LUISS

Management and Computer Science

Teaching:

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Retrieval vs. Comprehension in Long-Context LLMs: A Transfer Study Using MRCR and LongBench v2

Candidate:

Leonardo Azzi

ID 277941

Supervisor:

Prof. ITALIANO GIUSEPPE FRANCESCO

Academic Year 2024/2025

Abstract

Can a model that learns to find specific information in large documents also understand those documents well enough to answer questions? Long context windows encourage the belief that better retrieval yields better understanding. This thesis tests that assumption. I fine-tune a single small long-context model, Gemma-3 4B-IT, on MRCR, a multi-needle retrieval task with order disambiguation, and assess comprehension on LongBench v2 using the official harness with fixed evaluation settings. The baseline (without MRCR supervision) reaches 28.4 percent accuracy. After MRCR supervision, two checkpoints achieved 21.3 percent (format-agnostic) and 22.7 percent (format-aware). Format-aware targets reduce invalid outputs, yet the accuracy gap persists. The results indicate that retrieval-heavy supervision can strengthen surface matching without improving multiple-choice reasoning over long contexts. I provide a controlled, reproducible measurement with per-item logs and diagnostics that separate formatting effects from accuracy. For this model and training scale, retrieval skill does not transfer to comprehension; complementary alignment for reasoning is required.

Table of Contents

1 Introduction

This thesis asks a simple question: when a model learns to retrieve the right span inside a very long input, does that skill also help it answer questions about the input? Context windows now reach hundreds of thousands or even millions of tokens, which allow single-pass processing of long documents and repositories. However, support for long inputs does not guarantee strong long-context behavior. Recent evaluations report position sensitivity and sharp drops when tasks require more than span matching, and new benchmarks were designed to measure this gap under realistic settings (Liu et al., 2023; Hsieh et al., 2024; Goldman et al., 2024; Bai et al., 2024; Mai and Attia, 2025).

I frame the problem as retrieval versus comprehension. Retrieval is the ability to locate and reproduce the relevant span. Comprehension is the ability to integrate that span with a question, compare alternatives, and commit to a single option. Many long-context failures look like partial success on retrieval followed by weak decision making. LongBench v2 targets this gap with multiple-choice questions drawn from realistic sources and reports accuracy as a simple and reliable metric (Bai et al., 2024).

The study uses one model family, one supervision signal, and one benchmark. The model is Gemma-3 4B-IT with a 131 072 tokens window (Gemma Team, 2025). The supervision signal is MRCR, a multi-needle retrieval task with order disambiguation (OpenAI, 2025). The benchmark is LongBench v2, evaluated with the official harness and fixed settings (Bai et al., 2024). I compare the baseline model to two fine-tuned checkpoints that differ only in target format: a format-agnostic target that copies the retrieved span, and a format-aware target that appends a single Final Answer line intended to reduce formatting drift. All other settings remain equal.

The central result is negative transfer. On the official LongBench v2 harness, the baseline reaches 28.4 percent accuracy. After MRCR supervision, the two checkpoints achieve 21.3 percent (format-agnostic) and 22.7 percent (format-aware). Paired bootstrap confidence intervals for both deltas exclude zero. Format-aware targets reduce missing-letter predictions but do not restore accuracy. For this model and scale, retrieval-heavy supervision improves surface matching without improving multiple-choice reasoning over long contexts.

This thesis contributes a focused measurement and a transparent record. First, it provides a single-model transfer test with a fixed evaluation harness and a clear reporting protocol. Second, it adds diagnostics that separate formatting effects from accuracy, including item-level transitions from baseline to fine-tuned outcomes. Third, it supplies a reproducible artifact map: pinned harness commit and prompt, per-item predictions, aggregate tables, training run cards, and analysis scripts.

The scope is narrow by design. I do not introduce option-selection heads, constrained decoding, or preference optimization, and I do not report an MRCR held-out test beyond training loss. These choices keep the result interpretable and attributable to the supervision signal under test. Section 2 reviews long-context evaluation and the difference between retrieval and comprehension. Section 3 describes the model, MRCR data, and evaluation protocol. Section 4 reports results and diagnostics. Section 5 discusses implications, threats to validity, and limitations, and outlines directions for future work.

2 Background

2.1 Long-context models and why evaluation matters

Modern language models now accept inputs that span hundreds of thousands or even millions of tokens. This capability results from advances in attention efficiency, memory management, and positional scaling, such as FlashAttention-2, PagedAttention, and YaRN, combined with model and post-training recipes that target longer windows (Dao, 2023; Kwon et al., 2023; Peng et al., 2023; Gemma Team, 2025). As long-context support becomes common, the need for reliable evaluation grows. The HELM Long Context initiative argues that support for long inputs does not imply strong long-context behavior and calls for transparent, comparable evaluation across tasks that genuinely stress long-range use of information (Mai and Attia, 2025). This thesis follows that view: window size is a means, not an end, and measurement must test what the model actually does with the extra context.

2.2 Retrieval versus comprehension

Retrieval and comprehension are related but different skills. Retrieval is locating the relevant span inside a long input and reproducing it. Comprehension is using that span to answer a question, which requires comparing alternatives and committing to one option. Prior work shows why the distinction matters. Models can be sensitive to the position of evidence, using information near the beginning or end more reliably than information in the middle (Liu et al., 2023). They can also look strong on literal-match probes yet degrade when lexical overlap is reduced or when the task requires aggregating evidence and multi-hop reasoning (Hsieh et al., 2024; Modarressi et al., 2025). A recent taxonomy frames long-context difficulty in terms of how dispersed the relevant evidence is and how much of the context must be integrated. Success on a concentrated span does not guarantee success when evidence is scattered and the decision depends on the whole (Goldman et al., 2024). Many-shot studies reach a similar conclusion: tasks that benefit from retrieving similar examples behave differently from tasks that require learning from all examples in the prompt, with the latter degrading sooner as length grows (Zou et al., 2024).

2.3 The MRCR training signal

MRCR provides a long-context supervision signal that stresses retrieval under distractors. Inputs are long, synthetic conversations that contain multiple target spans. At the end, the model is asked to return the i-th instance of a requested span with a required random prefix; responses are scored by checking the prefix and the retrieved content (OpenAI, 2025; Mai and Attia, 2025). In this thesis MRCR is used in two target formats. The format-agnostic target asks the model to reproduce the retrieved span with the prefix. The format-aware target appends a short "Final Answer:" line after the span to introduce a consistent ending pattern for downstream formatting. Both variants preserve the long retrieval component. Two properties are relevant for transfer. First, answers are long, which reinforces continuation behavior rather than early termination after a decision. Second, the objective rewards accurate copying, not option selection. MRCR is therefore a focused manipulation for retrieval, not a joint objective for retrieval and choice.

2.3.1 Why a format-aware variant and instruction preservation

The format-aware target is a training-only augmentation that serves two purposes. First, it provides a consistent end to the response, which reduces formatting drift when a downstream task expects a short, final commitment. Second, it helps preserve instruction-following behavior during supervised fine-tuning by aligning the training prompt with a clear permission to end with a "Final Answer:" line. MRCR is primarily an evaluation benchmark, but adapting its target for training does not alter its retrieval demand and helps isolate whether any transfer failure is due to formatting and instruction loss, not retrieval itself. The official LongBench v2 evaluation remains unchanged.

