HALIGN: Hallucination-Aligned Span Correction for Retrieval-Augmented Generation

Anonymous ACL submission

Abstract

Retrieval-augmented generation (RAG) mitigates hallucination by grounding outputs in retrieved evidence, yet generated responses still frequently diverge from their source. Most prior work stops at *detecting* unfaithful spans and leaves the actual fix to the user. We argue that correction in deployed RAG systems is primarily a *precision* problem: a useful corrector must repair the small fraction of wrong content while leaving correct content untouched; unnecessary edits introduce new errors, break formatting, and inflate review cost. We present HALIGN (Hallucination-Aligned Span Correction), a four-stage pipeline (span detection, type classification, extractive source localisation, evidence-conditioned rewriting) scoped to Data2txt and QA, where hallucinations are correctable from retrieved text. The pipeline pairs a ModernBERT-large detector with a DeBERTa-v3-large QA model that localises evidence and, via its SQuAD 2.0 null-answer score, doubles as a parameter-free type classifier (90.3% vs. 83.6% for a trained MLP). On the 2,870-example RAGTruth test set, HALIGN + Llama-3.2-3B reaches **0.765** edit-decision F1 (balanced accuracy **0.81**) versus 0.545/0.616/0.668 for standalone Llama-3.2-3B, Qwen3.5-4B, and Qwen3.5-122B baselines (all using the same structured prompt) that rewrite 76–81% of all answers. The pipeline improves edit-decision F1 by **+22.1 pp** over same-model Llama standalone, edits only **12% of already-correct answers** (vs. 57–73% for standalones), and runs **3.4× faster** (1.9 s vs. 6.4 s). Document-level faithfulness on hallucinated examples is comparable to same-model standalone (MiniCheck 0.391 vs. 0.377) while the precision advantage is evident to human raters: HALIGN scores **4.90/5** on clean-answer preservation versus 3.54/4.28 for Qwen baselines (§6.2). Oracle analyses confirm detection recall — not correction-model scale — is the binding constraint.

1 Introduction

Retrieval-augmented generation (RAG) reduces hallucination by grounding language model outputs in retrieved documents (Gao et al., 2024), but the grounding is imperfect: generated responses still frequently contradict or fabricate beyond the retrieved evidence (Niu et al., 2024; Huang et al., 2025; Ji et al., 2023). In high-stakes deployments—medicine (Lozano et al., 2023), legal research (Magesh et al., 2024), enterprise knowledge systems—even an occasional hallucination erodes user trust and triggers manual review.

Several systems now address hallucination *detection*—identifying which parts of a generated response are unfaithful to the source. Approaches range from prompting commercial LLMs (Es et al., 2024; Friel and Sanyal, 2023) and training specialised encoder classifiers (Belyi et al., 2025; Zimmerman et al., 2024) to fine-tuning open-weight models on annotated data (Song et al., 2024; Niu et al., 2024). The recent LettuceDetect framework (Kovács and Recski, 2025) showed that a ModernBERT (Warner et al., 2024) encoder with an 8,192-token context window can match or exceed much larger LLM-based detectors on the RAGTruth benchmark, achieving 79.2% example-level F1 while processing 30–60 examples per second on a single GPU.

However, detection alone leaves the user with a highlighted problem but no solution. In practice, a flagged hallucination still requires manual intervention: the user must locate the relevant source passage, determine whether the error can be fixed from the context or must be removed, and rewrite the response accordingly.

We present HALIGN (Hallucination-Aligned Span Correction), a four-stage pipeline scoped to *source-grounded hallucination correction* for RAG, evaluated on RAGTruth. Given a generated response and its retrieved source docu-

ments, HALIGN (i) detects hallucinated token spans, (ii) classifies each span as replaceable (supported by source evidence) or deletable (entirely fabricated), (iii) locates the supporting evidence within the retrieved documents for replaceable spans, and (iv) generates a corrected span conditioned on the extracted evidence using a small instruction-tuned language model. The pipeline is scoped to *extractive correction*: cases where the correct text is derivable from the retrieved source. Data2txt (structured key–value fields) and QA (passage spans) are the primary targets. We deliberately include Summarisation as a *boundary condition*: summarisation hallucinations frequently involve abstractive reframing, negation, or hedging rather than factual substitution, making extractive grounding less applicable. We report Summary results throughout to characterise the scope of the approach and to identify where the method breaks down, not to claim strong performance on that task.

The QA model's built-in SQuAD 2.0 (Rajpurkar et al., 2018) answerability signal already implements the replace-vs-delete decision: if the model deems a hallucinated span "unanswerable" from the source, the span is extrinsic and must be deleted; otherwise, the extracted answer serves as the correction evidence. In our setting this parameter-free signal (90.3% accuracy) outperforms a dedicated trained MLP classifier (83.6%), avoiding an additional training stage at no cost.

Our main contributions are:

1. We observe that a QA model's SQuAD 2.0 answerability signal, already being computed for source localisation, can be reused as a parameter-free replace-vs-delete type classifier (90.3% accuracy, +6.7 pp over a trained MLP), removing the need for a dedicated trained head.
2. We package detection, type classification, evidence localisation, and evidence-conditioned rewriting into a single four-stage pipeline (HALIGN) and evaluate it end-to-end on RAGTruth across three task types (Data2txt, QA, Summarisation). With a 3B open-weight corrector, HALIGN reaches 0.765 edit-decision F1 against 0.545/0.616/0.668 for Llama-3.2-3B (3B), Qwen3.5-4B, and a 122B-parameter (MoE, 10B active) Qwen3.5-122B standalone baseline under the same structured prompt, while editing 36% vs. 76–81% of answers and running 1.9 s vs. 6.4–7.3 s per example — a deployment-relevant operating point that aggressive whole-answer rewriters do not occupy. We release per-stage baselines to support future work on each stage independently.
3. We release per-stage ablations and oracle analyses showing that detection recall, not correction-model scale, is the dominant residual bottleneck: gold detection adds +2.8 pp MiniCheck on hallucinated examples (0.391 → 0.419) and gold detection plus gold source spans adds only +0.5 pp more (0.424).

2 Related Work

Hallucinations in RAG. Hallucinations in LLMs are well documented (Ji et al., 2023; Huang et al., 2025); intrinsic hallucinations contradict the source while extrinsic ones introduce unsupported claims. In RAG both types persist despite grounding: either the model fails to condition on the retrieved context, or it overrides that context with parametric knowledge (Sun et al., 2025). RAGTruth (Niu et al., 2024) annotates both types at the token level across three generation tasks.

Hallucination detection. Prior work falls into three families. *Prompt-based* approaches (Es et al., 2024; Friel and Sanyal, 2023; Manakul et al., 2023) use LLM verification prompts; they avoid task-specific training but incur per-example LLM calls and reach moderate accuracy (GPT-4-turbo: 63.4% F1 on RAGTruth, per Kovács and Recski, 2025). *Encoder-based* methods train specialised classifiers: Luna (Belyi et al., 2025) (65.4%), LettuceDetect (Kovács and Recski, 2025) (ModernBERT (Warner et al., 2024), 79.2%, 30–60 ex/s), and FACTOID (Rawte et al., 2024). *Fine-tuned-LLM* methods reach the strongest detection scores at high inference cost: Niu et al. (2024) (Llama-2-13B, 78.7%) and RAG-HAT (Song et al., 2024) (Llama-3-8B with DPO, 83.9%). All of these stop at detection; we build on LettuceDetect and extend with type classification, evidence localisation, and correction.

Hallucination correction. Correction — automatically rewriting unfaithful content — is comparatively unexplored in the RAG setting. Existing work targets fact correction against external knowledge bases (Thorne and Vlachos, 2021), machine-translation post-editing, or self-correction loops (Self-RAG, Corrective RAG). In RAG the "ground truth" is the specific retrieved documents the model

was conditioned on, and corrections must be traceable to that evidence; prior correction systems do not report on RAG hallucination benchmarks. HALIGN packages detection, type classification, evidence localisation, and evidence-conditioned rewriting into a single pipeline scoped to extractive correction, and reports per-stage baselines on RAGTruth.

