# WHEN EASY CONTEXTS TEACH LESS: CONTEXTUAL ANSWERABILITY-GUIDED DATA REFORMULATION FOR LLM TRAINING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

The growing scarcity of high-quality natural corpora has made data reformulation and synthesis increasingly important for large language model training. However, existing document reformulation methods largely rely on human-designed heuristics, while identifying useful strategies typically requires expensive corpus generation and train-and-evaluate comparisons. In this work, we investigate whether feedback from the model itself can predict the knowledge-learning utility of reformulated data before training. We introduce Contextual Answerability (CA), which measures how readily a model can recover a target fact when a reformulated document is provided as context. Interestingly, we uncover a consistent and counter-intuitive relationship: **reformulations from which the model can more readily recover factual knowledge yield smaller improvements in closed-book factual recall after training.** In other words, higher CA indicates lower knowledge learnability. Our gradient analysis suggests that readily answerable reformulations favor context-dependent processing, whereas less answerable reformulations provide comparatively stronger signals for parametric knowledge acquisition. Building on this analysis, we propose CARVE, a self-evolving framework that uses CA as feedback to iteratively explore and refine effective corpus-level reformulation strategies under a factual-fidelity constraint. Experiments show that the reformulation selected by CARVE outperforms the original corpus and reformulations without feedback during continuous training, reaching a score comparable to the final baselines with **27% fewer training tokens**. The same reformulation also transfers to pretraining from scratch across multiple model scales, where it consistently achieves the highest average performance.

## 1 INTRODUCTION

Large language models (LLMs) have rapidly evolved from general text generation into systems capable of complex reasoning, tool use, and agentic interaction with external environments (Guo et al., 2025; Schick et al., 2023; Yao et al., 2022; Li, 2025). These capabilities ultimately depend on the representations and parametric knowledge of the underlying foundation model, whose continued improvement requires scaling model capacity, computation, and high-quality training data (Zhou et al., 2023; Lai et al., 2025; Hoffmann et al., 2022; Chen et al., 2025). However, as high-quality public text is repeatedly collected, filtered, and reused, LLM development is gradually entering a data-constrained regime: the demand for training tokens may grow faster than the supply of high-quality natural text, creating a potential data wall for further scaling (Villalobos et al., 2024; Muennighoff et al., 2023; Wang et al., 2026).

To mitigate this bottleneck, data reformulation and synthesis have become increasingly important for LLM training (Long et al., 2024; Kang et al., 2025). Within this broader trend, several studies have improved pretraining corpora by changing how their knowledge is organized and expressed. For instance, WRAP rewrites web documents into styles such as Wikipedia articles and question–answer pairs, improving data and compute efficiency (Maini et al., 2024). Nemotron-CC uses synthetic rephrasing to recover useful training signals from noisy web documents and create diverse variants of high-quality text (Su et al., 2025), while MGA expands curated corpora through diverse genre–audience reformulations (Xintong et al., 2026). Despite these promising results, current approaches



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

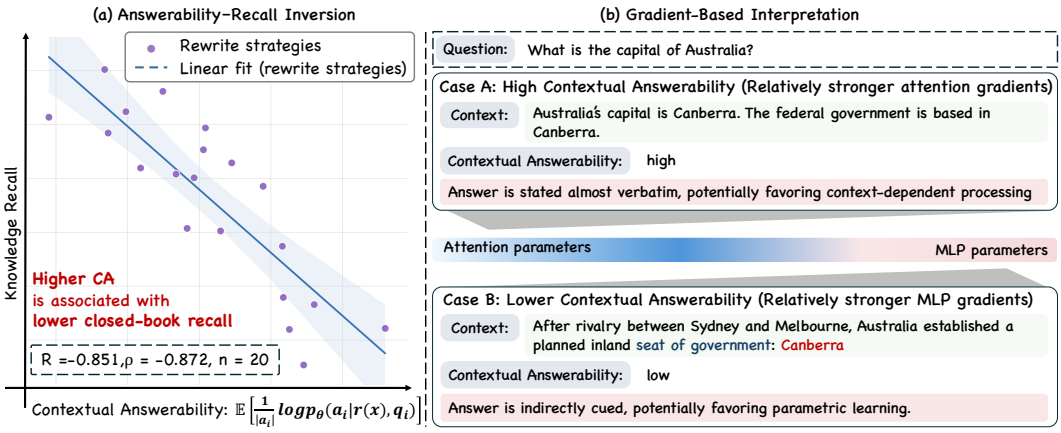


Figure 1: **Higher Contextual Answerability predicts lower knowledge learnability. (a)** Across 20 natural-text reformulation strategies, Contextual Answerability measured by a frozen model before training is strongly negatively correlated with closed-book recall performance after training. **(b)** Our gradient analysis suggests a possible explanation: high-CA reformulations are associated with relatively stronger attention gradients, whereas low-CA reformulations place greater relative optimization pressure on MLP modules associated with factual knowledge storage.

face two major limitations. First, reformulation strategies are largely specified through human-designed prompts and heuristics, with little feedback about how a given base model responds to the resulting data. Second, determining whether a strategy is useful typically requires generating large-scale data and training a separate model, making exhaustive train-and-evaluate search prohibitively expensive at the pretraining or mid-training scale (Long et al., 2024; Kang et al., 2025). This raises a natural question: *can we obtain a lightweight signal that predicts the knowledge acquisition utility of synthetic data for a given model, before any training is performed?*

In this paper, we approach this question from the perspective of model-dependent behavioral feedback. Rather than judging synthetic data solely by human-defined quality criteria, we ask how the base model itself responds to each candidate reformulation. Specifically, given a source corpus, we first extract a set of question–answer pairs about the facts it contains. We then place a candidate reformulation in the context of a frozen base model, ask the corresponding question, and compute the average token log-likelihood assigned to the correct answer. We refer to this score as *Contextual Answerability*. Unlike general measures of fluency or textual quality, Contextual Answerability captures how readily a specific model can use the factual information expressed by a particular reformulation. Intuitively, if a reformulation presents the target fact in a form that the model can easily identify and use, it should receive a higher Contextual Answerability score and might be expected to provide a more effective example for learning the same knowledge.

Surprisingly, our experiments reveal a consistent pattern in the opposite direction: **reformulations with higher Contextual Answerability yield *smaller* improvements in closed-book factual recall after training**. As shown in Figure 1(a), across twenty diverse reformulation strategies, Contextual Answerability exhibits a strong negative correlation with the resulting closed-book factual recall after training. This counter-intuitive finding reveals a distinction between contextual accessibility and parametric learnability, showing that *knowledge that is easy to recover from context is not necessarily easy to internalize in model parameters*. To investigate the optimization pattern associated with this inversion, we analyze how different reformulations distribute their gradients between the Attention and MLP modules of the Transformer (Vaswani et al., 2017). Interestingly, our analysis reveals that higher Contextual Answerability is associated with a lower MLP-to-attention gradient ratio, whereas lower-answerability reformulations place relatively greater optimization pressure on MLP parameters. This pattern potentially suggests that readily answerable reformulations rely more heavily on attention-mediated contextual processing, while less answerable reformulations provide comparatively stronger learning signals to MLP modules associated with factual knowledge storage.

Since Contextual Answerability provides a lightweight and training-free estimate of reformulation utility, we naturally introduce **CARVE** (**C**ontextual **A**nswerability-guided **R**eformulation **V**ia **E**volution), a self-evolving framework that uses model feedback to explore effective corpus-level

reformulation strategies. CARVE begins with open-ended exploration and evaluates each proposed strategy through Contextual Answerability and factual fidelity, representative rewrites, and diagnostic failures accumulated in an evolutionary strategy pool. As this experience grows, it identifies rewriting properties associated with favorable outcomes and translates them into guidance for subsequent generation. Repeating this generation–evaluation–analysis loop progressively develops promising rewriting directions, allowing the reformulation strategies to evolve across rounds.

Experiments show that the strategies discovered by CARVE improve both continuous training and pretraining from scratch scenarios. During 50B-token continuous training, CARVE-reformulated data consistently outperforms the original corpus and reformulations without any feedback throughout the training process, exhibits more stable training, and **reaches the same performance level with 27% fewer training tokens**. In addition, we show that the same reformulated data also transfers to pretraining from scratch scenario across multiple model scales, where it achieves the highest average performance across all evaluated model scales and consistently outperforms baselines. Together, these results demonstrate that CA feedback from the model can guide the discovery of effective and reusable data transformations before costly model training.

