

Universiteit Leiden
The Netherlands

# Master Computer Science

## Detecting and Repairing Hallucinations in Retrieval-Augmented Generation: A Pipeline Analysis of Conditional Model Scaling


| | |
|---|---|
| Name: | Saikrishna Reddy Mulakkayala |
| Student ID: | s4238206 |
| Date: | 6 July 2026 |
| Specialisation: | Data Science |
| 1st supervisor: | Prof. dr. Aske Plaat |
| 2nd supervisor: | Dr. Niki van Stein |


Master's Thesis in Computer Science


Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

# Acknowledgements

This thesis was supervised by the Leiden Institute of Advanced Computer Science (LIACS) at Leiden University. I would like to thank my supervisor, Prof. dr. Aske Plaat, for his guidance, feedback, and support throughout this project. I appreciate the discussions and written feedback that helped me refine the research direction, especially when the original experimental plan changed and the project moved to the RAGTruth benchmark.

I also thank Dr. Niki van Stein for serving as second reader and for taking the time to evaluate this thesis. The experimental work in this project was made possible by access to computational resources, including the Snellius supercomputer operated by SURF. I am grateful for the infrastructure that made the detector training, verifier inference, and judge-based evaluation feasible.

Finally, I thank my family and friends for their encouragement during my studies and during the writing of this thesis. Their support helped me complete this work.

# Abstract

Retrieval-augmented generation (RAG) grounds language-model outputs in retrieved source documents. Generated responses may nonetheless contain claims that the sources do not support. This thesis treats hallucination mitigation as a pipeline problem with three stages: detection, verification, and repair. It asks how unsupported content can be detected and repaired while preserving the supported content of the response, and under what conditions model scaling helps each stage. All experiments use the RAGTruth benchmark, and each verdict was locked by pre-registration before the corresponding evaluation ran.

Across seven documented phases, scaling is conditional rather than automatic. On the detection side, scaling to a 13-billion-parameter detector under a constrained quantised low-rank adaptation (QLoRA) training setup, with a short training context and a single epoch, fails the locked test and performs below the existing smaller baseline. Re-scaling the same model family under a revised training configuration, with the context length matched to the task's token statistics and longer training, succeeds: the scaled detector reaches a response-level F1 of 79.6, matching the published RAGTruth full fine-tuning baseline of 78.7, although the comparison should be made with care since training details differ. Error analysis attributes the remaining span-level gap to response-level misclassification rather than to span extraction quality.

On the repair side, substituting a large prompted verifier for a smaller cross-encoder changes repair behaviour: a claim-level pipeline passes all four locked repair gates and reduces judge-assessed hallucination by 9.15 percentage points on the held-out evaluation set. An ablation shows that this benefit is carried by removing claims that fail verification, not by rewriting. Prompting the verifier to reason step by step improves some verdict accuracy but does not produce the graded confidence profile that rewriting requires.

The thesis concludes that the variables that determine success are not raw parameter count alone, but interface choices: context length matched to task token statistics for detection, and repair granularity matched to verifier calibration for repair.

# Contents

**Acknowledgements** **i**

**Abstract** **ii**

**1 Introduction** **1**

1.1 Motivation . . . . . 1

1.2 Research questions . . . . . 1

1.3 Approach and scope . . . . . 3

1.4 Contributions . . . . . 3

1.5 Thesis outline . . . . . 4

**2 Related Work** **5**

2.1 Hallucination in retrieval-augmented generation . . . . . 5

2.2 Detection and repair . . . . . 5

2.3 Verifiers and calibration . . . . . 6

2.4 Chain-of-thought prompting . . . . . 6

2.5 Parameter-efficient fine-tuning . . . . . 7

2.6 Pre-registration in machine learning . . . . . 7

2.7 Positioning of this thesis . . . . . 7

**3 Methodology and Experimental Setup** **8**

3.1 Experimental design and evidential roles . . . . . 8

3.2 Dataset and experimental populations . . . . . 9

3.3 Detector task and metrics . . . . . 10

3.4 Repair pipelines and pipeline variants . . . . . 11

3.5 Verifier protocols . . . . . 12

3.6 Repair metrics and locked criteria . . . . . 12

3.7 Compute environment and reproducibility . . . . . 14

**4 Detector Scaling: Phases A and F** **15**

4.1 Purpose of the detector-scaling experiments . . . . . 15

4.2 Phase A: Detector Scaling Sweep . . . 15
4.3 Phase A result and locked verdict . . . 16
4.4 What Phase A did and did not show . . . 16
4.5 Phase F: Same-Family Detector Re-scaling . . . 16
4.6 Phase F result and locked verdict . . . 17
4.7 Shared mechanism: context length, epochs, and same-family scaling . . . 17
4.8 Detector-side design rule . . . 18

**5 Verifier Scaling and Repair: Phases B, C, D, and E 19**
5.1 Purpose of the verifier and repair experiments . . . 19
5.2 Phase B setup: verifier substitution and repair candidates . . . 19
5.3 Phase B locked result: V1f passes the four gates . . . 20
5.4 Why V1f wins: verifier confidence profile and repair granularity . . . 21
5.5 Phase C: Reasoning Verifier Swap . . . 22
5.6 Phase C result and locked failure . . . 22
5.7 Phase D: Tie-Break Counterfactual . . . 23
5.8 Phase E: Mechanism Ablation . . . 23
5.9 Repair-side design rule . . . 24

**6 Span Localisation Diagnostic: Phase G 25**
6.1 Purpose of the span diagnostic . . . 25
6.2 Phase G setup . . . 25
6.3 Diagnostic result . . . 26
6.4 Error taxonomy and cascade interpretation . . . 27
6.5 Implication for detector design . . . 27

**7 Discussion and Synthesis 28**
7.1 Answer to the primary research question . . . 28
7.2 RQ1: detection and detector scaling . . . 28
7.3 RQ2: verification and reasoning . . . 29
7.4 RQ3: repair action and mechanism . . . 29
7.5 RQ4: faithfulness and completeness trade-off . . . 29
7.6 RQ5: reliability of automatic judging . . . 30
7.7 Conditional model scaling in RAG hallucination mitigation . . . 30
7.8 Practical design rules . . . 31

**8 Limitations and Future Work 32**
8.1 Benchmark and generalisation limits . . . 32
8.2 Model, training, and compute limits . . . 32

8.3 Evaluation and judge limitations . . . 33
8.4 Evidence and reproducibility limitations . . . 33
8.5 Future work . . . 33

**9 Conclusion** **35**

**A Pre-registration and amendment summaries** **36**
A.1 Phase A: Detector Scaling Sweep . . . 36
A.2 Phase B: Verifier Substitution and Repair . . . 37
A.3 Phase C: Reasoning Verifier Swap . . . 37
A.4 Phase D: Tie-Break Counterfactual . . . 38
A.5 Phase E: Mechanism Ablation . . . 38
A.6 Phase F: Same-Family Detector Re-scaling . . . 38
A.7 Phase G: Span Localisation Diagnostic . . . 39

**B Supplementary tables and examples** **40**
B.1 Worked example of claim-level repair . . . 40
B.2 Per-task span diagnostic for Phase G . . . 40

**C Reproducibility details** **42**
C.1 Repository and commit . . . 42
C.2 Software environment . . . 42
C.3 Compute environment . . . 42
C.4 Detector training configuration . . . 43
C.5 Verifier and judge configuration . . . 43
C.6 Data splits and per-task counts . . . 43
C.7 Result file inventory . . . 44
C.8 Phase F training and adjudication note . . . 44

**D Independent-judge robustness check** **45**
D.1 Purpose and design . . . 45
D.2 What the check tests . . . 45
D.3 Result: a split outcome . . . 45
D.4 Reading of the robustness check . . . 46

# List of Figures

3.1 The hallucination-mitigation pipeline studied in this thesis. The detector produces a response-level label and hallucination spans; the verifier adjudicates claims against the source; the repair action keeps, drops, rewrites, or substitutes content; the judge assesses the repaired response. Each experiment chapter changes one part and holds the rest fixed. . . . . 9

4.1 Response-level F1 across the detector experiments. Under the constrained training configuration the scaled detector (Phase A, 68.9) falls below the existing Qwen-2.5-7B baseline (74.8); under the revised configuration the same scale step (Phase F, 79.6) matches the published full fine-tuning baseline (78.7, dotted line). Test split, $n$=2,700. . . . . 18

5.1 Confidence profile of the substituted Qwen-2.5-32B verifier on the development set. Most verification events fall at the confident extremes; the middle band, where a rewrite decision would apply, holds only 13.33%. This shape is why claim-level dropping, not rewriting, carries the repair result. . . . . 21

6.1 Span-level F1 on the full validation set versus the response-correct (oracle) subset, for both detector sizes. Full-set values are 34.96 (7B) and 44.50 (13B); oracle-subset values are 53.66 (7B) and 53.56 (13B). The two oracle bars are nearly equal, indicating that scale improved response-level judgement rather than span localisation. Validation split, $n$=942. . . . . 26

# List of Tables


1.1 Mapping between the research questions, experimental phases, and the original proposal. . . . 3

3.1 The seven phases of the thesis, their evidential roles, the data each touched, and their verdict labels. The roles are defined in Section 3.1; the verdicts are reported and interpreted in Chapters 4 to 6. . . . 9

3.2 The five experimental populations. Each population serves one role; the locked populations are evaluated under the one-touch discipline described in Section 3.1. 10

3.3 The four-gate success criterion shared by Phase B and Phase D. A winning pipeline must satisfy all four gates simultaneously on the locked population; the clause order is also the tie-break order. Thresholds are as fixed in the Phase B pre-registration, reproduced verbatim in Appendix A. . . . 13

4.1 Phase A locked-test result on the RAGTruth test split ($n$=2,700), with the project baseline and the published fully fine-tuned baseline for reference. The supplementary three-epoch run is discussed in Section 4.7. . . . 16

4.2 Phase F locked-test results on the RAGTruth test split ($n$=2,700). Both detectors share the revised training configuration (2,048-token context, three epochs); only the model size differs. . . . 17

5.1 Phase B locked-test result for V1f on the locked repair population ($n$=732 records, 5,439 claims). . . . 20

5.2 Confidence profile of the substituted Qwen-2.5-32B verifier on the verifier development set. The middle band is the region in which a rewrite action would be considered. . . . 21

5.3 Phase C verdict summary. The criteria are conjunctive: all parts must succeed for the phase to succeed. Values are measured on development data. . . . 22

5.4 Phase E mechanism ablation on the locked repair population ($n$=732). V1f and V1cdo share the same claim decomposition and the same verifier; they differ only in the repair actions available. . . . 24

6.1 Phase G span decomposition on the RAGTruth validation split ($n$=942), with test-split values for reference. Validation and test are different splits; values are not directly comparable across them. . . . 26

7.1 Synthesis of the research questions. Evidence sources use the evidential roles of Chapter 3; every answer is bounded by the stated limitation. . . . 31

B.1 Per-task span diagnostic for the F-13B detector on the RAGTruth validation split ($n$=942). Oracle counts are response-correct records; span F1 is reported on the full task set and on the oracle subset. . . . . 41

C.1 Compute summary per phase. GPU: NVIDIA A100 40 GB on the `gpu_a100` partition. The time column gives the requested SLURM wall-clock limit for the committed run, not measured elapsed time. . . . . 43

C.2 Experimental splits, repository paths, and per-task record counts. Paths are relative to the repository root. . . . . 44

D.1 Independent-judge robustness check on the locked repair records ($n$=732). Each judge re-scores the original responses and the V1f-repaired responses; counts are the number of responses each judge labels hallucinated. The primary-judge column repeats the Chapter 5 figures for reference. Agreement with human gold is reported as Cohen's $\kappa$ [2]. . . . . 46

# Chapter 1

# Introduction

## 1.1 Motivation

Retrieval-augmented generation (RAG) has become a dominant way of applying large language models to factual tasks. A retriever supplies source documents, and a generator produces an answer conditioned on those documents. This architecture reduces the model's reliance on parametric memory, but it does not guarantee source faithfulness. A response may still add a date, number, attribution, causal relation, or event detail that is not present in the retrieved material. This failure mode is difficult precisely because the response often appears grounded: the retrieved passage covers the general topic, the generated text is fluent and plausible, and yet one or more factual claims are unsupported. The practical problem is whether every factual statement can be traced to the supplied evidence, not merely whether the response looks relevant [14].

A mitigation pipeline for this problem can be decomposed into three stages. Detection identifies whether and where a response contains unsupported material. Verification adjudicates individual claims against the source. Repair uses these signals to produce a revised response, usually by dropping, rewriting, or abstaining on unsupported content. The stages interact: a repair pipeline can only act on the population flagged by the detector, and the kind of repair that is effective depends on how the verifier judges whether a sentence or claim is supported.

A natural expectation is that larger models improve every stage. This thesis tests that expectation directly and finds it incomplete: scaling helped under some configurations and harmed under others. The pattern of successes and failures is informative: model scaling helps when the architectural interface between the model and its pipeline stage is right, and under-performs when it is not. The chapters that follow develop this conditional view of scaling through a sequence of pre-registered experiments on a single benchmark.

## 1.2 Research questions

This thesis investigates how hallucinations in retrieval-augmented generation can be detected and repaired. The questions continue the line of inquiry of the original research proposal, submitted in January 2026, which asked how hallucinations in RAG systems can be detected and reduced while maintaining response quality and completeness. When the original interview-analysis experiment was cancelled, the experimental testbed changed to the RAGTruth benchmark. The questions, however, remained the same in spirit: how to detect unsupported content, how to repair it, and how to preserve useful supported content while doing so.

The primary research question is:

> **Primary RQ.** How can hallucinations in retrieval-augmented generation be detected and repaired while preserving the supported content of the response?

This primary question is divided into five sub-questions, each answered by named experimental phases that later chapters report in full. The first three (Detection, Verification, and Repair) carry the central empirical investigation; the last two (Trade-offs and Evaluation) are analysis-oriented questions that qualify and bound those results rather than introducing new experiments.

**RQ1 Detection.** How well can a parameter-efficient detector trained with QLoRA (quantised low-rank adaptation) identify hallucinated responses and spans, and does scaling the detector improve detection?

This question is addressed by the Detector Scaling Sweep (Phase A), the Same-Family Detector Re-scaling comparison (Phase F), and the Span Localisation Diagnostic (Phase G). The supporting analysis examines model size, model family, context length, training duration, and how response-level errors affect span-level localisation.

**RQ2 Verification.** Can a stronger verifier, or one prompted to reason step by step, improve the claim-verification decisions that drive repair?

This question is addressed by Verifier Substitution and Repair (Phase B) and the Reasoning Verifier Swap (Phase C). The supporting analysis examines the verifier's confidence distribution and how that distribution determines which repair actions the pipeline can use.

**RQ3 Repair.** Which repair action, rewriting, extractive substitution, or removing unsupported claims, most effectively reduces hallucinations?

This question is addressed by the repair-pipeline comparison under locked gates (Phase B), the Mechanism Ablation (Phase E), and the explicitly labelled post-hoc Tie-Break Counterfactual (Phase D). The thesis distinguishes the committed experimental verdicts from the counterfactual analysis throughout.

**RQ4 Trade-offs.** What is the trade-off between removing unsupported content and preserving a complete, useful response?

This question is addressed by the four-gate evaluation structure used in the repair experiments and described in Chapter 3. The change in unsupported rate and the damage rate measure faithfulness, while claim survival and length retention measure preservation of useful content. This question carries over directly from the original proposal's faithfulness-versus-completeness concern.

**RQ5 Evaluation.** How reliable is an automatic judge for measuring hallucination, compared with human annotation and across independent judges?

This question is addressed by the manual validation of the held-out judge and by the post-hoc independent-judge robustness check in Appendix D. It defines the scope of the repair claims in this thesis: judge-based results are interpreted as relative comparisons under a checked judge, not as absolute hallucination rates.

Table 1.1 summarises how the five sub-questions map onto the experimental phases and onto the questions of the original proposal. The phases themselves, and the evidential discipline under which they were run, are introduced in Chapter 3.

