathrm{tie}, \mathrm{reg}\}$; the objective and its pointwise terms are Eqs. (20) and (21), evaluated over the non-rejected, non-`KEEP` candidate rows $\mathcal{A}_i^+$. The scale $s_G = \max\{\mathrm{sd}_{\mathrm{train}}(g), 10^{-6}\}$ is computed over training candidate rows, and Smooth L1 transitions at normalized absolute error one, so normalization changes the loss scale but not the utility units of $\widetilde{G}$. Balanced cross-entropy weights each class by $N/\max(3N_c, 1)$, where N_c is the training-row count of class c and N the total. Exposure weights are

$$\widetilde{w}_{ia} = 1 + \mathbf{1}[q_{ia} > q_{i0}] + \mathbf{1}[q_{ia} = q_i^{\max}], \qquad w_{ia}^c = w_{ia}^g = \frac{\widetilde{w}_{ia}}{\sum_{b \in \mathcal{A}_i^+} \widetilde{w}_{ib}}, \tag{71}$$

where q_{ia} is reference-free QE and $q_i^{\max}$ its maximum over non-rejected, non-`KEEP` candidates before policy-level family exclusions; all candidates tied at the maximum receive the final increment. This emphasizes QE-attractive candidates without discarding the remaining training examples.

Ranking and tail objectives. The pair set $\mathcal{P}_i$ and the harmful set $\mathcal{H}_i$ are defined in Section 3; each candidate–candidate pair enters once, and `KEEP` participates with $g_{i0} = \widetilde{G}_{i0} = 0$ in Eq. (22). The evaluated tail configuration compares harmful candidates against a fixed raw-logit baseline of zero in Eq. (23). Empty pair or tail sets contribute zero, and the main configuration uses $\eta_g = \eta_r = \eta_t = 1$ and $\tau = 0.5$. The tail term is an auxiliary penalty on the raw degradation logit: a common shift of all logits leaves softmax probabilities unchanged, so this penalty does not itself enforce a degradation-probability margin or a calibrated risk constraint.

Optimization. The encoder and heads are trained jointly with AdamW for three epochs, using learning rate 2×10^{-5}, weight decay 0.01, gradient clipping at norm 1.0, and a linear schedule with warmup ratio 0.06. Training uses bfloat16, gradient checkpointing, record batch size one, and eight gradient-accumulation steps. With this record batch size, pair and tail terms are averaged within each record before accumulation, avoiding automatic upweighting of records with more comparisons.

C.2 From outcome predictions to benefit and degradation

Calibration data. The development tune set is partitioned by document with `GroupShuffleSplit` using policy fraction 0.5 and split seed 42; the fraction applies to document groups and not necessarily to equal numbers of records. One subset fits the calibration mappings and consequence tables; the other compares operating configurations before freezing. The calibration subset is disjoint from predictor training, and gain calibration, probability calibration, and consequence-table estimation reuse the same calibration rows rather than independent folds. The development-wide scale of Eq. (69) is determined before this partition.

Gain calibration. Let $d(a)$ denote the stored action category and define calibration residuals $r_{ia} = g_{ia} - \widetilde{G}_{ia}$. With global mean residual $\bar{r}$ and group count n_d, the gain correction is

$$\Delta_d = \frac{\sum_{(i,a):\, d(a)=d} r_{ia} + \alpha_g \bar{r}}{n_d + \alpha_g}, \qquad \widehat{g}_{ia} = \widetilde{G}_{ia} + \Delta_{d(a)}, \tag{72}$$

with $\alpha_g = 20$; an unseen group uses the global offset $\bar{r}$.

Degradation-probability calibration. The degradation component of $\mathrm{softmax}(\xi_a)$ is first calibrated by isotonic regression against $\mathbf{1}[g_{ia} < -m]$, giving p_a^{iso}. Candidates are stratified jointly by action category, initial-QE tertile (boundaries computed over calibration candidate rows), and edit-size group. Edit size is the sum over edit operations of the larger old/new token-span length, divided by the current translation's token count with denominator at least one and capped at four; its groups are $[0, 0.05]$, $(0.05, 0.15]$, and $(0.15, 4]$. For stratum j with n_j candidates and h_j degradation outcomes, the shrunk rate is

$$\widehat{p}_j = \frac{h_j + \alpha_p\, p_{\mathrm{global}}}{n_j + \alpha_p}, \qquad \alpha_p = 20, \tag{73}$$

where p_{global} is the calibration-wide degradation frequency, and the first-stage prediction is adjusted by

$$p'_a = \sigma\left[\mathrm{logit}(p_a^{\mathrm{iso}}) + \mathrm{logit}(\widehat{p}_j) - \mathrm{logit}(p_{\mathrm{global}})\right]. \tag{74}$$

An unseen stratum receives a zero log-odds adjustment and is then handled by the final stage. A final isotonic mapping, fitted against the same binary outcomes, maps p'_a to the calibrated degradation probability $\widehat{p}_a$. For numerical stability, logit inputs are clipped to $[10^{-6}, 1 - 10^{-6}]$ and sigmoid arguments to $[-30, 30]$; these safeguards apply at the transformation steps and do not restrict the final isotonic outputs to lie strictly between zero and one. The stages have distinct roles: the first isotonic mapping corrects the global score–frequency relationship, the stratum adjustment accounts for residual differences across candidate groups, and the final isotonic mapping recalibrates the adjusted score. The consequence table serves a different purpose, estimating degradation magnitude rather than degradation frequency.

Consequence targets and tables. For every calibration candidate we compute the realized bounded consequences

$$B_{ia} = \min\{\max(g_{ia}, 0), B_{\max}\}, \qquad L_{ia} = \min\{\max(-g_{ia} - m, 0), L_{\max}\}. \tag{75}$$

Benefit and degradation are therefore transforms of the same observed outcome, not independently annotated labels; positive gains within the tie interval still contribute benefit. The deployed estimator uses two separate one-dimensional tables: benefit is indexed by action category and calibrated gain, whereas degradation is indexed by action category and calibrated degradation probability. It does not use a joint gain–probability grid. Bin boundaries are calibration quantiles with duplicate boundaries removed, yielding at most 12 gain bins and at most 10 probability bins; predictions outside the stored range use the nearest bin. Within each group–bin cell $\mathcal{C}_{d,k}$, either target $T \in \{B, L\}$ is estimated by

$$\widehat{T}_{d,k} = \frac{\sum_{(i,a)\in\mathcal{C}_{d,k}} T_{ia} + \alpha\, \overline{T}_k}{n_{d,k} + \alpha}, \qquad \alpha = 20,$$

equivalent to Eq. (24), where $\overline{T}_k$ is the pooled bin mean. Empty cells and unseen action categories use the pooled bin estimate, and an empty pooled bin falls back to the global mean of that target. The degradation table averages excess-degradation magnitudes L_{ia}, not binary degradation labels; its output is an expected magnitude and is not multiplied by $\widehat{p}_a$ again.