## 2 CONTEXTUAL ANSWERABILITY PREDICTS KNOWLEDGE LEARNABILITY

Motivated by the reliance of existing reformulation methods on human-designed strategies and expensive train-and-evaluate comparisons, we seek a model-dependent signal that could predict reformulation utility before training. Specifically, we first formulate knowledge learnability and introduce Contextual Answerability in Section 2.1. We then investigate their relationship across diverse reformulation strategies in Section 2.2 and analyze the model-internal gradient patterns associated with the observed relationship in Section 2.3.

### 2.1 PRELIMINARIES

**Notations.** Let $\mathcal{D} = \{x_i\}_{i=1}^{N}$ denote a source corpus, where each document $x_i$ is associated with a set of question–answer pairs $\mathcal{Q}_i = \{(q_{ij}, a_{ij})\}_{j=1}^{m_i}$ covering its factual content. A reformulation strategy $r$ transforms $x_i$ into a synthetic document $\tilde{x}_i^r = r(x_i)$ while preserving the underlying facts, yielding a reformulated corpus $\widetilde{\mathcal{D}}^r = \{\tilde{x}_i^r\}_{i=1}^{N}$. To measure how effectively the model learns from strategy $r$, we train a base model $M_\theta$ on $\widetilde{\mathcal{D}}^r$ and obtain $M_{\theta_r}$. Let $\mathcal{E}(M)$ denote the closed-book factual recall performance of model $M$. We define the knowledge learnability of $r$ as the performance gain induced by its reformulated corpus,

$$\mathcal{U}(r; M_\theta) = \mathcal{E}(M_{\theta_r}) - \mathcal{E}(M_\theta), \qquad \theta_r = \mathrm{Train}\big(\theta, \widetilde{\mathcal{D}}^r\big). \tag{1}$$

**Contextual Answerability (CA).** For each reformulated document $\tilde{x}^r$, we use its associated question–answer pairs to examine whether the frozen base model can recover the corresponding facts from context. Let $\mathcal{D}_{\mathrm{QA}}$ denote the distribution of source documents and their fact-level question–answer supervision. We define the CA of strategy $r$ as

$$\mathrm{CA}_\theta(r) = \mathbb{E}_{(x,q,a)\sim\mathcal{D}_{\mathrm{QA}}}\left[\frac{1}{|a|}\log p_\theta\big(a \mid r(x), q\big)\right], \tag{2}$$

where $|a|$ denotes the number of tokens in the answer. Equivalently, $-\mathrm{CA}_\theta(r)$ is the expected length-normalized negative log-likelihood (NLL) of the correct answer. In essence, CA treats the reformulated document as context and measures how easily the model can extract the target fact to answer the corresponding question. A higher score indicates that the model assigns a higher probability to the correct answer and can recover the knowledge more readily from the reformulated context. Intuitively, such a reformulation appears to express knowledge in a form that the model can easily identify and use, and may therefore be expected to provide a more effective training example for learning the corresponding knowledge.

### 2.2 THE CONTEXTUAL ANSWERABILITY–LEARNABILITY INVERSION

To investigate whether CA computed before training can predict the knowledge acquired afterward, we construct multiple corpora through different reformulation strategies, compute CA score for each corpus, and compare with the knowledge learnability after training.

**Experimental setup.** We sample one million English Wikipedia documents from FineWiki (HuggingFaceFW, 2025) as the source data, covering approximately 3.8 million fact-level question–answer pairs extracted from their content. We use Qwen3.6-35B-A3B (Qwen Team, 2026) as an auxiliary model to propose and apply 20 natural-text reformulation strategies to this corpus. Using OLMo-3-7B (Olmo et al., 2025) as the base model, we compute $\mathrm{CA}_\theta(r)$ with the frozen checkpoint and then train the same checkpoint on each reformulated corpus under an equal budget and identical optimization settings. We measure knowledge learnability using SimpleQA (Wei et al., 2024), which evaluates a model's closed-book recall of factual world knowledge. We provide further implementation details in Appendix B.

Surprisingly, our experiments reveal a counter-intuitive phenomenon in which **reformulations with higher CA yield lower knowledge learnability after training**. As shown in Figure 1(a), across the 20 natural-text reformulations, CA has a Pearson correlation of $-0.851$ and a Spearman correlation of $-0.872$ with the resulting SimpleQA performance. In particular, the Spearman correlation indicates that strategies assigned lower CA tend to consistently rank higher after training, demonstrating that the frozen-model score can distinguish reformulations with different knowledge-learning utility. This finding is counter-intuitive because a highly answerable reformulation presents the target fact in a form that the model can immediately recover, yet this accessibility does not translate into stronger knowledge retention when the supporting context is removed. By contrast, reformulations that are more difficult to use in context lead to larger gains in closed-book recall. Our results therefore imply that, how easily a model can *use* knowledge from the current context and how effectively it can *internalize* that knowledge through training.

**Contextual Answerability provides a potential training-free proxy for reformulation utility.** As discussed in Section 1, current reformulation strategies are commonly selected through human heuristics or expensive train-and-evaluate comparisons. Our results suggest that Contextual Answerability can provide a lightweight, model-dependent signal for screening candidate strategies before large-scale training, where a lower score indicates greater expected knowledge learnability. Since the score only requires evaluating the frozen base model, it can be used to compare a broad strategy space without training a separate model for every candidate.

### 2.3 Why Does Contextual Answerability Inversely Predict Learnability?

To further investigate this counter-intuitive relationship, we turn to the model's internal optimization dynamics. Prior studies associate Transformer MLP modules with the storage and recall of factual knowledge (Geva et al., 2021; Dai et al., 2022; Meng et al., 2022), while mechanistic analyses show that attention heads route task-relevant information and entity attributes within the current context (Wang et al., 2023a; Ren et al., 2024; Ferrando et al., 2025). Motivated by these findings, we compare how reformulations with different Contextual Answerability scores distribute their gradients between the MLP and attention modules.

**Gradient-based analysis.** For a reformulation strategy $r$, we define the answer-prediction loss as

$$\mathcal{L}_{\mathrm{ans}}\big(r(x), q, a\big) = -\frac{1}{|a|} \log p_\theta\big(a \mid r(x), q\big). \tag{3}$$

For module family $c \in \{\mathrm{MLP}, \mathrm{Attention}\}$, let $W_c^{(l)}$ denote its parameters at the $l$-th Transformer layer, and let $\mathcal{L} = \{1, \ldots, L\}$ denote the set of layers. Following recent work that analyzes LLM training dynamics through layer-wise gradient matrix norms (Li et al., 2024; Yang et al., 2026), we use the Frobenius norm to define the strategy-level gradient magnitude of module family $c$ as

$$G_c(r) = \mathbb{E}_{\substack{(x,q,a)\sim\mathcal{D}_{\mathrm{QA}} \\ l\sim\mathrm{Unif}(\mathcal{L})}} \left[ \left\| \nabla_{W_c^{(l)}} \mathcal{L}_{\mathrm{ans}}\big(r(x), q, a\big) \right\|_F \right]. \tag{4}$$

We then summarize the relative gradient allocation using the MLP-to-attention gradient ratio $R_{\mathrm{M/A}}(r) = G_{\mathrm{MLP}}(r)/G_{\mathrm{Attention}}(r)$. Here, a smaller $R_{\mathrm{M/A}}$ indicates that the attention modules receive relatively larger gradients, suggesting that the optimization places greater emphasis on context-dependent processing. Conversely, a larger ratio indicates relatively stronger MLP gradients, suggesting greater emphasis on the acquisition and storage of factual knowledge.

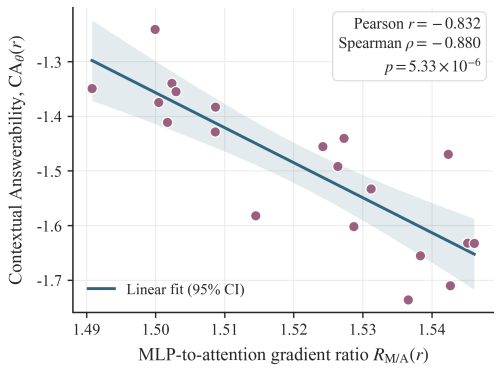


Figure 2: Contextual Answerability and relative gradient allocation.

