

Which Signal Did the Synthesis Import? Information, Re-presentation and Selection in Synthetic Pretraining, Why 'the Verifier Is Doing the Work' Holds Only Where Correctness Is the Target, and the Controls That Would Assign the Credit

Pranay Mahendrakar

SONYTECH · ORCID 0009-0003-7224-029X

pranaymahendrakar@sonytech.in · mahendrakarpranay@gmail.com · +91 6361723454

Abstract. The expectation that language models will keep improving after the supply of human text runs out rests on a claim about synthetic data: that a model, or a family of models, can produce training text worth more than the text it was trained on. A competing reading holds that no synthetic-data success is self-generated at all, because every pipeline that works draws on something outside the generating model, most visibly a verifier. In this analysis the author reviews and synthesises published results on synthetic data for pretraining, continued pretraining and mid-training, and argues that both readings are posed at the wrong grain. Read as a conservation law, the imported-signal claim is true and uninformative, since every generator's weights were themselves fitted to human text. Read as a claim about what explains a measured gain, it splits. The paper separates three contributions a synthetic corpus can make - new information, re-presentation of information already in an organic seed, and selection within the generator's own distribution - and four places the signal can come from: the seed text, the generator's prior, a selection signal, and the designer's prompts or grammars. Against that ledger, the best-controlled pretraining gains are mostly re-presentation: they beat repetition of the same corpus, they persist when the generator is small, and summarisation of web pages nearly matches teacher-written textbooks. There is no verifier in those pipelines. The verifier does the work where the target is correctness, in geometry, code and arithmetic. The most self-contained bootstrapping results use held-out human text as the scorer. No located pretraining result shows a gain from self-generation with none of the four sources present, and the pipelines that come closest score below the real-data baseline on factuality. The paper consolidates the reported values, states a credit-assignment protocol, and names the five controls that would settle the open part.

Keywords: synthetic data; language model pretraining; model collapse; data-constrained scaling; rephrasing; verification; self-improvement; credit assignment

SONYTECH ORCID: 0009-0003-7224-029X

1. Introduction

Forecasts of the stock of public human text put the crossing point, at which training sets equal the whole stock, somewhere between 2026 and 2032 if current trends hold (Villalobos et al., 2022). The same forecast names synthetic data as one of the ways progress might continue past that point. Since then the idea has moved from a forecast into production. A 14-billion-parameter model reports that synthetic text makes up most of its training data (Abdin et al., 2024); open pretraining corpora now ship rephrased subsets by default (Su et al., 2024; Niklaus et al., 2026); and a TMLR survey of self-improvement organises the field around a closed loop in which the model generates its own data, selects what to learn from, updates itself and refines its outputs, with an evaluation layer that is also meant to be automated (Yang et al., 2026). The motivation that survey gives is explicit: as models approach human level in some domains, it argues, human feedback may stop being informative enough to drive further improvement.

The opposing reading is easy to state and hard to dismiss. A model cannot learn from its own output

anything its output does not contain, and its output contains nothing its weights do not encode. Huang et al. (2024) put it in one line: self-improvement, they write, cannot "create information that is not already in the model". If that is right, then every reported synthetic-data success owes its gain to something outside the generating model - a verifier, an execution environment, a rubric, a stronger teacher, or human text that is merely restated - and where nothing outside is present the loop either concentrates the distribution it already has or degrades (Shumailov et al., 2024). Recent work has made that argument at two levels. A survey of 1,250 papers on recursive self-improvement orders feedback signals from formal verifiers down to intrinsic self-assessment, and reports that the strength of demonstrated self-improvement follows that ordering (Chen et al., 2026). A theoretical analysis of verifier-guided retraining shows that a verifier lets a loop escape collapse, and then shows that the loop converges to the verifier's own knowledge rather than to the truth (Yi et al., 2025).

Neither of those two papers is about pretraining corpora. Chen et al. (2026) is scoped to self-improvement loops in deployment, training and research automation; Yi et al. (2025) works in linear regression, a small autoencoder

and a 135-million-parameter summarisation fine-tune. The pretraining literature, which is where the "data wall" argument actually lives, has been read through them by analogy. This paper asks whether the analogy holds. The strong form of the imported-signal reading is the claim that "the verifier is doing the work". The evidence assembled here does not support it in that form.