Table 1.1: Mapping between the research questions, experimental phases, and the original proposal.

| Question | Addressed by | Proposal question refined |
|---|---|---|
| RQ1 Detection | Phases A, F, G | Parameter-efficient detection and span localisation |
| RQ2 Verification | Phases B, C | Mechanism behind repair decisions |
| RQ3 Repair | Phases B, D, E | Repair effect and repair action choice |
| RQ4 Trade-offs | Four-gate structure in B, D, E | Faithfulness versus completeness |
| RQ5 Evaluation | Manual validation; Appendix D | Judge reliability and generator-group validity |

## 1.3 Approach and scope

All experiments use the RAGTruth benchmark [14], which provides source passages, model-generated responses, and span-level hallucination annotations across question answering, summarisation, and data-to-text generation. The five sub-questions are answered through seven documented phases with different evidential roles. Phases A, B, and F provide locked-test results: their success criteria were committed before any test inference [15], and the test data was touched once per committed configuration. Phases C, E, and G provide diagnostic evidence under pre-registered criteria. Phase D is an explicitly labelled post-hoc counterfactual on Phase B's locked development data, with no new compute. A post-hoc independent-judge robustness check (Appendix D) bounds the repair claims under two judges from model families different from the primary judge.

The scope is bounded in three ways. First, all results concern one benchmark, RAGTruth, and the locked verdicts are specific to the tested training setups. Second, the detector experiments use QLoRA [3, 7], so the detector conclusions concern parameter-efficient adapter training, not full fine-tuning. Third, the repair evaluation relies primarily on an automatic judge built on Qwen-2.5-32B [16]; the robustness appendix quantifies how the repair claims transfer to independent judges, and where one independent judge does not confirm a result, that non-confirmation is reported explicitly.

## 1.4 Contributions

The thesis makes six contributions, listed below.

1. **A pre-registered negative detector-scaling result.** An eighteen-cell Llama-2-13B QLoRA sweep [17] reaches 68.9 response-level F1 and 20.6 span-level F1 on the locked test, below the existing smaller baseline (74.8 and 37.4), so the locked Phase A verdict is failure.

2. **A pre-registered repair result.** Under the substituted verifier, the claim-level repair pipeline called V1f passes all four locked repair gates on 732 held-out records, with a delta unsupported rate of $-7.50$ percentage points, a damage rate of 0.00%, claim survival of 84.20%, and length retention of 86.52%.

3. **Identification of the repair mechanism.** A pre-registered claim-drop-only ablation reproduces the same judge-assessed outcome, a reduction of 9.15 percentage points in response-level hallucination, indicating that the operative mechanism is the removal of claims that fail verification rather than rewriting, under a verifier whose confidence profile leaves only 13.33% of verification events in the uncertain middle band.

4. **A mixed verdict on reasoned verification.** Chain-of-thought prompting [19] raises the sentence-level baseline's response F1 from 44.79% to 54.06% but misses the locked

lead-narrowing criterion by 0.20 percentage points and leaves the near-binary confidence profile in place, so the combined Phase C verdict is failure.

5. **Candidate-set selection as a methodological decision.** Under an expanded candidate set, the locked tie-break order selects a different pipeline, V1e, via claim survival, in a post-hoc analysis that does not modify Phase B's committed verdict.

6. **A recipe-specific reading of the detector-scaling failure.** The same model family, trained with a context length matched to the task's token statistics and longer training, reaches 79.6 response-level F1, matching the published full fine-tuning baseline of 78.7 under adapter training only, while the span diagnostic locates the remaining span-level gap in response-level misclassification rather than span extraction.

The most important contributions are the first three: the negative detector-scaling result, the locked repair result, and the ablation that identifies removal as the operative repair mechanism. These carry the central empirical claim of the thesis. Contributions five and six are supporting: the candidate-set counterfactual is methodological, and the recipe-specific re-scaling result qualifies the Phase A failure rather than introducing a new headline finding.

Across all six contributions, the thesis shows the value of pre-registration in a machine-learning thesis: locked criteria, amendment discipline, and one-touch test evaluation allow a negative result, a positive result, a mechanism ablation, a mixed verdict, and a methodological counterfactual to coexist without post-hoc reinterpretation.

## 1.5 Thesis outline

Chapter 2 reviews related work on hallucination in RAG, detection and repair, verifiers and calibration, chain-of-thought prompting, parameter-efficient fine-tuning, and pre-registration. Chapter 3 describes the dataset, experimental populations, metrics, locked criteria, and compute environment shared by all phases, and defines the project-specific vocabulary used in the experiment chapters. Chapter 4 reports the detector-scaling experiments (Phases A and F). Chapter 5 reports the verification and repair experiments (Phases B, C, D, and E). Chapter 6 reports the span localisation diagnostic (Phase G). Chapter 7 synthesises the findings into the conditional-scaling claim and its design rules. Chapter 8 states limitations and future work, and Chapter 9 concludes. The appendices contain condensed pre-registration and amendment summaries (Appendix A), supplementary tables and examples (Appendix B), reproducibility details (Appendix C), and the independent-judge robustness check (Appendix D).

# Chapter 2

# Related Work

## 2.1 Hallucination in retrieval-augmented generation

Retrieval-augmented generation was introduced as an architecture that combines a parametric generator with a non-parametric retrieval memory, motivated by provenance and by the ability to update knowledge without retraining [10]. Lewis et al. reported that retrieval-augmented models produce more specific and more factual output than parametric-only baselines on knowledge-intensive tasks. Retrieval reduces the model's dependence on parametric memory, but it does not guarantee that every generated statement is supported by the retrieved documents, and that residual gap is the problem this thesis studies.

The hallucination literature distinguishes intrinsic hallucination, content that contradicts the source, from extrinsic hallucination, content that the source cannot verify [8]. This thesis adopts a source-faithfulness reading under which both kinds count as unsupported relative to the retrieved evidence: the operative question is not whether a statement is true in the world but whether it can be traced to the supplied passages. Human evaluation of abstractive summarisation showed that fluent neural summarisers frequently generate content unfaithful to the input document, and that a statement can be world-true yet still unfaithful [12]. That distinction motivates the thesis position that verification must be grounded in the source rather than in general knowledge.

RAGTruth supplies what this position requires empirically: source passages, responses generated by multiple language models, and span-level hallucination annotations across question answering, summarisation, and data-to-text generation [14]. Span-level annotation supports both response-level detection and localisation of unsupported content, and localisation is what targeted repair needs. All seven phases of this thesis run on this benchmark.

## 2.2 Detection and repair

Detection and repair are related but distinct tasks. Detection asks whether a response contains unsupported content and where it occurs. Repair asks what the system should do once unsupported content is identified. Prior systems combine the two in different ways. Self-RAG trains a single model to retrieve on demand, generate, and critique its own output through reflection tokens [1]. Chain-of-verification has the model draft a response, plan verification questions, answer them independently, and revise [5]. Both report reduced hallucination on the tasks they study; neither isolates how the benefit divides between detection quality, verification quality, and the repair action itself, which is the decomposition this thesis makes explicit.

A complementary line detects hallucination without any source at all. SelfCheckGPT samples

multiple responses and scores consistency among them, on the premise that hallucinated content diverges across samples [11]. In the RAG setting of this thesis the source is available, so verification can be grounded in the retrieved passages directly; SelfCheckGPT also appears among the baselines in the published RAGTruth comparison against which Phase F's detector is evaluated.

Detection quality does not imply localisation quality: a detector can have high response-level F1 while placing spans poorly, a gap Phase G diagnoses directly. Repair, in turn, can operate at different granularities. A sentence-level pipeline is simple and conservative but may remove supported content when only one claim inside a sentence fails. A claim-level pipeline can target unsupported material precisely, at the cost of depending on decomposition quality, verifier calibration, and reconstruction decisions. Which granularity wins under a given verifier is the question Phases B, D, and E answer.

## 2.3 Verifiers and calibration

Verifier-based factuality assessment is commonly framed as natural language inference: given a source as premise and a claim as hypothesis, the verifier estimates entailment. Cross-encoder verifiers descend from the transformer architecture and its bidirectional encoders [18, 4], with DeBERTaV3 refining the pre-training procedure [6]; the DeBERTa-v3-large natural-language-inference model that served as this project's prior verifier is a fine-tuned member of that family. Such verifiers are fast and produce probability-like outputs. Prompted large-language-model verifiers, such as the Qwen-2.5-32B-Instruct model adopted in Phase B [16], are more flexible, but their behaviour depends on prompt wording and on the output protocol used to extract a decision.

Calibration describes how well a model's confidence scores track the empirical probability of being correct. For repair, the shape of the verifier's output distribution matters as much as its accuracy, because it determines which repair actions are available in practice. A verifier that produces graded intermediate probabilities can support a three-way decision among keep, rewrite, and drop: claims in the uncertain middle band are candidates for rewriting rather than outright removal. A verifier that commits to near-binary probabilities leaves that middle band almost empty, so rewrite conditions rarely trigger and the pipeline is pushed towards keep-or-drop behaviour. Phase B measures this distribution shape directly on the substituted verifier, and Phase C tests whether a change of reasoning protocol moves it towards the graded profile that rewriting requires.

## 2.4 Chain-of-thought prompting

Chain-of-thought prompting elicits step-by-step reasoning before the model commits to an answer, and it improves accuracy on multi-step reasoning benchmarks when demonstrations are supplied [19]. Kojima et al. showed that a single zero-shot instruction can elicit similar behaviour without hand-crafted exemplars, indicating that the reasoning style is activated by prompting alone [9]. These results establish accuracy gains on reasoning tasks; they do not establish that chain-of-thought changes the calibration of a verifier performing source-grounded inference. Whether reasoned verification produces the graded probability output that would activate rewrite decisions is exactly the pre-registered question of Phase C.

## 2.5 Parameter-efficient fine-tuning

Adapting a large pre-trained model by updating all of its weights is expensive in memory and compute and requires storing a full model copy per task. Low-Rank Adaptation freezes the pre-trained weights and trains low-rank decomposition matrices injected into the network, reducing trainable parameters by orders of magnitude while matching full fine-tuning on many reported tasks [7]. QLoRA extends this to a quantised setting: the frozen base model is held in 4-bit precision and gradients flow through it into the adapters, making large-model fine-tuning feasible on a single GPU [3]. This thesis trains its detectors with QLoRA throughout (Phases A and F). The choice is consequential: whether adapter-only training can reproduce published full-fine-tune detection quality is what Phase A tests under one fixed training configuration, and Phase F separates the training-configuration variables, context length and epochs, from the capacity question by re-scaling within the same model family.

## 2.6 Pre-registration in machine learning

Pre-registration separates prediction from postdiction by committing hypotheses and analysis plans before observation [15]; in this thesis that practice is applied by also fixing explicit success criteria in advance. Machine-learning experiments carry many degrees of freedom: model choice, split construction, checkpoint selection, metric priority, thresholds, and supplementary runs. This thesis uses pre-registration to lock success criteria, define tripwires, and document amendments, with one-shot test evaluation per committed configuration. The discipline matters here because the verdicts include a locked failure (Phase A), a mixed verdict missed by 0.20 percentage points (Phase C), and an explicitly post-hoc counterfactual (Phase D); locking prevents any of these from being reframed after the results were known.

## 2.7 Positioning of this thesis

The work surveyed above supplies the components: an architecture that retrieves [10], terminology and evidence for source-unfaithful generation [8, 12], mitigation systems that intertwine detection, verification, and revision [1, 5], source-free detection [11], verifier families with different calibration behaviour, prompting techniques that alter reasoning, and adapter-based training that makes detector fine-tuning affordable. What it largely does not supply is pipeline-level evidence about when model scaling helps each stage, and it reports mostly positive results. This thesis contributes that missing piece on a single benchmark: a pre-registered, phase-wise analysis in which detector scaling fails under one training configuration and succeeds under a revised one, verifier substitution changes the ranking of repair pipelines, the winning mechanism is identified by ablation as claim-level drop-on-fail, and chain-of-thought fails to unlock graded calibration. The claims are bounded to RAGTruth and to the tested training setups, and the negative and mixed verdicts are reported with the same prominence as the wins.

# Chapter 3

# Methodology and Experimental Setup

This chapter introduces the experimental language of the thesis. It explains how the experiments are organised into phases, what data they use, what the detection and repair tasks are, how verification works, and what counted as success before any result was seen. Later chapters rely on the vocabulary established here, so each project-specific term is defined at its first appearance.

The object of study throughout is a single hallucination-mitigation pipeline, shown in Figure 3.1. Source documents and a generated response enter the pipeline. A detector decides whether the response contains unsupported content and where. A verifier adjudicates individual sentences or claims against the source. A repair action then produces a revised response, by keeping, rewriting, or removing content, and an automatic judge assesses whether the repaired response still hallucinates. Each phase of this thesis isolates one part of this pipeline, and each experiment chapter opens with a brief reminder of which part its phases change and which parts stay fixed.

## 3.1 Experimental design and evidential roles

The experiments of this thesis were not run as one undifferentiated batch. They were organised into seven named phases, each created to answer one question, and each assigned an evidential role before its results were interpreted. The reason for this structure is methodological. Machine-learning experiments leave their designer many degrees of freedom, such as which checkpoint to report, which metric to prioritise, and which threshold separates success from failure. If those choices are made after the results are visible, it becomes impossible to tell prediction apart from hindsight. Following the practice of pre-registration [15], this project committed its hypotheses, success criteria, and analysis plans in written documents before the corresponding evaluations ran.

Four terms describe the resulting evidential roles, and they recur throughout the thesis. A *locked test* is an evaluation whose success criteria were fixed in advance and whose evaluation data is touched only once per committed training configuration; its verdict, whether success or failure, cannot be renegotiated afterwards. A *diagnostic* phase also follows pre-registered criteria, but its purpose is to explain or bound a locked result rather than to declare a new winner. A *post-hoc* analysis is performed after results are known; the thesis labels such analyses explicitly so that the reader can weigh them accordingly. Finally, a *counterfactual* re-analyses outputs that already exist under an alternative decision input, without running any new model; it asks what would have been decided, not what would have been generated.

One consequence of this discipline is central to how the results should be read. Because verdicts




Figure 3.1: The hallucination-mitigation pipeline studied in this thesis. The detector produces a response-level label and hallucination spans; the verifier adjudicates claims against the source; the repair action keeps, drops, rewrites, or substitutes content; the judge assesses the repaired response. Each experiment chapter changes one part and holds the rest fixed.

Table 3.1: The seven phases of the thesis, their evidential roles, the data each touched, and their verdict labels. The roles are defined in Section 3.1; the verdicts are reported and interpreted in Chapters 4 to 6.

| Phase | Evidential role | Data touched | Verdict label |
|---|---|---|---|
| A: Detector Scaling Sweep | Locked test | Test split (one touch) | Failure |
| B: Verifier Substitution and Repair | Locked test | Locked population, $n$=732 | Win |
| C: Reasoning Verifier Swap | Diagnostic | Development data | Failure (combined) |
| D: Tie-Break Counterfactual | Post-hoc counterfactual | Phase B development data | Alternative winner |
| E: Mechanism Ablation | Mechanism ablation | Phase B locked population | Mechanism identified |
| F: Same-Family Detector Re-scaling | Locked test | Test split (two touches) | Win |
| G: Span Localisation Diagnostic | Diagnostic | Validation split | Diagnostic only |

were locked in advance, the thesis contains failures as well as successes, and it reports both. The failed detector-scaling experiment of Phase A and the mixed verdict of the reasoning experiment in Phase C are not omitted or softened, because the research question asks under what conditions scaling helps, and the failures are precisely what identify those conditions.

Table 3.1 gives the resulting map. Each row names a phase, states its role in the vocabulary just defined, records which data it touched, and gives its verdict label. The verdict labels are shorthand for outcomes that Chapters 4 to 6 report in full; nothing in this table should be read as a result yet. The research questions of Chapter 1 attach to the phases as follows: the detection question RQ1 is addressed by Phase A (Detector Scaling Sweep) and Phase F (Same-Family Detector Re-scaling), with Phase G (Span Localisation Diagnostic) examining what the successful detector still cannot do; the verification question RQ2 by Phase B (Verifier Substitution and Repair) and Phase C (Reasoning Verifier Swap); the repair question RQ3 by Phases B, D, and E; the trade-off question RQ4 by the four-gate structure of Section 3.6; and the evaluation question RQ5 by the judge validation and the independent-judge robustness check of Appendix D.