**Lower CA places relatively greater optimization pressure on MLP modules associated with factual storage.** Figure 2 shows that CA is strongly negatively correlated with $R_{\mathrm{M/A}}$. As CA increases, the MLP-to-attention gradient ratio decreases, indicating that the attention modules receive larger gradients relative to the MLP modules. Conversely, lower-answerability reformulations produce a larger ratio and place relatively more optimization pressure on the MLP modules. The gradient results provide a natural interpretation of the inverse relationship in Section 2.2. When a reformulation already allows the model to recover the target fact from context, the model can rely more heavily on its existing attention-based information flow, and the corresponding learning signal is relatively concentrated on *contextual processing*. When the same fact is less readily accessible, the loss places greater relative pressure on MLP parameters associated with *factual storage*.

# 3 CARVE: CONTEXTUAL ANSWERABILITY-GUIDED REWRITING VIA EVOLUTION

Section 2 shows that CA provides a training-free signal for estimating the knowledge-learning utility of a reformulation strategy. Building on this, we introduce CARVE, a self-evolving framework that uses feedback from a frozen target model to explore effective corpus-level reformulation strategies. We first present the framework overview in Section 3.1, then describe its core modules in Section 3.2.

## 3.1 FRAMEWORK OVERVIEW

The goal of CARVE is to discover effective and reusable reformulation strategies under the guidance of the target model's CA. Since Section 2.2 shows that lower CA is associated with higher knowledge learnability, we use this signal to explore promising rewriting directions without repeatedly training the target model or relying on human experience to manually design and compare strategies.

**Framework design.** We design CARVE as a self-evolving framework that iteratively explores reformulation strategies, with CA providing feedback to guide the evolution process. As illustrated in Figure 3, the framework comprises four main modules: a *reformulation generator*, a *contextual feedback evaluator*, an *evolutionary strategy pool*, and a *strategy analyzer*. Given the source corpus, the reformulation generator proposes a corpus-level strategy and applies it to produce the corresponding rewritten texts, after which the contextual feedback evaluator measures their CA and factual fidelity. Each strategy and its evaluation results, together with representative rewrites and diagnostic cases, are stored in the evolutionary strategy pool. Since the pool is initially empty, CARVE begins with several rounds of open-ended exploration to accumulate diverse observations. Once sufficient experience has been collected, the strategy analyzer reviews the complete pool, identifies rewriting properties associated with favorable outcomes and recurrent failures, and generates guidance for the next round. The reformulation generator then translates this guidance into a new strategy, whose evaluation results are returned to the pool, forming an iterative loop that progressively explores new ones when existing strategies cease to improve. Algorithm 1 summarizes the complete workflow.

## 3.2 CONTEXTUAL ANSWERABILITY-GUIDED STRATEGY SELF-EVOLVING

We now describe the four core modules that implement one round of strategy synthesis. Each module consumes the output of the preceding module, so that a proposed strategy is converted into model-dependent feedback and then into reusable experience for the next round.

**Reformulation generator.** Given natural-language guidance $h_k$ and the strategies in $\mathcal{P}_{k-1}$, the reformulation generator $G_\phi$ proposes a corpus-level strategy $r_k$ and applies it to every source docu-

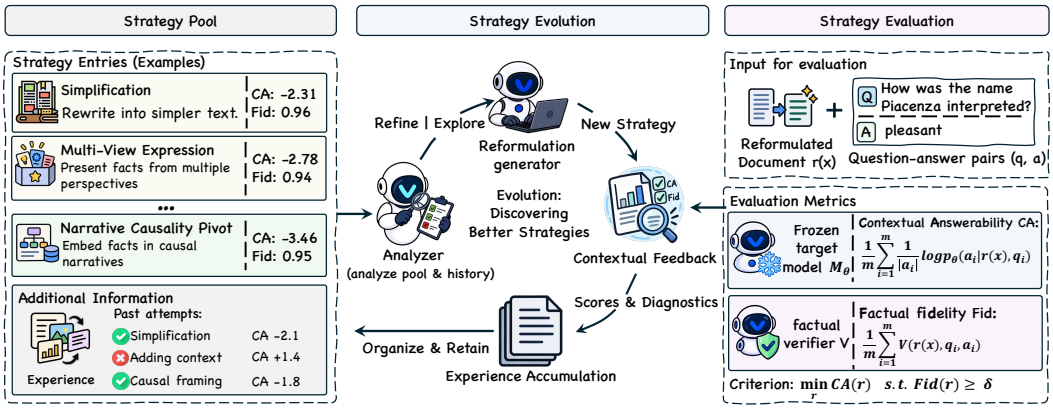


Figure 3: **Overview of CARVE.** A reformulation generator proposes corpus-level strategies and rewrites the source data. A frozen target model and a factuality verifier evaluate the resulting texts, after which the strategy record is added to the evolutionary pool. The strategy analyzer uses the accumulated history to guide subsequent generation.

ment:

$$r_k = G_\phi^{\text{plan}}\big(h_k, \mathcal{P}_{k-1}\big), \qquad \widetilde{\mathcal{D}}^{r_k} = G_\phi^{\text{rewrite}}\big(r_k, \mathcal{D}\big). \tag{5}$$

Here, $h_k$ is empty during initial exploration and otherwise contains the analyzer's recommendation from the preceding search history. The strategy $r_k$ specifies a reusable principle for reorganizing factual knowledge, while $\widetilde{\mathcal{D}}^{r_k}$ contains its document-level realizations. To avoid repeatedly exploring, the generator is also instructed to revise proposals that duplicate a strategy already in $\mathcal{P}_{k-1}$.

**Contextual feedback evaluator.** The rewritten corpus is next passed to the contextual feedback evaluator, which examines both its potential knowledge-learning utility and its faithfulness to the original content. Following Section 2.2, the frozen target model provides the primary search signal

$$c_k = \widehat{\mathrm{CA}}_\theta\big(r_k; \mathcal{D}_{\mathrm{QA}}\big), \tag{6}$$

where a lower $c_k$ potentially indicates a strategy with greater predicted knowledge learnability. Since this signal alone can be trivially reduced by deleting or obscuring the target answer, the evaluator also measures whether the rewrite preserves each answer. Let $V(r(x), q, a) \in \{0, 1\}$ indicate whether $r(x)$ preserves the answer $a$ to question $q$. The factual fidelity of $r_k$ is defined as

$$f_k = \mathrm{Fid}\big(r_k; \mathcal{D}_{\mathrm{QA}}\big) = \mathbb{E}_{(x,q,a)\sim\mathcal{D}_{\mathrm{QA}}}\left[V\big(r_k(x), q, a\big)\right]. \tag{7}$$

When $f_k < \tau$, the strategy remains in the pool as negative evidence but is ineligible for final selection, allowing CA to guide the search without rewarding fact deletion or contradiction.

**Evolutionary strategy pool.** The evolutionary strategy pool $\mathcal{P}_k$ organizes the experience accumulated across search rounds, allowing subsequent decisions to build on previously explored strategies and the analyses. Specifically, after round $k$, CARVE constructs a strategy record

$$e_k = \big(r_k, h_k, c_k, f_k, \mathcal{Z}_k\big), \qquad \mathcal{P}_k = \mathcal{P}_{k-1} \cup \{e_k\}, \tag{8}$$

where $r_k$ is the proposed reformulation strategy, $h_k$ is the analyzer guidance that motivated it, $c_k$ and $f_k$ denote its CA and factual fidelity, respectively, and $\mathcal{Z}_k$ contains representative rewrites and diagnostic cases. The pool is initialized as $\mathcal{P}_0 = \varnothing$ and updated after every round, the pool records what has been explored, why it was proposed, and how it performed. The generator uses this history to avoid redundancy, while the analyzer identifies promising properties and failure patterns.