Relation to the theoretical quantities. The resulting lookups approximate the conditional consequences through the learned summaries,

$$\widehat{b}_a \approx \mathbb{E}[B_a \mid d(a), \mathrm{bin}(\widehat{g}_a)],$$
$$\widehat{\ell}_a \approx \mathbb{E}[L_a \mid d(a), \mathrm{bin}(\widehat{p}_a)].$$

These lookups replace conditioning on the full state by conditioning on action category and a binned predictor summary; shrinkage further regularizes the finite-sample estimates.

Operating configuration. With the mappings fixed, the policy subset compares a fixed grid of operating configurations: retaining all action categories versus excluding standalone `LOCAL`, crossed with an additive offset c to non-`KEEP` net values, $\widehat{V}_a(c) = \widehat{b}_a - \lambda\widehat{\ell}_a + c$ for $c \in \{0, 0.004, 0.008, 0.016, 0.024, 0.032\}$, with $\lambda = 1$ fixed. The offset is an operating parameter selected separately for each run on the development policy subset and frozen before test evaluation. Candidate policies must have positive empirical mean realized net value and nonzero edit coverage.

Table 4: Per-record call budget of the proposal workflow. "Current" denotes the current translation and "certs" the error certificates.

Calls	Family	Input	Output	Temp.
1–2	Evidence audit	src + current	two error certificates	0.20
3–4	Generic rewrite	src + current	two full revisions	0.55
5–6	Retranslation	src only	two new translations	0.65
7–8	Local repair	src + current + one cert	two minimal span edits	0.30
9	Fusion	src + current + certs + candidates	one fused translation	0.35

Within each mask, policies within one standard error of the largest mean net value (computed from record-level realized net values) are compared by lower conditional degradation rate, then higher coverage, then higher mean net value; the same rule compares winners across masks. Here the conditional degradation rate is the fraction of edited records with composite-utility gain $g < -m$, rather than the COMET-only automatic-harm measure reported in the experiments. The mask and offset selected on the policy subset are frozen before test evaluation. This is an empirical selection heuristic, not a test-time risk certificate.

Inference. At inference, the frozen tables return $\widehat{b}_a$ and $\widehat{\ell}_a$ without references. Among eligible repair candidates, RCAS selects the largest positive adjusted net value $\widehat{b}_a - \lambda\widehat{\ell}_a + c$; otherwise it selects `KEEP`. Equal positive values are resolved by higher calibrated gain, then higher reference-free QE, then candidate order. No additional positive-raw-gain requirement is imposed.

C.3 Proposal supply and the action set

Generation workflow. A fixed Qwen3-4B-Instruct-2507 model supplies nine logical completions per record with the budget and temperatures of Table 4; thinking is disabled and top-p is 0.30. Evidence completions are capped at 320 output tokens, and all translation completions at 384 output tokens. Evidence audits inspect the source and current translation and return structured certificates containing an edit/abstain decision, error type and severity, evidence spans, anchoring information, a replacement, and confidence. Aggregation compares signatures formed from the decision, error type, source and target spans, target anchor, placement, and replacement; with two audits, agreement is one for matching signatures and one half otherwise, so the threshold 0.75 prevents a discordant pair from supplying valid aggregated editing evidence. Structural validity does not establish semantic correctness. Generic rewrites condition on the source and current translation; retranslations condition on the source alone. Each local repair uses its corresponding individual certificate, and both local completions are requested even when the audits disagree. Validation checks certificate validity, anchoring, and edit scope. Out-of-scope generations and generations with an empty replacement inside the certified edit envelope retain a validation error and are rejected under the evaluated configuration, even if a mechanical fallback is stored. An entirely empty generation instead invokes mechanical substitution directly; its eligibility depends on the validation result of that substitution. Fusion receives the source, current translation, both certificates, and preceding non-rejected proposals presented anonymously, and generates a complete translation rather than concatenating candidate fragments.

Clause blocking and alignment. Composition adds candidates without further LLM completions and is enabled by a whole-sentence QE improvement or valid aggregated editing evidence. The current translation and candidates are split at the configured clause punctuation, retaining punctuation and merging fragments shorter than four characters with a neighbor. A monotone dynamic program aligns candidate units to the current translation using one-to-one, one-to-two, two-to-one, insertion, and deletion transitions; matched blocks cost one minus character-sequence similarity from `SequenceMatcher`, and gaps cost one. Shared block boundaries are merged where rewrites span multiple current-translation units. Zero-width insertions do not form independent composition blocks, and composition is skipped when the shared partition exceeds 12 blocks.

Clustering and support counting. Alternatives at each block are clustered greedily using character-trigram F1 with threshold 0.82 against the first member of each existing cluster; medoid

representatives are selected after cluster assignment, with QE used for tie breaking. Support counts distinct generation families from provenance, not multiple samples from the same family. QE-weighted support sums $\sigma(5\,\Delta q_f^{\max})$ over supporting generation families f, where $\Delta q_f^{\max}$ is the largest whole-sentence QE improvement among supporting candidates from family f. Clusters are ranked by family count, then QE-weighted support, then the best supporting candidate's QE improvement.

Composition variants and deduplication. Up to three variants are constructed: a strict variant requiring support from at least two families; a QE-supported variant that additionally permits a single-family alternative supported by a whole-sentence candidate with QE above `KEEP`; and an evidence-targeted variant that additionally permits single-family alternatives at blocks overlapping validated target evidence. Cross-family support is therefore not required for every composite. Each variant chooses at most one alternative per shared block and retains the current translation elsewhere; outputs equal to `KEEP` or to an existing candidate after whitespace normalization are omitted. Whole-sentence candidates remain available whether or not composition is enabled. The evaluated pools were generated without an additional filter rejecting empty block realizations or reductions in clause count; composition therefore does not guarantee preservation of clause count or semantic completeness.

Action categories and eligibility. Composed candidates are stored with action label `FUSION` and provenance `unit_composite`: they share the `FUSION` calibration group while remaining distinguishable through provenance and a dedicated predictor feature, and they are not a separate calibration group. For calibration, $d(a)$ denotes the stored action category. For composition, cross-family support is counted using generation provenance, falling back to the action category when provenance is unavailable. These two groupings need not coincide. Rejected proposals, candidates excluded by the frozen action mask, and candidates with non-finite required scores are unavailable for selection. The evaluated selector uses critical-risk and constraint summaries as features without an additional hard rejection based on these summaries. When the selected action mask excludes standalone `LOCAL`, this exclusion does not remove its earlier contribution to fusion or composition.