A *mechanism ablation*, the role of Phase E, is an experiment that switches off a component of a winning system and re-runs the evaluation; if the outcome is unchanged, the disabled component was not the source of the win. In addition to the seven phases, a post-hoc robustness check, reported in Appendix D, re-scores the repair outputs of Phase B under two independent judge models to test how far the repair conclusions depend on the particular judge used.

## 3.2 Dataset and experimental populations

All experiments use RAGTruth, a benchmark built specifically for hallucination research in retrieval-augmented generation [14]. Each record contains source passages, a response generated by one of several language models, and human annotations marking the exact spans of the response that the source does not support. The corpus covers three task types: question answering, summarisation, and data-to-text generation. The span-level annotation is what makes the benchmark suitable here, because it supports two distinct questions, whether a response contains unsupported content at all, and where that content lies, and later chapters show that a

Table 3.2: The five experimental populations. Each population serves one role; the locked populations are evaluated under the one-touch discipline described in Section 3.1.

| Population | $n$ | Role | Used by phases |
|---|---|---|---|
| RAGTruth test split | 2,700 | Locked detector evaluation | A, F |
| RAGTruth validation split | 942 | Span localisation diagnostic | G |
| Verifier development set | 1,046 | Verifier accuracy and calibration | B, C |
| Repair development set | 314 | Pipeline development and comparison | B, D |
| Locked repair population | 732 | Locked repair evaluation; mechanism re-use | B, E |

detector can be good at the first and weak at the second.

RAGTruth provides responses generated by several large language models across its three task types, with every response annotated at the span level by trained human annotators [14]. This thesis does not re-annotate the corpus; it uses the published annotations directly. The populations drawn from it are summarised in Table 3.2: a detector test split of 2,700 responses, a validation split of 942, a verifier development set of 1,046, a repair development set of 314, and a locked repair population of 732 responses comprising 5,439 individual claims, of which 269 responses (36.7%) carry at least one gold hallucination annotation. The corpus-level totals, the per-task record counts, and the identities of the generating models are as reported in the RAGTruth paper; the exact per-task counts used in each split of this thesis are listed in Appendix C.

Different experiments require different slices of this data, and keeping those slices separate is part of the locked-test discipline. The two detector phases evaluate on the official RAGTruth test split of 2,700 responses. Under the one-touch rule this split was evaluated exactly three times in the whole project, once for the committed configuration of Phase A and twice for the two committed configurations of Phase F, and no other phase reads it. The span diagnostic of Phase G works entirely on the validation split of 942 responses, so it adds no test exposure. On the repair side, pipelines were developed and compared on development populations, and the selected pipeline was then evaluated once on the locked repair population of 732 records. Phase D, the tie-break counterfactual, re-reads only the stored development outputs of Phase B and runs no new models. Phase E re-uses the locked repair population, which is legitimate for its purpose: it tests a mechanism claim about the pipeline that had already been selected, and selects nothing new, so the re-use creates no second selection event on locked data.

## 3.3 Detector task and metrics

The first stage of the pipeline is detection. A detector receives the source passages and a generated response and must answer two questions: does this response contain hallucinated content, and if so, where. Performance on the first question is measured by the F1 score over the binary hallucinated label, called *response-level F1*. Performance on the second is measured by the F1 score over the overlap between predicted and annotated character spans, called *span-level F1*. Both follow the evaluation protocol of the RAGTruth paper, whose published fully fine-tuned detector serves as the external baseline for the detector phases [14].

Formally, let a response be labelled hallucinated or clean. Writing TP, FP, and FN for true positives, false positives, and false negatives of the hallucinated class at the response level,

$$\mathrm{F1}_{\mathrm{resp}} = \frac{2\,\mathrm{TP}}{2\,\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}. \tag{3.1}$$

Span-level performance is scored on character offsets. For a response, let $P$ be the set of character positions the detector marks as hallucinated and $G$ the set of positions marked hallucinated in

the gold annotation. Precision and recall are computed on the overlap of these position sets, and the span-level F1 is their harmonic mean,

$$\text{prec} = \frac{|P \cap G|}{|P|}, \quad \text{rec} = \frac{|P \cap G|}{|G|}, \quad \text{F1}_{\text{span}} = \frac{2\,\text{prec} \cdot \text{rec}}{\text{prec} + \text{rec}}, \tag{3.2}$$

with the score aggregated over the evaluation population. A response on which the detector predicts no hallucinated span contributes no positive predictions, so a missed hallucinated response contributes zero span-level recall for that response, a point that becomes central to the diagnostic of Chapter 6.

The detectors in this thesis are not trained by updating all model weights. They use QLoRA, a parameter-efficient method in which the pre-trained model is frozen in quantised form and only small adapter matrices are trained [3, 7], applied to Llama-2 base models [17]. This choice makes 13-billion-parameter training affordable on the available hardware, and it also sharpens the research question: the detector phases ask whether scaling helps under adapter-based training specifically. Two training-configuration variables matter throughout. The *context length* is the number of tokens of the concatenated source and response the model sees during training; anything beyond it is truncated and never seen. The *epoch count* is how many passes the training makes over the data. Phase A, the Detector Scaling Sweep, fixes these at 1,024 tokens and one epoch while scaling the model; Phase F, the Same-Family Detector Re-scaling, varies them. Why these two variables turn out to matter is the subject of Chapter 4, not of this chapter.

## 3.4 Repair pipelines and pipeline variants

Detection alone does not help an end user; the flagged response must also be repaired. A *repair pipeline* takes a response, verifies its content against the source, and produces a revised response in which unsupported material has been removed or corrected. The central design decision in a repair pipeline is its granularity, the size of the unit it verifies and acts on. A *sentence-level* pipeline checks whole sentences and acts on any sentence that fails verification. This is simple and conservative, but blunt: if a single fact inside an otherwise correct sentence is wrong, the whole sentence is at risk. A *claim-level* pipeline first decomposes the response into atomic factual claims, verifies each claim separately, and then reconstructs a response from the claims that survive. This can target errors precisely, but it inherits new dependencies: the decomposition must be sound, the verifier must judge claims reliably, and the reconstruction must produce readable text. A worked example of claim decomposition and reconstruction is given in Appendix B.1.

The thesis compares five concrete pipeline variants, and since their short names appear throughout Chapters 5 and 6, they are introduced here once, in plain terms. The baseline, called **V0**, is the sentence-level pipeline inherited from the earlier stage of this project: it verifies sentences and removes those that fail. The main claim-level candidate, called **V1f**, decomposes the response into claims and drops any claim the verifier rejects; it additionally contains a rewriting pathway, in which a language model may rephrase problematic content instead of deleting it, and an extractive fallback, which assembles the output from verified source material when rewriting is unavailable. How often these extra pathways actually operate is an empirical question that Chapter 5 answers, and it matters for interpreting what the pipeline really does. The variant called **V1cdo**, short for claim-drop-only, is the deliberately impoverished version of V1f built for the mechanism ablation of Phase E: identical decomposition and verification, but with the rewriting and fallback machinery disabled, so that dropping failed claims is the only repair action available. Finally, two further candidates enter through the expanded candidate set of Phase D, the Tie-Break Counterfactual. The variant called **V1e** is a claim-level repair pipeline closely related to V1f, but without V1f's final do-no-harm rewrite gate, the last check that blocks a rewrite unless it is verified not to introduce new content. The variant called **V2** is an alternative

claim-level design that decomposes the response into more atomic propositions before verification, trading coarser claims for finer-grained checking. The verbatim pre-registered definitions of both are reproduced in Appendix A; what matters for this chapter is that both are evaluated under exactly the same criteria, on the same development population, as every other candidate.

## 3.5 Verifier protocols

Both pipeline granularities depend on a *verifier*, the component that decides whether the source supports a given sentence or claim. The standard framing is natural language inference [4]: the source acts as the premise, the sentence or claim as the hypothesis, and the verifier estimates whether the premise entails the hypothesis. The earlier stage of this project used a cross-encoder verifier of this kind, a DeBERTa-v3-large model fine-tuned for natural language inference [6, 4, 18]. Cross-encoders are fast and produce probability-like scores, but they are fixed-task models. Phase B, the Verifier Substitution and Repair experiment, replaces this verifier with a prompted large language model, Qwen-2.5-32B-Instruct [16], under a pre-registered amendment. The prompted verifier returns a structured verdict with an associated probability; outputs that cannot be parsed under the protocol are counted and reported rather than silently discarded.

One property of a verifier matters for repair beyond its accuracy: the shape of its probability distribution. Consider the fraction of all verification events whose probability falls strictly between 0.10 and 0.90, that is, in the uncertain middle of the scale rather than at the confident extremes. The thesis calls this fraction the verifier's *middle mass*. The concept matters because the available repair actions depend on it. A verifier with substantial middle mass expresses graded uncertainty, which can support a three-way policy: keep what is confidently supported, drop what is confidently unsupported, and rewrite what lies in between. A verifier whose probabilities pile up at the extremes behaves like a binary classifier; the middle band is nearly empty, rewriting conditions are rarely triggered, and the pipeline is pushed towards keep-or-drop behaviour regardless of how sophisticated its rewriting machinery is. On the development data, the substituted verifier of Phase B placed 13.33% of its verification events in this middle band, a figure that becomes the reference point for the calibration check of Phase C, described in the next section.

## 3.6 Repair metrics and locked criteria

Evaluating repair requires more than counting removed hallucinations, because the cheapest way to remove every hallucination is to delete the entire response. The thesis therefore measures repair at two levels, with metrics that pull against each other.

At the claim level, computed over the 5,439 claims of the locked repair population, four quantities are tracked. The *delta unsupported rate* is the change in the proportion of unsupported claims after repair compared with before; negative values mean the repair removed unsupported content, which is the goal. The *damage rate* is the proportion of repairs that introduce new unsupported content; an acceptable repair system must hold this near zero. The *claim survival rate* is the proportion of supported claims that the repair retains, and *length retention* is the proportion of the response's length that survives. The last two are the guard rails against degenerate deletion: a pipeline that empties responses would score well on the first metric and collapse on these.

Writing these over the locked population, let $U_{\text{orig}}$ and $U_{\text{rep}}$ be the number of unsupported claims before and after repair, $N$ the total claims, $S$ the number of supported claims in the original responses, and $S_{\text{kept}}$ the supported claims surviving repair. Let $L_{\text{orig}}$ and $L_{\text{rep}}$ be total response length before and after repair, and let $R$ be the number of repaired claims that introduce new

Table 3.3: The four-gate success criterion shared by Phase B and Phase D. A winning pipeline must satisfy all four gates simultaneously on the locked population; the clause order is also the tie-break order. Thresholds are as fixed in the Phase B pre-registration, reproduced verbatim in Appendix A.

| Clause | Gate quantity | Threshold | Reading |
|---|---|---|---|
| 1 | Delta unsupported rate | $\geq 1.0$ pp better than V0 | More negative is better |
| 2 | Damage rate | $\leq 5.0\%$ | No new hallucination |
| 3 | Claim survival rate | $\geq 70.0\%$ | Retain supported content |
| 4 | Length retention | $\geq 60.0\%$ | Retain response length |

unsupported content. Then

$$\Delta_{\text{unsup}} = \frac{U_{\text{rep}} - U_{\text{orig}}}{N}, \quad \text{damage} = \frac{R}{N}, \quad \text{survival} = \frac{S_{\text{kept}}}{S}, \quad \text{length} = \frac{L_{\text{rep}}}{L_{\text{orig}}}. \tag{3.3}$$

A negative $\Delta_{\text{unsup}}$ means repair removed unsupported content; the four-gate criterion of Table 3.3 requires $\Delta_{\text{unsup}}$ at least one percentage point below the V0 baseline, damage $\leq 0.05$, survival $\geq 0.70$, and length $\geq 0.60$ simultaneously.

At the response level, an independent post-hoc *judge*, built on the same Qwen-2.5-32B model family [16], labels every original and every repaired response as hallucinated or clean. Pipelines are then compared by the change in the judge-assessed hallucination rate, and the statistical significance of that change is assessed with McNemar's test, the standard test for paired binary labels [13]. Because this judge shares a model family with the verifier, Appendix D repeats the judging with two unrelated models to check robustness.

With the metrics in place, the meaning of a Phase B "win" can be made precise. The pre-registered success criterion is conjunctive across four gates: to win, a candidate pipeline must simultaneously satisfy all four conditions of Table 3.3 on the locked population. In summary, the pipeline must reduce the unsupported-claim rate by at least one percentage point more than the sentence-level baseline V0 achieves, must introduce new unsupported content in at most 5% of repairs, must retain at least 70% of supported claims, and must preserve at least 60% of response length. The first gate rewards effective repair; the remaining three are guard rails that disqualify destructive or careless repair. The verbatim pre-registration wording is reproduced in Appendix A. The same four clauses double as the tie-break order: if several candidates pass all gates, the clauses are compared in the listed order until one candidate is strictly better. This ordering mattered in Phase D, where the first two clauses tied across the development candidates and the third clause, claim survival, determined the selection.

Phase C, the Reasoning Verifier Swap (here *swap* denotes a change of prompting protocol on the same verifier model, as distinct from the model substitution of Phase B), carries one further pre-registered check that needs introducing here because its name is opaque without context. The phase asks whether prompting the verifier to reason step by step changes not only its accuracy but also its calibration, in the middle-mass sense of Section 3.5. The pre-registration therefore includes a *calibration tripwire*: a condition, fixed in advance, that records a specific failure if it occurs. The tripwire is defined against the 13.33% middle mass of the non-reasoning verifier: if the reasoning verifier's middle mass remains at or below 26.65%, the tripwire condition is met, recording that step-by-step reasoning did not produce the graded uncertainty profile that a rewriting policy would need. Whether it was met is reported in Chapter 5.

## 3.7 Compute environment and reproducibility

All model training and large-model inference ran as batch jobs on two GPU clusters, the Snellius national supercomputer operated by SURF and the ALICE cluster of Leiden University, scheduled through SLURM. Detector fine-tuning used QLoRA on single GPU nodes; verifier and judge inference ran as batched jobs over the populations of Section 3.2. Every phase is governed by a written pre-registration with committed criteria and, where applicable, amendments; these documents, together with the result files and the robustness outputs, are archived in the project repository. Commit identifiers, job configurations, and a file inventory appear in Appendix C, so that each number in Chapters 4 to 6 can be traced to a stored artefact. One labelling issue is recorded for transparency: the stored result file for the Phase F detector contains an internal description field referring to the smaller 7-billion-parameter model while holding the 13-billion-parameter results; the file inventory in Appendix C documents this. The cost of the locked-test discipline can be stated in one number: across the entire project, the RAGTruth test split was evaluated three times. The benchmark consists of publicly released model outputs with published human annotations and contains no personal or human-subject data, so no additional ethical approval was required.

# Chapter 4

# Detector Scaling: Phases A and F

## 4.1 Purpose of the detector-scaling experiments

The first stage of the pipeline introduced in Chapter 3 is detection: given the source passages and a generated response, decide whether the response contains unsupported content and locate it. Both phases of this chapter concern the detection stage only. Their input is the source passages together with the generated response, and their output is a response-level label and predicted hallucination spans; verification, repair, and judging play no role here. This chapter reports the two locked-test experiments that ask whether a larger detector is a better detector. The question is less obvious than it sounds. The detectors here are trained with QLoRA, the adapter-based method described in Section 3.3 in which the base model stays frozen and only small adapter matrices are trained [3, 7], and a larger frozen base does not automatically translate into better adapters. Phase A, the Detector Scaling Sweep, tested scaling under one fixed training configuration and failed its locked criteria. Phase F, the Same-Family Detector Re-scaling, tested scaling again under a revised training configuration and passed. The two verdicts are not in tension, and the burden of this chapter is to show why: what failed in Phase A was one specific training configuration, and what succeeded in Phase F was scaling once the training setup respected the data.