**Strategy analyzer.** The strategy analyzer $A_\psi$ translates the accumulated experience into guidance for the next round as

$$h_{k+1} = A_\psi\big(\mathcal{P}_k\big). \tag{9}$$

To produce $h_{k+1}$, the analyzer jointly examines the explored strategies, their CA and factual fidelity, the guidance that motivated them, and their representative successes and failures. It then summarizes which rewriting properties are associated with favorable outcomes, assesses whether the current direction should be further developed or replaced, and specifies what the generator should preserve, modify, or avoid in the next round.

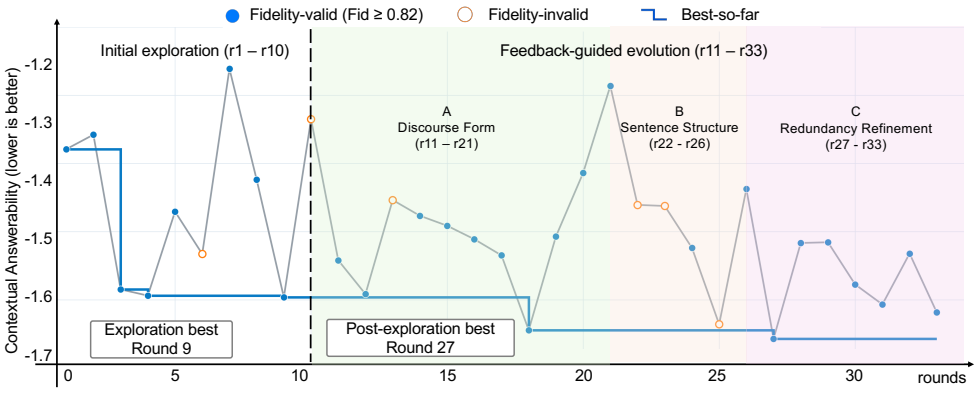


Figure 4: **Evolution of reformulation strategies in CARVE.** CA across the initial exploration stage and three subsequent feedback-guided stages. Filled and hollow points denote fidelity-valid and low-fidelity strategies, respectively, while the blue line tracks the best strategy found so far.

# 4 EXPERIMENTS

In this section, we first examine how the strategy population evolves under the feedback loop of CARVE. We then evaluate the discovered strategy against raw text and randomly sampled reformulations in continuous-training and pretraining-from-scratch settings.

## 4.1 EXPERIMENTAL SETUP

**Models and training settings.** Within CARVE, Qwen3.6-35B-A3B (Qwen Team, 2026) serves as both the reformulation generator and strategy analyzer, while OLMo-3-7B (Olmo et al., 2025) is the target model used to compute Contextual Answerability. We also use OLMo-3-7B for continuous training, initializing from its released checkpoint and training for 50B tokens. For pretraining from scratch, we randomly initialize several OLMo-3 series models and apply the strategies discovered to these models. Detailed optimization hyperparameters are provided in Appendix B.2.

**Training data.** The corpus used for strategy evolution is constructed from the English Wikipedia set of FineWiki (HuggingFaceFW, 2025). We compare the original text, denoted as RAW-WIKI, with the reformulation selected by CARVE and two random baselines, denoted as RANDOM-1 and RANDOM-2 (reformulation details are provided in Appendix B.1). To construct the random baselines, we use the same reformulation LLM to propose strategies under the same search budget without CA feedback or analyzer guidance, and randomly sample two resulting strategies. Each Wikipedia variant is combined with the same official OLMo data sources to construct a 50B-token mixture containing mathematics, code, reasoning traces, scientific documents, and high-quality web text. The total token budget and background-data composition are fixed across variants, with further details provided in Appendix B.3.

**Evaluation benchmarks.** We evaluate all models on ARC (Clark et al., 2018), GPQA (Rein et al., 2023), PubMedQA (Jin et al., 2019), HellaSwag (Zellers et al., 2019), AGIEval (Zhong et al., 2024), PIQA (Bisk et al., 2020), SimpleQA (Wei et al., 2024), and PopQA (Mallen et al., 2023). Further details of the benchmarks and evaluation protocols are provided in Appendix B.5.

## 4.2 STRATEGY EVOLUTION AND DISCOVERED REFORMULATIONS

Before evaluating the discovered strategies through model training, we first examine how the strategy population evolves under feedback and what rewriting properties emerge from the search. This analysis connects the iterative search process to the final strategies evaluated below.

**CARVE progressively moves the search from genre-level reformulation to the distribution of factual expressions.** As shown in Figure 4, the first $K_0 = 10$ rounds perform open-ended explo-

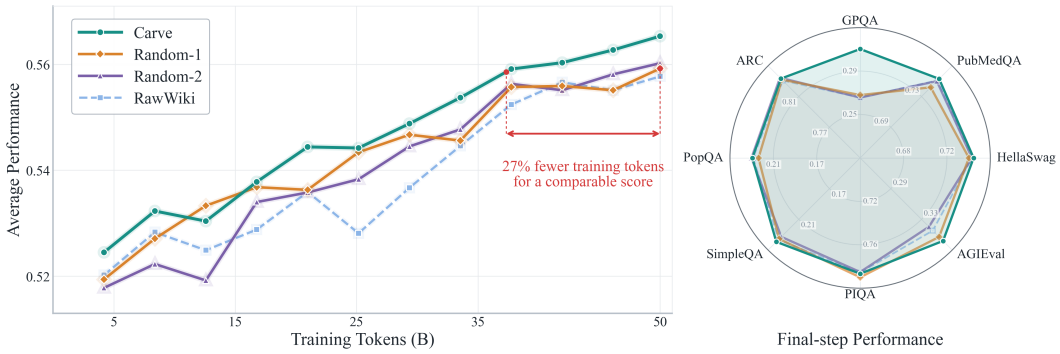


Figure 5: **Continuous-training (Mid-training) performance of CARVE-reformulated data.** Left: average performance throughout training. Right: per-benchmark performance at the final checkpoint.

ration without analyzer guidance, producing diverse strategies with widely varying CA. After that, the subsequent rounds form three stages that successively examine different levels of language:

- **Exploring alternative discourse forms, Rounds 11–21.** The analyzer explores procedural descriptions, relational maps, causal analyses, and investigative narratives, examining whether reorganizing the same facts into different discourse forms makes them less immediately accessible. Notably, Rounds 13–18 exhibit a monotonic decrease in CA as the strategies progress from multi-perspective narration toward relational organization and, eventually, syntactic restructuring.
- **Moving to syntactic and lexical expression, Rounds 22–26.** Motivated by the limited progress of genre-level variations, the analyzer shifts its attention to how individual facts are phrased, exploring register changes, syntactic inversion, and metaphorical expression. CA decreases continuously from Round 22 to Round 25, but several strategies obscure target answers and receive low fidelity scores, redirecting the search toward more faithful transformations.
- **Refining the distribution of factual expressions, Rounds 27–33.** Building on the fidelity failures observed in the previous stage, the search no longer relies on making individual answers more indirect. Instead, it explores how the same fact can be distributed across multiple syntactic realizations and embedded in denser surrounding contexts, including parallel restatement, structured redundancy, and lexical-density transformations. This intervention immediately produces the global minimum in CA, after which several rounds further develop the same general direction.

Taken together, the three stages form a progressive localization process rather than three unrelated collections of prompts. The search first rules out discourse form as the dominant factor, then identifies the potential and risk of syntactic or lexical obfuscation, and finally locates a fidelity-preserving solution at the distributional level.

## 4.3 MAIN RESULTS

### 4.3.1 CONTINUOUS TRAINING

We continue training OLMo 3–7B for 50B tokens on mixtures containing RAW-WIKI, RANDOM-1, RANDOM-2, or the strategy selected by CARVE. All runs use the same background data and optimization configuration. Figure 5 reports their average performance throughout training and their per-benchmark performance at the final checkpoint.