Candidate accounting. The nine completions produce seven proposed translations before deduplication and rejection; adding `KEEP` gives eight base candidates. Composition adds at most three more, while validation and deduplication can reduce the usable count. The nine-completion budget excludes initial MT generation, QE scoring, and selector computation. Once the pool is built, outcome prediction, calibration lookups, and the final `KEEP`/edit decision require no further LLM generation.

D Additional empirical results

This appendix follows the three empirical questions in the main text. It first checks how repair opportunity varies with initial quality, then probes the value and risk of frozen decisions, and finally reports additional human-audit outcomes. Post hoc diagnostics are distinguished from preregistered tests throughout. Opportunity analyses and the direct bound audit use the original deployment at seed 42; the decision component controls and the author code seed variation use five seeds. Transfer and threshold sweep diagnostics keep their separately specified three seed or single seed scope.

D.1 Opportunity and subgroup sensitivity

Within datasets, Spearman correlations between official human quality and RCAS gain are -0.66, -0.83, and -0.83; the corresponding oracle-gain correlations are -0.83, -0.94, and -0.77. With human quality z-normalised and both variables demeaned within dataset, the pooled slopes per quality standard deviation (six systems per dataset) are -2.43 COMET points for RCAS (source-document bootstrap 95% interval $[-2.83, -2.00]$) and -2.53 for the oracle ($[-2.92, -2.13]$). Resampling systems instead gives the wider intervals $[-3.07, -0.45]$ and $[-3.16, -0.61]$, respectively. A stratified one-sided permutation test gives $p < 10^{-4}$ for both. The same negative association appears for QE-best and a fixed generic rewrite.

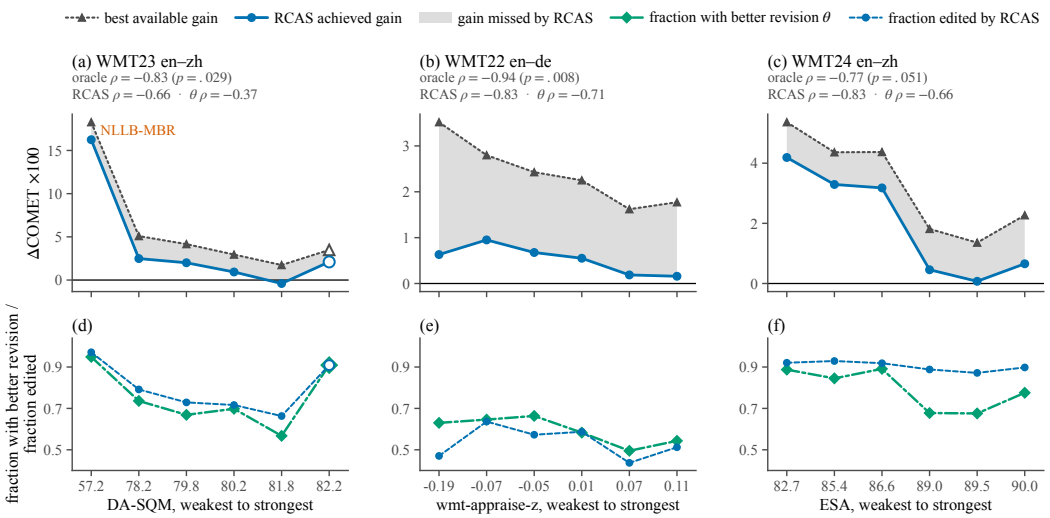


Figure 3: Repair value versus initial quality across 18 base-system cells from the original seed-42 deployment, ordered within each human-rated dataset. **Top:** best available and RCAS gain; shading is missed value. **Bottom:** opportunity frequency θ and RCAS coverage. Correlations are within-dataset Spearman; hollow markers have < 100 records.

The fraction with a better proposed revision also falls by 0.067 per quality standard deviation (document/system intervals $[-0.078, -0.054]/[-0.090, -0.004]$; permutation $p = 0.00069$), whereas the interval for the fraction edited by RCAS crosses zero. The pattern therefore concerns the revisions that are available, not merely how often RCAS edits. Because this analysis is post-hoc with six systems per dataset, we interpret it for these generated revisions and metrics.

Oracle gain falls by 2.532 COMET points per standard deviation. The absolute RCAS/QE-best oracle-gap slopes are only $-0.107/-0.060$ (intervals $[-0.218, 0.009]/[-0.149, 0.044]$), so the shortfall does not detectably widen. Their recovery ratios fall by 0.172/0.162 as useful revisions become rarer. We use these only as a secondary view because ratios become unstable near zero oracle gain.

In absolute terms, RCAS ranges from -0.40 to $+16.24$ COMET points on WMT23, from $+0.07$ to $+4.19$ on WMT24, and remains positive in all six WMT22 en–de cells. The policy can therefore regress on a strong-system subgroup despite positive aggregate gain.

The two preregistered test sets ask whether this decline depends on the initial three datasets. On en–de, the oracle slope is -4.50 $[-7.00, -0.14]$, but becomes -0.47 $[-0.81, +0.02]$ after removing the extreme MSLC system: the direction remains, but its magnitude is driven largely by that system. The zh–en design removes that structure; ten systems span a narrower DA-SQM range with no isolated weak system. Its oracle slope is -1.37 $[-1.76, -0.68]$ ($p = 0.00078$), remains negative after removing the weakest system ($[-1.76, -0.44]$), and has $\rho = -0.90$ ($p = 0.00034$). The decline in available improvement is therefore clearest where the confirmatory design does not rely on one weak system.

The aggregation is composition-sensitive. On WMT23, the weak NLLB system drives the positive aggregate BLEU and chrF results; most strong-system cells are negative. We therefore use per-system cells for the regime claim and treat the aggregate only as a deployment-mixture summary.

Candidate-level beneficial-action prior. The illustrative likelihood-ratio threshold in Eq. (60) depends on an individual-action improvement prior ϱ, not the record-level opportunity fraction θ. We therefore add a post-hoc check on the unchanged frozen pools. Within each base-system cell, ϱ is the fraction of unrejected non-KEEP candidate rows whose COMET score strictly exceeds KEEP. Its Spearman association with independently rated initial quality is -0.75 on WMT23 en–zh, -0.89 on WMT22 en–de, and -0.83 on WMT24 en–zh; exact one-sided permutation probabilities over

the six system labels are 0.047, 0.017, and 0.029. Candidate counts reflect deduplication and vary by record, and the WMT23 Yishu cell contains only 11 records. We therefore treat this as frozen-pool support for the direction predicted by the evidence illustration, not as an input to the deployed policy or a prospective calibration result. COMET improvement here is a proxy for the theoretical utility-specific event.