## 4.2 Phase A: Detector Scaling Sweep

At the start of the project, the working detector was based on Qwen-2.5-7B-Instruct [16], reaching 74.8 response-level F1 and 37.4 span-level F1 on the RAGTruth test split of 2,700 responses (the two metrics are defined in Section 3.3: the first scores the binary hallucinated-or-not decision, the second scores how well the predicted spans overlap the annotated ones). This model is the incumbent project baseline: the system the project already had, not a scaling comparator chosen for experimental cleanliness. The published RAGTruth baseline, a fully fine-tuned 13-billion-parameter detector, stood considerably higher, at 78.7 and 52.7 [14]. Phase A asked whether moving to a 13-billion-parameter base model, Llama-2-13B [17], trained with QLoRA, could close that gap.

The phase swept eighteen training configurations of the Llama-2-13B adapter on the development data, varying adapter and optimisation settings while holding two configuration variables fixed: a training context length of 1,024 tokens and a single training epoch. The best configuration by development performance was then committed, and the test split was evaluated once for that committed configuration, under the one-touch discipline of Section 3.1. The pre-registered success criterion required the scaled detector to reach at least 76.7 response-level F1 or 42.4

Table 4.1: Phase A locked-test result on the RAGTruth test split ($n$=2,700), with the project baseline and the published fully fine-tuned baseline for reference. The supplementary three-epoch run is discussed in Section 4.7.

| Detector | Response F1 | Span F1 |
|---|---|---|
| Project baseline (Qwen-2.5-7B, existing) | 74.8 | 37.4 |
| Phase A winner (Llama-2-13B QLoRA, 1,024 ctx, 1 epoch) | 68.9 | 20.6 |
| Supplementary run (as above, 3 epochs) | 72.2 | 25.0 |
| Published baseline (Llama-2-13B, full fine-tune) [14] | 78.7 | 52.7 |

span-level F1, improvements over the existing baseline rather than mere parity. (The criterion is disjunctive, as locked in Appendix A.1; the Phase A winner met neither bound.)

## 4.3 Phase A result and locked verdict

Table 4.1 reports the locked-test outcome alongside the two reference points. The scaled detector reached 68.9 response-level F1 and 20.6 span-level F1. It did not approach the pre-registered thresholds; it fell below the smaller model it was meant to replace, by 5.9 points at the response level and by 16.8 points at the span level, and it fell far below the published fully fine-tuned baseline. The locked verdict of Phase A is therefore FAILURE, and that verdict is final under the pre-registration discipline: no post-hoc adjustment of thresholds or re-evaluation was permitted, and none was performed.

## 4.4 What Phase A did and did not show

Read precisely, Phase A established one thing: under this specific training setup, a 13-billion-parameter QLoRA detector trained with a 1,024-token context for one epoch, scaling did not help and actively hurt, most severely at span level. That is a real and useful result, because this configuration is the kind a practitioner with a fixed compute budget would plausibly choose first.

Phase A did not establish that larger detectors are worse in general, and it could not, for two reasons that were visible in its own design. The first is a confound: the comparison crossed model families, setting a Llama-2 candidate against a Qwen-2.5 incumbent, so the effect of scale was entangled with the effect of switching families. The comparison was inherited rather than chosen: the incumbent happened to be a Qwen model, and the scaling candidate followed the published RAGTruth setup in using Llama-2. The second is that the fixed configuration variables were themselves suspects. A 1,024-token window must hold the source passages and the response together, and a single epoch gives the randomly initialised adapters one pass over the data. Either constraint, rather than scale, could explain the collapse. The span-level damage is also larger than the response-level damage, a pattern in which a wrong response-level decision forfeits the span predictions for that response entirely; this failure mode is examined directly in Chapter 6. Distinguishing the training setup from the scaling hypothesis required a second, separately pre-registered experiment with a clean within-family comparison, which is exactly what Phase F is.

## 4.5 Phase F: Same-Family Detector Re-scaling

Phase F re-tested detector scaling with both suspects addressed. The comparison was kept within a single model family, Llama-2 at 7 and 13 billion parameters [17], so that scale is the only model

Table 4.2: Phase F locked-test results on the RAGTruth test split ($n$=2,700). Both detectors share the revised training configuration (2,048-token context, three epochs); only the model size differs.

| Detector | Response F1 | Span F1 |
|---|---|---|
| Llama-2-7B QLoRA (revised configuration) | 66.0 | 36.6 |
| Llama-2-13B QLoRA (revised configuration) | 79.6 | 46.0 |
| Published baseline (Llama-2-13B, full fine-tune) [14] | 78.7 | 52.7 |

variable. The training configuration was revised on both axes flagged by Phase A: the context length was doubled to 2,048 tokens, and training ran for three epochs. As in Phase A, this is a detection-only experiment; no repair component is involved. Both model sizes were trained under this configuration, each was committed in advance, and the test split was evaluated once per committed configuration, bringing the project's cumulative test exposure to three touches in total (Section 3.7). The pre-registered primary criterion asked whether the 13-billion-parameter detector beats its 7-billion-parameter sibling at response level; the secondary criterion concerned span-level improvement.

## 4.6 Phase F result and locked verdict

Table 4.2 reports the outcome, and Figure 4.1 places the two phases side by side. Under the revised configuration, scaling delivered a large gain on both pre-registered criteria: the 13B detector reached 79.6 response-level F1 against 66.0 for the 7B detector, an improvement of 13.6 percentage points (pp), and 46.0 span-level F1 against 36.6, an improvement of 9.4 points. The locked verdict of Phase F is WIN on both criteria. Two readings of the table deserve note. First, the 13B QLoRA detector's response-level score of 79.6 matches the published fully fine-tuned baseline of 78.7 (a margin of 0.9 points), although the comparison should be made with care since training details differ; adapter-based training at matched context proved sufficient for response-level detection on this benchmark. Second, the span-level score of 46.0, improved by 9.4 points, remains 6.7 points below the published baseline's 52.7. Detection of whether a response hallucinates has caught up; localisation of where has not, and Chapter 6 diagnoses why.

## 4.7 Shared mechanism: context length, epochs, and same-family scaling

Why did the same scaling step fail in one phase and succeed in the other? The key variable is what the detector is allowed to see during training. Detector training concatenates the source passages and the response into one sequence; whatever exceeds the context window is truncated, simply cut off, and the model never observes it. At the 1,024-token window of Phase A, this was not a marginal effect: 33.1% of summarisation training examples and 47.7% of data-to-text training examples were truncated. For nearly half of the data-to-text data, the detector was being taught to judge faithfulness against evidence it could not read, and span-level supervision pointing into the truncated region was lost outright. This is consistent with the signature of the Phase A failure, a collapse most severe at span level. Doubling the window to 2,048 tokens in Phase F directly reduces the share of truncated examples and restores the correspondence between what the annotations assert and what the model sees.

The number of epochs is the second, smaller factor, and the project has direct evidence separating the two. A supplementary Phase A run, documented in the phase's amendments, kept the

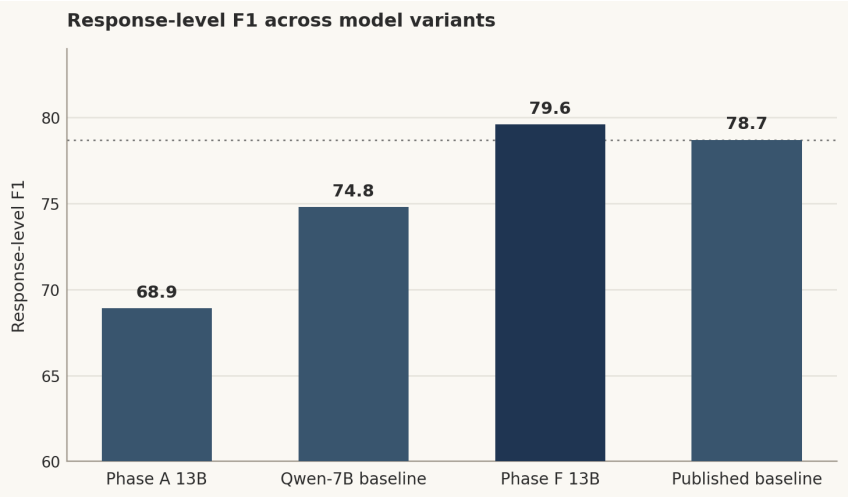



Figure 4.1: Response-level F1 across the detector experiments. Under the constrained training configuration the scaled detector (Phase A, 68.9) falls below the existing Qwen-2.5-7B baseline (74.8); under the revised configuration the same scale step (Phase F, 79.6) matches the published full fine-tuning baseline (78.7, dotted line). Test split, $n$=2,700.

1,024-token window but trained for three epochs: performance improved from 68.9 to 72.2 response-level F1 and from 20.6 to 25.0 span-level F1, yet remained below even the existing baseline (Table 4.1). Longer training helps the adapters converge, but it cannot teach the model content it never sees. Context length is the binding constraint; epochs are secondary.

The third element, keeping the comparison within one model family, contributes no points by itself but makes the win interpretable: because the 7B and 13B detectors of Phase F differ only in scale, the 13.6-point gap is attributable to scale, in a way the cross-family comparison of Phase A never allowed. Taken together, the two phases support a conditional conclusion rather than a contradiction. Scaling the detector does help, substantially, but only once the training configuration lets the larger model see the data the task is actually about.

## 4.8 Detector-side design rule

The detector phases reduce to one design rule for practitioners building hallucination detectors with parameter-efficient training. Before spending parameters, match the training context length to the token statistics of the task: measure the source-plus-response lengths, and choose a window that the training examples actually fit. Then train adapters to convergence rather than for a fixed single epoch, and when evaluating a scaling decision, compare within a model family so that scale is not confounded with architecture. Under those conditions, on this benchmark, scaling from 7 to 13 billion parameters was worth 13.6 response-level F1 points; ignoring them, the same 13-billion-parameter model lost 5.9 points to a smaller incumbent. The remaining detector weakness, span-level localisation, is not a training-configuration problem of the same kind, and Chapter 6 returns to it after the repair side of the pipeline has been examined in Chapter 5.

# Chapter 5

# Verifier Scaling and Repair: Phases B, C, D, and E

## 5.1 Purpose of the verifier and repair experiments

Chapter 4 examined the first stage of the pipeline, detection. This chapter examines the remaining stages: verification and repair. The full repair pipeline, introduced in Chapter 3, runs as follows. A generated response is decomposed into units, either sentences or individual factual claims. A verifier checks each unit against the source documents. A repair action is then applied to each unit, keeping it, dropping it, rewriting it, or, where rewriting is unavailable, substituting verified source material through an extractive fallback. The reassembled response is finally assessed by an automatic judge, which labels it hallucinated or clean. The four phases of this chapter probe different parts of this pipeline while the detector of Chapter 4 plays no role: Phase B substitutes a much larger verifier and runs the locked repair comparison; Phase C changes only the verifier's reasoning protocol; Phase D is a post-hoc counterfactual about which candidate pipelines were allowed to compete; and Phase E disables components of the winning pipeline to identify the mechanism behind its result.

The story these phases tell can be stated in advance, so the reader knows what the evidence is building towards. Scaling the verifier changed what repair could achieve, and a claim-level pipeline passed every locked criterion. But the successful behaviour was not rewriting. The substituted verifier judges with near-binary confidence, and under such a verifier the action that does the work is the simple one: removing claims that fail verification. Reasoning prompts improved some verdicts without changing that confidence shape, and the ablation of Phase E shows that removing the sophisticated machinery leaves the headline outcome intact.

## 5.2 Phase B setup: verifier substitution and repair candidates

The earlier stage of this project evaluated repair with a sentence-level pipeline, called V0, in which a cross-encoder verifier, a DeBERTa-v3-large natural-language-inference model [6], checked each sentence of a response and failing sentences were removed. Phase B, the Verifier Substitution and Repair experiment, asked what changes when the verifier is replaced by a model two orders of magnitude larger: Qwen-2.5-32B-Instruct [16], prompted to verify each unit against the source and to return a structured verdict with a probability. The substitution was committed in a pre-registered amendment before any locked evaluation ran.

The substitution had an immediate measurable consequence on the sentence-level baseline itself.

Table 5.1: Phase B locked-test result for V1f on the locked repair population ($n$=732 records, 5,439 claims).

| Gate | Threshold | Achieved | Outcome |
|---|---|---|---|
| Delta unsupported rate | $\geq 1.0$ pp better than V0 | $-7.50$ pp | Pass |
| Damage rate | $\leq 5.0\%$ | 0.00% | Pass |
| Claim survival rate | $\geq 70.0\%$ | 84.20% | Pass |
| Length retention | $\geq 60.0\%$ | 86.52% | Pass |

On the verifier development set of 1,046 records, V0's response-level agreement with the gold annotations fell from 62.40% under the previous cross-encoder to 44.79% under the new verifier. This was not a malfunction but a change of judging standard: the prompted verifier applies stricter and differently distributed judgements than the cross-encoder it replaced, and every pipeline in the phase is evaluated under the same new standard.

Against this baseline, the main claim-level candidate is the pipeline called V1f. As defined in Chapter 3, V1f decomposes the response into atomic factual claims, verifies each claim, and drops claims that fail; it also carries a rewriting pathway, in which a language model may rephrase problematic content, and an extractive fallback that assembles output from verified source material when rewriting is unavailable. Development and comparison used the repair development set of 314 records; the final evaluation ran once on the locked repair population of 732 records, of which 269 (36.7%) carry gold hallucination annotations, under the four-gate criterion of Section 3.6. To restate the gates briefly: the winning pipeline must reduce the unsupported-claim rate by at least one percentage point more than V0, introduce new unsupported content in at most 5% of repairs, retain at least 70% of supported claims, and preserve at least 60% of response length, all simultaneously.

## 5.3 Phase B locked result: V1f passes the four gates

Table 5.1 reports the locked-test outcome for V1f on the 732-record population, covering 5,439 claims. All four gates pass, none narrowly. The unsupported-claim rate fell by 7.50 percentage points. The damage rate was 0.00%: no repair introduced new unsupported content. Claim survival was 84.20%, well above the 70% floor, and length retention was 86.52%, well above the 60% floor; in total the pipeline removed 17.14% of claims. For comparison, the same pipeline's development-set reduction was 4.18 percentage points on 314 records; the development and locked figures are different numbers on different populations and are never interchanged in this thesis.

The response-level judge tells the same story from outside the pipeline. Before repair, the judge labelled 399 of the 732 responses (54.5%) as hallucinated; after V1f repair, 332 of 732 (45.4%), a reduction of 9.15 percentage points. Under McNemar's test for paired labels the change is statistically significant ($p = 1.7 \times 10^{-5}$). The locked verdict of Phase B is therefore WIN for V1f: the claim-level pipeline, under the substituted verifier, satisfied every pre-registered criterion.

One further measurement from the locked run matters more than any other for interpreting this verdict: across the entire locked evaluation, V1f's rewriting pathway triggered on 0.00% of claims. Not a single repair was produced by the language-model rewriter. The reduction therefore cannot be attributed to rewriting; Section 5.4 identifies which component carried it.

Table 5.2: Confidence profile of the substituted Qwen-2.5-32B verifier on the verifier development set. The middle band is the region in which a rewrite action would be considered.

| Verifier probability region | Share of verification events |
|---|---|
| Below 0.10 (confident fail) | 15.60% |
| Middle band, 0.10 to 0.90 | 13.33% |
| Above 0.90 (confident pass) | 71.08% |

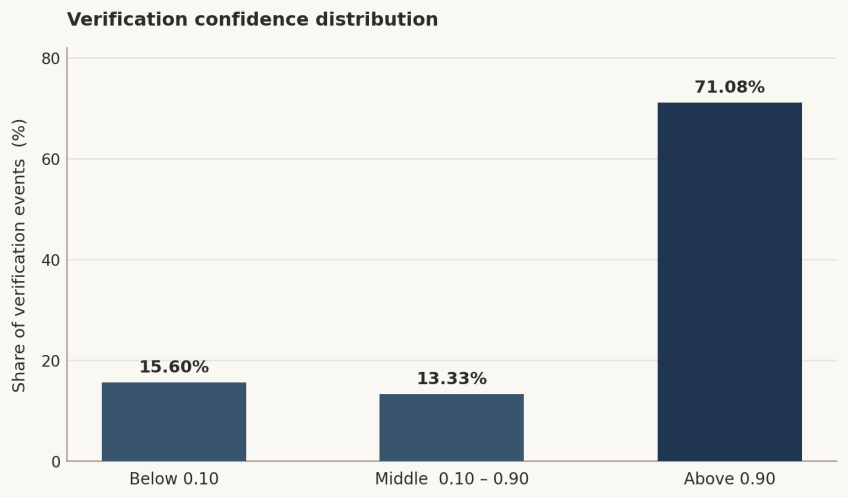



Figure 5.1: Confidence profile of the substituted Qwen-2.5-32B verifier on the development set. Most verification events fall at the confident extremes; the middle band, where a rewrite decision would apply, holds only 13.33%. This shape is why claim-level dropping, not rewriting, carries the repair result.