Extractive QA and answerability. We cast evidence localisation as extractive QA over the source prompt and reuse the SQuAD 2.0 (Rajpurkar et al., 2018) answerability signal as a zero-cost type classifier. DeBERTa (He et al., 2021) with sliding-window inference outperforms ModernBERT for this stage despite the latter's longer native context (§5.3); we adopt a two-model design accordingly.

3 Method

3.1 Overview

HALIGN processes a RAG response through four sequential stages. Given an LLM response and its retrieved source chunks, the pipeline (i) identifies hallucinated tokens, (ii) classifies each span as *replace* (contradicts source) or *delete* (baseless), (iii) locates supporting evidence in the source for replaceable spans, and (iv) rewrites the response. Figure 1 illustrates the pipeline.

The pipeline uses two specialised encoders and an LLM corrector: ModernBERT-large (Warner et al., 2024) (394.8M params) with a token-level classification head for detection; a DeBERTa-v3-large (He et al., 2021) model fine-tuned for extractive QA (434.0M params) handling type classification and evidence localisation in a single forward pass; and Llama-3.2-3B-Instruct as Stage 4 backend. Total end-to-end budget is ∼3.8B parameters (829.9M for stages 1–3).

Data2txt and QA are the primary targets, with Summarisation included as a boundary condition. We chose specialised models over a single end-to-end architecture: an early cross-attention pointer head trained jointly with detection had only ∼6,800 replace-span signals and underperformed a dedicated QA model pre-trained on SQuAD 2.0 (Rajpurkar et al., 2018).

3.2 Hallucination Detection

We frame detection as token-level binary classification on the answer, encoding input as `[CLS] prompt [SEP] answer [SEP]` following LettuceDetect (Kovács and Recski, 2025). ModernBERT-large is chosen for its long context (RAGTruth averages 793 tokens, 95th percentile 1,410); DeBERTa-v3-large's 512-token limit truncated 75% of examples in an earlier attempt. The head adds 1.05M parameters ($1024 \rightarrow 1024$ dense, GELU, LayerNorm, dropout, $1024 \rightarrow 2$). We initialise from LettuceDetect (Example F1 = 79.2%) and continue training. Detection recall was the binding bottleneck (QA 75.0%, Summary 55.9%), so we augmented training with 1,850 synthetic hallucinations generated by swapping entities and numbers; this lifted QA recall +4.6 pp and Summary +4.4 pp, bringing overall Example F1 to 79.6%. Consecutive hallucinated tokens form contiguous spans for downstream stages.

3.3 Type Classification via QA Answerability

Hallucinations split into two kinds: *intrinsic* (a correct version exists somewhere in the source — replace) and *extrinsic* (no basis in the source — delete). Getting this wrong matters: trying to fix a baseless claim from source yields garbage, and deleting a merely inaccurate one discards recoverable information. We initially trained a dedicated classifier on the detection encoder's hidden states (linear head: 73.6%; two-layer MLP $1024 \rightarrow 256 \rightarrow 2$: 83.6%), but the approach that worked best required no additional training at all. The Stage 3 DeBERTa QA model is pre-trained on SQuAD 2.0, which explicitly includes unanswerable questions, so its forward pass already produces an answerability signal. We compare null_score $= s_0 + e_0$ against best_score $= \max_{i,j}\ s_i + e_j$ (where s_i, e_i are start/end logits and position 0 is `[CLS]`); if null_score $>$ best_score, the model deems the span unanswerable and we delete it, otherwise we replace. This signal reaches 90.3% accuracy overall (91.5% D2T, 91.8% QA, 83.1% Summary), outperforming the dedicated MLP by +6.7 pp at zero extra parameter or training cost. The lower Summary accuracy reflects the blurred line between unsupported inference and source contradiction in that task.

3.4 Evidence Localization

For spans classified as *replace*, we locate the supporting passage in the source chunks by casting it as extractive QA: the hallucinated text is the question and the full source prompt is the context. We fine-tune `deepset/deberta-v3-large-squad2` (He et al., 2021) on QA pairs derived from the Qwen-bootstrapped source spans of §4.1 (ques-

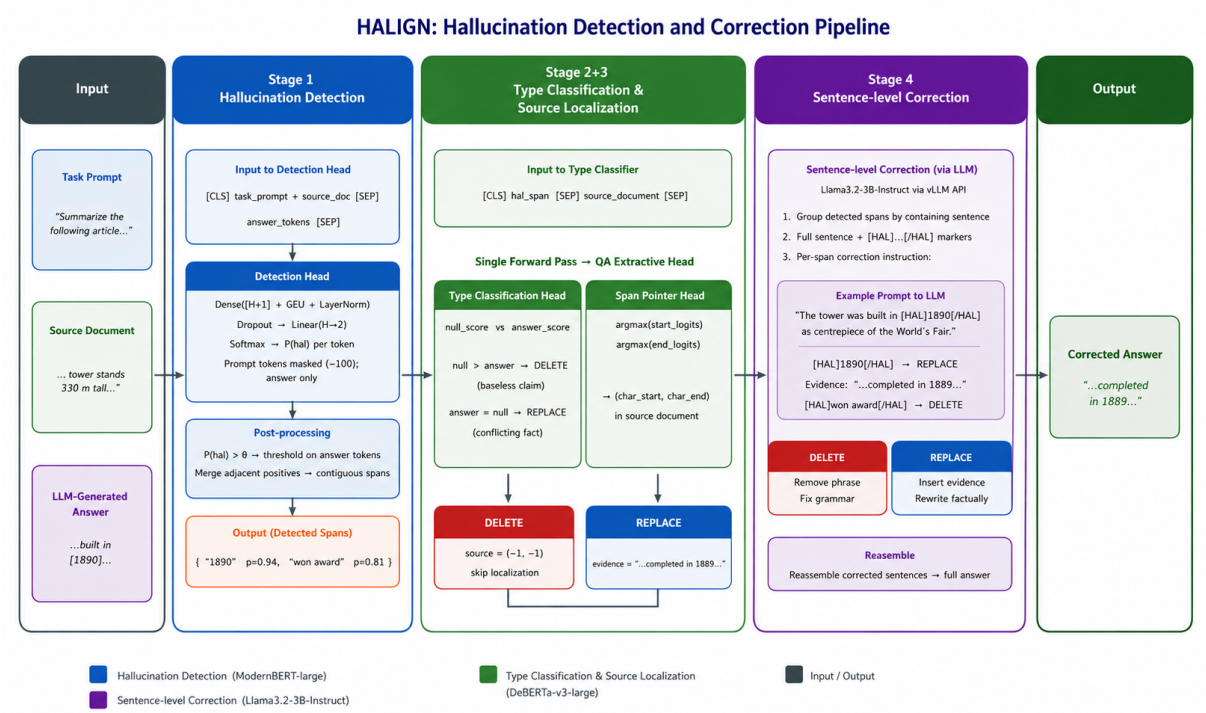

Figure 1: The HALIGN pipeline. Given a RAG context and LLM response, Stage 1 detects hallucinated tokens with a ModernBERT-large classifier. Stage 2 classifies each span as replace or delete using the QA model's null-score answerability signal. Stage 3 locates supporting evidence in the source with a DeBERTa-v3-large QA span finder and post-processing heuristics. Stage 4 rewrites the full answer with detected spans marked, using Llama-3.2-3B conditioned on the extracted source evidence (the diagram shows a generic LM icon; the deployed model is Llama-3.2-3B-Instruct).

tion = hallucinated text, answer = cleaned source span); this is the same model used for type classification, both signals coming from a single forward pass. RAGTruth source prompts often exceed 512 tokens (especially Data2txt JSON), so we use sliding-window inference with stride 128 and keep the span with the highest aggregate start+end logit across windows. For Data2txt, raw QA predictions frequently clip at wrong KV boundaries (extra quote, missing comma, mid-value cut); we apply two inference-time fixes mirroring the annotation cleanup: (i) snap each prediction to the nearest `'key': value` pair if their IoU exceeds 0.3, and (ii) expand across contiguous same-value fields when at least three adjacent fields match and the hallucinated text is >40 characters. The stage returns character offsets of the supporting evidence; delete spans bypass it entirely.