**Training on CARVE-reformulated data leads to more stable and data-efficient performance improvements.** As shown in Figure 5(a), the performance of RAW-WIKI and RANDOM-2 fluctuates noticeably during training, including several intervals in which additional tokens reduce the benchmark average. In contrast, the CARVE reformulated corpora exhibit a substantially steady upward trend, with fewer regressions as training proceeds. It also outperforms the baselines under the same token budget throughout most of the training trajectory. By approximately 37B tokens, CARVE already reaches a score comparable to the final random baselines at 50B tokens, using **27% fewer training tokens**. With the full budget, CARVE achieves the strongest average performance,

| Training Data | Knowledge and QA | | | Commonsense | | | Factuality | | Avg. |
|---|---|---|---|---|---|---|---|---|---|
| | ARC | GPQA | PubMedQA | Hella. | AGIEval | PIQA | SimpleQA | PopQA | |
| *OLMo 3–370M* | | | | | | | | | |
| RAW-WIKI | 40.99 | 24.11 | 49.40 | 36.28 | 25.06 | **65.61** | 12.80 | **9.02** | 32.91 |
| RANDOM-1 | 39.69 | 24.33 | 42.10 | **36.45** | 25.35 | 64.42 | 12.71 | 8.85 | 31.74 |
| RANDOM-2 | 40.23 | 23.88 | 46.40 | 36.01 | 24.83 | 64.20 | 12.64 | 8.56 | 32.09 |
| CARVE | **41.66** | **24.55** | **51.10** | 36.26 | **25.47** | 64.64 | **12.96** | 8.93 | **33.20** |
| *OLMo 3–670M* | | | | | | | | | |
| RAW-WIKI | **42.48** | 24.55 | 52.50 | 38.97 | **26.35** | 65.61 | 12.91 | 9.98 | 34.17 |
| RANDOM-1 | 41.90 | 22.99 | **55.70** | 38.51 | 26.27 | 65.61 | 13.19 | 9.73 | 34.24 |
| RANDOM-2 | 42.36 | 23.21 | 53.20 | 38.18 | 25.74 | 65.61 | 13.45 | 10.50 | 34.03 |
| CARVE | 42.30 | **25.45** | 54.20 | **39.57** | 26.08 | **66.70** | **13.77** | **10.58** | **34.83** |
| *OLMo 3–760M* | | | | | | | | | |
| RAW-WIKI | 43.60 | 24.11 | 46.70 | 41.17 | 25.97 | 67.14 | 13.86 | 10.29 | 34.11 |
| RANDOM-1 | 44.41 | 28.35 | 56.50 | 40.20 | 25.31 | 66.49 | 13.49 | 10.40 | 35.64 |
| RANDOM-2 | 43.45 | 27.23 | 52.00 | 40.86 | 25.81 | **68.01** | 14.12 | **10.44** | 35.24 |
| CARVE | **45.14** | **28.79** | **57.10** | **41.66** | **26.58** | 66.32 | **14.56** | 10.06 | **36.28** |

Table 1: **Results of pretraining from scratch across model scales.** Each block compares RAW-WIKI, RANDOM-1, RANDOM-2, and CARVE. All scores are reported as percentages.

and Figure 5(b) shows that its advantage is distributed across most benchmarks. Taken together, these results show that CARVE-reformulated data enables more stable and token-efficient training for LLMs, validating the effectiveness of the proposed strategy evolving framework.

#### 4.3.2 PRETRAINING FROM SCRATCH

The preceding results evaluate the discovered strategies in the continuous-training setting used during strategy search. We further train randomly initialized models on the same Wikipedia variants to examine whether the resulting data remains effective under a different initialization and training regime. The strategies are applied directly without rerunning CARVE, and all data variants within each model block use identical token budgets and optimization settings.

**The benefits of CARVE-reformulated data generalize across model scales and to pretraining from scratch.** As shown in Table 1, CARVE achieves the highest average performance at all three model scales and consistently outperforms reformulations without feedback. In contrast, RANDOM-1 and RANDOM-2 both underperform RAW-WIKI at 370M and provide gains only at the two larger scales, whereas CARVE improves over the original corpus at every scale, with its advantage becoming more pronounced as model capacity increases. These results show that reformulation alone does not guarantee better training data; instead, feedback-guided evolution identifies a strategy whose benefits transfer more consistently across model scales and initialization regimes. Overall, the effectiveness of CARVE beyond continuous training suggests that it discovers a reusable principle for organizing factual data rather than a strategy narrowly adapted to a single training configuration.

## 5 CONCLUSION

In this work, we introduce Contextual Answerability as a training-free signal and uncover an inverse relationship between contextual accessibility and knowledge learnability, where reformulations that make facts easier to recover from context produce smaller gains in closed-book factual recall after training. Gradient analysis suggests that lower-answerability reformulations place greater relative optimization pressure on MLP modules associated with factual storage. Building on this finding, CARVE uses Contextual Answerability and factual fidelity to evolve corpus-level reformulations. Compared with randomly sampled strategies, the selected reformulation provides more stable and token-efficient continuous training and transfers more reliably to pretraining from scratch across model scales, demonstrating that frozen-model feedback can identify effective transformations beyond arbitrary rewriting.

## AI USE STATEMENT

In this work, we used Qwen3.6-35B-A3B to propose corpus-level reformulation strategies, rewrite source documents, and analyze the outcomes accumulated during strategy evolution, as described in Sections 3 and 4. The remaining required-disclosure tasks are not applicable beyond the uses described above. Additionally, generative AI tools were used for language polishing and grammar checking, and to assist with experimental visualization. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

## REPRODUCIBILITY STATEMENT

We provide the formal definition and computation of Contextual Answerability in Section 2, the complete design of CARVE in Section 3, and its multi-round procedure in Algorithm 1. Appendix B reports the model configurations, optimization hyperparameters, training-data composition, reformulation strategies, and evaluation benchmarks used in our experiments. All comparisons within a model block use matched initialization, data order, token budget, and optimization settings.

## REFERENCES

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.1, knowledge storage and extraction. *arXiv preprint arXiv:2309.14316*, 2023.

Loubna Ben Allal, Anton Lozhkov, and Daniel van Strien. Cosmopedia: How to create large-scale synthetic data for pre-training. Hugging Face Blog, March 2024. URL `https://huggingface.co/blog/cosmopedia`.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 7432–7439, 2020.

Zhengyu Chen, Siqi Wang, Teng Xiao, Yudong Wang, Shiqi Chen, Xunliang Cai, Junxian He, and Jingang Wang. Revisiting scaling laws for language models: The role of data quality and training strategies. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 23881–23899, 2025.

Daixuan Cheng, Yuxian Gu, Shaohan Huang, Junyu Bi, Minlie Huang, and Furu Wei. Instruction pre-training: Language models are supervised multitask learners. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 2529–2550, 2024.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8493–8502, 2022.

Javier Ferrando, Oscar Obeso, Senthooran Rajamanoharan, and Neel Nanda. Do i know this entity? knowledge awareness and hallucinations in language models. In *International Conference on Learning Representations*, volume 2025, pp. 77965–78000, 2025.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 5484–5495, 2021.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, et al. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*, 2023.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

HuggingFaceFW. FineWiki. Hugging Face dataset, 2025. URL `https://huggingface.co/datasets/HuggingFaceFW/finewiki`. Accessed September 2026.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. Pubmedqa: A dataset for biomedical research question answering. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pp. 2567–2577, 2019.

Feiyang Kang, Newsha Ardalani, Michael Kuchnik, Youssef Emad, Mostafa Elhoushi, Shubhabrata Sengupta, Shang-Wen Li, Ramya Raghavendra, Ruoxi Jia, and Carole-Jean Wu. Demystifying synthetic data in llm pre-training: A systematic study of scaling laws, benefits, and pitfalls. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 10750–10769, 2025.

Hanyu Lai, Xiao Liu, Junjie Gao, Jiale Cheng, Zehan Qi, Yifan Xu, Shuntian Yao, Dan Zhang, Jinhua Du, Zhenyu Hou, et al. A survey of post-training scaling in large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2771–2791, 2025.

Ming Li, Yanhong Li, and Tianyi Zhou. What happened in llms layers when trained for fast vs. slow thinking: A gradient perspective. *arXiv preprint arXiv:2410.23743*, 2024.

Xinzhe Li. A review of prominent paradigms for llm-based agents: Tool use, planning (including rag), and feedback learning. In *Proceedings of the 31st international conference on computational linguistics*, pp. 9760–9779, 2025.

Zhenghao Lin, Zhibin Gou, Yeyun Gong, Xiao Liu, Yelong Shen, Ruochen Xu, Chen Lin, Yujiu Yang, Jian Jiao, Nan Duan, et al. Not all tokens are what you need for pretraining. *Advances in Neural Information Processing Systems*, 37:29029–29063, 2024.