## 5.4 Why V1f wins: verifier confidence profile and repair granularity

The explanation begins with the shape of the verifier's confidence. Recall from Section 3.5 that the middle mass of a verifier is the fraction of its verification events whose probability falls strictly between 0.10 and 0.90, the uncertain region where a rewrite decision would live. Table 5.2 and Figure 5.1 show the substituted verifier's profile on the development data: 71.08% of verification events fall above 0.90, 15.60% below 0.10, and only 13.33% in the middle band. The verifier behaves, for most practical purposes, like a binary classifier: it commits.

This shape determines which repair actions can operate. A rewriting policy needs the verifier to say "uncertain" often enough for rewrite conditions to trigger; a verifier that almost always commits leaves the rewriter idle. That is exactly what the 0.00% trigger rate records. What remains active is the simplest action in the pipeline: *claim-level drop-on-fail*, meaning that each atomic claim the verifier rejects is removed and the rest of the response is kept. The granularity is what makes this action safe. When a sentence-level pipeline drops a failing unit, it loses the whole sentence, supported material included; when a claim-level pipeline drops a failing unit, it loses one atomic claim. Under a committing verifier, claim-level drop-on-fail removes unsupported content precisely while leaving supported content largely untouched, which is the combination the four gates reward: a large reduction in unsupported claims alongside high survival and length retention. The headline result of Phase B should therefore be read carefully: it is a win for matching repair granularity to verifier behaviour, not a win for sophisticated repair machinery. This mechanism claim is tested directly, by ablation, in Phase E (Section 5.8).

Table 5.3: Phase C verdict summary. The criteria are conjunctive: all parts must succeed for the phase to succeed. Values are measured on development data.

| Criterion | Requirement | Observed | Outcome |
|---|---|---|---|
| Verdict accuracy | Improve over 44.79% | 54.06% | Met |
| Lead narrowing | Claim-level lead $\leq 3.12$ pp | 3.32 pp | Missed by 0.20 pp |
| Calibration tripwire | Middle mass $> 26.65\%$ | 15.98% | Tripwire fired (failed) |

## 5.5 Phase C: Reasoning Verifier Swap

If the obstacle to rewriting is the verifier's near-binary confidence, an obvious response is to change how the verifier thinks. Phase C, the Reasoning Verifier Swap, tested whether prompting the same Qwen-2.5-32B verifier to reason step by step, in the chain-of-thought style [19], would both improve verdict accuracy and produce the graded confidence profile that rewriting requires. The pipeline architecture is unchanged from Phase B; the only modification is the verifier's prompting protocol, which now elicits explicit reasoning before the verdict. The phase is diagnostic and ran on development data only.

The pre-registered criteria were conjunctive, with two parts and a tripwire. The accuracy part asked whether reasoning improves the sentence-level baseline's agreement with gold annotations. The lead-narrowing part asked whether the improvement is large enough to matter for pipeline choice: if reasoning lifts the sentence-level baseline sufficiently, the claim-level pipeline's advantage should shrink below a pre-registered ceiling of 3.12 percentage points. The calibration tripwire of Section 3.6 records a failure if the reasoning verifier's middle mass remains at or below 26.65%, that is, if reasoning fails to at least double the 13.33% middle mass of the non-reasoning verifier.

## 5.6 Phase C result and locked failure

Table 5.3 summarises the three criteria. Reasoning did improve accuracy: the sentence-level baseline's response-level agreement with gold rose from 44.79% to 54.06%, a substantial and statistically significant change in individual verdicts. It also improved repair: under reasoning verdicts, the sentence-level pipeline's unsupported-claim reduction reached 3.61 percentage points, against 6.93 for the claim-level pipelines. But that is exactly the problem: the claim-level lead stood at 3.32 percentage points, above the 3.12-point ceiling, missing the lead-narrowing criterion by 0.20 percentage points. And the calibration tripwire was met: the reasoning verifier's middle mass was 15.98%, barely above the non-reasoning 13.33% and far below the 26.65% threshold. Step-by-step reasoning changed many verdicts without changing the verifier's character; it still commits.

Because the criteria are conjunctive, the combined Phase C verdict is FAILURE, and the thesis reports it as such, including the 0.20-point margin. The result is two-sided. Reasoning is not useless: it improved verdict accuracy considerably, and it narrowed the gap between cheap sentence-level repair and claim-level repair. But it did not deliver the specific property it was hired for. The rewrite pathway needs graded uncertainty, and reasoning produced more accurate commitment instead. For the question of this chapter, the mechanism of Section 5.4 stands: under this verifier family, with or without reasoning, repair lives in the keep-or-drop regime.

## 5.7 Phase D: Tie-Break Counterfactual

Phase B's locked comparison evaluated a committed set of candidate pipelines, and V1f won. Phase D, the Tie-Break Counterfactual, asks a methodological question about that setup: how much did the outcome depend on which candidates were allowed to compete? It is an explicitly labelled post-hoc analysis, performed after the Phase B results were known, and it is a counterfactual in the strict sense of Section 3.1: it re-reads the stored development outputs of Phase B, on the 314-record development set, and runs no new models. Because it is post-hoc, it cannot and does not modify Phase B's committed verdict; its value is in documenting how a defensible alternative choice, made before locking, would have changed the selection.

The counterfactual expands the candidate set with two further claim-level variants defined in Chapter 3: V1e, which is V1f without the final do-no-harm rewrite gate, and V2, which decomposes responses into finer-grained propositions before verification. The selection procedure is the locked one, applied unchanged: each candidate faces the four gates, and ties are broken by comparing the gate quantities in clause order. On the development data the decision flow runs as follows. All three candidates reduce the unsupported-claim rate by the same 4.18 percentage points, so clause 1 ties; all satisfy the damage bound, so clause 2 ties; clause 3, claim survival, then decides, and there the candidates separate cleanly: 88.55% for V1e, 84.60% for V1f, and 74.36% for V2. Under the expanded set, the locked tie-break selects V1e. The clause-by-clause walkthrough, including length retention, is given with the Phase D documentation in Appendix A.

The finding is methodological rather than operational. The same locked rules, fed a different but defensible candidate set, select a different winner, and the deciding criterion is not repair power, on which the candidates tied, but how much supported content each preserves. Candidate-set choice is therefore itself a consequential pre-registration decision, of the same kind as a threshold or a metric priority, and it deserves the same advance commitment.

## 5.8 Phase E: Mechanism Ablation

Section 5.4 advanced a mechanism claim: V1f's locked result is carried by claim-level drop-on-fail, not by its rewriting or fallback machinery. Phase E, the Mechanism Ablation, tests that claim in the most direct way available, by switching the machinery off. The ablated pipeline, V1cdo (claim-drop-only, defined in Chapter 3), uses the identical claim decomposition and the identical verifier as V1f, but its rewrite pathway and extractive fallback are disabled; removing failed claims is the only action it has. If the mechanism claim is right, V1cdo should reproduce V1f's outcome; if the machinery mattered, removing it should hurt.

Phase E re-uses the 732-record locked population of Phase B. The re-use is legitimate for this purpose and was pre-registered as such: the phase selects nothing, declares no new winner, and exists only to attribute an already-committed result to a component, so it creates no second selection event on locked data.

Table 5.4 reports the comparison. At the claim level, V1cdo reduced the unsupported rate by 6.67 percentage points against V1f's 7.50, a difference of 0.83 points, within the pre-registered equivalence bound of 1.00 point. Its claim survival (78.19%) and length retention (79.30%) sit below V1f's, as expected for a pipeline with no way to salvage content except keeping it. At the response level the two pipelines are aggregate-equivalent under the judge: each leaves exactly 332 of 732 responses labelled hallucinated, the same 9.15-percentage-point reduction. Aggregate equivalence is not per-record identity, and the Phase E results do not support that stronger reading: the two pipelines agree on 634 of 732 records (86.6%), and the 98 disagreements flip in both directions in exactly balanced numbers, 49 records where only V1f's output is judged hallucinated and 49 where only V1cdo's is.

Table 5.4: Phase E mechanism ablation on the locked repair population ($n$=732). V1f and V1cdo share the same claim decomposition and the same verifier; they differ only in the repair actions available.

| | V1f (full pipeline) | V1cdo (ablation) |
|---|---|---|
| Repair actions available | Drop, rewrite, extractive fallback | Drop only |
| Delta unsupported rate | −7.50 pp | −6.67 pp |
| Claim survival rate | 84.20% | 78.19% |
| Length retention | 86.52% | 79.30% |
| Judge: hallucinated after repair | 332 of 732 (−9.15 pp) | 332 of 732 (−9.15 pp) |

The detail behind the balanced flips explains where V1f's extra machinery went. The rewrite pathway, as Section 5.3 reported, never fired. The extractive fallback did fire, on 45.1% of the locked records, and it changes text: only 402 of the 732 record pairs are byte-identical between the two pipelines, and on those identical records the judge agrees with itself completely. On the 330 records where the fallback or related reconstruction produced different text, the judge's verdicts flip in both directions in equal measure. The fallback, in other words, is judge-neutral: it alters the wording of repaired responses without shifting the judged hallucination outcome in either direction. The verdict of Phase E is that the mechanism is identified: under this verifier, the component doing the repair work in V1f is claim-level drop-on-fail, and the remaining machinery changes form, not outcome. This is stated as what the ablation shows on this population and judge, not as a universal claim about rewriting.

## 5.9 Repair-side design rule

The four phases of this chapter compress into one design rule, the repair-side counterpart of Chapter 4's: match the repair granularity to the verifier's calibration. A verifier that commits, as this one does with 86.68% of its verification events at the confident extremes, supports keep-or-drop repair and rarely triggers rewriting; under such a verifier, claim-level dropping is both the safest and, by the evidence of Phase E, the operative action. Building rewrite machinery ahead of a verifier that can feed it is effort spent on a pathway that never runs. If rewriting is wanted, the binding investment is calibration, and Phase C indicates that prompting the same model to reason is not by itself sufficient to provide it.

Two scope notes bound these conclusions. First, the response-level evidence in this chapter comes from a judge built on the same model family as the verifier, Qwen-2.5-32B; family-shared blind spots are therefore possible in principle. The thesis treats judge-based results as relative comparisons under a checked judge, and Appendix D re-scores the repair outputs under two independent judge models from different families, confirming part of the picture and failing to confirm another part, reported there in full. Second, all repair results concern the RAGTruth populations and the tested pipeline implementations. With the repair side established, one detector question remains open from Chapter 4: why span-level localisation still trails response-level detection. Chapter 6 closes that loop.

# Chapter 6

# Span Localisation Diagnostic: Phase G

## 6.1 Purpose of the span diagnostic

Chapter 4 left one detector question open. The revised 13-billion-parameter detector of Phase F reached a response-level F1 of 79.6 on the test split, matching the published baseline, yet its span-level F1 of 46.0 still trailed the published 52.7. Phase G, the Span Localisation Diagnostic, asks why. Like the detector phases, it concerns only the detection stage of the pipeline: the input is the source passages and the generated response, and the output is a response-level label together with predicted hallucination spans. The verifier, repair, and judge stages of Chapter 5 play no role here. Unlike the detector phases, Phase G is a diagnostic, not a locked test: it declares no winner and changes no verdict. Its job is to separate two possible explanations for the span-level gap. Either the detector localises poorly, placing spans badly even when it knows a response is hallucinated, or its response-level mistakes cascade downward, because a response wrongly judged clean produces no useful spans at all. Which explanation dominates determines where improvement effort should go, and that is the question this chapter answers.

## 6.2 Phase G setup

The diagnostic works entirely on the RAGTruth validation split of 942 responses, using the stored detections of the Phase F detectors. No new training was performed, and the test split was not touched, so the project's test exposure remains the three evaluations accounted for in Section 3.7. Two metrics from Section 3.3 are involved: response-level F1, which scores the binary decision of whether a response contains hallucinated content, and span-level F1, which scores how well the predicted character spans overlap the annotated ones.

The central device is an *oracle subset analysis*. The 942 validation responses are divided by whether the detector's response-level decision was correct. On the full set, span-level F1 reflects every kind of error at once. On the response-correct subset, the response-level decision is right by construction, so the remaining span-level error measures localisation quality in isolation. Comparing the two values decomposes the span-level deficit: the gap between them is the part attributable to response-level misclassification, and the shortfall that remains on the oracle subset is the part attributable to span extraction itself. The phase was pre-registered with a wider set of diagnostic analyses and was terminated early under its Amendment G1 once this decomposition had answered the motivating question; the analyses not run are recorded in the Appendix A summary, and nothing from them is reported here.

Table 6.1: Phase G span decomposition on the RAGTruth validation split ($n$=942), with test-split values for reference. Validation and test are different splits; values are not directly comparable across them.

| Quantity | Split | Population | Span F1 |
|---|---|---|---|
| F-13B, full set | Validation | all 942 | 44.50 |
| F-13B, oracle response-correct subset | Validation | 793 of 942 | 53.56 |
| F-7B, full set | Validation | all 942 | 34.96 |
| F-7B, oracle response-correct subset | Validation | 724 of 942 | 53.66 |
| F-13B locked result (Chapter 4) | Test | all 2,700 | 46.0 |
| Published full fine-tune baseline [14] | Test | all 2,700 | 52.7 |

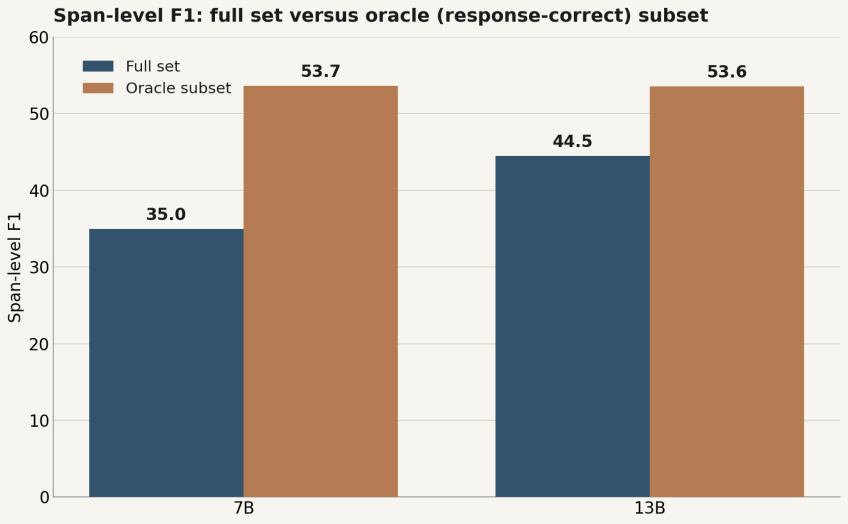



Figure 6.1: Span-level F1 on the full validation set versus the response-correct (oracle) subset, for both detector sizes. Full-set values are 34.96 (7B) and 44.50 (13B); oracle-subset values are 53.66 (7B) and 53.56 (13B). The two oracle bars are nearly equal, indicating that scale improved response-level judgement rather than span localisation. Validation split, $n$=942.