2.4 LongBench v2 as the transfer target

LongBench v2 evaluates long-context comprehension under a multiple-choice format. It contains 503 questions drawn from realistic sources across six task categories, with contexts ranging from roughly 8k to 2M words and the majority below 128k. Humans reach about 53.7 percent accuracy under a 15-minute time limit. The best direct-answering models are around 50 percent, while models that use longer reasoning

perform higher, which underscores the role of reasoning and inference-time compute in long-context settings (Bai et al., 2024; LongBench v2 Project, 2025). The official harness used in this thesis prepares a fixed prompt per item and expects a single letter. Invalid responses are counted as incorrect. The per-item schema exposes length and difficulty buckets that support descriptive breakdowns without changing the scoring rule. LongBench v2 complements MRCR: MRCR tests targeted retrieval in the presence of distractors; LongBench v2 tests whether a model can use available evidence to select one option.

2.5 Why transfer is non-trivial in practice

Even perfect retrieval does not guarantee correct option selection. First, multiple-choice questions require mapping retrieved content to a discrete label, which a copying objective does not teach. Second, long-answer supervision encourages continued generation rather than early termination after a committed choice, so decoding can drift past the single letter. Third, formatting and parsing matter in practice, since small deviations from the expected letter are scored as incorrect by reliable harnesses. The broader literature reinforces these points. When lexical overlap is limited or evidence must be aggregated across many locations, models degrade sharply, especially as context length grows (Hsieh et al., 2024; Modarressi et al., 2025). Many-shot evaluations also show that tasks demanding learning from the whole prompt suffer earlier and more severely with increasing length than tasks solvable by retrieving similar examples (Zou et al., 2024). These effects make retrieval-to-comprehension transfer an empirical question rather than an assumption.

2.6 Design principles used in this thesis

Principle 1. Isolate the training signal. I use one model family and one supervision signal. The only difference between fine-tuned checkpoints is whether the target includes a “Final Answer” line. This avoids conflating retrieval supervision with other forms of instruction tuning or preference modeling.

Principle 2. Fix the evaluation harness. All headline results come from the official LongBench v2 harness with fixed settings. The same scripts and prompt are used for the

baseline and both fine-tuned checkpoints, enabling paired comparisons on identical items.

Principle 3. Treat invalids as incorrect and log diagnostics. Counting invalids as incorrect prevents optimistic scoring when formatting drifts. Recording diagnostic counters supports separating formatting issues from decision errors.

Principle 4. Quantify uncertainty and per-item flows. I report bootstrap confidence intervals for accuracies and paired intervals for deltas versus the baseline, and I analyze transitions across correct, incorrect, and invalid states.

2.7 What this background implies for the rest of the thesis

If retrieval-only supervision transfers to multiple-choice comprehension, I expect non-negative deltas versus the baseline, fewer formatting failures, and a visible flow from baseline incorrect to fine-tuned correct in the transitions view. If transfer does not occur, I expect negative deltas, improved formatting at most, and a dominant flow from baseline correct to fine-tuned incorrect. The experiments test these expectations using the official LongBench v2 outputs and the transitions table derived from per-item predictions.

2.8 Scope and naming notes

All references to the benchmark use the name LongBench v2. Headline results refer to the official LongBench v2 harness with fixed evaluation settings. Path names that contain earlier internal labels are historical and do not affect the naming used in the thesis. The terms “MRCR SFT (format-agnostic)” and “MRCR SFT (format-aware)” identify the two fine-tuned checkpoints.

2.9 Summary

Long-context evaluation must consider both retrieval and decision making. MRCR provides a clear retrieval-centric signal under distractors. LongBench v2 provides a multiple-choice decision test under long inputs. Using one model family, one supervision signal, and one fixed harness allows a clean measurement of transfer from

retrieval to comprehension. The next chapters describe the experimental setup and report the results under that frame.

2.10 Why MRCR and why LongBench v2 for this study

The goal is to test whether strengthening retrieval transfers to multiple-choice comprehension under long inputs. MRCR targets retrieval explicitly by requiring the model to find the correct needle among distractors and to return it with a prefix. LongBench v2 targets decision making explicitly by requiring a single letter after reading a long context. This pair cleanly separates the skills of interest. MRCR's synthetic structure yields controlled retrieval pressure without artifacts from external knowledge, while LongBench v2's data and categories reflect realistic usage and provide a simple accuracy metric that is easy to interpret and compare (OpenAI, 2025; Bai et al., 2024; LongBench v2 Project, 2025). The format-aware MRCR variant further minimizes sources of bias by preserving instruction-following behavior during training so that transfer failure is not explained by formatting loss.

2.11 Inference-time reasoning and long tasks

Several recent evaluations suggest that allocating extra inference-time compute and using explicit reasoning steps can improve long-context performance. On LongBench v2, chain-of-thought style answering outperforms direct answers (Bai et al., 2024). Independent assessments of long task completion also report gains that track reliability and the ability to recover from mistakes, which align with deliberate reasoning and tool use rather than raw window size (METR, 2025). This thesis focuses on supervised retrieval signals without additional reasoning budgets to keep the experiment interpretable. Future work can test whether the same retrieval-only SFT behaves differently when combined with explicit inference-time reasoning.

3 Methods

3.1 Model and software stack

3.1.1 Base Model

I use Gemma-3 4B-IT with a 131 072 tokens context window as the baseline instruction-tuned checkpoint (Gemma Team, 2025). The choice balances long-context support with a small footprint suitable for controlled SFT and evaluation. Gemma-3 extends context via training and scaling recipes described in its technical report; I rely on the released instruction-tuned weights and chat template without modifying the base architecture (Gemma Team, 2025; Peng et al., 2023).

3.1.2 Parameter-efficient fine-tuning

Both fine-tuned checkpoints are trained with parameter-efficient adapters using LoRA with QLoRA quantization to reduce memory while preserving model quality (Hu et al., 2021; Dettmers et al., 2023). I set $r = 16$ and $\alpha = 32$ and enable 4-bit loading for the frozen base model. Adapters are applied to attention and MLP modules; biases are not trained. Text-only fine-tuning; any vision tower parameters are frozen and asserted non-trainable. $r = 16$ and $\alpha = 32$ are standard LoRA settings that balance adapter capacity and stability at long sequence length while keeping trainable parameters small (Hu et al., 2021; Dettmers et al., 2023).

3.1.3 Attention kernel and precision

Training runs use FlashAttention-2 (script defaults to it when available and falls back to SDPA if not detected) with bf16 compute for stability at long sequence lengths (Dao, 2023). Gradient checkpointing is enabled and key-value cache reuse is disabled during training to control memory. Runs use Unsloth's SFT stack with the Gemma chat template, response-only masking, and FA2 integration, with W&B logging (Unsloth Docs, 2025; Unsloth, 2025b).

3.1.4 Hardware and runtime

The final SFT runs were executed on a single NVIDIA H100 GPU (PCIE). Training summaries record a peak reserved memory of 75.562 GB and average throughput

between 3899 and 3955 tokens per second. The training environment and library versions are captured in a run card and saved alongside checkpoints (NVIDIA, 2022).

Given a single-GPU budget with 128k sequences, I use QLoRA, FlashAttention-2, response-only masking to keep memory and wall-time within bounds without altering the evaluation harness.

3.1.5 Model variants under test

I evaluate three models:

- Baseline (Gemma-3 4B-IT).
- MRCR SFT (format-agnostic).
- MRCR SFT (format-aware).

All three expose a 131 072 tokens context window. The evaluation harness applies a 128 000 tokens trim cap for consistency and template safety.

3.2 Training data: MRCR

MRCR is a long-context multi-needle retrieval dataset with order disambiguation. Inputs are synthetic multi-turn conversations containing 2, 4, or 8 “needles.” The target requires returning the i-th requested needle preceded by a unique 10-character alphanumeric prefix (OpenAI, 2025). I use the inspected split with 2 400 rows balanced across needle counts.