The concession comes first, because it is real. The imported-signal reading is correct as a conservation statement. No result located for this paper shows a pretraining gain from synthetic text produced with no organic seed, no externally trained generator, no selection signal and no human-designed structure. Where the organic data is replaced by generated data rather than kept, the literature reports degradation (Section 7). But a conservation statement cannot carry the weight the reading puts on it, because it is satisfied trivially: every generator in every pipeline was pretrained on human text, so every synthetic token is "imported" in the sense that its distribution was learned from people. A reading that no result could contradict is not yet a reading of the results.

The question that can be answered is narrower and more useful. Holding the organic corpus fixed, what accounts for the measured gain of a synthetic-data pipeline over the best control that uses only that corpus? Asked that way, the literature divides. The largest and best-controlled pretraining gains come from re-representation: the same content, restated in forms a model learns more efficiently from. These pipelines contain no verifier. Their gains do not grow with the generator's knowledge - several systematic studies find that rephrasers beyond a few billion parameters add little (Kang et al., 2025; DatologyAI et al., 2025; Niklaus et al., 2026). Yet the gains are not reducible to the seed, because they beat repetition of the same seed by wide margins (Yu and Xiong, 2026; Yang et al., 2025b). Verification does the work in a different, narrower class: problems with a checkable answer, where a symbolic engine, an executor or a ground-truth label sorts generated candidates (Trinh et al., 2024; Zhao et al., 2025; Singh et al., 2023). And the results closest to genuine bootstrapping at pretraining scale use the organic corpus itself as the scorer: generated text is kept or weighted by how well it predicts, or relates, held-out human documents (Zelikman et al., 2024; Ruan et al., 2025; Yang et al., 2025b; Dong et al., 2025).

The paper makes four contributions. First, it states a ledger: three things synthetic data can contribute (information, re-representation, selection) and four places the signal can come from (seed, generator prior, selection signal, design), and shows that the two readings in the title are claims about different cells of it (Section 3). Second, it places the located pretraining and post-training results in that ledger, consolidating the measured values in Table 1 and the clearest single-source case in Figure 1 (Sections 4 to 7). Third, it states a

credit-assignment protocol, Algorithm 1, that says which control isolates which cell, and audits which of the protocol's controls the located studies run (Section 9). Fourth, it names five measurements that would settle the part that is still open (Section 10).

Two limits of scope are stated here so that they are not discovered later. The paper is about text pretraining and the stages adjacent to it (continued pretraining, mid-training, and pre-pretraining on procedural data). It draws on post-training self-improvement only where those results bear on the mechanism, and it does not adjudicate the reinforcement-learning debate over whether verifiable-reward training expands what a model can solve; that debate is treated as evidence about selection, not reopened. And the paper's units of evidence are reported gains on benchmarks chosen by each source. Where a gain might reflect proximity to the evaluation suite rather than general capability, the paper says so, but it does not resolve that confound, which needs its own treatment.

2. Two readings, and why each needs a denominator

2.1 The self-generation reading

The strong form of the self-generation reading holds that a model can produce, from its own distribution, training data that raises its own capability, and that iterating this is a route past the data wall. The weak form holds only that synthetic text, from whatever source, can substitute for scarce human text at a useful exchange rate. The two are often run together. The phrase "bootstrapping" appears in titles describing both a model that learns a synthesiser from its own pretraining corpus (Yang et al., 2025b) and a model that improves its own latent-thought annotations by expectation-maximisation (Ruan et al., 2025). Absolute Zero describes a reasoning system trained "with zero data" (Zhao et al., 2025). The TMLR overview of self-improvement frames the destination as fully autonomous improvement (Yang et al., 2026).

What the strong form needs is a result in which the gain cannot be traced to anything the generating model did not already have. What the weak form needs is only an exchange rate: how many organic tokens a synthetic token is worth, under what mixture, at what scale. Most published pretraining results are evidence for the weak form, which is not in dispute. The strong form is what the imported-signal reading denies.