D.2 Direct audit of the opportunity–edit bound

Table 5 evaluates the excess-coverage certificate in Section 2 on the four frozen scored pools for which record-level RCAS selections are available. An opportunity means that at least one non-rejected candidate has strictly positive reference-based gain over KEEP. Every row satisfies the bound. The result is more informative than the inequality alone: even when $e_\pi \leq \theta$ on en–de, 14.6–18.7% of the population is edited without a better candidate under the corresponding utility, confirming that edit volume cannot diagnose decision quality. Because the bound is algebraic, this is a finite-sample consistency diagnostic, not a hypothesis test.

Table 5: **Frozen-pool audit of the excess-coverage implication (seed 42).** Unsupported is $\Pr(\pi > 0, o = 0)$; bound is $(e_\pi - \theta)_+$; slack is their difference. All entries are percentages of the fixed test population.

Dataset	Utility	e_π	θ	Unsupported	Bound	Slack
WMT23 en–zh	COMET	77.9%	73.0%	15.8%	5.0%	10.8%
	Q_ω	77.9%	68.0%	20.5%	9.9%	10.6%
WMT22 en–de	COMET	53.1%	58.9%	14.6%	0.0%	14.6%
	Q_ω	53.1%	53.1%	18.7%	0.0%	18.7%
WMT24 en–zh	COMET	90.4%	79.2%	16.0%	11.2%	4.8%
	Q_ω	90.4%	74.8%	20.2%	15.7%	4.6%
WMT23 zh–en	COMET	84.2%	77.8%	14.8%	6.3%	8.4%
	Q_ω	84.2%	75.7%	16.6%	8.5%	8.1%

D.3 Frozen decision-component controls

All controls below are post-hoc and use the already consumed en–zh and zh–en tests. Each deployed selection is first reproduced with zero candidate mismatches at seeds 42/7/13/17/19. Paired differences are averaged over the five seeds within each record, then source documents are resampled 10,000 times. The percentile intervals in Table 6 condition on these trained selectors; they do not resample training seeds and are not adjusted for multiple comparisons.

Table 6: **Five-seed paired decision-component contrasts.** Each entry is left minus right, with a 95% source-document bootstrap interval. Quality differences are metric points; regression differences are percentage points of all test records.

Contrast	Direction	ΔQ_ω	ΔCOMET	ΔReg. (pp)
KEEP vs forced	en–zh	$+0.945\ [+0.711, +1.109]$	$+0.219\ [+0.145, +0.291]$	$-10.18\ [-11.73, -8.35]$
	zh–en	$+0.235\ [+0.055, +0.437]$	$+0.149\ [+0.018, +0.302]$	$-5.71\ [-7.15, -2.71]$
RCAS vs QE rank	en–zh	$+0.809\ [+0.340, +1.259]$	$-0.383\ [-0.695, -0.157]$	$-2.56\ [-3.93, -1.17]$
	zh–en	$-0.093\ [-0.441, +0.250]$	$-0.194\ [-0.328, -0.107]$	$-2.22\ [-3.85, -0.91]$
Net vs signed gain	en–zh	$-0.018\ [-0.190, +0.208]$	$-0.077\ [-0.167, +0.021]$	$+0.64\ [-0.01, +1.34]$
	zh–en	$-0.182\ [-0.353, +0.024]$	$-0.140\ [-0.230, -0.027]$	$+0.40\ [-0.84, +1.46]$
Net vs benefit only	en–zh	$+0.094\ [-0.100, +0.280]$	$-0.124\ [-0.254, -0.029]$	$-0.30\ [-1.14, +0.66]$
	zh–en	$+0.516\ [+0.290, +0.838]$	$+0.182\ [+0.007, +0.358]$	$-5.51\ [-6.44, -4.52]$

For E1, retaining KEEP improves both quality metrics and cuts population regression by 10.18 and 5.71 percentage points in the two directions. This is the largest regression reduction among the component contrasts in Table 6, so the retain decision is the main driver of population regression. Forced editing reaches 99.6/99.2% coverage; KEEP lowers coverage by 26.9/13.6 percentage points on en–zh and zh–en. The candidate on each record already accepted by RCAS is unchanged, iso-

lating the boundary at action zero. Records without a feasible candidate other than KEEP remain unchanged even in the arm with forced edits.

For E2, the accepted records and candidate eligibility mask are fixed; only ranking changes. RCAS ranking strategically prioritizes regression reduction, successfully lowering population regression by 2.56 and 2.22 percentage points in both directions while maintaining competitive composite utility. The two rankings pick different candidates on about 55–64% of accepted en–zh records and 36–43% of accepted zh–en records over the five seeds. Both rules exceed the random candidate bands on the same records for both quality metrics; these bands describe candidate randomization rather than uncertainty from document sampling.

For E3, alternative thresholds approximate deployed coverage on tune and are then frozen for test. Against signed gain, the net score gives composite and regression intervals that cross zero in both directions but lowers COMET on zh–en with an interval excluding zero (-0.140). The factored benefit and degradation form therefore matches the signed gain case on composite and regression, and adds asymmetric consequence control (cf. Appendix B, Eq. (61)), without dominating it. Against benefit alone, the degradation term improves both quality metrics and cuts regression by 5.51 percentage points on zh–en; on en–zh it lowers COMET (-0.124, interval excluding zero) with composite and regression unresolved. So the degradation term matters against benefit alone on zh–en, but does not dominate signed gain in both directions. Benefit score quantization leaves coverage gaps on tune of at most 1.64 percentage points, and the coverage difference on test is not significant between scoring variants.

Matched-edit random controls. In the original seed-42 diagnostic, at RCAS's edit count it exceeds all three random-control 95% bands on both quality scales and falls below every harm band in the three initial tests. The closest quality control, random records with QE-best candidates on WMT24 en–zh, reaches 1.824 $[1.722, 1.923]$ COMET points above KEEP, versus RCAS's 1.985. The three controls separately randomize record choice, candidate choice, or both, diagnosing whether each choice carries information beyond chance.

Frozen cross-generator transfer. We replace only the proposal pool with candidates from Qwen2.5-7B-Instruct on WMT23 en–zh; all trained components remain fixed. Across seeds, RCAS gains 3.709 COMET $[3.117, 4.455]$ and 5.214 composite $[4.460, 6.122]$ at a 0.631 edit rate. Relative to same-pool QE-best, the frozen RCAS policy lowers overall automatic regression by 0.049 $[0.026, 0.072]$ and the edit rate by 0.128 $[0.088, 0.174]$, while maintaining competitive composite utility ($+0.384$ $[-0.313, +0.967]$); the corresponding COMET difference of -0.628 $[-0.953, -0.376]$ reflects the same risk-aware trade-off observed in the main comparison. A selector trained and calibrated on the target pool achieves a $+0.865$ $[0.276, 1.494]$ composite advantage over QE, confirming that the frozen policy's transfer is substantive rather than incidental.