Lin Long, Rui Wang, Ruixuan Xiao, Junbo Zhao, Xiao Ding, Gang Chen, and Haobo Wang. On llms-driven synthetic data generation, curation, and evaluation: A survey. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 11065–11082, 2024.

Ziyang Luo, Kaixin Li, Hongzhan Lin, Yuchen Tian, Mohan Kankanhalli, and Jing Ma. Tree-of-evolution: tree-structured instruction evolution for code generation in large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 297–316, 2025.

Pratyush Maini, Skyler Seto, Richard Bai, David Grangier, Yizhe Zhang, and Navdeep Jaitly. Rephrasing the web: A recipe for compute and data-efficient language modeling. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14044–14072, 2024.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 9802–9822, 2023.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372, 2022.

Niklas Muennighoff, Alexander Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin A Raffel. Scaling data-constrained language models. *Advances in Neural Information Processing Systems*, 36:50358–50376, 2023.

Team Olmo, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, et al. Olmo 3. *arXiv preprint arXiv:2512.13961*, 2025.

Qwen Team. Qwen3.6-35B-A3B. Hugging Face model card, 2026. URL https://huggingface.co/Qwen/Qwen3.6-35B-A3B. Accessed September 2026.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*, 2023.

Jie Ren, Qipeng Guo, Hang Yan, Dongrui Liu, Quanshi Zhang, Xipeng Qiu, and Dahua Lin. Identifying semantic induction heads to understand in-context learning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 6916–6932, 2024.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *Advances in neural information processing systems*, 36:68539–68551, 2023.

Dan Su, Kezhi Kong, Ying Lin, Joseph Jennings, Brandon Norick, Markus Kliegl, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. Nemotron-cc: Transforming common crawl into a refined long-horizon pretraining dataset. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2459–2475, 2025.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Pablo Villalobos, Anson Ho, Jaime Sevilla, Tamay Besiroglu, Lennart Heim, and Marius Hobbhahn. Position: Will we run out of data? limits of llm scaling based on human-generated data. In *Forty-first International Conference on Machine Learning*, 2024.

Lean Wang, Lei Li, Damai Dai, Deli Chen, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. Label words are anchors: An information flow perspective for understanding in-context learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 9840–9855, 2023a.

Shaobo Wang, Xiangqi Jin, Ziming Wang, Jize Wang, Jiajun Zhang, Kaixin Li, Zichen Wen, Zhong Li, Conghui He, Xuming Hu, et al. Data whisperer: Efficient data selection for task-specific llm fine-tuning via few-shot in-context learning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 23287–23305, 2025a.

Shaobo Wang, Xuan Ouyang, Tianyi Xu, Yuzheng Hu, Jialin Liu, Guo Chen, Tianyu Zhang, Junhao Zheng, Kexin Yang, Xingzhang Ren, et al. Opus: towards efficient and principled data selection in large language model pre-training in every iteration. *arXiv preprint arXiv:2602.05400*, 2026.

Yixuan Wang, Shiqi Zhou, Chuanzhe Guo, and Qingfu Zhu. Tag-evol: Achieving efficient instruction evolving via tag injection. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 7856–7869, 2025b.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, pp. 13484–13508, 2023b.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. Measuring short-form factuality in large language models. *arXiv preprint arXiv:2411.04368*, 2024.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. Less: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*, 2024.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy S Liang, Quoc V Le, Tengyu Ma, and Adams Wei Yu. Doremi: Optimizing data mixtures speeds up language model pretraining. *Advances in Neural Information Processing Systems*, 36:69798–69818, 2023a.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy S Liang. Data selection for language models via importance resampling. *Advances in Neural Information Processing Systems*, 36: 34201–34227, 2023b.

Hao Xintong, Rui-Jie Zhu, Ge Zhang, Ke Shen, and Chenggang Li. Reformulation for pretraining data augmentation. In *International Conference on Learning Representations*, volume 2026, pp. 92459–92486, 2026.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, Qingwei Lin, and Daxin Jiang. Wizardlm: Empowering large pre-trained language models to follow complex instructions. In *International Conference on Learning Representations*, volume 2024, pp. 30745–30766, 2024.

Junyao Yang, Chen Qian, Kun Wang, Linfeng Zhang, Quanshi Zhang, Yong Liu, and Dongrui Liu. Entropy-gradient inversion: Moving toward internal mechanism of large reasoning models. *arXiv preprint arXiv:2605.17770*, 2026.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022.

Zichun Yu, Spandan Das, and Chenyan Xiong. Mates: Model-aware data selection for efficient pretraining with data influence models. *Advances in Neural Information Processing Systems*, 37: 108735–108759, 2024.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 4791–4800, 2019.

Weihao Zeng, Can Xu, Yingxiu Zhao, Jian-Guang Lou, and Weizhu Chen. Automatic instruction evolving for large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 6998–7018, 2024.

Wanjun Zhong, Ruixiang Cui, Yiduo Guo, Yaobo Liang, Shuai Lu, Yanlin Wang, Amin Saied, Weizhu Chen, and Nan Duan. Agieval: A human-centric benchmark for evaluating foundation models. In *Findings of the association for computational linguistics: NAACL 2024*, pp. 2299–2314, 2024.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36:55006–55021, 2023.

Fan Zhou, Zengzhi Wang, Qian Liu, Junlong Li, and Pengfei Liu. Programming every example: Lifting pre-training data quality like experts at scale. *arXiv preprint arXiv:2409.17115*, 2024.

Jiang Zhou, Yunhao Wang, Xing Wu, Tinghao Yu, and Feng Zhang. Wrap++: Web discovery amplified pretraining. *arXiv preprint arXiv:2604.06829*, 2026.

# A Related Work

**Data Synthesis and Reformulation for LLMs.** Data synthesis and reformulation improve pretraining corpora through knowledge enrichment, quality refinement, and diverse presentation. Textbook-style pretraining (Gunasekar et al., 2023) combines curated code with synthetic explanations, while Cosmopedia (Ben Allal et al., 2024) generates educational documents from topics and web-page seeds. Instruction Pre-Training (Cheng et al., 2024) augments raw corpora with synthetic instruction–response pairs. For quality refinement, ProX (Zhou et al., 2024) executes example-specific editing programs, and Nemotron-CC (Su et al., 2025) rephrases noisy documents and creates diverse variants of high-quality text. WRAP (Maini et al., 2024) rewrites documents into styles such as Wikipedia articles and question–answer pairs, improving data and compute efficiency when combined with the original text. MGA (Xintong et al., 2026) expands stylistic diversity through genre–audience combinations. Beyond individual documents, WRAP++ (Zhou et al., 2026) discovers hyperlink-based relations and synthesizes cross-document question–answer pairs to enrich factual associations. Using synthetic biographies, Allen-Zhu & Li (2023) show that paraphrasing and sentence shuffling during pretraining improve subsequent knowledge extraction; Kang et al. (2025) further demonstrate that synthetic-data benefits depend on data type, mixing ratio, model size, and training budget. Our work instead investigates whether frozen target-model feedback can predict the knowledge-learning utility of reformulations and guide their automatic discovery.

**Model-Based Data Evaluation.** Data evaluation estimates training utility at the domain, document, or token level using distributional and model-derived signals. DoReMi (Xie et al., 2023a) optimizes domain mixtures through distributionally robust proxy training, while DSIR (Xie et al., 2023b) selects documents through density-ratio importance resampling. RHO-1 (Lin et al., 2024) selects high-excess-loss tokens relative to a curated-data reference model. MATES (Yu et al., 2024) learns data influence from local training effects and adapts sample selection as pretraining progresses, while LESS (Xia et al., 2024) selects instruction examples using optimizer-aware gradient similarity to target-task examples. Data Whisperer (Wang et al., 2025a) evaluates demonstrations through a frozen model's in-context prediction performance, with attention-based utility attribution. Related instruction-data methods expand or refine examples through model-generated transformations: Self-Instruct (Wang et al., 2023b) bootstraps a seed task pool, Evol-Instruct (Xu et al., 2024) applies predefined operators to increase complexity and diversity, and Auto Evol-Instruct (Zeng et al., 2024) revises strategies based on evolution failures. Tag-Evol (Wang et al., 2025b) guides instruction evolution with knowledge tags, while Tree-of-Evolution (Luo et al., 2025) searches branching code-instruction trajectories through quality-guided beam search. In contrast, CARVE discovers fact-preserving corpus-level strategies using frozen-model Contextual Answerability, which inversely predicts closed-book recall gains across the document-style reformulations studied.