## 6.3 Diagnostic result

Table 6.1 reports the decomposition for both Phase F detectors, alongside the test-split reference values, and Figure 6.1 shows the four validation bars; the validation and test figures come from different splits and are not directly comparable, which the table marks explicitly. The 13-billion-parameter detector classified 793 of the 942 validation responses correctly at response level (84.2%), leaving 149 responses (15.8%) misclassified. Its span-level F1 on the full validation set is 44.50. On the response-correct subset, the same detector's span-level F1 rises to 53.56, an improvement of 9.05 percentage points obtained purely by conditioning on a correct response-level decision.

The smaller detector makes the same point more sharply. The 7-billion-parameter model's full-set span F1 is 34.96, but on its own response-correct subset (724 of 942 responses) it reaches 53.66, a gap of 18.69 percentage points. The two oracle values, 53.56 and 53.66, are nearly identical: once the response-level decision is correct, the two model sizes localise spans about equally well. This indicates that the span-level advantage the larger model showed in Phase F is carried mostly by its better response-level decisions, not by better span extraction within correctly judged responses.

## 6.4 Error taxonomy and cascade interpretation

The mechanism behind the decomposition is simple. Span prediction is conditioned on the response-level decision. If the detector wrongly judges a hallucinated response to be clean, it predicts no spans for that response, and every annotated span in it is missed outright; if it wrongly judges a clean response to be hallucinated, any spans it predicts are spurious. In both cases the span-level score is damaged before span extraction quality ever enters the picture. The thesis calls this the *response-misclassification cascade*: an error at the coarse decision level that propagates into the fine-grained metric. The remaining error category, imperfect localisation, covers the cases where the response-level decision is right but the predicted spans are misplaced, too short, too long, or incomplete. The oracle subset isolates exactly this category, and per-task breakdowns of both categories are given in Appendix B.

The numbers of Section 6.3 bound how much each category explains. For the 13-billion-parameter detector, conditioning away the cascade recovers 9.05 of the points separating its full-set value from the published baseline's range, and the near-equality of the two models' oracle values suggests that scale bought response-level judgement rather than localisation skill. The interpretation is deliberately bounded. The oracle analysis is a decomposition of correlated error sources, not a causal experiment: it shows that response-level misclassification accounts for a substantial part of the span-level deficit on this split, and that span extraction is responsible for the rest, since even the oracle values sit in the low fifties rather than near the ceiling. Phase G therefore supports the view that span localisation is not the only bottleneck, without claiming that it is no bottleneck at all.

## 6.5 Implication for detector design

The design implication completes the detector story of Chapter 4. The training-configuration fixes of Phase F, context length matched to the task's token statistics and longer training, repaired response-level detection to the point of matching the published baseline under adapter training. The diagnostic of this chapter indicates that closing the remaining span-level gap requires work on two fronts at once: further improving response-level classification, because each misclassified response forfeits its spans entirely, and improving span extraction itself, because a deficit remains even when the response-level decision is correct. Effort aimed at span extraction alone would leave the cascade losses untouched, and on this evidence those losses are the larger share for detectors at this performance level. With the detection and repair sides both examined, Chapter 7 brings the two design rules together into the conditional-scaling claim of the thesis.

# Chapter 7

# Discussion and Synthesis

## 7.1 Answer to the primary research question

The primary research question asked how hallucinations in retrieval-augmented generation can be detected and repaired while preserving the supported content of the response. The answer this thesis supports is specific. Detection is best served, under parameter-efficient training, by a detector whose training context is long enough to hold the source and the response together, trained past a single epoch, and scaled within a model family; under those conditions a 13-billion-parameter QLoRA detector matched the published full fine-tuning baseline at response level. Repair is best served by matching the unit of repair to the verifier's behaviour: under a large prompted verifier that judges with near-binary confidence, removing individual claims that fail verification reduced judge-assessed hallucination by 9.15 percentage points while retaining 84.20% of supported claims and 86.52% of response length, which is precisely the preservation the question demanded. The answer is conditional rather than universal, and the rest of this chapter unpacks the conditions, one sub-question at a time, using the evidential vocabulary of Chapter 3: locked tests carry the committed verdicts, diagnostics explain them, the one post-hoc counterfactual is weighed as such, and the robustness check bounds the judging. The failures among these results are part of the evidence, not weaknesses in it; they are what locate the boundary between the configurations in which scaling helps and those in which it does not.

## 7.2 RQ1: detection and detector scaling

The detection finding is that a parameter-efficient detector can match and exceed published full fine-tuning at response-level hallucination detection, and scaling helps, but only under a training configuration that lets the model see the data, and span localisation remains the weaker capability. The evidence is the pair of locked tests of Chapter 4. Under a constrained training setup with a 1,024-token context and one epoch, the scaled detector failed its locked criteria, scoring 68.9 response-level F1 against the existing 74.8 baseline (Phase A). With the context doubled, training extended, and the comparison kept within one model family, the same scale step produced a 13.6-point response-level gain and a final score of 79.6 against the published 78.7 (Phase F). The diagnostic of Chapter 6 then showed that the remaining span-level gap (46.0 against 52.7 on test) is driven substantially by response-level misclassification cascading into span scoring: on validation responses where the response-level decision is correct, both model sizes localise about equally well (53.56 and 53.66), which indicates that scale bought judgement rather than localisation skill. The limitation is that all of this concerns adapter training on one benchmark, and the Phase A comparison crossed model families, so its negative verdict is specific

to that tested setup.

## 7.3 RQ2: verification and reasoning

On verification, the evidence shows that substituting a much larger prompted verifier changed what repair could achieve, but prompting that verifier to reason step by step improved its verdicts without changing the property repair design actually depends on, its confidence profile. The substitution (Phase B) shifted the judging standard itself, and under it a claim-level pipeline passed every locked gate. The verifier, however, commits: only 13.33% of its verification events fall in the uncertain middle band where a rewrite decision could live. Chain-of-thought prompting (Phase C) raised the sentence-level baseline's agreement with gold annotations from 44.79% to 54.06%, yet the middle band moved only to 15.98%, the pre-registered calibration tripwire condition was met, and the lead-narrowing criterion was missed by 0.20 percentage points, so the combined diagnostic verdict is failure. The synthesis reading is that reasoning made the verifier more accurate at committing, not more graded; under the tested conditions, accuracy and calibration moved independently. The limitation is that one prompting protocol on one verifier family was tested; other elicitation methods or verifier families could behave differently.

## 7.4 RQ3: repair action and mechanism

For repair, the central result is that among the tested repair actions, removing claims that fail verification did the work, rewriting never ran, and extractive substitution changed wording without changing judged outcomes. The locked Phase B result established that the claim-level pipeline V1f passes all four gates with a 7.50-percentage-point reduction in unsupported claims and a 0.00% damage rate, while its language-model rewriting pathway triggered on 0.00% of claims. The mechanism ablation (Phase E) then indicated why: a claim-drop-only variant reproduced the aggregate judge outcome exactly, 332 of 732 responses still judged hallucinated under both pipelines, with the claim-level difference of 0.83 percentage points inside the pre-registered equivalence bound. The post-hoc counterfactual (Phase D) adds a methodological qualification rather than an operational one: under an expanded candidate set, the same locked rules would have selected a sibling variant on claim survival (88.55%), which shows that candidate-set choice is itself a consequential pre-registration decision. The limitation is that aggregate equivalence is not per-record identity, the ablation evidence is specific to this verifier's committing profile, and the counterfactual, being post-hoc, modifies no committed verdict.

## 7.5 RQ4: faithfulness and completeness trade-off

On the trade-off, the answer is that removing unsupported content while preserving a useful response proved manageable at claim granularity, because the four-gate structure forced both sides of the trade-off to be satisfied at once. The gates of Chapter 3 institutionalise the tension: the unsupported-rate gate rewards removal, while the damage, survival, and length gates penalise careless or destructive repair. The winning pipeline cleared all four simultaneously on the locked population, and the ablation showed the cost of giving up content-salvaging machinery in exactly the expected place, lower survival and length retention at equal judged outcome. Sentence-level repair, by contrast, removes supported material whenever one claim inside a sentence fails, which is why granularity, not removal aggressiveness, is the operative design variable. This question carries directly over from the original proposal's faithfulness-versus-completeness concern, and the thesis answer is that the trade-off is governed by the unit of repair. The limitation is that

survival and length are proxies for usefulness; no human evaluation of the repaired responses' utility was performed.

## 7.6 RQ5: reliability of automatic judging

On judging, the finding is that the automatic judge is reliable enough for the relative comparisons this thesis makes, and not reliable enough to be read as an absolute measure of hallucination, which is exactly how the thesis uses it. Three strands of evidence bound the claim. Manual validation of the held-out judge found moderate agreement with human judgement, sufficient for comparative use. The primary judge shares a model family with the verifier, so family-shared blind spots are possible in principle; this is disclosed wherever judge results are used. The post-hoc independent-judge robustness check (Appendix D) then re-scored the repair outputs under two unrelated judge models: one confirmed the direction and rough size of the Phase B reduction, while the other did not confirm it, and that non-confirmation is attributable to the second judge's poor calibration against human gold rather than to a failure of the repair. The thesis reports the non-confirmation explicitly and scopes every repair claim accordingly: judge-based results are relative comparisons under a checked judge. The limitation is the flip side of the same coin: absolute hallucination rates in this thesis are judge-dependent quantities and should not be quoted without that qualifier.

## 7.7 Conditional model scaling in RAG hallucination mitigation

Across the five sub-questions, one pattern repeats, and it is the central claim of the thesis: model scaling helped exactly where the interface between the model and its pipeline stage was right, and failed or idled where it was not. The interfaces are the ones defined in Chapter 3. At the detector interface, the model meets the task through its training context window; when that window truncated a third to a half of the training sources, a larger model performed worse than a smaller incumbent, and when the window matched the task's token statistics, the same scale step was worth 13.6 response-level points. At the verifier interface, the model meets the pipeline through its confidence profile; scaling the verifier changed verdicts and shifted the judging standard, but its near-binary profile determined which repair actions could operate at all, and no amount of additional machinery downstream could compensate, as the idle rewriter and the matching ablation outcome indicate. At the repair interface, the available actions only matter insofar as the verifier can feed them; claim-level dropping succeeded because it is the action a committing verifier supports.

Three disciplined distinctions make this pattern interpretable rather than anecdotal. The locked tests (Phases A, B, F) carry verdicts that were fixed before evaluation, so the failure and the wins have equal evidential standing and the conditional claim is not a post-hoc selection of favourable results. The diagnostics (Phases C, E, G) explain the locked outcomes without generating new winners. The post-hoc analyses (Phase D and the robustness check) are labelled as such and weighed accordingly. Pre-registration is what allows a thesis containing a locked failure, a mixed verdict missed by 0.20 percentage points, and an unconfirmed robustness cell to present all of them without reframing; the mixed results are interpretable precisely because the criteria predate the results. The claim that scaling is conditional is therefore bounded but well supported: under the tested conditions, on this benchmark, parameter count was never sufficient by itself, and the deciding variables were interface choices.

Table 7.1: Synthesis of the research questions. Evidence sources use the evidential roles of Chapter 3; every answer is bounded by the stated limitation.

| Question | Evidence source | Answer | Limitation |
|---|---|---|---|
| RQ1 Detection | Locked tests A, F; diagnostic G | Scaling helps once context matches task token statistics; span lag driven largely by response-level cascade | One benchmark; adapter training; A crossed model families |
| RQ2 Verification | Locked test B; diagnostic C | Larger verifier changes repair surface; reasoning improves verdicts, not calibration | One prompting protocol and verifier family |
| RQ3 Repair | Locked test B; ablation E; post-hoc D | Claim-level drop-on-fail carries the result; rewriting never triggers | Aggregate, judge-based; specific to committing verifier |
| RQ4 Trade-offs | Four-gate structure (B, D, E) | Granularity governs the faithfulness–completeness trade-off | Survival and length are proxies; no human utility study |
| RQ5 Evaluation | Manual validation; robustness check (App. D) | Judge fit for relative comparisons, not absolute rates | One of two independent judges does not confirm |

## 7.8 Practical design rules

Table 7.1 condenses the chapter, and two design rules condense the table. For detection: before spending parameters, match the training context length to the source-plus-response token statistics of the task, train adapters past a single epoch, and evaluate scaling within one model family; then expect response-level gains first, and treat span localisation as a separate problem requiring work on both response-level classification and span extraction. For repair: measure the verifier's confidence profile before designing repair actions, because the profile determines which actions can run; under a committing verifier, claim-level drop-on-fail is the operative and safest action, and investment in rewriting is premature until the verifier can produce graded uncertainty, which, on the evidence of Phase C, step-by-step reasoning alone does not provide. Both rules are statements about interfaces, not about scale; scale paid off in this thesis only after the interfaces were right. Their boundaries, and what was not tested, are the subject of Chapter 8.

# Chapter 8

# Limitations and Future Work

The conclusions of this thesis are bounded, and this chapter states the bounds. They are organised by where they arise: the benchmark, the models and training, the evaluation machinery, and the evidence base itself. None of the limitations below weakens a committed verdict; each defines the region within which that verdict has been demonstrated, and the final section turns the bounds into a programme of future work.

## 8.1 Benchmark and generalisation limits

All experiments in this thesis use a single benchmark, RAGTruth, with its three task types and its fixed source-and-response records. This was a deliberate design choice: one benchmark with span-level annotations made the locked-test discipline and the cross-phase comparisons possible. The cost is generalisation. The results may not transfer directly to other RAG datasets, to other domains, to different retrieval quality, or to live systems in which retrieval errors, prompt variation, and distribution shift interact with detection and repair. In particular, every record here begins from a given source and a given response; the pipeline never faces the upstream question of whether the right documents were retrieved at all. The design rules of Chapter 7 are therefore stated as rules for this setting, and their transfer is an empirical question, not an assumption.

## 8.2 Model, training, and compute limits

The detection conclusions concern parameter-efficient adapter training. Every detector in this thesis was trained with QLoRA under the compute available on shared clusters, so the conclusions speak to what adapter training achieves, not to what full fine-tuning would achieve under the same conditions; the published full fine-tuning baseline enters only as an external reference point. Within that scope, three further bounds apply by design. The negative verdict of Phase A, the Detector Scaling Sweep, is bounded to its tested setup, a cross-family comparison under a constrained training configuration, and the thesis draws no conclusion from it about larger detectors in general. The positive verdict of Phase F, the Same-Family Detector Re-scaling, rests on two model sizes in one family, 7 and 13 billion parameters; it demonstrates a large gain at one scaling step under the tested conditions and is not a scaling law. And the reasoning result of Phase C concerns one prompting protocol on one verifier family; other elicitation methods, other model families, or models trained specifically for calibrated verification could behave differently.

## 8.3 Evaluation and judge limitations

The repair conclusions rest primarily on automatic judging, and three limitations follow. First, the response-level judge is a model, and judge-based results are therefore relative comparisons under a checked judge rather than absolute hallucination rates; the thesis uses them only in that relative sense. Second, the primary judge and the substituted verifier share a model family, so blind spots common to that family could in principle affect both the repair decisions and their assessment; this is disclosed wherever judge results are used. Third, the independent-judge robustness check that addresses this concern is itself post-hoc and its outcome is mixed: one unrelated judge confirms the direction of the Phase B repair effect and the other does not, with the non-confirmation traced in Appendix D to that judge's low agreement with human gold rather than to a failure of the repair. The repair claims of this thesis are scoped accordingly and should not be quoted without that qualifier. Human validation of the judge exists but is limited in size and annotator count; it supports comparative use of the judge and should not be presented as a full human evaluation of the system.

## 8.4 Evidence and reproducibility limitations

Two scope limits on the repair mechanism deserve restating as limitations because they are easy to misread as broader claims. The success of the claim-level pipeline is not a rewriting success: its rewriting pathway never triggered during the locked evaluation, so the results say nothing about what rewriting would contribute under a verifier that could feed it. And the mechanism finding of Phase E is aggregate and judge-based: the ablated pipeline reproduces the same judged outcome in total, not record by record, so the claim is about which component carries the result, not about output equivalence.