D.4 System heterogeneity in frozen transfer

Figure 4 stratifies the frozen RCAS–QE contrast by the initial MT system. RCAS lowers overall automatic regression and the edit rate for five of six systems. The exception is NLLB_MBR_BLEU, the weakest initial system, where RCAS edits 6.6 points more and the harm difference is $+0.013$ $[-0.012, +0.033]$. This is substantively compatible with greater repair opportunity, although RCAS also loses 1.06 composite points there. Across all systems, RCAS changes overall automatic regression by -0.049 $[-0.072, -0.026]$ and the edit rate by -0.128 $[-0.175, -0.089]$; both directions survive all six leave-one-system-out analyses. This post-hoc analysis uses frozen selections and no refitting. The Yishu subgroup has only 11 records from nine documents, so its intervals are descriptive.

Joining the same frozen records to their prespecified text-type labels gives a second heterogeneity check (Table 7). Frozen RCAS has document-bootstrap intervals above zero for absolute COMET and composite gain in all four types: its composite gains are $5.67/6.54/4.34/4.33$ points for news/social/speech/user-review. Relative to QE, RCAS's lower-regression point estimate recurs in all four types, with intervals excluding zero for news and speech and reaching zero for user reviews; the social interval crosses zero. Composite differences favor RCAS in news and speech but cross zero in social and user reviews. Leaving out any one type preserves the aggregate lower-regression and lower-edit directions. These post-hoc slices diagnose frozen cross-genre robustness

Table 7: **Frozen cross-genre transfer.** RCAS-minus-QE differences average the three frozen selectors within record; brackets are 95% source-document bootstrap intervals (10,000 resamples). Quality entries are points; regression and edit entries are percentage points.

Text type (n/docs)	ΔCOMET	ΔQ_ω	Δregression	Δedit
News (397/5)	-0.29 $[-0.43, +0.01]$	$+1.21$ $[+0.17, +3.12]$	-8.40 $[-12.10, -6.00]$	-21.49 $[-31.11, -14.11]$
Social (346/11)	-1.04 $[-1.55, -0.42]$	-0.55 $[-1.33, +0.14]$	-1.93 $[-8.65, +3.62]$	-7.51 $[-12.48, -3.41]$
Speech (376/3)	-0.23 $[-0.43, +0.32]$	$+1.09$ $[+0.98, +1.35]$	-4.43 $[-5.71, -3.52]$	-6.65 $[-10.98, +2.38]$
User review (349/8)	-1.04 $[-1.75, -0.48]$	-0.39 $[-2.33, +1.17]$	-4.49 $[-8.55, 0.00]$	-14.80 $[-22.97, -8.01]$

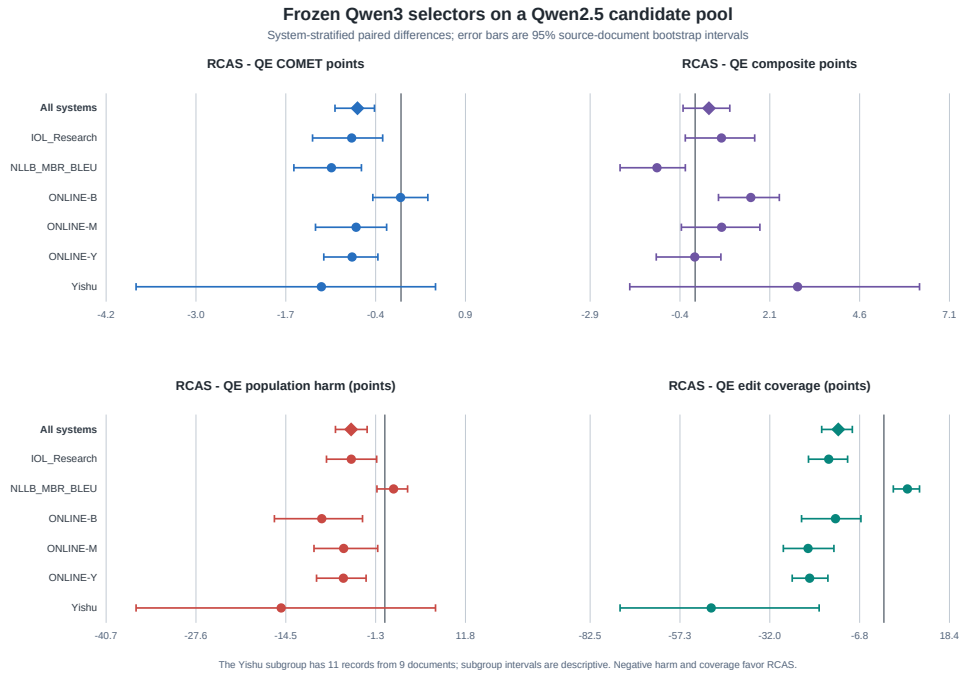


Figure 4: **Frozen cross-generator robustness across initial MT systems.** Paired RCAS-minus-QE differences average the three frozen selectors within each record; intervals use 10,000 source-document bootstrap resamples. Diamonds show the aggregate. Negative regression and edit-rate differences favor RCAS.

within WMT23; because development and test are not genre-disjoint, they are not an unseen-domain transfer test. News and speech contain only five and three source documents, so their cluster intervals are descriptive rather than strong genre-level evidence.

D.5 Frozen candidate-call frontier

We construct a dependency-valid nested frontier from the three initial scored pools without new generation or fitting. One call supplies the first generic rewrite; four supply both generic and both retranslation proposals; six add two evidence calls; eight add their two local repairs; and nine add fusion. Clause composites are available only when all of their source candidates and any required evidence are available. Frozen action masks remain in force.

Table 8 compares the four-call point with the complete pool. Oracle gain measures candidate supply. "Replay" keeps the deployed full-pool RCAS action when that exact candidate remains available and otherwise returns KEEP; it does not rescore the smaller pool. The four-call subset preserves 91.2–97.0% of full oracle composite gain, and replay on this subset retains 87.9–94.0% of deployed composite gain, and population automatic regression is *lower* than the full-pool rate on all three pools, by 1.9, 1.2, and 2.2 percentage points on WMT23 en–zh, WMT22 en–de, and WMT24 en–zh, respectively. The five additional calls (evidence audits, local repairs, and fusion) therefore contribute

only the remaining 3.0–8.8% of oracle composite gain in these pools. Replay retains these gains without rescoring or refitting on the smaller pool. The full 1/4/6/8/9 curves and document-bootstrap intervals are archived with the analysis. Because Q_ω is dataset-specific, ratios are computed within, not across, datasets. Together with the matched-risk frontiers (Appendix D.6), these results show how candidate supply and selection affect realized gain and regression.