# B Experimental Implementation Details

This section provides the implementation details for the experiments in Sections 2.2 and 4. Unless otherwise specified, runs within the same model block differ only in the raw or reformulated Wikipedia corpus included in the training mixture.

## B.1 Reformulation Details

**Selected CARVE reformulation.** The strategy selected by CARVE is *Dual-Track Factual Redundancy*, discovered in Round 27 of strategy evolution. It expresses each key fact twice within the same coherent passage: first through a relatively direct statement and then through a syntactically distinct realization that places the same entity–attribute relation in a causal, comparative, or elaborative context. For example, a fact expressed as "$X$ serves as $Y$" may subsequently be restated as "when considering the relevant function, it is $X$ that fulfills the role of $Y$." The two realizations preserve the same factual content but differ in sentence structure and surrounding context, providing multiple linguistic exposures to each entity–fact association. Among the fidelity-valid strategies explored by CARVE, this strategy achieves the lowest Contextual Answerability and is therefore used to construct the CARVE training corpus in the main experiments.

| Hyperparameter | From Scratch | Continuous Training |
|---|---|---|
| Model initialization | Random | OLMo 3–7B checkpoint |
| Global batch size | 4,194,304 tokens | 2,097,152 tokens |
| Training steps | 11,921 | 23,842 |
| Learning-rate schedule | Cosine | Linear |
| Warmup | 1,000 steps | None |
| Peak learning rate | Size-dependent | $2.0712 \times 10^{-4}$ |
| Final/peak learning rate | 0.1 | 0 |
| Tensor parallelism | None | Degree 2 |
| Activation checkpointing | None | Feed-forward modules |

Table 2: **Key hyperparameters for the two training settings.** The from-scratch peak learning rate follows the OLMo ladder rule in Equation 10.

**Random reformulation baselines.** We additionally consider two strategies randomly selected from independent open-ended exploration without CA feedback or analyzer guidance. RANDOM-1, *Conversational Didactic Recitation*, rewrites each document as a natural pedagogical dialogue in which a novice reconstructs the source facts through questions and tentative statements, while an expert provides precise explanations and the novice subsequently confirms or restates the key information. RANDOM-2, *Generative Causality Morphological Transformation*, presents static attributes as outcomes of formative or evolutionary processes and connects them through a continuous narrative using generative expressions such as "evolved," "crystallized," and "anchored." Both strategies preserve the source facts but differ substantially from the parallel factual restatement used by the reformulation selected by CARVE.

### B.2 MODELS AND OPTIMIZATION

**Model initialization.** For pretraining from scratch, we randomly initialize the OLMo 3 models reported in Table 1. For continuous training, we initialize OLMo 3–7B from its released checkpoint at step 1,413,814. All data variants within a model block share the same initialization, data order, and optimization configuration.

**Training configuration.** Both settings use the Dolma 2 tokenizer, a sequence length of 8,192, and a 50B-token training budget. We optimize with SkipStepAdamW using $\beta_1 = 0.9$, $\beta_2 = 0.95$, $\epsilon = 10^{-8}$, weight decay $0.1$ with no decay on embedding parameters, and global gradient clipping at $1.0$. The model and data-loader seeds are both set to 1,337. Training uses hybrid sharded data parallelism with bfloat16 communication and float32 reduction, while model parameters are stored in float32, and `torch.compile` is enabled in both settings. The 7B continuous-training runs additionally use tensor parallelism with degree two and selective activation checkpointing for feed-forward modules. Table 2 summarizes the setting-specific hyperparameters.

For a model with $N_{\text{nonemb}}$ non-embedding parameters, the peak learning rate used for pretraining from scratch is

$$\eta_{\max} = \frac{0.0047}{2} \left( \frac{N_{\text{nonemb}}}{108 \times 10^6} \right)^{-1/3}. \tag{10}$$

The learning rate increases linearly during the 1,000-step warmup and then follows cosine decay to $0.1\eta_{\max}$. Continuous training starts directly from its peak learning rate and linearly decays to zero. It loads the model and optimizer states from the released checkpoint but resets the trainer state so that the new schedule begins at step zero.

### B.3 TRAINING DATA

**Wikipedia variants.** The source documents are drawn from the English Wikipedia subset of FineWiki (HuggingFaceFW, 2025). The same documents are used to construct RAW-WIKI, the reformulation selected by CARVE, and the two randomly sampled variants RANDOM-1 and RANDOM-2. Each variant is tokenized with the same tokenizer and subsampled to exactly 6B tokens using evenly spaced blocks over the full corpus, preserving broad document coverage while matching the number of Wikipedia tokens across conditions.

| Data Category | Approx. Tokens |
|---|---|
| Mathematics | 9.0B |
| Code | 9.4B |
| QA and knowledge | 6.5B |
| Reasoning traces | 3.9B |
| Instruction data | 2.9B |
| Scientific documents | 2.4B |
| High-quality web text | 12.9B |
| Raw or reformulated Wikipedia | 3.0B |
| **Total** | **50.0B** |

Table 3: **Approximate composition of the 50B-token training corpus.** Every condition uses the same background data and changes only the Wikipedia component.

**Training mixture.** For each run, approximately 3B tokens are sampled from one Wikipedia variant and combined with 47B tokens from the official OLMo training sources (Olmo et al., 2025), forming the 50B-token training corpus summarized in Table 3. The background data covers mathematics, code, question answering and knowledge, reasoning traces, instruction data, scientific documents, and high-quality web text. Its composition and sampling weights are fixed across conditions, so the raw or reformulated Wikipedia component is the only data-level difference.

### B.4 Reformulation Strategy Set

Table 4 summarizes twenty non-fallback strategies selected from the explored strategy pool to cover distinct reformulation dimensions. The descriptions are condensed from the strategy records used by the rewriting model; all strategies preserve the source facts while changing how they are organized or expressed.

Table 4: **Twenty representative reformulation strategies explored in our study.**

| ID | Strategy | Description |
|---|---|---|
| 1 | Contextual Embedding via Narrative Displacement | Re-express the same facts in several unrelated narrative genres, placing the subject in varied syntactic and discourse contexts. |
| 2 | Lexical Disambiguation via Etymological Anchoring | Bind each fact to the entity through its canonical name, aliases, expanded forms, and naming origin. |
| 3 | Epistemic Framing via Scholarly Discourse | Embed facts in an academic analysis in which they support arguments, critiques, or historical interpretations. |
| 4 | Thematic Resonance via Analogical Weaving | Explain factual attributes through concrete cross-domain analogies while retaining the entity as the semantic anchor. |
| 5 | Chronological Disambiguation via Temporal Layering | Arrange facts along a linear timeline and attach each attribute to an explicit historical stage or transition. |
| 6 | Conversational Pedagogical Reconstruction | Present facts through a coherent teacher–student dialogue containing questions, explanations, and contextualized restatements. |
| 7 | Perspectival Relativization via Embodied Experience | Cast facts as practical constraints or solutions within first-person experiences involving the target entity. |
| 8 | Polyphonic Historiography | Present the topic through multiple complementary or competing voices that use the same facts to support different interpretations. |
| 9 | Causal-Functional Narrative Weaving | Integrate facts into causal and functional chains in which attributes become causes, effects, or enabling conditions. |
| 10 | Sensory-Anchored Descriptive Realism | Ground factual attributes in visual, auditory, spatial, and tactile descriptions within an immersive passage. |
| 11 | Procedural Integration via Craftsmanship | Express facts as necessary steps, materials, or constraints in an expert's instructional or behind-the-scenes account. |
| 12 | Biographical Immersion via Anachronistic Memoir | Recast facts as first-person memories or reflections narrated by a personified historical subject or entity. |
| 13 | Speculative Counterfactual Recontextualization | Contrast the preserved facts with an explicitly marked alternative scenario and explain the consequences of changing them. |
| 14 | Archival Epistolary Reconstruction | Distribute facts across letters, diaries, telegrams, memoranda, or other correspondence in which they serve pragmatic purposes. |

| ID | Strategy | Description |
|---|---|---|
| 15 | Symbiotic Contextualization via Integrated Prose | Build a dense, coherent paragraph that restates each fact through multiple syntactic forms and overlapping retrieval paths. |
| 16 | Chronoscape Layering | Reveal facts through non-linear temporal vignettes in which recurring motifs connect different moments in the subject's history. |
| 17 | Linguistic Register Shifting | Preserve the factual core while moving between colloquial, technical, bureaucratic, and literary registers within one narrative. |
| 18 | Geospatial Narrative Anchoring | Tie factual attributes to locations, spatial relations, and interactions with a specific physical environment. |
| 19 | Systemic Interconnectivity via Ecological Webbing | Describe the subject as a node in a causal, spatial, and temporal network of surrounding entities and processes. |
| 20 | Diagnostic Narrative Reconstruction | Introduce facts as evidence and intermediate deductions in a clinical, technical, or historical investigation. |