3.5 Correction

The final stage rewrites the full answer conditioned on Stage 3 evidence. The model sees the full answer with detected spans wrapped in `[HAL]...[/HAL]` tags plus a per-span instruction (replace with provided evidence, or delete), and emits the corrected answer in a single generation call; passing the whole answer as context lets the model avoid duplicating claims already stated elsewhere. The prompt instructs the model to incorporate replace-evidence naturally rather than copy it verbatim, remove delete spans cleanly, and leave non-hallucinated text untouched. For Data2txt the evidence is a structured KV pair, so the prompt adds a single few-shot example demonstrating the marker-stripped rewrite (`'OutdoorSeating': False` → "does not offer outdoor seating"). The deployed Stage 4 backend is Llama-3.2-3B-Instruct; as comparison baselines we evaluate Llama-3.2-3B-Instruct and Qwen3.5-4B standalone, and Qwen3.5-122B (Qwen Team, 2026) as an aggressive-rewrite reference.[1] Full prompts in Appendix J.

[1] Qwen3.5-122B is an MoE model with 10B active / 122B total parameters; the 41× ratio cited elsewhere is total-parameter, the active-parameter ratio against HALIGN's 3.8B is 2.6×.

4 Experimental Setup

4.1 Dataset

We evaluate on RAGTruth (Niu et al., 2024), a benchmark of LLM-generated responses grounded in retrieved documents, spanning three task types: structured data-to-text generation (**Data2txt**), open-domain question answering (**QA**), and document summarisation (**Summary**). Each response is annotated with token-level hallucination spans and a label type (Evident Conflict, Subtle Conflict, Evident Baseless Info, Subtle Baseless Info). RAGTruth does *not* provide source span annotations or correction types; we generate these using Qwen3.5-122B as described below.

The full corpus contains 16,940 train and 2,870 test examples roughly evenly split across the three tasks; the test set has 1,703 annotated hallucination spans (1,054 Data2txt, 355 QA, 294 Summary). Of the 14,543 training-set spans, 6,824 are replaceable from source and 7,719 must be deleted. Per-task statistics are in Appendix I.

Source span annotation. Since RAGTruth provides no source span annotations, we use Qwen3.5-122B (Qwen Team, 2026) to generate them. For each hallucinated span, the LLM is prompted with the source document, the response, and the hallucinated text, and asked to (i) classify whether the span should be replaced (the source contains a correct version) or deleted (the claim is entirely fabricated), and (ii) for replace spans, copy the exact contiguous substring from the source prompt that contains the correct information. This yields 6,824 replace spans with associated source substrings and 7,719 delete spans.

For Data2txt, where source prompts are structured key–value dictionaries, Qwen occasionally copies slightly off-boundary substrings (e.g., missing an opening quote, including a trailing comma). Because these artifacts arise from our annotation generator rather than from the downstream task itself, we post-process the generated spans with a simple KV-boundary snap: each generated source span is snapped to the nearest complete `'key': value` pair boundary in the prompt when their overlap (IoU) exceeds 0.3. This is a data-cleanup step on machine-generated labels — specific to structured-prompt Data2txt and to the choice of using an LLM to bootstrap source-span annotations — not a generalisable technique; we report its effect purely as an ablation on our training data in §5.3.

Additionally, 1,850 synthetic hallucination examples (1,100 QA + 750 Summary) are generated via entity and number swapping to improve detection recall on under-represented task types.

4.2 Training Details

Stage 1: Detection. We initialise from LettuceDetect (Kovács and Recski, 2025) weights (ModernBERT-large fine-tuned for token-level hallucination detection) and further fine-tune on RAGTruth with the 1,850 synthetic examples. Training uses a learning rate of 5×10^{-6}, batch size 8, 10 epochs with early stopping at epoch 4 (patience 4), and 10% linear warmup.

Stage 2/3: QA Span Locator. We fine-tune `deepset/deberta-v3-large-squad2` (He et al., 2021; Rajpurkar et al., 2018) on extractive QA pairs derived from the Qwen-generated and KV-boundary-snapped source span annotations (question = hallucinated text, context = source prompt, answer = cleaned source span). Training uses a learning rate of 2×10^{-5}, effective batch size 8, max sequence length 512 with stride 128 for sliding-window inference, and runs for 10 epochs. The same model simultaneously serves as the type classifier via its null-score signal (§3.3).

Stage 4: Correction. The correction stage is modular and not trained end-to-end with the detection/localisation components. The deployed system uses Llama-3.2-3B-Instruct conditioned on pipeline-localised evidence. As comparison baselines we evaluate three standalone correctors that receive the source and answer without pipeline evidence: Llama-3.2-3B-Instruct (same model as the pipeline corrector), Qwen3.5-4B, and Qwen3.5-122B (Qwen Team, 2026) as an aggressive-rewrite reference. Prompts are listed in Appendix J.

4.3 Evaluation Metrics

We report the following metrics:

- **Detection F1**: example-level and character-span F1 for hallucination detection, following the evaluation protocol of Kovács and Recski (2025).
- **Type Accuracy**: fraction of detected spans correctly classified as replace vs. delete.
- **Source Span EM**: exact match between the predicted source span and the gold annotation (for replace spans).
- **Document-level MiniCheck Faithfulness**: $P(\text{source supports answer})$ for the full cor-

Table 1: Stage-wise performance for stages 1–3 on RAGTruth test (correction reported separately in §6.1).

Stage	Metric	D2T	QA	Sum.	All
Detection	Ex. F1	88.2	75.5	63.6	79.6
	Char F1	58.7	62.9	43.6	57.7
Type cls.	Acc	91.5	91.8	83.1	90.3
Source span	EM	67.2	74.3	45.1	69.2

rected response, using Bespoke-MiniCheck-7B (Tang et al., 2024), a 7B NLI model that determines whether all claims in the answer are substantiated by the source. This metric captures whether the *whole* response becomes more source-faithful after correction, not just the modified spans.

5 Results

5.1 Stage-wise Results

Table 1 reports performance across all pipeline stages on the 2,870-example RAGTruth test set (1,703 annotated hallucination spans). Detection achieves 79.6% example-level F1, closely matching the LettuceDetect baseline (79.2%); the QA null-score type classifier reaches 90.3% accuracy at zero training cost; and source span finding achieves 69.2% exact match with Qwen-generated and KV-boundary-snapped training spans (§5.3). Detection is the primary bottleneck of the cascade: each downstream stage operates on a strict subset of spans passed forward from Stage 1, and missed detections account for the largest share of end-to-end failures (§6.2). Correction performance is evaluated in §6.1 via document-level faithfulness scoring.

5.2 Comparison with Prior Work

Detection is not a novel contribution of this work — we reuse the public LettuceDetect ModernBERT-large checkpoint — but our augmentation step matters for the downstream cascade: fine-tuning with 1,850 synthetic examples lifts QA F1 by +5.3 pp (70.2 → 75.5) and Summary F1 by +4.7 pp (58.9 → 63.6) over LettuceDetect-large, driven by recall gains of +4.6 and +4.4 pp respectively (§3.2). These gains reduce the fraction of hallucinations that never reach stages 2–4; overall F1 moves from 79.2% to 79.6%, and Data2txt F1 is essentially flat (88.2 vs. LettuceDetect-large's 88.5). RAG-HAT (Song et al., 2024) reaches 83.9% F1 with a fine-tuned 8B LLM but is 20× larger; our focus is on what happens *after* detection. A full per-task comparison against prior detectors is in Appendix D. All prior methods stop at detection; HALIGN additionally classifies, localises, and corrects, and we report per-stage baselines for each downstream stage on RAGTruth.

5.3 Ablation Studies

Type classifier comparison. We compare three approaches to delete-vs-replace classification on the test set: a linear head on the detector's hidden states (73.6%), a two-layer MLP $1024 \to 256 \to 2$ (83.6%), and the QA model's built-in null-score signal (90.3%, zero extra parameters). The null-score signal — which reflects SQuAD 2.0 answerability training — outperforms both trained heads (full table in Appendix B). The error pattern is asymmetric: the bulk of mistakes come from delete spans misclassified as replace (the QA model finds plausible but irrelevant evidence) rather than the reverse.

Source span annotation quality. Because RAGTruth provides no source span labels, we bootstrapped them with Qwen3.5-122B (§4.1). On Data2txt, 37–43% of generated spans clipped across structured KV boundaries; KV-boundary snapping cleaned these artifacts and produced the largest EM gain from any single data-side change. This is an ablation on our generated training labels, not a general localisation technique, but the effect is large enough to document.