Table 8: **Post-hoc four-call frontier on frozen pools (seed 42).** The four-call subset retains 91.2–97.0% of oracle composite gain; replay retains 87.9–94.0% of deployed composite gain while lowering population automatic regression on every pool. Composite values are four-call/full-pool points; parentheses give the retained percentage. Regression is population automatic regression at four/full calls.

Dataset	Oracle Q_ω	RCAS replay Q_ω	Regression
WMT23 en–zh	11.038/11.385 (97.0%)	5.413/5.756 (94.0%)	**14.2**/16.1%
WMT22 en–de	3.844/3.983 (96.5%)	0.213/0.242 (87.9%)	**9.7**/10.9%
WMT24 en–zh	5.479/6.008 (91.2%)	2.587/2.865 (90.3%)	**11.7**/13.9%

D.6 Matched-risk and matched-coverage frontiers

Theorem 2 gives the minimum conditional regression risk at each feasible edit coverage. We therefore reconstruct post-hoc threshold curves on the four tests whose complete frozen candidate pools are present in the local snapshot. Both RCAS and QE use the same eligibility mask; QE ranks each record's feasible non-KEEP candidates by reference-free QE margin and is then thresholded against KEEP. RCAS is not refit. The WMT23 curves average the three frozen seeds; the other two use the primary seed. WMT24 en–de is omitted because its candidate-level pool is unavailable locally.

At matched coverage, the QE threshold with closest holdout coverage is used. At matched risk, we select the QE point with maximum coverage whose conditional COMET-margin regression does not exceed the deployed RCAS risk. The thresholds are chosen on the same holdout, and the matched-risk choice uses reference outcomes. The comparison is therefore descriptive and gives QE test-set hindsight; it is not a new frozen-policy claim. Intervals use 10,000 source-document resamples with the selected thresholds held fixed.

Table 9: **Post-hoc matched operating points.** At matched risk, RCAS attains 2.8–38.8 percentage points higher coverage than the most permissive QE threshold held to RCAS's own deployed risk; at matched coverage, RCAS attains lower conditional regression and higher composite utility in point estimate on all four pools. Every entry is RCAS minus QE. Quality differences are metric points; Δr and Δc are percentage-point differences in conditional automatic regression and coverage. Bold estimates have intervals excluding zero.

	Matched coverage			Matched risk		
Dataset	ΔCOMET	ΔQ_ω	Δr (pp)	ΔCOMET	ΔQ_ω	Δc (pp)
WMT23 en–zh	**−0.475**	**+1.209**	**−3.9**	**−0.493**	**+0.590**	**+20.0**
WMT22 en–de	−0.387	+0.911	**−7.4**	−0.244	−0.032	**+38.8**
WMT24 en–zh	−0.106	**+0.615**	−0.5	−0.140	**+0.553**	+2.8
WMT23 zh–en	**−0.187**	+0.076	**−2.4**	**−0.155**	+0.083	**+25.2**

The matched operating-point analysis is where RCAS's risk-control advantage becomes most decisive, and it reframes the primary-metric comparison. *At matched risk*, even after granting QE full test-set hindsight to maximize coverage under RCAS's own deployed risk budget, QE still attains 2.8–38.8 percentage points less coverage than RCAS, with three of four coverage intervals excluding zero. *At matched coverage*, RCAS lowers conditional regression in point estimate on all four pools (-0.5 to -7.4 percentage points, intervals excluding zero on three) and raises composite utility in point estimate on all four ($+0.076$ to $+1.209$ points). The price of this risk profile is small and often statistically unresolved: across the eight matched operating points the COMET difference spans only 0.106–0.493 points, with intervals excluding zero on only two of four pools per panel, whereas the regression and coverage advantages exclude zero on three of four pools per panel. The

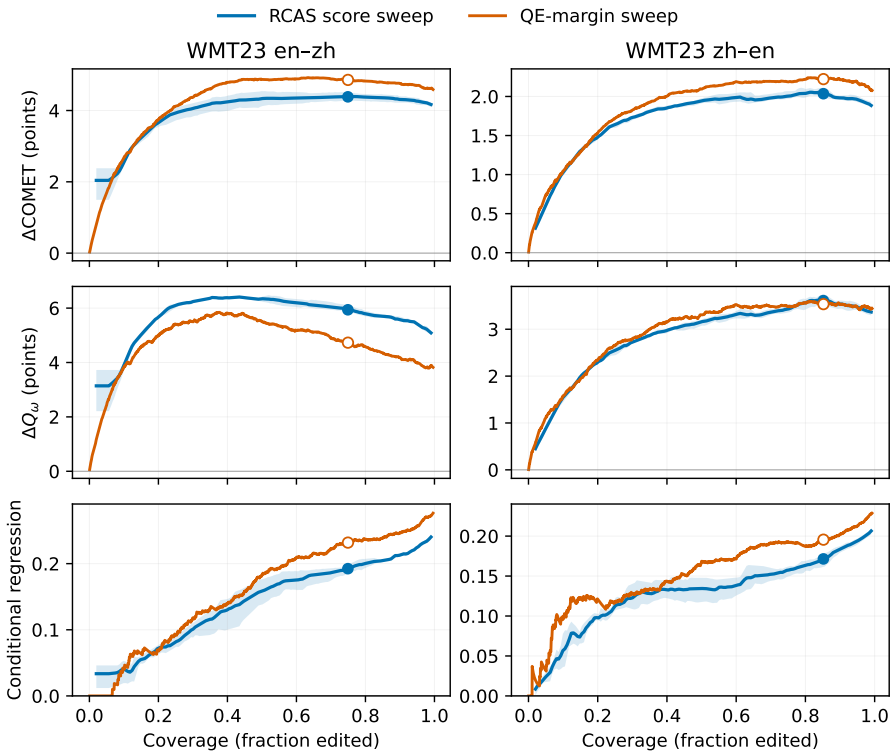


Figure 5: **Descriptive score-sweep frontiers on the bidirectional WMT23 pools.** Solid blue circles are deployed RCAS; open orange circles are QE at matched coverage. Blue bands show the range over three frozen RCAS seeds. Every threshold is swept after the fact; reference outcomes evaluate the curves but are not score inputs. Conditional regression is a COMET drop exceeding 0.01, not human harm.

two rules therefore occupy different regions of the quality–risk frontier: QE spends regression risk to buy COMET, while RCAS spends marginal COMET to buy coverage and lower regression. In deployment, every edit replaces an already usable translation, so the operative cost is damage to content that was already correct—the failure mode that population regression screens for, and which the blinded human audit at the end of this appendix links to human-judged worsening (flagged edits are judged worse on 13.5% of occasions versus 4.5% of unflagged edits). Under that priority ordering, sub-half-point COMET differences are secondary to the multi-point regression and coverage advantages in Table 9. QE remains the stronger COMET maximizer on these pools; the matched frontiers show that this strength is purchased with regression risk that RCAS declines to take.