3.2.1 Format-aware training-only augmentation

To reduce formatting drift and preserve instruction following, without changing the retrieval demand, I add a training-only augmentation in one SFT run: after reproducing the retrieved span with the prefix, the model appends a single line “Final Answer: {prefix}”. The corresponding user instruction is updated to authorize exactly that extra line. This preserves the long retrieval component while introducing a consistent termination pattern and instruction following task, intended to reduce formatting drift at inference, and reduce any noise introduced by training.

3.2.2 Length filtering and final train set

Many raw conversations exceed the context budget. I render examples to the Gemma chat template and compute token counts; after filtering to ≤131 072 tokens, the usable set contains 1 464 items. This count matches the training run cards for both SFT runs and reflects the "all_needles_128k" stage that includes 2/4/8-needle items.

3.2.3 Chat templating and masking

Examples are rendered with the Gemma-3 chat template. SFT uses response-only masking so that gradients flow through the model's generated span, and in the format-aware run also through the appended final line. This isolates the retrieval signal and keeps the objective focused on copying the requested content and adhering to the minimal formatting.

3.3 Training objective

Supervised fine-tuning minimizes token-level cross-entropy over the assistant response with response-only masking. No auxiliary losses, option heads, or preference optimization are introduced.

3.3.1 Training procedure, monitoring, and throughput

Hyperparameters are kept minimal to emphasize the supervision signal: batch size per device = 1, gradient accumulation = 4, learning rate = 2e-4, linear decay with 0.03 warmup ratio, AdamW with 8-bit optimizer states, bf16 compute, sequence length = 131 072, packing disabled, and one epoch over the filtered set. Logging records tokens per second and step time. Training summaries report:

- Format-agnostic: loss 1.7102, runtime 17 884.80 s, peak reserved 75.562 GB, 3899 tokens/s.
- Format-aware: loss 1.7095, runtime 17 641.25 s, peak reserved 75.562 GB, 3955 tokens/s.

Run cards persist environment, hyperparameters, and per-bin audits for reproducibility.

3.4 Evaluation benchmark and protocol

3.4.1 Benchmark

All headline results use LongBench v2. I evaluate with the official LongBench v2 harness and fixed evaluation settings using the official scripts pred.py and result.py and the default zero-shot prompt prompts/0shot.txt. The harness produces one per-item JSONL file per model and aggregates overall and bucketed accuracies (Bai et al., 2024; LongBench v2 Project, 2025).

3.4.2 Evaluation anchor (single source of truth)

Canonical runs use the LongBench v2 official harness with fixed evaluation settings (scripts longbench/pred.py, longbench/result.py, prompt longbench/prompts/0shot.txt), harness commit 2e00731f8d0bff23dc4325161044d0ed8af94c1e, and evaluation run id 2025-09-17T14:18:34Z. A frozen copy of all evaluation artifacts is mirrored under thesis/references/eval/lb2/20250917_141834/. The harness metadata records trim_cap_tokens: 128000 and max_tokens_per_request: 128.

3.4.3 Serving configuration

Models are served behind an OpenAI-compatible endpoint. No tool calls or external retrieval are used by the harness. The harness trims inputs at 128 000 tokens to respect the template budget and convention.

3.4.4 Scoring policy and schema

The harness expects a single-letter answer. Invalid responses are counted as incorrect. The per-item schema records the gold letter and the parsed predicted letter. Missing single-letter predictions are reported by the harness as pred_none and are treated as incorrect in all accuracy computations.

An additional flag “placeholder” is computed in a separate custom pipeline to tag canned responses that are not valid letters; it is reported in Results for interpretation and defined in Appendix B, and it does not affect scoring.

3.4.5 Evaluation incident

A stricter diagnostic that enforced a “Final Answer:” line and a very small decoding budget inflated apparent accuracy due to parser artifacts. It is documented in Appendix B and excluded from canonical reporting.

3.5 Metrics

The primary endpoint is accuracy on LongBench v2, reported overall and by Length buckets (Short, Medium, Long) and Difficulty buckets (Easy, Hard). Invalid responses count as incorrect. I also compute transitions from Baseline to each SFT checkpoint across {correct, incorrect, invalid} states to quantify per-item changes.

3.6 Statistical analysis

Uncertainty is estimated using non-parametric bootstrap 95 percent confidence intervals for per-model accuracies and paired bootstrap 95 percent confidence intervals for deltas versus the Baseline on matched items; the corresponding CI tables are included with the thesis artifacts (Efron, 1979).

3.7 Reproducibility and provenance

I pin the harness commit, scripts, prompt, and evaluation settings in a repository note. I fix model context windows and decoding caps across models. I export per-item JSONLs and aggregate CSVs used to produce tables and figures. Training run cards, summaries, and monitoring logs are saved next to checkpoints. All figure and table captions reference the same evaluation anchor string.

4 Experiments and Results

This chapter reports measurements on LongBench v2 using the official harness with fixed evaluation settings. All models are evaluated with the same scripts and prompt; the pinned harness commit and the evaluation run id are recorded once in Methods and repeated in captions for traceability.

4.1 Reporting conventions

Accuracies count invalid outputs as incorrect. Confidence intervals are 95 percent bootstrap. Paired bootstrap intervals are used for deltas versus the Baseline on matched items. Buckets follow the fields in the official per-item JSONL (Length in {Short, Medium, Long}, Difficulty in {Easy, Hard}). The official schema does not expose coverage, so coverage is omitted from the main tables. All captions carry the unified source string.

4.2 Overall accuracy and deltas

Table 1 and Figure 1 report overall accuracy with 95 percent confidence intervals for the Baseline, MRCR SFT (format-agnostic), and MRCR SFT (format-aware). The Baseline reaches 28.43 percent [24.45, 32.60]. MRCR SFT (format-agnostic) achieves 21.27 percent [17.69, 24.85]. MRCR SFT (format-aware) achieves 22.66 percent [19.09, 26.44]. The paired delta versus the Baseline is negative in both cases and the confidence intervals do not include zero: −7.16 percentage points [−11.93, −2.39] for the format-agnostic checkpoint and −5.77 percentage points [−10.54, −0.99] for the format-aware checkpoint.

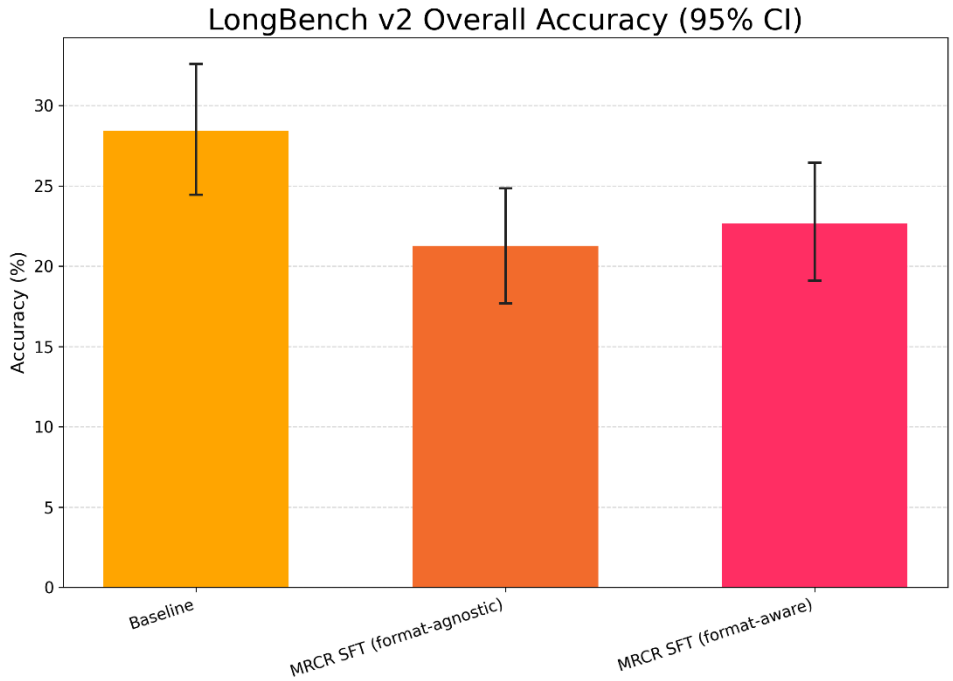

Figure 1. LongBench v2 Overall Accuracy (95% CI).
Source: Author's computation using the LongBench v2 official harness with fixed evaluation settings; run

id 2025-09-17T14:18:34Z; harness commit 2e00731f8d0bff23dc4325161044d0ed8af94c1e; artifacts: thesis/references/eval/lb2/20250917_141834/.