DeBERTa vs. ModernBERT for QA. Despite ModernBERT's 8,192-token native context, DeBERTa-v3 with 512-token sliding windows outperforms it by 17.3 pp on source span EM (69.2% vs. 51.9%). The DeBERTa architecture's disentangled attention and SQuAD 2.0 pretraining dominate the benefit of longer context. This negative result led us to adopt the two-model design (ModernBERT for detection, DeBERTa for QA).

6 Analysis

6.1 Faithfulness Analysis

We frame correction as a precision problem: repair what is wrong, leave what is correct untouched. We report two complementary metrics: *edit-decision F1* (Table 2), which scores whether and where a system edits, and *document-level MiniCheck* (Table 3), which scores $P(\text{source supports answer})$ via Bespoke-MiniCheck-7B (Tang et al., 2024). MiniCheck captures global consistency but rewards

Table 2: Edit-decision (example-level) and edit-region (character-level) metrics. Bal-Acc = balanced accuracy. CleanFP = fraction of clean examples with any edit. **Bold**: best among deployable systems. †Ablation: detection output with spans deleted and no LLM rewriting (produces incomplete answers); isolates detection quality from correction quality.

System	P	R	F1	Bal-Acc	CleanFP
Edit decision (binary)					
HALIGN + Llama-3.2-3B	**.797**	.736	**.765**	**.808**	**11.8%**
Llama-3.2-3B (stand.)	.411	.809	.545	.538	73.3%
Qwen3.5-4B (stand.)	.459	.938	.616	.618	69.3%
Qwen3.5-122B (stand.)	.505	**.987**	.668	.687	57.4%
Edit decision — ablation					
Det.+delete only†	.789	.797	.793	.831	13.5%
Edit region (character-level)					
HALIGN + Llama-3.2-3B	**.482**	.322	**.386**	—	—
Llama-3.2-3B (stand.)	.075	.364	.124	—	—
Qwen3.5-4B (stand.)	.101	.642	.174	—	—
Qwen3.5-122B (stand.)	.171	**.664**	.272	—	—
Edit region — ablation					
Det.+delete only†	.620	.632	.626	—	—

aggressive paraphrase of already-correct content, so edit-decision F1 carries the load.

Edit decision and edit region. On the 2,870-example test set (1,113 hallucinated, 1,757 clean), we score an example-level binary edit decision against the hallucinated/clean label and a character-level region match (edit regions via `difflib.SequenceMatcher`) against gold span annotations. Precision is over all 2,870 examples; recall is over the 1,113 hallucinated.

HALIGN + Llama-3.2-3B attains 0.765 edit-decision F1 (balanced accuracy 0.81) and 0.386 edit-region F1, both above all three standalone baselines including the 41×-larger Qwen3.5-122B. Standalones reach high recall (0.81–0.99) but precision collapses (0.41–0.51 on the binary decision, 0.08–0.17 on edit regions) because most of their edits land on already-clean answers (57–73% CleanFP). The Llama standalone is the cleanest comparison: same base model and same structured prompt as the Qwen baselines, but without the pipeline's detector or localised evidence it edits 76% of answers at a near-chance 0.54 balanced accuracy. Wrapping the HALIGN pipeline around that same corrector lifts edit-decision F1 by +22.1 pp (0.545 → 0.765), edit-region F1 by +26.2 pp, and edit-region precision by +40.7 pp. The corrector model still matters — Qwen-4B and Qwen-122B outscore standalone Llama-3.2-3B on hallucinated MiniCheck (Table 3) — but the pipeline benefit over the *same-model, same-prompt* standalone holds, suggesting that evidence localisation drives a precision improvement largely independent of corrector size.

A *detect-and-delete* ablation that deletes the pipeline's predicted spans without any LLM rewriting reaches 0.793 edit-decision F1 and 0.626 edit-region F1 — both higher than HALIGN + Llama — confirming that detection drives targeting precision. The +24 pp edit-region gap comes from incidental LLM rewriting churn around the targeted span and from replace-type spans requiring new evidence-conditioned text. Delete-only output is not deployable (replace spans leave semantic gaps), so we evaluate the LLM rewrite via MiniCheck and human ratings (§6.1, §6.2).

Why edit footprint matters. Each unnecessary rewrite is a new generation that can introduce errors on content the user had no reason to doubt, breaks stylistic consistency, and doubles the audit surface. HALIGN modifies only **36%** of all examples — less than half any standalone LLM baseline — at 3.4× lower latency than the same Llama corrector standalone (Appendices F, J.3).

Document-level MiniCheck. Edit-decision F1 measures *whether and where* a system edits, but not whether the rewritten answer is well-supported by source as a whole document. We therefore also report document-level faithfulness with Bespoke-MiniCheck-7B (Tang et al., 2024). Table 3 restricts to the 1,113 hallucinated test examples (on clean examples all methods equal the original) and compares the original answers, HALIGN + Llama-3.2-3B, three structured LLM baselines (Llama-3.2-3B, Qwen3.5-4B, Qwen3.5-122B without pipeline evidence), and two oracle HALIGN variants. MiniCheck has a known bias: it rewards source-faithful paraphrase and does not penalise unnecessary edits, so aggressive whole-answer rewriters score artificially high (as the precision collapse in Table 2 makes explicit).

The pipeline raises MiniCheck on hallucinated examples from 0.240 to 0.391 (+15.1 pp); oracle detection takes it to 0.419 and oracle detection+localisation to 0.424. On the 1,757 clean examples it moves only +0.3 pp despite false-positive detections, confirming that evidence-grounded edits preserve quality under imperfect detection. Against the *same* Llama-3.2-3B used standalone, HALIGN is comparable on document-level MiniCheck (0.391 vs. 0.377; paired bootstrap

Table 3: Document-level MiniCheck on the 1,113 hallucinated test examples (clean examples equal the original). "Mod." = fraction of all 2,870 examples whose output differs from the original. Standalone baselines use the same structured prompt without pipeline evidence. Oracle variants and full Δ breakdown in Appendix C.

Condition	Mod.	D2T	QA	Sum.	All
Original	—	.188	.284	.309	.240
HALIGN + Llama-3.2-3B	36%	.283	.526	.487	.391
Llama-3.2-3B (standalone)	76%	.287	.518	.429	.377
Qwen3.5-4B (standalone)	81%	.421	.683	.620	.533
Qwen3.5-122B (standalone)	76%	.413	.725	.675	.551

95% CI [-0.2, +2.9] pp, $B = 10{,}000$) while editing 36% of examples vs. 76%. The Qwen baselines score higher on raw MiniCheck by rewriting aggressively from source (R = 98.5–99.2%, P = 44.1–49.9%), inflating the metric without correcting anything; Table 2 captures the precision side of this trade-off.

Bootstrap CIs. The **+22.1 pp edit-decision F1** gain over same-model standalone Llama has bootstrap 95% CI [+19.5, +24.5] pp ($B = 10{,}000$), clear of zero by a wide margin. The **+1.4 pp MiniCheck** delta has CI [-0.2, +2.9] pp; per task, it is null on extractive targets (Data2txt [-2.2, +1.4]; QA [-2.8, +4.5]) and positive only on Summarisation ([+2.4, +9.3]). MiniCheck cannot distinguish a surgical edit from a broad paraphrase on extractive tasks; edit-decision F1 and the clean-answer human-eval gap (HALIGN 4.90 vs. 3.54/4.28 for Qwen-4B/122B, Table 4) capture that precision difference. Oracle ablation pinpoints the residual gap: gold detection alone lifts HALIGN from 0.391 to 0.419, implicating detection recall as the dominant bottleneck.

6.2 Error Analysis

Failures cascade through the pipeline in stage order. Detection misses dominate: QA recall is 79.6% and Summary 60.3%, so 20–40% of hallucinated examples never reach downstream stages. Type confusion concentrates in Summary (83.1% vs. 91–92% elsewhere), where the line between a fabricated claim and an unsupported inference is genuinely fuzzy; source-span errors are also heaviest there (45.1% EM), where evidence is scattered across long passages.

Out-of-domain (HaluEval). Zero-shot, HALIGN transfers well to QA (F1 = 0.736; MiniCheck 0.274 $\rightarrow$ 0.448, ceiling 0.743) and weakly to Summary (F1 = 0.485; 0.357 $\rightarrow$ 0.420, ceiling 0.642) (Li et al., 2023), mirroring the within-domain pattern (Appendix E).