Historical tune-threshold transfer audit. One earlier WMT23 en–zh tune artifact records QE thresholds before the present test-side frontier analysis. Applying its tune-matched-coverage threshold once to the frozen test pool gives QE/RCAS coverage of 71.4/74.9%. RCAS minus QE is -0.485 COMET points $[-0.873, -0.213]$, $+1.113$ Q_ω points $[+0.503, +1.711]$, and -0.033 conditional regression $[-0.0659, +0.00004]$. Thus the composite/risk direction survives a threshold that did not inspect test, while the COMET trade-off remains. The threshold selected to match risk on tune does not remain risk-matched on test (17.2% QE versus 19.2% RCAS), despite RCAS retaining 27.1 points more coverage. This one-direction retrospective audit is not a new preregistration or a substitute for a prospectively frozen matched-risk study; it shows why prospective comparisons should freeze risk budgets with conservative calibration rather than promise exact test-side matching.

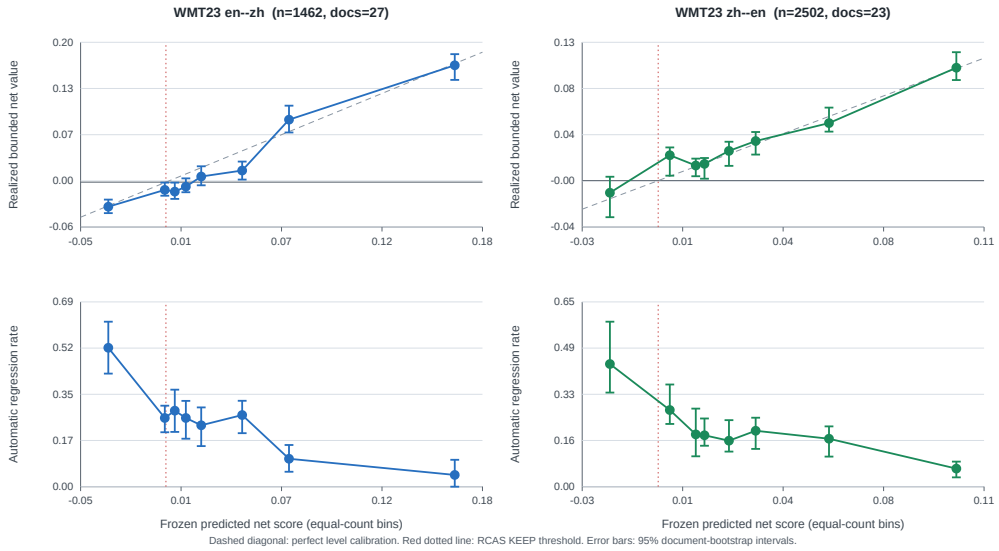


Figure 6: **Frozen score–outcome alignment with all fitted components frozen.** Points are eight equal-count score bins for the top feasible non-KEEP action on each record; error bars are 95% source-document bootstrap intervals (4,000 resamples). The dashed diagonal indicates perfect level calibration, and the red dotted line marks KEEP's zero threshold. Top: realized bounded net value shows a positive association with the frozen predicted score. Bottom: conditional automatic regression concentrates in the lowest-score bins.

D.7 Frozen decision-score alignment

Figure 6 evaluates the deployed net-value scores against realized outcomes with every fitted component frozen: the diagnostic uses the frozen seed-42 forced-edit arm, which retains each record's highest-scoring feasible non-KEEP action before the zero threshold is applied, divides that score into eight equal-count bins, and compares it with the realized bounded net target without refitting any model, threshold, or calibration table. The frozen scores exhibit both properties the net-value rule requires. *Ordering:* mean score and realized net value increase together (record-level Pearson $r = 0.60$ for en–zh and 0.40 for zh–en), and conditional automatic regression concentrates in the lowest-score bins, i.e., in the score region where the rule assigns little or negative value to editing. For level and scale, calibration in the large (realized minus predicted net value) is $-0.0092/+0.0031$, linear calibration slopes are $1.06/0.94$, and the weighted binned absolute gaps are $0.0133/0.0048$ for en–zh and zh–en. Aggregate scale on the executed candidate set is therefore approximately unbiased, although the middle bins are not perfectly monotone, especially on zh–en. Because the frozen records keep only the top feasible revision, this stays a post hoc reliability diagram rather than calibration across all candidates or prospectively. With the stability bound of Section 2.4, the alignment supports the zero threshold acceptance rule but does not certify it: among forced edits, the worst realized outcomes fall on records whose frozen scores sit in the region where the deployed rule retains the current translation.

D.8 Author-code comparisons and seed variation

The pipeline comparisons in Table 1(a) use the author-code adaptations specified in Appendix E.1, run on the shared Qwen3 backbone with frozen initial translations. Under this common setup RCAS attains the highest mean COMET on four of the five datasets, with margins of 0.209–0.712 points over the strongest baseline; on the fifth (WMT23 zh–en) it remains within 0.051 points of the best baseline, a gap smaller than RCAS's own seed standard deviation (0.067). These runs compare complete repair pipelines; the decision layer itself is isolated by the frozen-pool controls in Appendix D.3.

Table 10 reports five-seed variation. RCAS's standard deviations (0.067–0.112) reflect full selector retraining, while baseline deviations reflect generation only; on the four datasets where RCAS leads, its mean margin exceeds its seed standard deviation in every case. As scoped in the caption, these deviations describe repeatability variation and are not confidence intervals for method differences.

Table 10: **Seed variation for the main pipeline comparison.** COMET-22 $\times 100$, mean $\pm$ sample SD ($n = 5$, denominator $n - 1$). RCAS varies selector training; author-code baselines vary generation. RCAS results are aggregated as described in Appendix E.2.