### B.5 Evaluation

We evaluate the models on several benchmarks. ARC (Clark et al., 2018) measures grade-school science knowledge and reasoning through multiple-choice questions. GPQA (Rein et al., 2023) contains expert-written graduate-level questions in biology, physics, and chemistry. PubMedQA (Jin et al., 2019) evaluates biomedical question answering based on research abstracts. HellaSwag (Zellers et al., 2019) tests commonsense inference by requiring models to select a plausible continuation of an everyday situation. AGIEval (Zhong et al., 2024) uses standardized and professional examinations to assess general reasoning ability. PIQA (Bisk et al., 2020) evaluates physical commonsense reasoning about everyday actions and objects. SimpleQA (Wei et al., 2024) measures short-form factual recall through fact-seeking questions. PopQA (Mallen et al., 2023) evaluates factual question answering over entities with different levels of popularity. We use the same evaluation and decoding configurations for every data variant within a model block, and evaluate intermediate continuous-training checkpoints at matched token counts.

## C Additional Experimental Results

### C.1 Correlation under an Equal-Item Setting

The main analysis in Section 2.2 controls the number of training tokens across reformulation strategies. As a complementary control, we train on complete rewritten corpora generated from the same source documents, preserving document and fact coverage while allowing corpus length to vary with each strategy. As shown in Figure 6, Contextual Answerability remains negatively correlated with post-training SimpleQA performance in this equal-item setting.

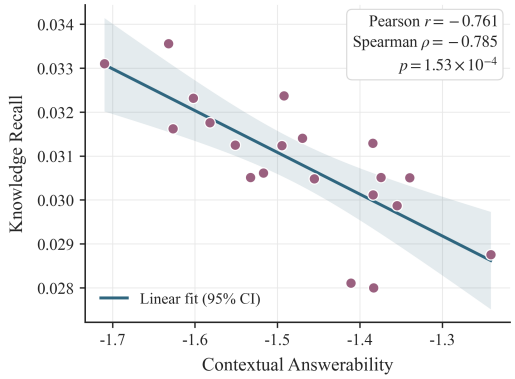


Figure 6: The relationship between CA and knowledge recall performance under full coverage.

### C.2 What Kinds of Reformulations Lower Contextual Answerability?

**Effective strategies modify how factual associations are encoded, whereas changing the surrounding genre alone is usually insufficient.** Grouping the 33 strategies explored during CARVE's evolution into broad families reveals that dialogue, screenplay, archival, and pedagogical formats generally remain easy to answer because they alter the document wrapper while retaining explicit entity–attribute statements. Narrative and temporal organization provide a stronger intermediate direction by introducing longer dependencies, whereas lower Contextual Answerability emerges from transformations that directly modify factual expression, including syntactic inversion, lexical-density manipulation, metaphorical mapping, and redundant restatement. Factual fidelity is nevertheless essential: aggressive syntactic and metaphorical strategies can obtain low scores by obscuring the answer, while redundancy-based strategies preserve the source facts and produce the strongest valid candidates. A useful reformulation must therefore reduce the immediate accessibility of factual associations without weakening them as training signals.

## D Additional Methodological Details and Discussion

### D.1 Complete Multi-Round Evolution Workflow

Algorithm 1 summarizes how CARVE coordinates its components over multiple rounds. The process begins with an initial exploration stage that establishes the evidence needed for strategy analysis, followed by feedback-guided evolution based on the accumulated search history.

**Initial exploration provides a diverse basis for strategy analysis.** At the beginning of the search, the evolutionary strategy pool is empty, leaving the strategy analyzer without sufficient observations to determine which rewriting properties are promising. We therefore reserve the first $K_0$ rounds for open-ended exploration. During these rounds, the reformulation generator proposes diverse and non-redundant strategies without analyzer guidance, while the contextual feedback evaluator assesses each strategy using the same Contextual Answerability and factual-fidelity criteria as in subsequent rounds. The resulting strategies, scores, rewritten examples, and diagnostic cases are added to the pool, providing the analyzer with initial experiences.

**Historical analysis guides subsequent strategy evolution.** Starting from round $K_0+1$, each round begins with the strategy analyzer reviewing the current evolutionary strategy pool and producing guidance for the next candidate. The reformulation generator translates this guidance into a new strategy, whose rewritten corpus is evaluated and returned to the pool as a new strategy record. As the loop continues, the analyzer revises its guidance according to the newly observed outcomes, further developing promising rewriting properties while correcting or replacing ineffective and low-fidelity directions. After $K$ rounds, CARVE retains the strategies that satisfy the fidelity threshold and ranks them in ascending order of Contextual Answerability. The top-ranked strategies are subsequently used to construct the final training corpora and evaluated through actual model training.

**Algorithm 1** CARVE: Contextual Answerability-Guided Rewriting via Evolution

**Require:** Source corpus $\mathcal{D}$ and fact-level QA pairs $\mathcal{D}_{\mathrm{QA}}$
**Require:** Frozen target model $M_\theta$, factuality verifier $V$, reformulation generator $G_\phi$, and strategy analyzer $A_\psi$
**Require:** Total rounds $K$, initial exploration rounds $K_0$, and fidelity threshold $\tau$
**Ensure:** Fidelity-valid reformulation strategies ranked by Contextual Answerability

Initialize the strategy pool $\mathcal{P}_0 \leftarrow \varnothing$
**for** $k = 1, \ldots, K$ **do**
  **if** $k \leq K_0$ **then**
    $h_k \leftarrow \varnothing$ ▷ Open-ended exploration
  **else**
    $h_k \leftarrow A_\psi(\mathcal{P}_{k-1})$ ▷ Analyze accumulated outcomes
  **end if**
  $r_k \leftarrow G_\phi^{\mathrm{plan}}(h_k, \mathcal{P}_{k-1})$
  **while** $r_k$ duplicates a strategy in $\mathcal{P}_{k-1}$ **do**
    Revise $r_k$ using the existing strategy descriptions
  **end while**
  $\widetilde{\mathcal{D}}^{r_k} \leftarrow G_\phi^{\mathrm{rewrite}}(r_k, \mathcal{D})$
  $c_k \leftarrow \widehat{\mathrm{CA}}_\theta(r_k; \mathcal{D}_{\mathrm{QA}})$
  $f_k \leftarrow \mathrm{Fid}(r_k; \mathcal{D}_{\mathrm{QA}})$
  Collect representative rewrites and diagnostic cases in $\mathcal{Z}_k$
  $e_k \leftarrow (r_k, h_k, c_k, f_k, \mathcal{Z}_k)$
  $\mathcal{P}_k \leftarrow \mathcal{P}_{k-1} \cup \{e_k\}$
  **if** $f_k < \tau$ **then**
    Mark $e_k$ as ineligible for final selection ▷ Retain as negative evidence
  **end if**
**end for**
$\mathcal{R}_{\mathrm{valid}} \leftarrow \{r_k \mid k \in \{1, \ldots, K\},\ f_k \geq \tau\}$
**return** $\mathcal{R}_{\mathrm{valid}}$ ranked by ascending $c_k$