Human evaluation. Three NLP/CS graduate students, blind to system identity, rated 50 examples (37 hallucinated, 13 clean) on a 1–5 faithfulness scale; inter-rater reliability was substantial (Krippendorff's $\alpha = 0.696$). HALIGN + Llama-3.2-3B leads on clean answers (4.90 vs. 3.54/4.28 for Qwen-4B/122B) — the axis that most directly tests over-editing. On hallucinated examples HALIGN trails (2.86 vs. 3.08/4.07), mirroring the $\sim$20% detection miss; 122B leads overall (4.13) through near-perfect recall. Full table in Appendix A.

7 Conclusion

HALIGN shows that a small open-weight model in a precision-oriented pipeline can deliver the editing behaviour deployed RAG systems need: surgical edits on the answers that need them, silence on those that do not. On RAGTruth, HALIGN + Llama-3.2-3B reaches 0.765 edit-decision F1 versus 0.545/0.616/0.668 for Llama-3.2-3B, Qwen3.5-4B, and Qwen3.5-122B standalone, edits 36% of answers vs. 76–81%, and runs 3.4$\times$ faster than standalone Llama (1.9 s vs. 6.4 s). The +22.1 pp gain over the same Llama corrector (CI [+19.5, +24.5]) isolates the pipeline's contribution: localised evidence and a span-level interface let a 3B model act as a careful editor rather than an aggressive rewriter, an operating point human raters prefer on clean answers (4.90/5 vs. 3.54/4.28). The QA model's SQuAD 2.0 answerability signal implements replace-vs-delete at zero parameter cost (+6.7 pp over a trained MLP). Oracle analyses show detection recall and source-span quality are the binding constraints: stronger detectors yield more gain than scaling correction.

Limitations

Extractive scope. The pipeline is built around the assumption that a hallucinated span can be repaired by inserting a contiguous substring lifted from the retrieved source. This assumption holds well for structured Data2txt fields and for QA, but breaks down for the kind of corrections that summarisation often needs: hedging an over-confident claim, negating a claim that the source does not support, or simply removing an editorial flourish that is plausible but unattested. Our Summary results (45.1% source-span EM; failure mode analysis in

Appendix H) reflect that mismatch, and we deliberately frame Summary as a boundary condition rather than a target task.

Detection is the binding constraint. The oracle analysis (§C) shows that, relative to predicted-detection HALIGN + Llama-3.2-3B (0.391 MiniCheck), gold detection adds +2.8 pp (0.391 → 0.419) and gold source spans add a further +0.5 pp (0.424); the gap to the Qwen3.5-122B paraphrase baseline (0.551, an aggressive-rewrite reference rather than a faithful-edit ceiling) is larger than either of these contributions, suggesting the detector is the dominant bottleneck but not the only one within the precision-preserving regime. In practice this means the system's headline numbers are bounded by what a 395M-parameter token classifier can recall from a single ModernBERT pass; we did not explore stronger detectors (multi-pass, retrieval-augmented, or LLM-based) and the gains from such directions are likely larger than further tuning of stages 2–4.

Single benchmark and language. We evaluate on RAGTruth and (out-of-domain) on HaluEval. Both are English-only and both reflect a snapshot of the kinds of hallucination patterns that current English LLMs produce. Multilingual RAG, domain-specific corpora (medical, legal, scientific), and longer multi-document contexts may exhibit failure modes that this evaluation does not exercise. We have not measured calibration on retrieval failures, where the retrieved documents themselves are incorrect or stale — the system will faithfully "correct" the answer to match a wrong source.

Label provenance. RAGTruth provides hallucination spans but no source-span annotations, so we bootstrap source spans with Qwen3.5-122B and clean structured-data artifacts with KV-boundary snapping (§4.1). The localiser's ceiling is therefore set by the quality of these machine-generated labels rather than by human annotation. A subset of disagreements between our model and the labels may reflect label noise rather than model error, and our reported EM numbers should be read in that light.

MiniCheck as a faithfulness proxy. Document-level MiniCheck rewards answers whose remaining claims are entailed by the source, but it does not penalise rewriting already-correct content into source-paraphrased prose. This is why the LLM baselines outscore us on raw MiniCheck even though they edit 2× more answers; we view this as a property of the metric rather than the systems, and report edit-decision F1 and edit-region precision alongside MiniCheck to make this trade-off explicit. Concretely, the +1.4 pp MiniCheck gain of HALIGN over same-model standalone Llama-3.2-3B is within bootstrap noise on the extractive tasks the pipeline targets (Data2txt 95% CI [−2.2, +1.4]; QA [−2.8, +4.5]) and only clearly positive on Summarisation ([+2.4, +9.3]); document-level entailment alone does not separate a surgical edit from a fluent paraphrase on the extractive tasks, which is why our headline claim rests on edit-decision F1 (CI [+19.5, +24.5] pp) and human evaluation rather than on MiniCheck.

Stage 4 backend. We evaluated only Llama-3.2-3B-Instruct as the Stage 4 correction backend. Whether the pipeline's precision benefit transfers to stronger correctors (e.g. Qwen3.5-4B in the pipeline role) is an open question; if a stronger corrector already edits more conservatively under structured prompting, the marginal gain from pipeline evidence may be smaller.

Latency measurements. The 1.9 s end-to-end latency we report is measured on a single A100 with 8-way concurrent requests to the vLLM-served corrector (§F); we apply the same methodology to all baseline correctors so the comparison is consistent. Production deployments routinely use throughput acceleration techniques (continuous batching, MTP, speculative decoding); the absolute numbers will shift depending on the deployment setup. The relative comparison to a same-model standalone baseline is the load-bearing claim, not the absolute seconds.

Ethical Considerations

Intended use. HALIGN is intended as a post-hoc faithfulness layer for RAG systems whose outputs are already grounded in retrieved documents. It is not a substitute for retrieval quality, document curation, or human review in high-stakes settings. A corrected answer that is faithful to a retrieved source can still be wrong about the world if the retrieved source is wrong; we make no claim about the truthfulness of the underlying corpora.

Risk of mis-correction. Any automatic editor can introduce new errors. We chose a precision-oriented design partly to limit this risk: the system declines to edit on roughly 64% of test ex-

amples, and when it does edit it touches a small region rather than rewriting the surrounding sentence. False-positive detection still occurs (12.0% of clean examples receive at least one flag), and in those cases the Stage 4 correction can substantively rewrite content that was already correct — the replacement may be source-faithful but is still an unintended edit. For deployments where an unintended edit is costly — medical, legal, regulated domains — we recommend a human review step on every flagged span before surfacing the corrected answer.

Models and data. All components are built on publicly released artefacts: RAGTruth (CC BY 4.0), HaluEval (MIT), ModernBERT-large, DeBERTa-v3-large, the LettuceDetect detector checkpoint, Llama-3.2-3B-Instruct for pipeline correction, and Qwen3.5 (4B and 122B) for the baseline correction comparison and synthetic label generation. Synthetic hallucination injections were derived from the same RAGTruth answers by perturbing entities and numbers; they do not introduce new personal information. We did not collect new human annotations for this work.

Compute footprint. Training stages 1 and 3 each ran for under 24 hours on a single A100. Stage 2's classifier is parameter-free and requires no additional training. Synthetic data generation and source-span bootstrapping used Qwen3.5-122B via a hosted endpoint; we estimate the one-time labelling cost at approximately 60 GPU-hours of large-model inference. Inference for the deployed pipeline runs at 1.9 s per example on a single A100 and skips the heaviest stages on the majority of clean inputs, which lowers the per-example compute relative to whole-answer LLM rewriting.

Use of AI writing assistance. We used a general-purpose LLM-based assistant to paraphrase and copy-edit portions of the manuscript for clarity and concision. All technical claims, experimental results, numbers, citations, and conclusions are the authors' own; the assistant was not used to generate research ideas, design experiments, analyse data, or write substantive content.