On the evidence base itself, the thesis distinguishes the strength of its sources. The headline locked results and the robustness check trace to committed result files and recomputable raw outputs. A minority of supporting values, including parts of the Phase A sweep detail and the Phase C measurements, are carried from the locked thesis text while their raw result files are queued for archival back-fill; none of these affects a committed verdict, and the file inventory in Appendix C records the status of each. Finally, two presentation choices trade detail for readability within the page budget: the verbatim pre-registrations are condensed into structured summaries in Appendix A, preserving criteria, thresholds, amendments, and commit identifiers so that auditability survives the compression, and several raw tables live in Appendix B rather than the main text. Readers who need the full artefacts are directed to the project repository throughout.

## 8.5 Future work

The limitations above convert directly into a research programme, organised by pipeline stage.

On the benchmark side, the first step is breadth: future work should test the detector findings on additional RAG hallucination benchmarks and domains, and should move from fixed source-and-response records towards end-to-end RAG systems in which retrieval errors are part of the evaluated behaviour. Per-task and per-domain behaviour, summarised only at headline level here, deserves deeper study in its own right.

On the detection side, the design rule invites sharper tests: longer context windows and newer model families would probe whether the context-matching principle continues to govern scaling benefit, and a comparison of QLoRA against full fine-tuning under matched compute would

resolve how much of the remaining span-level gap is attributable to adapter training itself. Independently of training method, the diagnostic of Chapter 6 points to a concrete target: improving span localisation directly, after controlling response-level errors, since a localisation deficit remains even on response-correct subsets.

On the verification and repair side, the most consequential direction is calibration. The repair results indicate that the binding constraint on richer repair is not repair machinery but the verifier's confidence profile, so future work should develop verifiers with genuinely graded middle-band confidence, whether through training objectives, ensembling, or elicitation methods beyond the single reasoning protocol tested here. With such a verifier, repair pipelines in which rewriting actually triggers could be evaluated for the first time on their merits, rather than idling behind a committing verifier.

On the evaluation side, two strands would most strengthen claims of the kind this thesis makes: human evaluation of repaired responses, assessing usefulness and completeness rather than only judged hallucination, and stronger judge validation, with larger human-annotated samples and pre-registered multi-judge protocols in place of post-hoc checks. Under the tested conditions, the conditional-scaling claim stands; these directions determine how far beyond those conditions it reaches.

# Chapter 9

# Conclusion

This thesis asked how hallucinations in retrieval-augmented generation can be detected and repaired while preserving the supported content of the response, and under what conditions scaling the models in that pipeline helps. The answer, developed across seven pre-registered phases on the RAGTruth benchmark, is that scaling is conditional on the interface between each model and its pipeline stage.

On detection, scaling a quantised low-rank adaptation detector from 7 to 13 billion parameters helped only once the training context length matched the source-and-response token statistics of the task. Under a short context and a single epoch the larger detector failed its locked criteria and fell below a smaller incumbent; under a context matched to the data and longer training, the same scale step produced a large gain and a response-level F1 that exceeded the published full fine-tuning baseline under adapter training alone. A diagnostic located the remaining span-level gap mainly in response-level misclassification cascading into span scoring, rather than in span extraction.

On repair, substituting a large prompted verifier for a small cross-encoder allowed a claim-level pipeline to pass every locked repair gate and to reduce judge-assessed hallucination while preserving most supported content. The behaviour that produced this result was not rewriting, which never triggered, but the removal of claims that failed verification, an action suited to a verifier that judges with near-binary confidence. An ablation attributed the result to that mechanism, a reasoning protocol failed to give the verifier the graded confidence that rewriting would need, and a labelled post-hoc counterfactual showed that the choice of competing candidates was itself a consequential design decision.

The contribution can be read at two levels. The six specific contributions of Chapter 1 group into two: a set of substantive, interface-level design rules and a methodological demonstration. Substantively, the thesis offers two design rules, one for detection and one for repair, that hold under the tested conditions and that frame scaling as a question of matching models to stages rather than of parameter count alone. Methodologically, it shows that a pre-registered, phase-based design lets a negative result, a positive result, a mechanism ablation, a mixed verdict, and a counterfactual coexist in one account without any being reframed after the fact. The claims are bounded to one benchmark and to the configurations tested, and the directions that would extend them, additional benchmarks, calibrated verifiers, end-to-end retrieval, and human evaluation, are set out in Chapter 8. Within those bounds, the thesis supports a single conclusion: in a retrieval-augmented generation pipeline, model scaling repays the investment only when the model's interface to its stage is right.

# Appendix A

# Pre-registration and amendment summaries

This appendix condenses the seven phase pre-registrations and their amendments into structured records. Each entry states the locked hypothesis, the committed success criterion, the locked-test discipline, the amendments with their timestamps and pre- or post-result status, and the adjudicated verdict. The verbatim pre-registration markdown is archived in the repository (Appendix C); these summaries preserve every locked threshold, amendment, and verdict while compressing the originals to readable length.

## A.1 Phase A: Detector Scaling Sweep

Locked 12 May 2026, before any sweep run. Six pre-experiment amendments (A1–A6) were committed after code inspection surfaced discrepancies with the original draft; all are pre-experiment.

**Hypothesis.** An eighteen-cell QLoRA grid over learning rate, LoRA rank, and alpha on Llama-2-13B can find a configuration that improves the existing detector's locked-test scores of 74.8 response F1 and 37.4 span F1.

**Sweep dimensions (locked).** Eighteen cells: learning rate $\in \{1\text{e-}4, 3\text{e-}4, 5\text{e-}4\}$, rank $\in \{16, 32, 64\}$, alpha $\in \{32, 64\}$, with warmup ratio (A1), epoch count (A2), validation split (A3), and seed (A4) fixed by amendment.

**Tie-break order (A5).** Ties on highest validation F1 resolve by: (1) average validation F1 across response and span, (2) lower learning rate, (3) lower rank, (4) lower alpha, (5) analyst judgement with rationale.

**Locked success criterion.** SUCCESS if the detector reaches at least one of response F1 $\geq 76.7$ (within 2 pp of the paper's 78.7) or span F1 $\geq 42.4$ (within about 10 pp of the paper's 52.7; the bound is 10.3 pp below) on test. PARTIAL if it improves by $\geq 3$ pp over 74.8 or 37.4 on either metric without meeting SUCCESS. FAILURE otherwise, in which case the existing detector remains the configuration of record.

**Locked-test discipline.** The 2,700-record test split is touched exactly once, after the winning validation configuration is locked. Re-running with a different grid after seeing results is prohibited.

**Verdict.** FAILURE. Response F1 = 68.9 on test under the winning cell, below the PARTIAL threshold.

## A.2 Phase B: Verifier Substitution and Repair

**Scope.** Re-run V0, the V1 ablation cells (V1a, V1b, V1e, V1f), and V2 on the RAGTruth development data ($n$=314 for V1 and V2, $n$=1,046 for V0) with Qwen-2.5-32B-Instruct (4-bit NF4, threshold $\tau = 0.50$) replacing the prior English DeBERTa-v3-large NLI verifier; all other components unchanged.

**Verifier protocol (locked).** Per-claim probability aggregated across premise chunks by maximum; supported flag is $P(\text{supported}) \geq 0.50$; the yes and no token identifiers were resolved at initialisation and logged.

**Four-gate WIN criterion (locked).** A pipeline wins against V0 if and only if all four hold simultaneously: delta unsupported rate better than V0's by $\geq 1$ pp; damage rate $\leq 5.0\%$; claim survival $\geq 70\%$; length retention $\geq 60\%$. PARTIAL if all four pass but the delta is within 1 pp of V0's. FAILURE otherwise. V2 additionally carried its own stricter criterion (absolute delta unsupported $\leq -6.3$ pp, same retention floors).

**Binding candidate set (locked).** {V1f, V2}; V1a, V1b, and V1e are ablation cells, revisited in Phase D.

**Amendments.** B1 (pre-result): switch from the v1 prompt to the v2 strict source-grounded NLI prompt. B2 (pre-result): retain Qwen-32B as verifier after the V0 replication tripwire fired, a substantive behavioural shift relative to the prior cross-encoder baseline. Both timestamped and committed before any inference under the v2 prompt.

**Verdict.** V1f wins the binding test: all four gates pass on development, and the locked test ($n$=732) yields a $-9.15$ pp aggregate judge-level hallucination reduction.

## A.3 Phase C: Reasoning Verifier Swap

**Scope.** The Phase B Qwen-2.5-32B verifier, swapped to a chain-of-thought protocol: the model reasons step by step, then emits a structured verdict line, and the yes/no token logits at the post-verdict position are read for scoring, preserving comparability with Phase B.

**Parse-failure fallback (locked).** Chunks without a parseable verdict, or with verdict-token probability mass below 0.10, are marked as parse failures, scored 0.0, and logged; a parse-failure rate above 5% on the development V0 run triggers protocol review before continuing.

**Cross-phase criterion (locked).** Hypothesis WIN if V0 response F1 falls in $[50\%, 62.40\%]$; STRONG SUPPORT above 62.40%; PARTIAL in $[44.79\%, 50\%)$; FAILURE below 44.79%. Lead-narrowing WIN if the V1f lead is $\leq 2$ pp; PARTIAL in $(2, 3.12]$ pp; FAILURE above 3.12 pp or below 0. The combined verdict is FULL SUPPORT only if both components succeed, PARTIAL if at least one is partial with no failure, and FAILURE if either component fails. The reference values 62.40% (prior cross-encoder V0) and 44.79% (Qwen-era V0) are locked from existing data; the WIN band and the 3.12 pp ceiling were fixed before any chain-of-thought inference.

**Calibration tripwire (locked).** Fires if the reasoning verifier's middle mass in $[0.10, 0.90]$ is at most twice the Phase B middle mass. The doubling multiplier is locked; amendment C1 filled the Phase B baseline, fixing the threshold at 26.65%. Firing triggers a discussion paragraph but does not invalidate the primary verdict. Amendments C1–C3 are all pre-result: C1 records the baseline middle mass, C2 locks the verdict-extraction fallback, C3 documents a sharded-execution detail.

**Locked-test conditionality.** If the combined development verdict is FULL SUPPORT or PARTIAL, the locked test runs automatically under reasoning verification; if FAILURE, it does not run. The conditional is fully pre-registered, with no analyst judgement at the decision point.

**Verdict.** Combined FAILURE. The V0 component met the WIN band (response F1 54.06%), but lead-narrowing missed the PARTIAL ceiling by 0.20 pp (3.32 pp against 3.12 pp), and the calibration tripwire fired (middle mass 15.98% against 26.65%). The locked-test run was skipped per the pre-registration.

## A.4 Phase D: Tie-Break Counterfactual

**Scope.** Apply the Phase B tie-break order to the expanded candidate set {V1e, V1f, V2} on Phase B's existing development data. Phase D is post-hoc on locked data and does not modify Phase B's V1f verdict.

**Tie-break order (Phase B-locked).** Clause 1, delta unsupported improvement over V0 (larger wins); clause 2, damage rate (lower wins); clause 3, claim survival (higher wins), which resolves Phase D; clause 4, length retention; clause 5, simplicity preference; clause 6, analyst judgement with rationale.

**Unified-criterion note.** Phase D applies V1f's WIN criterion uniformly to all three candidates so that the ranking is coherent. Under this unified criterion V2 passes all four gates, which differs from Phase B's V2 verdict under V2's own stricter criterion; Phase D does not modify that Phase B verdict, and both criteria are recorded.

**Verdict.** V1e wins by tie-break at clause 3 (survival 88.55% against V1f's 84.60%). Phase B's locked V1f win under the {V1f, V2} binding set is unchanged. The contribution is methodological: binding-test candidate-set selection is itself a pre-registration decision.

## A.5 Phase E: Mechanism Ablation

**Scope.** Re-run V1f's mechanism on the same $n$=732 locked population with the rewrite pathway force-converted to drop and the extractive fallback disabled; all other components verbatim from Phase B.

**Success criteria (locked, two parallel adjudications).** MECHANISM CONFIRMED if the claim-level delta differs from V1f's by $\leq 1.00$ pp; PARTIAL up to 2.50 pp; NOT CONFIRMED beyond. JUDGE REPLICATED if the judge-level change differs from V1f's by $\leq 2.00$ pp; PARTIAL up to 4.00 pp; NOT REPLICATED beyond. A PUBLISHABLE MECHANISM verdict requires both CONFIRMED and REPLICATED. The 1.00 pp threshold was locked before any inference.

**Verdict.** PUBLISHABLE MECHANISM. The claim-level delta is aggregate-equivalent to V1f's, and the judge replicated exactly (332 of 732 hallucinated under both). Phase B's V1f win is preserved.

## A.6 Phase F: Same-Family Detector Re-scaling

**Scope.** Disentangle the Phase A cross-family confound by training two same-family detectors, Llama-2-7B and Llama-2-13B QLoRA, under a corrected unified configuration.

**Unified configuration (locked).** Identical to Phase A except maximum sequence length 2,048 (Phase A: 1,024) and three epochs (Phase A: one). The longer context is motivated by the Phase A summarisation truncation analysis.

**Locked-test discipline.** The test split is touched at most twice in Phase F, once per model;

with Phase A's prior touch, cumulative test touches total three.

**Success criteria (locked).** PRIMARY WIN if the 13B detector exceeds the 7B by $\geq 2.0$ pp on both response and span F1; PARTIAL on one metric only; FAILURE on neither. SECONDARY SUCCESS if the 13B response F1 $\geq 76.7$ or span F1 $\geq 42.4$.

**Amendments.** F1 (2026-05-25, pre-training, pre-result): relaxes the thesis-submission-before-training dependency in exchange for three safeguards, a thesis PDF hash freeze, a no-peek commitment on Phase F outputs before submission, and the Phase A failure verdict locked regardless of outcome. F2 (2026-05-26, post-result): overrides F1 safeguards (i) and (iii); the thesis gains a new Phase F chapter carrying the locked win, while the Phase A failure chapter remains byte-identical and Phase F reframes Phase A's configuration-specific scope without altering its locked text.

**Verdict.** PRIMARY WIN (13B response 79.6 against 7B 66.0, +13.6 pp; 13B span 46.0 against 7B 36.6, +9.4 pp). SECONDARY SUCCESS (79.6 $\geq$ 76.7).

## A.7 Phase G: Span Localisation Diagnostic

**Scope.** Four sub-experiments targeting the 6.7 pp test span-F1 gap between Phase F (46.0) and the paper baseline (52.7), bundled under one pre-registration with a locked end-of-program test evaluation.

**Sub-experiments and locked thresholds.** G1, post-processing rules over emitted spans, must close $\geq 1.5$ pp (test span F1 $\geq 47.5$). G2, a second-pass prompted refiner, must reach $\geq 50.0$. G3, the oracle response diagnostic on the validation split, is diagnostic only with no test evaluation and no win/fail threshold. G4, a new BIO token-classification detector (DeBERTa-v3-large; AdamW, learning rate 2e-5, five epochs, seed 42, batch 16, maximum length 512), must reach $\geq 52.7$.

**Test-touch discipline.** No test touches during development; all four sub-experiments develop on the validation split. One end-of-program test evaluation would run all advancing sub-experiments together.

**Amendment G1.** Timestamped 2026-05-26 (post-G3 result, pre-G1/G2/G4 result). Early-terminates Phase G after G3. Rationale: G3 reached span F1 53.56 on the F-13B response-correct subset, already above the 52.7 baseline, and the per-task breakdown localises the residual gap on question answering and summarisation as response-misclassification cascade rather than span-extraction quality. G1 and G2 cannot recover spans on records the response classifier mispredicts, and G4 would replace the whole detector at high cost when the diagnostic shows the span side is adequate. The G1, G2, and G4 thresholds remain locked but un-adjudicated, valid for any future researcher. Cumulative test touches remain three; the pre-registered Phase G test evaluation is not consumed.