Method	WMT23 en–zh	WMT22 en–de	WMT24 en–zh	WMT24 en–de	WMT23 zh–en
RCAS	86.342 ± 0.112	86.126 ± 0.098	85.687 ± 0.112	81.156 ± 0.093	83.090 ± 0.067
KiC-G	85.111 ± 0.010	85.712 ± 0.017	84.975 ± 0.012	80.725 ± 0.019	82.763 ± 0.011
KiC-X	85.447 ± 0.009	85.750 ± 0.015	84.942 ± 0.013	80.007 ± 0.015	82.809 ± 0.008
TEaR	86.133 ± 0.054	85.657 ± 0.060	84.607 ± 0.056	80.027 ± 0.040	83.141 ± 0.049

Call budget and deployment cost. Call counts reflect structurally different pipelines: nine proposal calls for RCAS, one repair call for KiC, and one Estimate plus at most one Refine call for one-round TEaR before format recovery; they exclude the shared initial translation and are not token- or latency-matched. Two facts put this budget in perspective. First, all RCAS generation uses a single 4B-parameter model (Qwen3-4B-Instruct-2507), and its candidate scoring and selection are non-generative forward passes, whereas KiC-X additionally requires XCOMET-XL annotation inference. Second, the nine-call pool is a research configuration that maps the heterogeneous action space: once the pool exists, the decision layer requires no further LLM generation, and the frozen frontier in Appendix D.5 shows that replay on the four-call subset preserves 87.9–94.0% of deployed composite gain while lowering population regression.

D.9 Human and automatic outcomes

Before adjudication, the two experts agree on 67.6% of overall-preference labels (Cohen's $\kappa = 0.524$) and 81.6% of major-error-presence labels ($\kappa = 0.546$). A third blinded expert resolves all 113 records with a disagreement in either field. The resulting design-weighted estimates in Table 3 use 248 rateable preference comparisons and all 250 major-error comparisons.

The audit also calibrates the automatic outcome used above. After adjudication, experts judge an edit worse on 9/198 (4.5%) automatically unflagged records and 7/52 (13.5%) flagged records, so the flag enriches for regressions; yet 13/52 (25.0%) flagged edits are judged better, so it is a screening signal rather than a human harm rate.

E Experimental protocol and reproducibility

Data and runs. The five document-partitioned test sets are WMT23 en–zh (1,468 records, 27 documents), WMT22 en–de (1,366, 26), WMT24 en–zh (900, 26), WMT24 en–de (1,345, 23), and WMT23 zh–en (2,521, 23). Opportunity and human analyses use seed 42; pipeline and decision tables summarize seeds 42, 7, 13, 17, and 19. RCAS seeds repeat selector training and calibration on fixed candidate pools; author-code baseline seeds repeat generation.

Generation uses Qwen3-4B-Instruct-2507 with an 8,192-token context. References supervise training and calibration and score test outcomes; test-time candidate generation and action selection use the source and initial MT without references. All operating choices use train/tune data.

Metrics and uncertainty. Official human system scores index initial quality within each dataset: DA-SQM for WMT23 en–zh and zh–en, appraise-z for WMT22 en–de, ESA for WMT24 en–zh, and MQM for WMT24 en–de. COMET-22 ("`Unbabel/wmt22-comet-da`") is the primary reporting metric (Rei et al., 2020; 2022); reference-free QE uses "`Unbabel/wmt20-comet-qe-da`" under COMET v2.2.2. The policy is trained on $Q_\omega = \mathrm{COMET22} + \omega\,\mathrm{chrF}/100$, with ω fixed from training-split gain scales. We report COMET-22, Q_ω, BLEU, and chrF; BLEU and chrF use SacreBLEU (Post, 2018; Popović, 2015). Displayed gains are multiplied by 100, and Q_ω is compared within datasets.

An automatic regression is an edit whose raw COMET-22 falls by more than 0.01. We report its rate among edits and over all records, separately from the human major-error outcome. The value rule uses $\lambda = 1$; the COMET-scale margin 0.01 and benefit/degradation caps 0.10 are converted by the frozen tune scale ratio ρ as $m = 0.01\rho$ and $B_{\max} = L_{\max} = 0.10\rho$ (Eq. (69)). Descriptive quality intervals resample source documents 400 times; paired mechanism contrasts use 10,000 resamples. Seed SD uses denominator $n - 1$.

Decision controls. Same-pool comparisons include QE-best with KEEP, fixed families, forced editing, and candidate oracles. On frozen WMT23 en–zh and zh–en pools at five seeds, E1 forces the top feasible revision, E2 fixes accepted records while changing candidate ranking, and E3 compares signed gain, benefit, and benefit-minus-degradation scores at tune-matched coverage. The deployed selections are reproduced candidate-for-candidate before each contrast. The cross-generator analysis applies three original Qwen3 selectors to an existing Qwen2.5-7B-Instruct candidate pool without refitting or recalibration; its intervals resample source documents 10,000 times.

E.1 Author-code pipeline comparisons

Table 1(a) adapts the released Ki–Carpuat and TEaR components (Ki & Carpuat, 2024; Feng et al., 2025) to the shared Qwen3 backbone and frozen initial translations. KiC-G uses the author's generic prompt; KiC-X adds automatic reference-free XCOMET-XL feedback. Both use the author's first-line extraction. TEaR retains the author's zero-shot Estimate prompt, beta Refine prompt, parser, and no-error rule. The shared initial MT replaces its Translate stage; one Estimate–Refine round is used, with no Refine call when Estimate finds no error.

TEaR's reference-independent format audit covered all 38,000 record–seed evaluations. Five formatting exceptions received a deterministic additional Estimate call; all yielded valid no-error assessments under the author's rule. All 25 final runs have complete record coverage and pass author-component replay. KiC and TEaR are scored with the same COMET-22 checkpoint and metric signature, and their seed summaries are computed from complete segment scores.

E.2 Score provenance and reproducibility

The supplementary artifact provides RCAS source code, configuration settings, a SHA-256 manifest, and 25 text-free per-record RCAS selection files covering five datasets and five seeds. The supplied aggregation script reproduces the reported RCAS COMET-22 and Q_ω means, edit rates, automatic regression rates, and seed standard deviations using these records and the included fixed KEEP scores.

E.3 Human semantic audit

Two bilingual experts independently reviewed 250 seed-42 RCAS edits sampled from the 1,144 executed WMT23 en–zh edits. Six strata cross retrospective automatic regression with selected action family. Experts saw the source and both translations in blinded order and judged overall preference and major-error presence; a third expert blindly adjudicated 113 disagreements. A major error materially changes meaning, key facts, or usability rather than style alone. The 12 hidden repeats assess consistency and are excluded from outcome rates. Estimates weight each sampled stratum by its executed population N_s/n_s; intervals resample source documents. Preference uses 248 rateable comparisons, and major-error rates use all 250 edits. Repair and introduction indicate the removal or addition of major-error presence.