Reproducibility. Code, model checkpoints, and the post-processed RAGTruth derivatives needed to reproduce the reported numbers are released at the following anonymous links: code at https://anonymous.4open.science/r/halign-27E3/, model checkpoints at https://anonymous-hf.up.railway.app/a/5ahc0yh3kj6m/, and training data at https://anonymous-hf.up.railway.app/a/b7fryyr6lfd1/. We follow the Responsible NLP Checklist accompanying this submission.

References

Masha Belyi, Robert Friel, Shuai Shao, and Atindriyo Sanyal. 2025. Luna: A lightweight evaluation model to catch language model hallucinations with high accuracy and low cost. In *Proceedings of the 31st International Conference on Computational Linguistics: Industry Track (COLING)*, pages 398–409.

Shiqi Chen, Yiran Zhao, Jinghan Zhang, I-Chun Chern, Siyang Gao, Pengfei Liu, and Junxian He. 2023. FELM: Benchmarking factuality evaluation of large language models. *Advances in Neural Information Processing Systems*, 36.

Shahul Es, Jithin James, Luis Espinosa Anke, and Steven Schockaert. 2024. RAGAS: Automated evaluation of retrieval augmented generation. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations (EACL)*, pages 150–158.

Robert Friel and Atindriyo Sanyal. 2023. ChainPoll: A high efficacy method for LLM hallucination detection. *Preprint, arXiv:2310.18344*.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-augmented generation for large language models: A survey. *Preprint, arXiv:2312.10997*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. DeBERTa: Decoding-enhanced BERT with disentangled attention. *Preprint, arXiv:2006.03654*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12).

Ádám Kovács and Gábor Recski. 2025. LettuceDetect: A hallucination detection framework for RAG applications. In *Preprint, arXiv:2502.17125*.

Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yong Nie, and Ji-Rong Wen. 2023. HaluEval: A large-scale hallucination evaluation benchmark for large

language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6449–6464.

Alejandro Lozano, Scott L. Fleming, Chia-Chun Chiang, and Nigam Shah. 2023. Clinfo.ai: An open-source retrieval-augmented large language model system for answering medical questions using scientific literature. *Preprint, arXiv:2310.16146*.

Varun Magesh, Faiz Surani, Matthew Dahl, Mirac Suzgun, Christopher D. Manning, and Daniel E. Ho. 2024. Hallucination-free? assessing the reliability of leading AI legal research tools. *Preprint, arXiv:2405.20362*.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9004–9017.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, Kai Zang, Yang Zheng, Juntong Song, Randy Zhong, KaShun Shum, Tong Song, and Tong Zhang. 2024. RAGTruth: A hallucination corpus for developing trustworthy retrieval-augmented generation language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 10862–10878.

Qwen Team. 2026. Qwen3.5 language models. https://huggingface.co/Qwen/Qwen3.5-4B. Model series available at https://huggingface.co/Qwen; blog: https://qwen.ai/blog?id=qwen3.5.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 784–789.

Vipula Rawte, S. M. Towhidul Islam Tommoy, Krishnav Rajbangshi, Nag Shravani, Chadha Ama, Amit P. Sheth, and Amitava Das. 2024. Factoid: A framework for factual entailment for hallucination detection. In *Preprint, arXiv:2403.19113*.

Juntong Song, Xingguang Wang, Juno Zhu, Yuanhao Wu, Xuxin Cheng, Randy Zhong, and Cheng Niu. 2024. RAG-HAT: A hallucination-aware tuning pipeline for LLM in retrieval-augmented generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track (EMNLP)*, pages 1548–1558.

Zhongxiang Sun, Xiaoxue Zang, Kai Zheng, Yang Song, Jun Xu, Xiao Zhang, Weijie Yu, Yang Song, and Han Li. 2025. Redeep: Detecting hallucination in retrieval-augmented generation via mechanistic interpretability. *Preprint, arXiv:2410.11414*.

Liyan Tang, Philippe Laban, and Giuseppe Carenini. 2024. MiniCheck: Efficient fact-checking of LLMs on grounding documents. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8818–8847.

James Thorne and Andreas Vlachos. 2021. Evidence-based factual error correction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 3298–3309.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *Preprint, arXiv:2412.13663*.

Ilana Zimmerman, Jadin Tredup, Ethan Selfridge, and Joseph Bradley. 2024. Two-tiered encoder-based hallucination detection for retrieval-augmented generation in the wild. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track (EMNLP)*, pages 8–22.

Table 4: Human evaluation (50 examples × 3 annotators = 150 ratings). Row n = ratings. The HALIGN column is HALIGN + Llama-3.2-3B; Qwen 4B and 122B are the same standalone correctors as in Tables 2–3.

Condition	HALIGN+Llama	Qwen 4B	Qwen 122B
Hallucinated ($n = 111$)	2.86	3.08	**4.07**
Non-hallucinated ($n = 39$)	**4.90**	3.54	4.28
Overall	3.39	3.20	**4.13**

A Human Evaluation Protocol and Table

Three graduate students in NLP/CS, unaffiliated with this project, were blind to system identity and rated 50 examples (37 hallucinated, 13 clean; oversampled toward hallucinated) on a 1–5 faithfulness scale. System order was independently randomised per example, and raters saw the source, the original answer, and the three corrected outputs side-by-side. Inter-rater reliability was substantial (Krippendorff's $\alpha = 0.696$ ordinal, quadratic $\kappa = 0.694$); row n in Table 4 counts ratings, not examples.

B Type Classifier Ablation

Table 5 compares the three delete-vs-replace classifiers summarised in §5.3.

C Oracle Detection Analysis

To isolate each stage from detection errors, we feed *gold* hallucination spans directly into stages 2–4 (Table 6). Source-span EM under oracle detection is 61.0%, slightly below the 69.2% obtained

Table 5: Type classification ablation. Accuracy on the test set.

Classifier	Accuracy	Extra params
Linear head	73.6%	2,050
MLP (1024→256→2)	83.6%	262,402
QA null-score	**90.3%**	**0**

Table 6: Oracle detection results (gold detection spans fed to stages 2–4; localisation and correction remain predicted). Predicted-detection values in parentheses for reference.

Metric	D2T	QA	Sum.	All
Type accuracy (pred.: 90.3%)	90.5%	91.3%	80.6%	89.0%
Source span EM (pred.: 69.2%)	64.0%	69.1%	44.5%	61.0%
MiniCheck — oracle det. (pred.: .391)	.274	.582	.567	.419

with predicted detection (Table 1). This is not a regression of the QA model: gold detection surfaces additional hallucinated spans that the detector would otherwise miss, and those previously-missed spans are systematically harder to localise (concentrated in Summary, where evidence is scattered), pulling the average down. The interpretation is that detection misses are not silently failing on easy spans — they are silently dropping the spans the localiser would have struggled with anyway. Type accuracy is essentially unchanged (89.0% under oracle detection vs. 90.3% predicted, a 1.3 pp drop driven by the same Summary-heavy tail). Faithfulness reaches 0.419 with oracle detection (+2.8 pp over 0.391) and 0.424 with oracle detection + localisation (+0.5 pp more), still below the Qwen3.5-122B aggressive-rewrite baseline (0.551, which edits 76% of answers to get there) — detection recall and source-span quality are both binding constraints.

D Detection Baselines (Full Table)

E Generalisation to HaluEval (Out-of-Domain)

To assess generalisation, we evaluate HALIGN zero-shot on HaluEval (Li et al., 2023), a hallucination benchmark covering QA and Summarisation. Our training data (RAGTruth) shares neither distribution nor task formatting with HaluEval. We sample 500 hallucinated examples per task (`hallucinated_answer`/`hallucinated_summary`) and pair each with the corresponding ground-truth answer to form a balanced 500+500 mixed set per task, which lets us report unbiased precision, recall, and F1.

Table 7: Example-level detection F1 (%) on RAGTruth. Prior numbers from Kovács and Recski (2025). Best encoder-based result in **bold**; best overall underlined.