Table 1 reports point estimates; Figure 1 visualizes the same with error bars.

Table 1. Overall accuracy on LongBench v2 (95% CI).
Source: Author's computation using the LongBench v2 official harness with fixed evaluation settings; run id 2025-09-17T14:18:34Z; harness commit 2e00731f8d0bff23dc4325161044d0ed8af94c1e; artifacts: thesis/references/eval/lb2/20250917_141834/.

Model	n	Accuracy (%)	95% CI low	95% CI high
Baseline	503	28.43	24.45	32.60
MRCR SFT (format-agnostic)	503	21.27	17.69	24.85
MRCR SFT (format-aware)	503	22.66	19.09	26.44

4.2.1 Note on invalids and placeholders

The official JSONLs report missing letter predictions as pred_none. Counts are 3 for the Baseline, 59 for MRCR SFT (format-agnostic), and 11 for MRCR SFT (format-aware). A separate diagnostic label placeholder is produced by my auxiliary pipeline and is not part of the official schema or scoring; it is used only to aid interpretation and is defined in Appendix B. It confirms that the format-aware target greatly reduces placeholders and missing letters, yet the accuracy gap versus Baseline remains.

4.3 Bucketed results by Length and Difficulty

Table 2 and Figure 2 report accuracy by Length. Baseline reaches 30.56 on Short, 27.91 on Medium, and 25.93 on Long. MRCR SFT (format-agnostic) yields 18.89, 22.79, and 22.22. MRCR SFT (format-aware) yields 22.22, 23.72, and 21.30. Paired delta CIs versus Baseline show that the Short bucket declines significantly for both checkpoints: −11.66 percentage points [−19.44, −3.89] for format-agnostic and −8.29 percentage points [−16.11, −0.56] for format-aware. Medium and Long buckets are negative on average but their 95 percent intervals include zero.

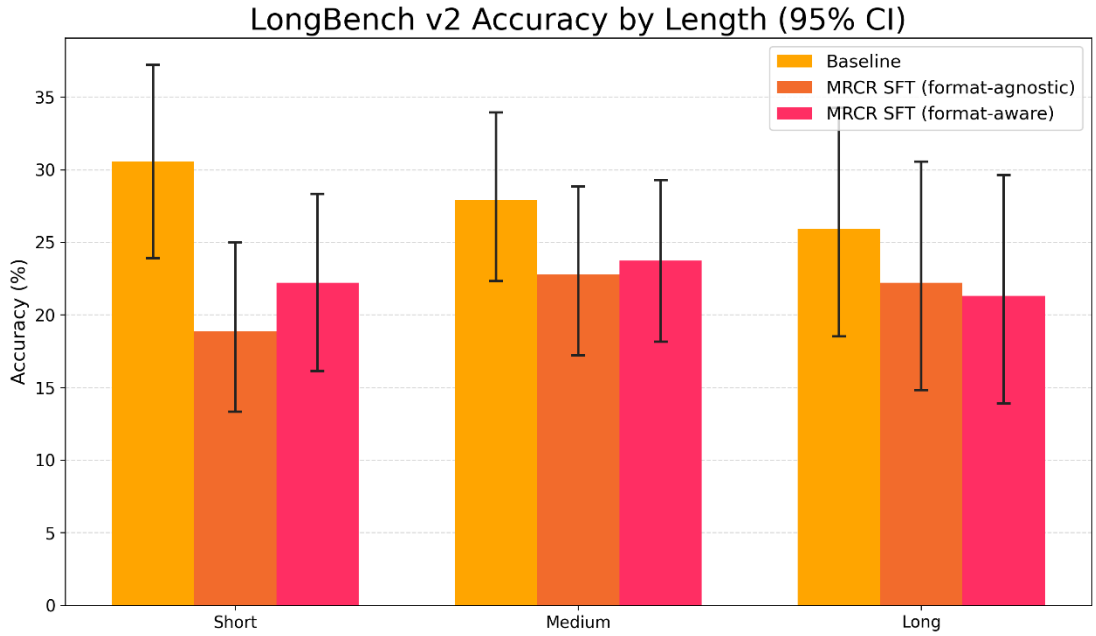

Figure 2. LongBench v2 Accuracy by Length (95% CI).
Source: Author's computation using the LongBench v2 official harness with fixed evaluation settings; run id 2025-09-17T14:18:34Z; harness commit 2e00731f8d0bff23dc4325161044d0ed8af94c1e; artifacts: thesis/references/eval/lb2/20250917_141834/.

Table 2. Accuracy by Length on LongBench v2 (95% CI).
Source: Author's computation using the LongBench v2 official harness with fixed evaluation settings; run id 2025-09-17T14:18:34Z; harness commit 2e00731f8d0bff23dc4325161044d0ed8af94c1e; artifacts: thesis/references/eval/lb2/20250917_141834/.

Length	n	Baseline (Acc, 95% CI)	Format-agnostic (Acc, 95% CI)	Format-aware (Acc, 95% CI)
Short	180	30.56 (23.89–37.22)	18.89 (13.33–25.00)	22.22 (16.11–28.33)
Medium	215	27.91 (22.31–33.95)	22.79 (17.21–28.84)	23.72 (18.14–29.30)
Long	108	25.93 (18.52–34.26)	22.22 (14.81–30.56)	21.30 (13.89–29.63)

Table 3 and Figure 3 report accuracy by Difficulty. Baseline reaches 29.69 on Easy and 27.65 on Hard. MRCR SFT (format-agnostic) yields 22.92 and 20.26. MRCR SFT (format-aware) yields 22.92 and 22.51. Paired delta CIs indicate a significant drop on Hard for the format-agnostic checkpoint, −7.43 percentage points [−13.50, −1.29],

while the format-aware delta on Hard, −5.13 percentage points [−11.25, 0.96], includes zero. Easy deltas are about −6.7 percentage points with intervals that include zero.