**Verdict.** G3 diagnostic only; no test verdict.

# Appendix B

# Supplementary tables and examples

## B.1 Worked example of claim-level repair

To make the claim-level pipeline concrete, consider a short response generated against a source passage about a public building. Suppose the response reads: “The museum opened in 1995, houses over four thousand works, and is the most visited gallery in the region.” Claim decomposition splits this into three atomic claims: (1) the museum opened in 1995; (2) the museum houses over four thousand works; (3) the museum is the most visited gallery in the region. The verifier checks each against the source. Suppose the source states the opening year and the size of the collection but says nothing about regional visitor rankings. Claims (1) and (2) are then supported and claim (3) is unsupported. Under claim-level drop-on-fail, claim (3) is removed and the response is reconstructed from the survivors: “The museum opened in 1995 and houses over four thousand works.” The unsupported ranking has been removed while both supported facts are retained, which is the behaviour the survival and length metrics of Section 3.6 reward. This example is illustrative; it is not drawn from the RAGTruth annotations.

## B.2 Per-task span diagnostic for Phase G

Table B.1 gives the per-task breakdown of the Phase G oracle subset analysis of Chapter 6 for the F-13B detector on the validation split, recording both the response-correct oracle counts and the span F1 on the full and oracle subsets. The cascade share of the span-level deficit is concentrated in the tasks where the oracle gap is largest: question answering and summarisation show oracle gaps of 12.31 and 11.97 percentage points, against only 3.88 for data-to-text, which is the evidence on which the Phase G Amendment G1 early-termination decision rested (the residual gap on those two tasks is response-misclassification cascade, not span-extraction quality).

Table B.1: Per-task span diagnostic for the F-13B detector on the RAGTruth validation split ($n$=942). Oracle counts are response-correct records; span F1 is reported on the full task set and on the oracle subset.

| Task | Overall $n$ | Overall span F1 | Oracle $n$ | Oracle span F1 | Gap |
|---|---|---|---|---|---|
| Question answering | 308 | 44.37 | 264 | 56.68 | +12.31 pp |
| Summarisation | 300 | 31.71 | 238 | 43.69 | +11.97 pp |
| Data-to-text | 334 | 50.26 | 291 | 54.15 | +3.88 pp |
| All tasks | 942 | 44.50 | 793 | 53.56 | +9.05 pp |

# Appendix C

# Reproducibility details

This appendix records the information needed to trace each reported number to a stored artefact. Bulk artefacts (training logs, raw prediction files, full job scripts) are archived in the project repository; this appendix gives the traceability spine and points to that repository for the rest.

## C.1 Repository and commit

The code, pre-registration documents, result files, and robustness outputs are archived at:

- Repository: `https://github.com/Saikrishna-cy/Projects_Datascience` (directory `MasterThesis`).
- Reproducibility-bundle commit: `c6bbd93` (2026-06-11), under which all result files, pre-registrations, and scripts referenced here are archived.
- Random seed: 42, fixed across all training, inference, and data-splitting scripts.

## C.2 Software environment

The independent-judge robustness check of Appendix D was run in a Python 3.11.3 virtual environment with the following pinned versions: `torch` 2.12.0 (CUDA build cu130), `transformers` 5.10.2, `accelerate` 1.13.0, and `bitsandbytes` 0.49.2. The full `pip freeze` snapshot of this environment (57 packages) is archived as `pip_freeze_judges_20260613.txt` alongside the robustness data. The QLoRA detector training of Phases A and F used the `peft` library for low-rank adapter management in a separate, earlier training environment; its exact `peft` version is recorded in the training-time SLURM logs rather than in the inference environment above.

## C.3 Compute environment

All training and large-model inference ran as SLURM batch jobs on the `gpu_a100` partition of the Snellius national supercomputer operated by SURF and the ALICE cluster of Leiden University. The Snellius `gpu_a100` nodes provide NVIDIA A100 40 GB SXM4 GPUs; the recorded driver version is 580.82.07 with CUDA runtime 13.0, captured from `nvidia-smi` in the SLURM logs. Detector training and pipeline inference each used a single A100; the Llama-family robustness judge used two. Table C.1 lists the SLURM time limits requested per phase (wall-clock limits, not necessarily elapsed time).

Table C.1: Compute summary per phase. GPU: NVIDIA A100 40 GB on the `gpu_a100` partition. The time column gives the requested SLURM wall-clock limit for the committed run, not measured elapsed time.

| Phase | Cluster | GPUs | SLURM time limit |
|---|---|---|---|
| A: Detector Scaling Sweep | Snellius / ALICE | 1 × A100 | 12 h per sweep cell |
| B: Verifier Substitution and Repair | Snellius / ALICE | 1 × A100 | up to 24 h (locked run) |
| C: Reasoning Verifier Swap | Snellius / ALICE | 1 × A100 | 12 h per array cell |
| D: Tie-Break Counterfactual | No new compute; re-reads stored Phase B development outputs | | |
| E: Mechanism Ablation | Snellius / ALICE | 1 × A100 | 12 h |
| F: Same-Family Detector Re-scaling | Snellius / ALICE | 1 × A100 | 12–18 h (see note) |
| G: Span Localisation Diagnostic | Re-scores stored Phase F validation detections; no training | | |

## C.4 Detector training configuration

Detectors were trained with QLoRA on frozen 4-bit base models. Base models: `meta-llama/Llama-2-7b-hf` and `meta-llama/Llama-2-13b-hf`.

- Phase A (Detector Scaling Sweep): Llama-2-13B, context length 1,024 tokens, 1 epoch. The eighteen-cell sweep varied LoRA rank $r \in \{16, 32, 64\}$, LoRA alpha $\in \{32, 64\}$, and learning rate $\in \{1\text{e-}4, 3\text{e-}4, 5\text{e-}4\}$; seed 42. The supplementary run repeated the committed cell for 3 epochs.
- Phase F (Same-Family Detector Re-scaling): both Llama-2-7B and Llama-2-13B trained at LoRA rank 32, alpha 64, learning rate 5e-4, context length 2,048 tokens, 3 epochs, seed 42 (output directories `F_7B_qlora_r32_a64_lr5e-4_ctx2048` and `F_13B_qlora_r32_a64_lr5e-4_ctx2048`)

## C.5 Verifier and judge configuration

- Phase B and Phase E verifier: `Qwen/Qwen2.5-32B-Instruct`, greedy decoding (temperature 0.0).
- Phase C reasoning verifier: `Qwen/Qwen2.5-32B-Instruct` with chain-of-thought prompting, `max_new_tokens` 300 (locked in the Phase C pre-registration).
- Prior cross-encoder verifier: `MoritzLaurer/DeBERTa-v3-large-mnli-fever-anli-ling-wanli`, maximum sequence length 512. The configuration key and source filename use the legacy token `mdeberta`; this refers to the English DeBERTa-v3-large NLI checkpoint above, not to the multilingual mDeBERTa-v3 model.
- Repair planner model: `Qwen/Qwen2.5-7B-Instruct`, `max_new_tokens` 256, temperature 0.0.
- Primary judge: `Qwen/Qwen2.5-32B-Instruct`, greedy decoding.
- Independent robustness judges (Appendix D): `Llama-3.1-70B-Instruct` (two A100s) and `Gemma-2-27B-it` (single A100), both greedy decoding, temperature 0, seed 42.

## C.6 Data splits and per-task counts

Table C.2 lists each population, its size, its path in the repository, and its per-task record counts. The 1,046, 942, and 314 splits record task labels in TitleCase (QA, Summary, Data2txt); the 732

Table C.2: Experimental splits, repository paths, and per-task record counts. Paths are relative to the repository root.

| Split | $n$ | QA | Summary | Data2txt |
|---|---|---|---|---|
| Verifier development | 1,046 | 175 | 559 | 312 |
| Repair development | 314 | 52 | 168 | 94 |
| Locked repair test | 732 | 123 | 391 | 218 |
| Phase F/G validation | 942 | 308 | 300 | 334 |

locked test records them in lowercase (qa, summarization, data2text). Both refer to the same three RAGTruth tasks and were normalised at evaluation time.

The split files are stored at: verifier development, `results/step10_judge_qwen32b_fs.jsonl`; repair development, `v1_repair_project/results/data/dev_gold_spans.jsonl`; locked repair test, `01_robustness/inputs/locked_test_records.jsonl`; validation, `data/sweep_val.jsonl`.

## C.7 Result file inventory

Each reported result traces to a stored file under `MasterThesis/03_phase_results/`:

- Phase F detector results: `phaseF/test_response_f1.json` and `phaseF/test_span_f1.json`.
- Phase B headline metrics: `phaseB_headline_metrics.json`.
- Phase G span diagnostics: `phaseG/G3_F_13B_val_diagnostic.json` and `phaseG/G3_F_7B_val_diagnost`
- Robustness re-scoring outputs (Appendix D): `01_robustness/judge_gemma/` and `01_robustness/judge_` over the input set `01_robustness/inputs/locked_test_records.jsonl`.

## C.8 Phase F training and adjudication note

Two facts about the Phase F runs are recorded for transparency, drawn from the phase's lock-provenance file. First, the F-7B training job was cancelled at its initial 8-hour SLURM limit at step 2600 of 2625 (99% complete); it was re-run with a 12-hour limit and completed cleanly. The reported F-7B numbers are from the clean re-run; the incomplete output is preserved separately under a directory name marked `failed_8h_cap`. Second, the Phase F verdict was computed manually because the evaluation job's adjudicator encountered a JSON-key mismatch; the underlying per-task metrics are unaffected and are the values reported in Chapter 4.

# Appendix D

# Independent-judge robustness check

## D.1 Purpose and design

The repair results of Chapter 5 are measured by an automatic judge built on the Qwen-2.5-32B model family, the same family as the substituted verifier of Phase B. A judge and a verifier drawn from one family may share systematic blind spots, so a reduction in judge-assessed hallucination could in principle reflect agreement between two related models rather than a genuine improvement in faithfulness. This appendix reports a post-hoc robustness check that addresses that concern directly. It is post-hoc: it was designed and run after the Phase B and Phase E verdicts were known, and it modifies none of them. Its role is to bound how far the repair conclusions depend on the particular judge used, by re-scoring the same repair outputs under two judges from model families unrelated to the verifier.

Two independent judges were used: `Llama-3.1-70B-Instruct` and `Gemma-2-27B-it`. Each judge re-scored the original responses and the V1f-repaired outputs over the locked repair records, under the same labelling protocol as the primary judge: each response is labelled hallucinated or clean against its source. The judges ran with greedy decoding, temperature 0, and seed 42, with no per-judge prompt tuning, so that any difference between them reflects the judge model rather than a difference in how it was invoked. The stored re-scoring outputs are archived as described in Appendix C (the `judge_gemma` and `judge_llama` record sets over the locked input file).

## D.2 What the check tests

The check addresses two claims from Chapter 5 separately, because they have different evidential standing. The first is the Phase B directional claim: that the claim-level pipeline reduces response-level hallucination relative to the original responses. The second is the Phase E mechanism claim: that the claim-drop-only ablation reproduces the full pipeline's judged outcome, which in Chapter 5 rested on per-record agreement between the two pipelines under the primary judge. A robustness check can confirm or fail to confirm each of these independently, and the honest reporting requirement is to say which held under which judge.

## D.3 Result: a split outcome

The two independent judges did not agree with each other, and the thesis reports that split rather than averaging it away. The Llama-3.1-70B judge confirmed the Phase B directional result: it labelled 414 of the 732 original responses hallucinated and 316 of the repaired responses, a

Table D.1: Independent-judge robustness check on the locked repair records ($n$=732). Each judge re-scores the original responses and the V1f-repaired responses; counts are the number of responses each judge labels hallucinated. The primary-judge column repeats the Chapter 5 figures for reference. Agreement with human gold is reported as Cohen's $\kappa$ [2].

| | Primary judge Qwen-2.5-32B | Independent judge 1 Llama-3.1-70B | Independent judge 2 Gemma-2-27B |
|---|---|---|---|
| Original, hallucinated | 399 | 414 | 632 |
| V1f-repaired, hallucinated | 332 | 316 | 650 |
| Change | −9.15 pp | −13.39 pp | +2.46 pp |
| Direction | Reduction | Reduction (confirms) | Not reduced |
| $\kappa$ vs human gold | — | 0.434 | 0.158 |

reduction of 13.39 percentage points, in the same direction as the primary judge and somewhat larger. Its agreement with human gold annotations was moderate ($\kappa = 0.434$), comparable to the primary judge's. The Gemma-2-27B judge did not confirm the result: it labelled 632 of the originals and 650 of the repaired responses hallucinated, a change of +2.46 percentage points. The three judges' counts are collected in Table D.1.

The Gemma non-confirmation should be read as a calibration artefact rather than as evidence against the repair. Gemma labels 86.3% of the original responses hallucinated, against 54.5% for the primary judge and 56.6% for the Llama judge, and its agreement with human gold is low ($\kappa = 0.158$). A judge that labels almost every response hallucinated, before any repair, has little room to register a reduction and discriminates poorly between repaired and unrepaired text; its disagreement with the human gold confirms that it is the weaker judge here, not that the repair failed. The robustness check therefore returns one confirmation from the better-calibrated independent judge and one non-confirmation from a judge whose own gold agreement is low.

On the Phase E mechanism claim, the picture is likewise mixed and is reported as such. The per-record agreement between the full pipeline and the claim-drop-only ablation, which under the primary judge stood at 634 of 732 records, held under the better-calibrated Llama judge and was not corroborated under the low-agreement Gemma judge, in the same pattern as the directional result. The interpretation is conservative. The mechanism finding of Chapter 5, that claim-level dropping carries the repair result, is supported under the primary judge and under one independent judge, and is not corroborated under the second; it is therefore stated throughout the thesis as a result under a checked judge, not as a judge-independent fact.

## D.4 Reading of the robustness check

The value of this appendix is precisely that it does not return a clean confirmation. A single judge, especially one sharing a model family with the verifier, could overstate the reliability of the repair results; running two unrelated judges shows that the central directional finding reproduces under the better-calibrated of the two and not under the judge whose own agreement with human gold is low. This bounds the repair claims rather than overturning them: the results do not support an absolute, judge-independent hallucination reduction, and the split outcome here is the empirical reason for that restraint. Wherever Chapter 5 and Chapter 7 state a repair result, they state it as a relative comparison under a checked judge, and this appendix is the evidence that the qualifier is necessary rather than decorative. Strengthening this check, with more independent judges and with human adjudication of the records on which the judges disagree, is listed among the evaluation directions of Chapter 8.

# Bibliography


[1] Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *International Conference on Learning Representations*, 2024. arXiv:2310.11511.

[2] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46, 1960. doi: 10.1177/001316446002000104.

[3] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient finetuning of quantized LLMs. In *Advances in Neural Information Processing Systems*, volume 36, 2023. arXiv:2305.14314.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186. Association for Computational Linguistics, 2019. doi: 10.18653/v1/N19-1423.

[5] Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. Chain-of-verification reduces hallucination in large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3563–3578. Association for Computational Linguistics, 2024. arXiv:2309.11495.

[6] Pengcheng He, Jianfeng Gao, and Weizhu Chen. DeBERTaV3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *International Conference on Learning Representations*, 2023. arXiv:2111.09543.

[7] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. arXiv:2106.09685.

[8] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12), 2023. doi: 10.1145/3571730.

[9] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*, volume 35, pages 22199–22213, 2022. arXiv:2205.11916.

[10] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474, 2020. arXiv:2005.11401.

[11] Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 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*, pages 9004–9017, Singapore, 2023. Association for Computational Linguistics. arXiv:2303.08896.

[12] Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.acl-main.173.

[13] Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947. doi: 10.1007/BF02295996.

[14] Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, KaShun Shum, Randy Zhong, Juntong Song, and Tong Zhang. RAGTruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10862–10878. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.585.

[15] Brian A. Nosek, Charles R. Ebersole, Alexander C. DeHaven, and David T. Mellor. The preregistration revolution. *Proceedings of the National Academy of Sciences*, 115(11): 2600–2606, 2018. doi: 10.1073/pnas.1708274114.

[16] Qwen Team. Qwen2.5 technical report. arXiv preprint, 2024. arXiv:2412.15115.

[17] Hugo Touvron, Louis Martin, Kevin Stone, et al. Llama 2: Open foundation and fine-tuned chat models. arXiv preprint, 2023. arXiv:2307.09288.

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008, 2017. arXiv:1706.03762.

[19] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837, 2022. arXiv:2201.11903.