Method	Type	QA	D2T	Sum.	All
Prompt GPT-4-turbo	Prompt	45.6	78.3	47.6	63.4
SelfCheckGPT	Prompt	43.7	74.8	40.1	58.8
ChainPoll	Prompt	40.5	49.6	46.9	46.7
Luna	Encoder	51.3	75.9	52.5	65.4
LettuceDetect-base	Encoder	65.5	87.9	50.5	76.1
LettuceDetect-large	Encoder	70.2	88.5	58.9	79.2
Finetuned Llama-2-13B	LLM	68.2	88.1	59.1	78.7
RAG-HAT (Llama-3-8B)	LLM	74.8	91.6	67.6	83.9
HALIGN (ours)	Encoder	**75.5**	**88.2**	**63.6**	**79.6**

Table 8: HaluEval (out-of-domain). Detection on a balanced mixed set of 500 hallucinated + 500 clean examples per task. Faithfulness (MiniCheck) on the 500 hallucinated examples per task; "Ceiling" is the MiniCheck score of the ground-truth correct answers.

	Detection (mixed set)			Faithfulness (hal. only)			
Task	P	R	F1	Orig.	HALIGN	Δ	Ceiling
QA	.845	.652	.736	.274	.448	**+.174**	.743
Summary	.564	.426	.485	.357	.420	+.063	.642
Overall	.706	.539	.611	.315	.434	**+.119**	.692

Table 8 reports example-level detection on the 2,000-example mixed set and MiniCheck faithfulness on the 1,000 hallucinated examples. HALIGN transfers well to QA (F1 = 0.736) and weakly to Summary (F1 = 0.485), mirroring the within-domain pattern. On faithfulness, the pipeline raises MiniCheck from 0.274 to 0.448 on QA (gold-answer ceiling 0.743) and from 0.357 to 0.420 on Summary (ceiling 0.642).

F Inference Latency

Table 9 reports per-example latency measured with 8-way concurrent requests to the vLLM-served corrector, averaged over 100 hallucinated examples. Stage 4 rewrites the full answer in one API call per example and dominates end-to-end latency. Clean examples without detected hallucinations skip Stages 2–4 and complete in ≈79 ms. The Qwen3.5-122B baseline averages 4.1 s per example — 2.1× slower than HALIGN + Llama-3.2-3B but faster than the Qwen-4B standalone despite being 30× larger; its hosted deployment uses Multi-Token Prediction (MTP), which substantially closes the per-token decoding gap to smaller models.

Table 9: Per-example inference latency (single A100). †Hosted with Multi-Token Prediction.

Stage	Mean	Median
Stage 1 Detection (ModernBERT)	79 ms	67 ms
Stage 2–3 Type + Source (DeBERTa)	213 ms	155 ms
Stage 4 Correction (Llama-3.2-3B)	1,600 ms	1,445 ms
E2E (hallucinated example)	**1,892 ms**	**1,737 ms**
Llama-3.2-3B E2E (standalone)	6,369 ms	5,442 ms
Qwen3.5-4B E2E (standalone)	7,288 ms	6,558 ms
Qwen3.5-122B E2E†	4,060 ms	3,590 ms

Table 10: Training hyperparameters for stages 1–3. Stage 2 (type classification) is parameter-free and inherits the Stage 3 DeBERTa-v3-large checkpoint at inference; only Stages 1 and 3 are trained.

Parameter	Detection	Type + Source Span
Base model	ModernBERT-large	DeBERTa-v3-large
Learning rate	5e-6	2e-5
Batch size (eff.)	8 (2×4)	8 (4×2)
Epochs	10	10
Max sequence length	4096	512 (stride 128)
Warmup ratio	0.1	0.1
Weight decay	0.01	0.01
Patience	4	4
Grad norm clip	1.0	1.0

G Hyperparameters

Table 10 lists the key hyperparameters used for each training phase. All experiments use a single NVIDIA A100 GPU with mixed-precision (bf16) training.

H Qualitative Error Analysis: Summary Task

The Summary task exhibits the lowest end-to-end accuracy (29.6%) across all three task types. We identify three systematic failure modes, illustrated with test set examples below.

Failure Mode 1: Detection Misses. The detector fails to flag hallucinated spans that are plausible continuations of the source context. In 82 of 204 hallucinated Summary examples, the pipeline produces no hallucination prediction at all.

> **Answer:** "Three women, including Keonna Thomas of Philadelphia, were charged with attempting to join ISIS this week."
>
> **Missed span:** "Three women… were charged with attempting to join ISIS this week."
> **Gold correction:** "A Philadelphia woman was charged with trying to fight for ISIS, making her one of three women arrested this week on terror charges."
>
> *Analysis:* The claim is factually close to the source but subtly reframes the structure (three women charged → one woman charged, among three arrested). The detector likely scores it below threshold because most tokens are individually supported.

Failure Mode 2: No Source Evidence. Unlike Data2txt and QA, where the source document contains verbatim or near-verbatim evidence for corrections, Summary replace spans have *zero* extractable source text in the RAGTruth annotations. All 165 Summary replace spans lack a `source_text` field. This means the QA-based evidence localiser cannot find an anchor span, and the correction backend has no source evidence to condition on.

> **Hallucinated:** "was arrested before her trip"
> **Gold correction:** "it remains unclear when or where she was arrested"
>
> **Hallucinated:** "She was arrested on March 26"
> **Gold correction:** "It is unknown when or where she was taken into custody"
>
> *Analysis:* The correct text negates or hedges the original claim. This requires reasoning over the *absence* of information in the source—the correction is that the fact is unknown—which an extractive QA model cannot produce.

Failure Mode 3: Fabricated Claims (Delete Spans). Summary answers frequently contain entirely fabricated details that must be deleted rather than replaced. The 129 delete spans in Summary include invented attributions, invented specifics, and editorial embellishments:

> **Fabricated:** "they claim their marriage works due to their shared values and love for each other"
> *Context:* A summary about a controversial age-gap marriage invents a direct quote about "shared values" that appears nowhere in the source articles.
>
> **Fabricated:** "love and redemption"
> *Context:* A summary characterises a TV interview as being about "love and redemption" when the source describes the tone differently.

These fabrications often sound editorially plausible, making them difficult for token-level detectors to flag.

Summary. The Summary task exposes architectural limitations: the pipeline assumes corrections can be grounded in extractable source evidence, which holds for Data2txt (structured fields) and QA (passage spans) but breaks down for summarisation, where hallucinations often involve subtle reframing, hedging, or invented specifics that require abstractive reasoning to correct.

I Dataset Representativeness

We evaluate on RAGTruth (Niu et al., 2024), which is currently the largest human-annotated hallucination benchmark for RAG systems. While this constitutes a single benchmark, RAGTruth provides substantial diversity along several axes:

- **Task diversity:** Three distinct generation tasks—Data2txt (structured-to-text), QA (passage-based question answering), and Summarisation—each presenting different hallucination patterns and correction challenges.
- **Model diversity:** Responses generated by six LLMs spanning different architectures and scales (GPT-4, GPT-3.5-turbo, Llama-2-7B/13B/70B-chat, Mistral-7B-Instruct), ensuring the benchmark captures diverse hallucination styles rather than artefacts of a single model.
- **Annotation quality:** Word-level span annotations with correction types (replace vs. delete) by trained human annotators, providing fine-grained supervision suitable for correction (not just detection).
- **Scale:** 2,870 test examples (of which 1,113 contain at least one annotated hallucination, for a total of 1,703 spans)—substantially larger than prior benchmarks such as FELM (Chen et al., 2023) or HaluEval (Li et al., 2023), which provide only sentence-level labels.

Extending evaluation to additional benchmarks (e.g., multilingual RAG datasets, domain-specific corpora) remains important future work, but the diversity within RAGTruth makes it a rigorous single-benchmark evaluation.

J Prompts

We list the correction prompt used by HALIGN (§J.1) and the structured baseline prompt shared by all three standalone correctors (Llama-3.2-3B, Qwen3.5-4B, Qwen3.5-122B; §J.2).

J.1 Pipeline Correction Prompt

This is the correction prompt used by HALIGN + Llama-3.2-3B for pipeline correction (§3.5). The model receives the full answer with all detected hallucination spans marked with `[HAL]...[/HAL]` tags, along with a per-span instruction list (REPLACE with evidence, or DELETE). A single Data2txt few-shot example demonstrates the marker-stripped full-answer rewrite; the model outputs the complete corrected answer in one generation call, handling all spans simultaneously.

System: You are a factual text editor. The answer below contains hallucinated phrases marked with `[HAL]...[/HAL]` tags. Rewrite the full answer so that every `[HAL]...[/HAL]` phrase is either replaced with the correct information from the provided source evidence, or removed cleanly if it has no basis in the source.