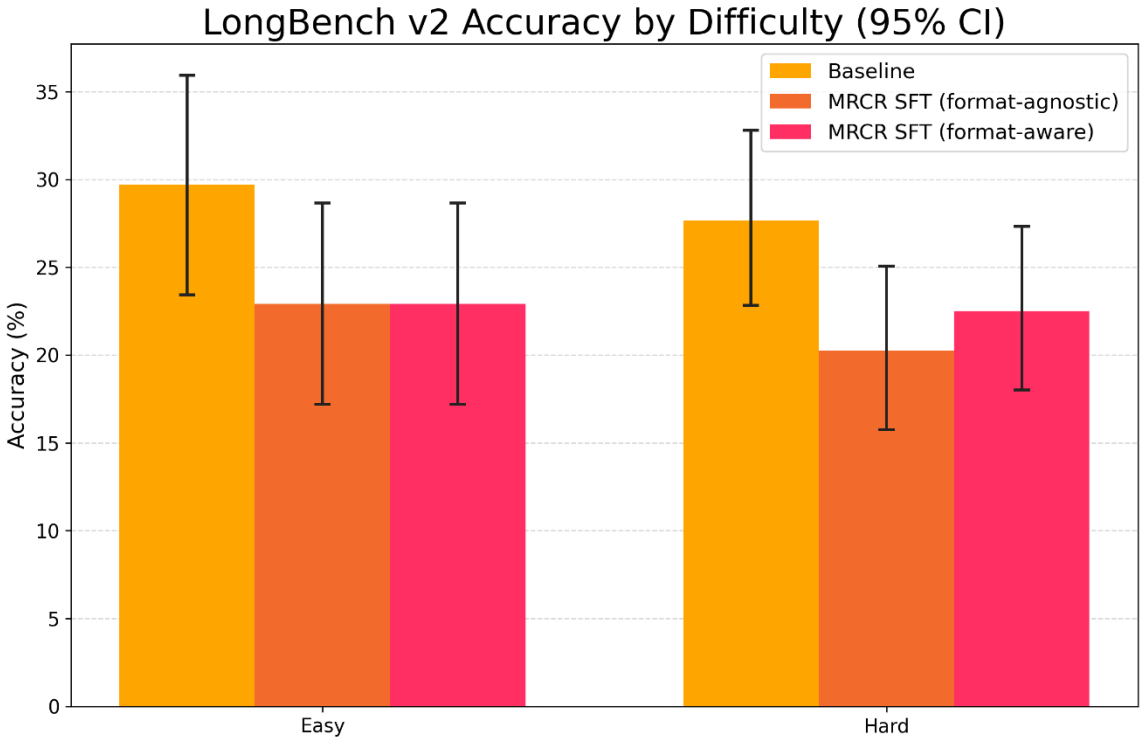

Figure 3. LongBench v2 Accuracy by Difficulty (95% CI).
Source: Author's computation using the LongBench v2 official harness with fixed evaluation settings; run id 2025-09-17T14:18:34Z; harness commit 2e00731f8d0bff23dc4325161044d0ed8af94c1e; artifacts: thesis/references/eval/lb2/20250917_141834/.

Table 3. Accuracy by Difficulty on LongBench v2 (95% CI).
Source: Author's computation using the LongBench v2 official harness with fixed evaluation settings; run id 2025-09-17T14:18:34Z; harness commit 2e00731f8d0bff23dc4325161044d0ed8af94c1e; artifacts: thesis/references/eval/lb2/20250917_141834/.

Difficulty	n	Baseline (Acc, 95% CI)	Format-agnostic (Acc, 95% CI)	Format-aware (Acc, 95% CI)
Easy	192	29.69 (23.44–35.94)	22.92 (17.19–28.65)	22.92 (17.19–28.65)
Hard	311	27.65 (22.83–32.80)	20.26 (15.76–25.08)	22.51 (18.01–27.33)

4.4 Transitions from Baseline to MRCR SFT

To understand how item outcomes change, I compute transitions from Baseline states to SFT outcomes across the nine possible flows between {correct, incorrect, invalid}. The dominant pattern is a net increase in correct→incorrect flows that outweighs incorrect→correct recoveries. Flows into invalid, driven by formatting behavior, are lower for the format-aware checkpoint. See Appendix 7.5.4 for the condensed transition table; the full CSV remains in Appendix D's artifact map.

4.5 Error patterns and formatting diagnostics

The official harness reports two counters: with_pred (number of parsed single-letter predictions) and pred_none (missing letter). Figure 4 summarizes these, and the auxiliary placeholder diagnostic from my pipeline. Counts are: Baseline with_pred 500, pred_none 3, placeholder 0; format-agnostic with_pred 444, pred_none 59, placeholder 48; format-aware with_pred 492, pred_none 11, placeholder 6. The format-aware target improves compliance and reduces both pred_none and placeholder, yet the accuracy gap relative to the Baseline remains. Placeholder is defined and discussed in Appendix B and is never used for scoring.

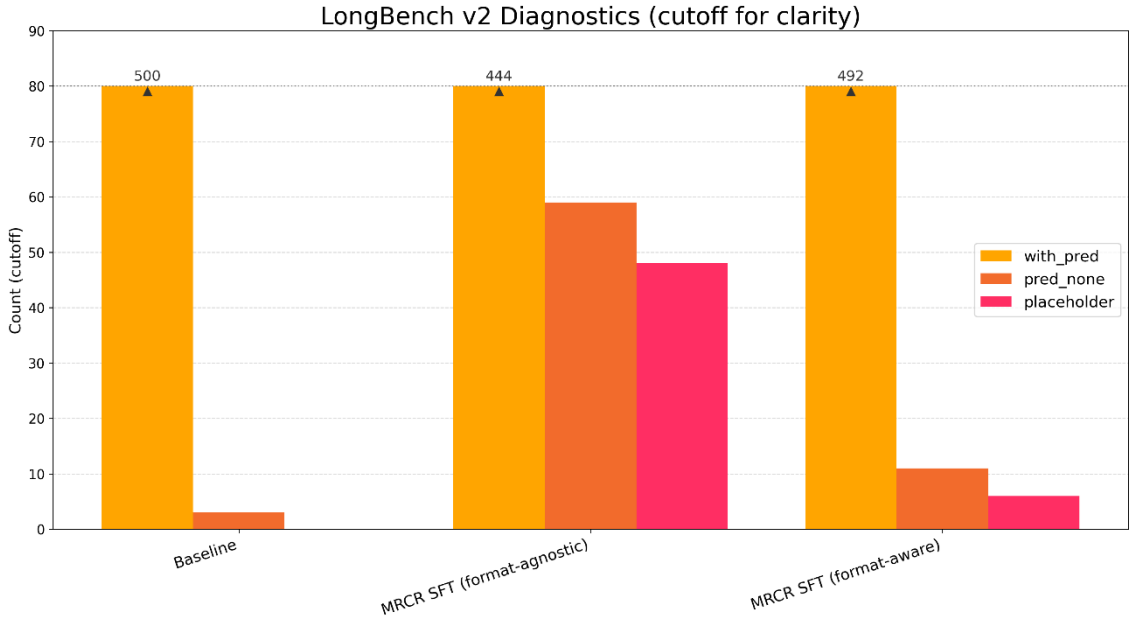

Figure 4. LongBench v2 diagnostics (cutoff for clarity): with_pred, pred_none, and placeholder counts. Bars for with_pred are clipped at 80 to improve readability. Black markers and labels show the true counts (Baseline = 500, Format-agnostic = 444, Format-aware = 492)
Source: Author's computation using the LongBench v2 official harness with fixed evaluation settings; run

id 2025-09-17T14:18:34Z; harness commit 2e00731f8d0bff23dc4325161044d0ed8af94c1e; artifacts: thesis/references/eval/lb2/20250917_141834/.

4.6 Domain-level accuracy

I compute accuracy by domain for descriptive purposes. The intent is to show that the negative transfer appears across domains rather than concentrating in a single category.

See Appendix 7.5.5; the full CSV remains in Appendix D.

4.7 Summary of findings

MRCR supervision does not improve LongBench v2 accuracy for this model at this training scale. The overall deltas versus the Baseline are negative and statistically supported. The decrease is most pronounced for Short items and present for both Easy and Hard items. Transitions show that correct→incorrect flows dominate, while incorrect→correct recoveries are insufficient to offset the loss. Format-aware targets reduce placeholders and missing predictions but do not restore accuracy to the Baseline level. All results use the official LongBench v2 harness with fixed evaluation settings and with the scripts and prompt pinned to the commit recorded in the evaluation note.

5 Discussion and Conclusion

5.1 Interpretation of the main result

This study asked whether supervised training for multi-needle retrieval transfers to long-context multiple-choice comprehension. Under the official LongBench v2 harness with fixed evaluation settings, the answer is negative for the configuration tested here. The baseline Gemma-3-4B-IT reaches 28.43 percent accuracy on 503 items; the MRCR-SFT checkpoints reach 21.27 percent (format-agnostic) and 22.66 percent (format-aware). Paired bootstrap confidence intervals for the deltas versus baseline exclude zero for both fine-tuned models, which supports a real drop rather than sampling noise. The run uses the official LongBench v2 harness with a pinned commit and recorded run id.

The format-aware target improves compliance, reducing missing single-letter predictions from 59 to 11 and placeholders from 48 to 6, but it does not recover accuracy relative to the baseline. Item-level transitions show that the dominant flow is baseline-correct to SFT-incorrect, and the reduction in invalids does not compensate for the rise in wrong-letter commitments. These diagnostics indicate that formatting contributed to early failures, yet the residual gap is mainly decision error rather than a parsing artifact.

5.2 Where the loss concentrates

By length, the largest and statistically supported decrease appears in the Short bucket. Medium and Long trend negative on average but their paired intervals include zero. By difficulty, both Easy and Hard subsets drop relative to baseline, with a clearer effect on Hard for the format-agnostic checkpoint. This matches a supervision signal that strengthens copying but does not teach the short, discrete option mapping required by LongBench v2.

The compliance improvements from the format-aware target align with the diagnostics. It reduces missing letters and boilerplate placeholders while leaving the error profile dominated by wrong-letter choices. In practice, once the model emits a letter it is more often the wrong one, especially on Short items where retrieval pressure is lower and option selection is central.

5.3 Why retrieval did not transfer to comprehension

Mechanism 1 - objective mismatch. MRCR supervision rewards reproducing the requested span under long distractors. With response-only masking, gradients flow through the generated span rather than a discrete choice. The format-aware variant adds a short final line that improves instruction-following and reduces formatting drift, and it succeeds on that goal, yet it does not add supervision on option selection. The model learns to copy long spans reliably but not to compare alternatives and commit to a single letter.

Mechanism 2 - data regime and behavior shaping. After filtering to the 131 072 tokens budget, the training pool contains about 1 464 MRCR items. Answers are long, so SFT

encourages continuation and verbosity. That combination is enough to change generation habits but is modest for re-learning a robust short-answer decision rule on diverse LongBench v2 items. Run cards and dataset stats document both the size and length profile of the filtered set.

These findings also align with recent reports that long-context success requires more than access to the right span and that additional inference-time reasoning can help on LongBench v2 tasks. When relevant evidence is not lexically obvious or is dispersed, models tend to drop without extra reasoning time.

5.4 Threats to validity

5.4.1 Single model family

Results are based on Gemma-3-4B-IT only. Larger Gemma-3 variants or other instruction-tuned families could respond differently to the same MRCR supervision. LongBench v2 reports that models with longer reasoning budgets can outperform direct-answering peers, which suggests that test-time compute interacts with training signals.

5.4.2 Single training signal

The study intentionally isolates MRCR SFT to answer the research question cleanly. Adding multiple-choice supervision or preference objectives could change outcomes, but would mix effects and weaken the causal reading we aim for. The training recipe, run cards, and monitoring confirm that both SFT variants share identical hyperparameters and environment, which limits alternative explanations.

5.4.3 Evaluation harness

All headline results use the official LongBench v2 harness and prompt with fixed settings, pinned commit, and recorded run id. A stricter mid-study pipeline briefly inflated scores due to parser shortcuts. Manual review and strict rescoring showed those runs were not reliable, so they are treated as diagnostic only. We therefore report the canonical harness results. See Appendix B for the incident timeline and strict rescoring note.

5.4.4 Run-time effects

Throughput and memory appear stable according to the training summaries and logging, but performance characteristics can vary with server configuration. The repository contains a simple Weights and Biases export that records tokens per second during training and supports the stated throughput ranges.

5.5 Limitations

5.5.1 No MRCR manipulation check is reported beyond training loss

There is no held-out MRCR test to quantify retrieval gains directly. Training summaries and run cards show stable training and throughput but do not replace an explicit retrieval evaluation.

5.5.2 No decoding constraints

The official harness expects a single letter, and we did not apply logit bias or constrained decoders to force that emission. The format-aware SFT already removes most formatting failures, so stricter decoding would likely offer limited gains while changing the task.

5.5.3 No coverage reporting

The official per-item schema does not expose coverage. Inputs are trimmed to 128 000 tokens to protect the template budget, and the trim cap is recorded in run metadata, but coverage cannot be analyzed in the main tables.

5.6 Implications and recommendations

For a small long-context model, retrieval-only SFT is not sufficient to improve multiple-choice comprehension on LongBench v2. If the goal is higher accuracy under the same harness, training and decoding should teach or enforce the decision step.

1. Add an option-aware objective. Mix a small MC head or a token-level bias toward the letter set into SFT so the model learns the mapping from retrieved content to a discrete option. Keep long contexts in the prompt to preserve the retrieval burden.

2. Light decoding constraints. Apply a minimal logit bias to {A,B,C,D}, emit the first valid letter, then stop. The format-aware run already reduced invalids; this step reduces variance further without changing the evidence used.
3. Preserve instruction-following during SFT. Keep the format-aware target or an equivalent short final-answer contract in the SFT mix to avoid regressions in compliance.
4. Allocate test-time reasoning budget when allowed. LongBench v2 reports higher scores when models spend more time reasoning before answering. Short scratchpads can help where the harness permits them.

These steps address the decision-making gap seen in the transitions and diagnostics while keeping the canonical harness unchanged for comparability.

5.7 Future work

5.7.1 Constrained decoding

Test letter-biased decoding with immediate stop to measure the ceiling from format control alone. Keep the same harness and report paired deltas with bootstrap confidence intervals.

5.7.2 Cross-model replication

Repeat the study on a second small model, a medium model, and optionally a comparable open family, to measure scale sensitivity and instruction-tuning interactions under identical evaluation.

Great candidates for this are from Gemma's own model family: Gemma 3 12B and 27B IT (Instruction Tuned).

As for choosing another model family, the Qwen 3 family is the closest in performance and size for comparability, for expanding this study: especially the 4B variant.

5.7.3 Reasoning models and budgets

Evaluate whether retrieval-only SFT behaves differently when paired with explicit inference-time reasoning, given LongBench v2 evidence that longer reasoning helps.

5.7.4 Rigorous MRCR manipulation check

Add a held-out MRCR split to directly measure retrieval improvements and relate them to LongBench v2 flows by length and difficulty.

5.7.5 Evaluation pipeline hardening

Finalize a parser that enforces Final Answer: {letter} without fragile shortcuts, then re-run baseline and the format-aware SFT to confirm the conclusions under a stricter but reliable rule set. The status report tracks this plan.

5.8 Conclusion

This thesis measured whether MRCR training transfers to long-context multiple-choice comprehension for a small instruction-tuned model. Using the official LongBench v2 harness with fixed evaluation settings, both MRCR SFT variants underperform the Baseline, and the negative deltas are statistically supported. Diagnostics and closer examination show that the format-aware checkpoint improves format compliance, correcting many “pred_none”, but do not change the main outcome. The findings indicate that retrieval skill alone does not translate into option selection accuracy in long contexts. Effective transfer likely requires explicit supervision or architectural support for mapping retrieved content to choices. The repository includes per-item outputs, run cards, training summaries, and an evaluation environment note that make the results reproducible and set a clear baseline for future work.