Rules:

1. For each DELETE: remove the `[HAL]...[/HAL]` phrase and fix the surrounding text naturally.
2. For each REPLACE: substitute the `[HAL]...[/HAL]` phrase with correct information from the source. Express it as natural English — do not copy key-value pairs verbatim.
3. Leave every word NOT inside `[HAL]...[/HAL]` exactly as written.
4. Do NOT output `[HAL]` or `[/HAL]` tags in your response.
5. Output ONLY the corrected full answer. No explanation, no preamble, no markdown.

User (few-shot example):
Answer:
Zizzi is a `[HAL]`French`[/HAL]` restaurant located in the centre of Snaresbrook. It has a customer rating of `[HAL]`4 out of 5`[/HAL]`.
Corrections:
REPLACE `"[HAL]`French`[/HAL]"` with information from: "Zizzi is an Italian restaurant"
REPLACE `"[HAL]`4 out of 5`[/HAL]"` with information from: "customer rating of 1 out of 5"
Corrected answer:

Assistant (few-shot example):
Zizzi is an Italian restaurant located in the centre of Snaresbrook. It has a customer rating of 1 out of 5.

User (test):
Answer:
{answer with `[HAL]...[/HAL]` markers}
Corrections:
REPLACE `"[HAL]`{phrase_1}`[/HAL]"` with information from: "{source_evidence_1}"
DELETE `"[HAL]`{phrase_2}`[/HAL]"`
Corrected answer:

The prompt is deliberately short and free of any chain-of-thought scaffold: Llama-3.2-3B-Instruct is not a reasoning model, so explicit thinking tokens degrade rather than help output quality. KV-pair interpretation rules (e.g. `'Day': '0:0-0:0'` → CLOSED) are encoded implicitly through the few-shot demonstration rather than enumerated in the system message, which we found reduced the rate of verbatim KV copying in the output.

J.2 Structured Standalone Baseline Prompt (shared)

Shared by all three standalone correction baselines (Llama-3.2-3B-Instruct, Qwen3.5-4B, Qwen3.5-122B; §6.1). The model receives the full source context and LLM-generated answer, identifies hallucinated *sentences* (not spans), and replaces only those sentences while leaving all others verbatim. Corrections are applied mechanically: we first attempt to use the model's `<corrected_answer>` block, falling back to exact string replacement from the parsed JSON list when that block is missing or truncated.

System: You are a factual accuracy checker for AI-generated text. Given a source document and an AI-generated answer, your task is to:

1. Identify every sentence in the answer that contains a hallucination — a claim that contradicts or is not supported by the source.
2. For each hallucinated sentence, write a corrected version grounded in the source. If the entire sentence should be removed, use type "delete".
3. Output the corrected answer by replacing ONLY the hallucinated sentences. Every other sentence must remain exactly as written, word-for-word.

Output format — two blocks, in this order:

```
<hallucinations>
[
  {
    "original": "<exact hallucinated sentence>",
    "corrected": "<corrected or empty>",
    "type": "replace" or "delete"
  }
]
</hallucinations>

<corrected_answer>
<answer with hallucinations corrected>
</corrected_answer>
```

If the answer contains no hallucinations, output `<hallucinations>[]</hallucinations>`.

Rules:

- Only flag sentences with clear factual errors that contradict the source
- Ignore vague, abstractive, or stylistic differences — only flag verifiable factual errors
- Corrections must be directly supported by the source text
- Non-hallucinated sentences: copy them EXACTLY — no paraphrasing, no additions, no reordering

User: Source document: {source}

AI-generated answer: {answer}

J.3 Qualitative Comparison: HALIGN vs. Qwen3.5-122B

Table 11 illustrates the surgical vs. broad editing behaviour of the two correction approaches on a representative QA example. The gold annotation marks only one span for deletion; HALIGN + Llama-3.2-3B removes precisely that span, while Qwen3.5-122B additionally rewrites two other sentences that were factually acceptable.

Table 11: Qualitative comparison on a QA example (RAGTruth test set). Deletions are shown ~~struck through~~; rewrites/insertions are underlined. HALIGN's edit is verbatim Llama-3.2-3B output (only one clause excised); Qwen-122B's row condenses two deleted disclaimer sentences for space.

Condition	Answer
Original	Automotive technicians can get paid in different ways. Pay varies by location, with the highest in Alaska ($23.70/hr) and **the lowest in Mississippi ($18.60/hr or $38,900/yr)** [hallucinated]. Some may earn more in aerospace manufacturing ($32/hr). *However, the passages do not provide specific information… Therefore, I am unable to answer…*
HALIGN + Llama 3.2-3B	Automotive technicians can get paid in different ways. Pay varies by location, with the highest in Alaska ($23.70/hr) ~~and the lowest in Mississippi ($18.60/hr or $38,900/yr)~~. Some may earn more in aerospace manufacturing ($32/hr). However, the passages do not provide specific information… Therefore, I am unable to answer…
Qwen-122B	Automotive technicians can get paid in different ways. Pay varies by location, with the highest in Alaska ($23.70/hr) and techs in Connecticut, Maryland, Massachusetts and Washington D.C. averaging $20.50–$21.50/hr. Some may earn more in aerospace manufacturing ($32/hr). ["However…" and "Therefore…" sentences deleted.]

HALIGN makes exactly one change (the gold deletion), while Qwen3.5-122B makes three: it rewrites the pay-variation sentence with additional salary data, and deletes two disclaimer sentences that were factually acceptable (the model judged them as unsupported even though they accurately reflected the QA context). This pattern — higher MiniCheck scores through broader editing rather than more precise correction — is consistent with the quantitative result that the 122B baseline modifies 76% of all test examples vs. 36% for HALIGN.

K HaluEval Correction Examples

The following examples illustrate cases where HALIGN successfully detected and corrected hallucinations in HaluEval samples. Each example shows the source passage (or document), the hallucinated answer produced by an LLM, and the corrected answer produced by our pipeline. MiniCheck scores before and after correction are given in parentheses.

Example 1 — QA: fabricated authorship (MiniCheck: 0.002 → 0.974). Question: Who is the author of the fairy tale that the 1948 movie by the Archers is based on?

Source passage (excerpt): …The film is about a ballerina who joins an established ballet company

and becomes the lead dancer in a new ballet called "The Red Shoes", itself based on the fairy tale "The Red Shoes" by Hans Christian Andersen.
Hallucinated answer: The fairy tale that the 1948 movie by the Archers is based on is by the Brothers Grimm.
Corrected answer: The fairy tale that the 1948 movie by the Archers is based on is by **Hans Christian Andersen**.
The detector flags "the Brothers Grimm" as a hallucinated span; the QA stage localises the correct author from the passage; the correction substitutes it in place.

Example 2 — QA: incomplete answer corrected (MiniCheck: 0.014 → 0.971). **Question:** Allison Brooks Janney is an American actress — in which two years did she win the Emmy for Outstanding Supporting Actress in a Comedy Series for her role as Bonnie Plunkett on the CBS sitcom "Mom"?
Source passage (excerpt): In 2014 and 2015, she won the Emmy for Outstanding Supporting Actress in a Comedy Series for her role as Bonnie Plunkett on the CBS sitcom "Mom".
Hallucinated answer: She won the award only in 2015.
Corrected answer: She won the Emmy for Outstanding Supporting Actress in a Comedy Series in **2014 and 2015** for her role as Bonnie Plunkett on the CBS sitcom "Mom".
The hallucinated span drops one of the two winning years. The pipeline detects the under-specified claim, localises the correct multi-year span from the passage, and completes the answer.

Example 3 — Summary: fabricated embellishment corrected (MiniCheck: 0.012 → 0.915).
Document (excerpt): The father of one of three teenagers arrested in Turkey . . . works for the Ministry of Defence . . . it is believed the man . . . works in the military's post office, where he could have had access to the names and address of all military personnel at home and overseas.
Hallucinated summary: The father . . . works as a high-ranking official in the Ministry of Defence's Anti-Terrorism department, allowing him access to classified information.
Corrected summary: The father . . . works in the **military's post office, where he could have had access to the names and address of all military personnel at home and overseas**.
The hallucinated span invents a specific role (anti-terrorism, high-ranking) not present in the source. The pipeline localises the actual description from the document and substitutes it, removing the fabricated embellishment.