6 References

Author’s computation. 2025 [Internal bundle]. Official LongBench v2 evaluation environment. Files: OFFICIAL_EVAL_ENV.md; longbench.COMMIT (2e00731f8d0bff23dc4325161044d0ed8af94c1e); longbench/pred.py; longbench/result.py; longbench/prompts/0shot.txt. Path: thesis/references/env/. Date: 17 Sep 2025.

Author’s computation. 2025 [Internal bundle]. LongBench v2 outputs and summaries for Baseline, MRCR SFT (format-agnostic), MRCR SFT (format-aware): result.txt; lb2_overall_with_ci.csv; lb2_overall_deltas_with_ci.csv; lb2_length_acc_with_ci.csv;

lb2_difficulty_acc_with_ci.csv; lb2_transitions_overall.csv; table2a_lb2_overall.csv; table2b_lb2_by_length.csv; table2c_lb2_by_difficulty.csv; table3_lb2_transitions.csv; per-item JSONL predictions (503 items per model). Path: thesis/references/eval/lb2/20250917_141834/. Date: 17 Sep 2025.

Author's computation. 2025 [Internal bundle]. Figures generated from the official LongBench v2 outputs: fig1_lb2_overall_accuracy.png; fig2_lb2_by_length.png; fig3_lb2_by_difficulty.png; fig4_lb2_transitions.png; figA1_lb2_by_domain.png. Path: thesis/references/fig/. Date: 19 Sep 2025.

Author's computation. 2025 [Internal bundle]. MRCR SFT training artifacts for Gemma-3 4B-IT: run_card.json; train_summary.json; wandb-summary.json (format-aware run); MRCR dataset stats (mrcr_dataset_stats.json). Paths: thesis/references/train/gemma3_mrcr_all_128k/; thesis/references/train/gemma3_mrcr_all_128k_v2/. Dates: 09 Sep 2025 and 17 Sep 2025.

Bai, Y. et al. 2024. LongBench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. arXiv. doi: 10.48550/arXiv.2412.15204.

Dao, T. 2023. FlashAttention-2: Faster attention with better parallelism and work partitioning. arXiv. doi: 10.48550/arXiv.2307.08691.

Dettmers, T. et al. 2023. QLoRA: Efficient finetuning of quantized LLMs. arXiv. doi: 10.48550/arXiv.2305.14314.

Efron, B. 1979. Bootstrap methods: Another look at the jackknife. The Annals of Statistics, 7(1), 1–26. Available at: https://projecteuclid.org/journals/annals-of-statistics/volume-7/issue-1/Bootstrap-Methods-Another-Look-at-the-Jackknife/10.1214/aos/1176344552.full (Accessed: 20 Sep 2025).

Gemma Team. 2025. Gemma 3 Technical Report. arXiv. doi: 10.48550/arXiv.2503.19786.

Goldman, O., Jacovi, A., Slobodkin, A., Maimon, A., Dagan, I. and Tsarfaty, R. 2024. Is it really long context if all you need is retrieval? Towards genuinely difficult long context NLP. arXiv. doi: 10.48550/arXiv.2407.00402.

Hsieh, C.-Y. et al. 2024. RULER: What is the real context size of your long-context language models. arXiv. doi: 10.48550/arXiv.2404.06654.

Hu, E. J. et al. 2021. LoRA: Low-rank adaptation of large language models. arXiv. doi: 10.48550/arXiv.2106.09685.

Kwon, W. et al. 2023. Efficient memory management for large language model serving with PagedAttention. Proceedings of the ACM Symposium on Operating Systems Principles. doi: 10.1145/3600006.3613165.

Liu, N. F. et al. 2023. Lost in the middle: How language models use long contexts. Transactions of the ACL. doi: 10.48550/arXiv.2307.03172.

LongBench v2 Project. 2025. LongBench v2 website. Available at: https://longbench2.github.io/ (Accessed: 20 Sep 2025).

Mai, Y. and Attia, J. 2025. HELM Long Context. HELM Blog. Available at: https://crfm.stanford.edu/2025/09/09/helm-long-context.html (Accessed: 20 Sep 2025).

METR. 2025. Measuring AI ability to complete long tasks. arXiv. doi: 10.48550/arXiv.2503.14499.

Modarressi, A. et al. 2025. NoLiMa: Long-context evaluation beyond literal matching. arXiv. doi: 10.48550/arXiv.2502.05167.

NVIDIA. 2022. NVIDIA H100 Tensor Core GPU architecture whitepaper. Available at: https://www.nvidia.com/it-it/data-center/h100/ (Accessed: 20 Sep 2025).

OpenAI. 2025. MRCR: Multi-needle retrieval dataset. Available at: https://huggingface.co/datasets/openai/mrcr (Accessed: 20 Sep 2025).

Peng, B. et al. 2023. YaRN: Efficient context window extension of large language models. arXiv. doi: 10.48550/arXiv.2309.00071.

Unsloth. 2025b. Gemma 3 finetuning with Unsloth. Available at: https://unsloth.ai/blog/gemma3 (Accessed: 20 Sep 2025).

Unsloth Docs. 2025. Unsloth documentation. Available at: https://docs.unsloth.ai/ (Accessed: 20 Sep 2025).

Zou, K., Khalifa, M. and Wang, L. 2024. Retrieval or global context understanding? On many-shot in-context learning for long-context evaluation. arXiv. doi: 10.48550/arXiv.2411.07130.

7 Appendices

7.1 Appendix A. Coverage and truncation (qualitative)

Why coverage is omitted: The official LongBench v2 harness used in this thesis does not emit a coverage field in its per-item JSONL schema, so there is no authoritative post-truncation token-coverage value to aggregate. In line with the project's evaluation note, canonical tables therefore report accuracy and uncertainty only; coverage is discussed qualitatively here.

Truncation policy (what actually ran): For all canonical runs, we used the LongBench v2 runner and pinned it to commit 2e00731f8d0bff23dc4325161044d0ed8af94c1e (recorded in longbench.COMMIT). The runner's pred.py applies center truncation when prompts exceed a model's declared maximum length: it keeps the first half and the last half of tokens around a max_len cap. In our canonical configuration, the harness metadata records trim_cap_tokens: 128000 (model window 131 072) and max_tokens_per_request: 128. This leaves a safety margin for the chat template while preserving document head and tail.

Length categories vs. truncation: LongBench v2's length categories (short, medium, long) are defined at dataset construction time (by word-length ranges) and shipped in each instance; the harness uses those labels for stratified reporting. Our center-truncation step is an evaluation-time safeguard and does not alter those labels. The official README describes the benchmark's focus (8k–2M-word contexts; majority under 128k) and the standardized multiple-choice format used for reliable scoring.

What we did track qualitatively: As a proxy for formatting/termination issues, we logged the harness's pred_none counter (responses where no single letter could be parsed) alongside a small "placeholder" diagnostic from our auxiliary scripts. Aggregated over all 503 items, pred_none was 3 (baseline), 59 (MRCR SFT v1), 11 (MRCR SFT v2, format-aware); placeholders fell from 48 → 6 with the format-aware run. These improvements confirm that the format-aware target materially reduced formatting drift, without reversing the accuracy gap. (Per-bucket pred_none tallies were

inspected during analysis but are not part of the harness output; see also Appendix B for the strict rescoring discussion.)

Reproducibility pointers: The "Official LongBench v2 Evaluation Environment" note documents the canonical sequence: serve model with vLLM, run pred.py with the default zero-shot prompt, then aggregate with result.py. The note also stresses that invalid responses count as incorrect, and that we must not mix strict/diagnostic pipelines with canonical reporting, this thesis follows both conventions.

7.2 Appendix B. Evaluation incident (16 Sep 2025) and strict rescoring

On 16 Sep 2025 I tested a stricter evaluation variant that enforced a "Final Answer:" line with very low decoding budgets (4 or 32 tokens). Manual checks showed that the parser occasionally credited partial markdown bullets, inflating apparent accuracy. The model sometimes repeated the options and the parser picked up the letter from it. I therefore treated these runs as diagnostic only and reverted to the official LongBench v2 harness with fixed evaluation settings for canonical reporting. A strict rescoring pass confirmed that the inflated scores were artefacts of formatting and parsing. The canonical caption provenance in the main text names the evaluation run id 2025-09-17T14:18:34Z and the pinned harness commit 2e00731f8d0bff23dc4325161044d0ed8af94c1e for reproducibility.

In this thesis, placeholder is an auxiliary diagnostic tag produced by my analysis script for canned, non-letter boilerplate (e.g., generic preambles or markdown bullets) that the official harness would not parse as a single letter; it is never used for scoring.

7.3 Appendix C. Training-loss (format-aware SFT)

Figure 5 plots training loss over steps for the format-aware MRCR SFT run on Gemma-3 4B-IT. Both SFT runs share the same setup: LoRA $r = 16$, $\alpha = 32$; bf16 with 4-bit loading; max sequence length 131 072; stage all_needles_128k. The training subset filtered to ≤131 k tokens contains 1 464 examples. The format-aware run completed in ~17 640 s at ~3 956 tokens/s, with stable throughput. The curve shows a smooth decrease without instability, supporting the claim that training proceeded normally; this

appendix is provided for monitoring transparency and does not change the interpretation of the LongBench results.

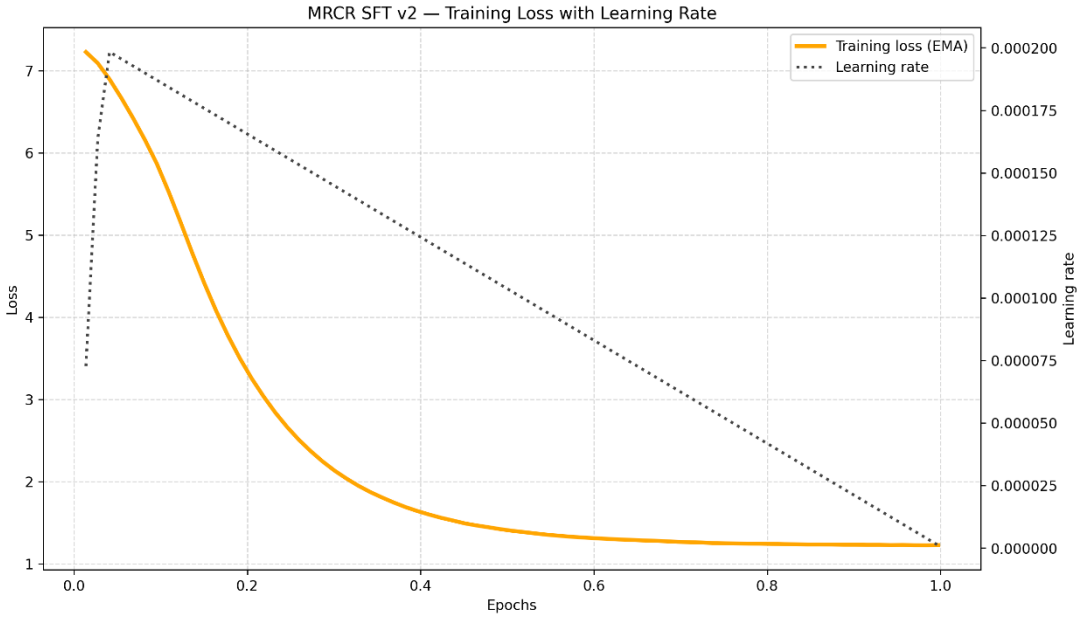

Figure 5. Training Loss with Learning Rate from format-aware checkpoint training Source: Author's computation from training logs and W&B export (thesis/references/train/gemma3_mrcr_all_128k_v2/train_summary.json, thesis/references/train/gemma3_mrcr_all_128k_v2/wandb-summary.json), run: gemma3-4b_mrcr_all_128k_v2_1ep_ga4, 17 Sep 2025.

7.4 Appendix D. Reproducibility checklist and artifact map

Harness and settings (canonical):

- Benchmark name in text: LongBench v2; phrase: "official LongBench v2 harness with fixed evaluation settings".
- Pinned harness commit: 2e00731f8d0bff23dc4325161044d0ed8af94c1e.
- Scripts and prompt: longbench/pred.py, longbench/result.py, longbench/prompts/0shot.txt.
- Canonical evaluation run id: 2025-09-17T14:18:34Z.
- Policy: invalid responses count as incorrect; coverage omitted by schema.
- Server: OpenAI-compatible endpoint; context window 131 072; trim cap 128 000; max tokens per request 128.

Artifacts included with this thesis (paths in my repository):

- Per-item outputs for each model; aggregate CSVs for overall and buckets; paired bootstrap CI CSVs for deltas.
- Training run cards and summaries for both SFT runs; W&B export (format-aware run) for Figure 5.
- Diagnostic materials (strict rescoring, manual checks) referenced from the incident note.

7.5 Supplementary Tables (condensed)

Source: Author's computation using the LongBench v2 official harness with fixed evaluation settings; run id 2025-09-17T14:18:34Z; harness commit 2e00731f8d0bff23dc4325161044d0ed8af94c1e; artifacts: thesis/references/eval/lb2/20250917_141834/.

7.5.1 Overall accuracy (95% CI)

Model	n	Accuracy (%)	95% CI low	95% CI high
Baseline	503	28.43	24.45	32.60
MRCR SFT (format-agnostic)	503	21.27	17.69	24.85
MRCR SFT (format-aware)	503	22.66	19.09	26.44

7.5.2 Accuracy by Length (95% CI)

Length	n	Baseline (Acc, 95% CI)	Format-agnostic (Acc, 95% CI)	Format-aware (Acc, 95% CI)
Short	180	30.56 (23.89–37.22)	18.89 (13.33–25.00)	22.22 (16.11–28.33)
Medium	215	27.91 (22.31–33.95)	22.79 (17.21–28.84)	23.72 (18.14–29.30)
Long	108	25.93 (18.52–34.26)	22.22 (14.81–30.56)	21.30 (13.89–29.63)

7.5.3 Accuracy by Difficulty (95% CI)

Difficulty	n	Baseline (Acc, 95% CI)	Format-agnostic (Acc, 95% CI)	Format-aware (Acc, 95% CI)
Easy	192	29.69 (23.44–35.94)	22.92 (17.19–28.65)	22.92 (17.19–28.65)
Hard	311	27.65 (22.83–32.80)	20.26 (15.76–25.08)	22.51 (18.01–27.33)

7.5.4 Transitions from Baseline to MRCR SFT (overall flows, counts)

Note: negative "net Δ correct" indicates loss of correct answers vs. baseline on matched items.

Model	Correct to incorrect	Incorrect to correct	Correct to invalid	Net Δ correct	pred_none
MRCR SFT (format-agnostic)	79	61	18	-36	59
MRCR SFT (format-aware)	89	64	4	-29	11

7.5.5 Accuracy by Domain

domain	n	Baseline Acc (%)	Format-agnostic Acc (%)	Format-aware Acc (%)
Code Repository Understanding	50	20.00	22.00	26.00
Long In-context Learning	81	33.33	22.22	23.46
Long Structured Data Understanding	33	27.27	36.36	30.30

Long-dialogue History Understanding	39	35.90	23.08	23.08
Multi-Document QA	125	32.00	16.80	19.20
Single-Document QA	175	24.57	20.57	22.29