2.2 The imported-signal reading

The imported-signal reading holds that the information in a synthetic corpus is bounded by the information in what produced it, and that any gain therefore comes from the producer, not from the act of synthesis. Its theoretical anchors are well defined. Huang et al. (2024) formalise self-improvement as sharpening: the model uses its own verification ability to shift probability mass

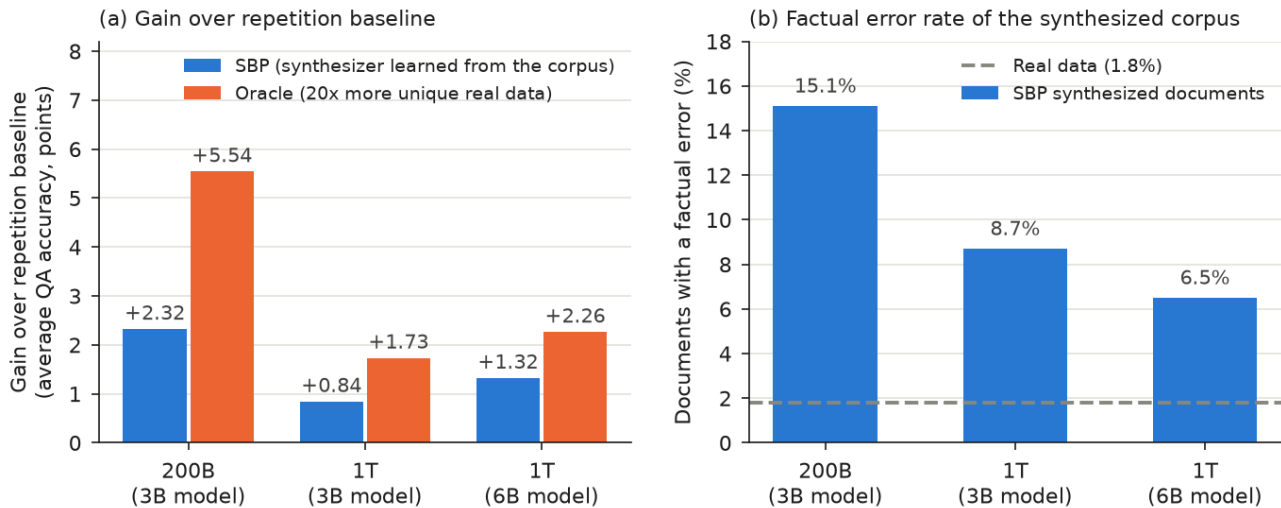


Figure 1. Synthetic Bootstrapped Pretraining as reported by Yang et al. (2025b), the most self-contained corpus-bootstrapping result located. (a) Average question-answering accuracy gain over a compute-matched repetition baseline, for the SBP corpus and for an oracle trained on 20 times more unique real data, at three scales; at the 1T scale the source's oracle is a proxy, 482 billion unique tokens repeated twice. (b) Share of synthesised documents containing a verifiable factual error at the same three scales, with the real-data rate as a dashed line. Values from Yang et al. (2025b), Table 1 (panel a) and Table 2 (panel b).

onto sequences it already rates highly, amortising inference-time search into the weights. Under that account, self-improvement can raise the quality of what the model reliably produces, but cannot extend what it can produce at all. Song et al. (2025) measure the corresponding quantity, the gap between a model's ability to verify and its ability to generate, and report that a variant of that gap grows monotonically with pretraining compute - which is to say that the self-improvement available is itself a product of pretraining on human data. Gan and Liu (2024) analyse synthetic post-training data from an information-theoretic standpoint and tie the generalisation benefit to the information the generator contributes.

On the empirical side, the reading draws on two kinds of evidence. One is the collapse literature, which shows that replacing real data with generated data across generations erodes the tails of the original distribution (Shumailov et al., 2023; Shumailov et al., 2024; Alemohammad et al., 2023). The other is the verification literature, which shows that a verifier, even an imperfect one, can rescue a loop that would otherwise collapse (Feng et al., 2024; Yi et al., 2025; Gillman et al., 2024).

2.3 Why both need a denominator

The two readings are less opposed than they look, because they usually answer different questions. The self-generation reading is typically defended with a comparison against natural data at matched compute: synthetic data is shown to reach a given loss sooner, or a given accuracy with fewer organic tokens. The imported-signal reading is typically defended with a comparison against the information content of the producer: whatever the synthetic data contains, the generator, the seed or the verifier contained first. Both comparisons can

hold at once. A pipeline can make a fixed corpus five times more useful without adding a single proposition to it.

What neither reading supplies by itself is the denominator that would make a gain attributable. A gain of three points over natural web text does not say whether the points came from facts the generator knew and the web did not, from the same facts in a more learnable form, from discarding the generator's bad outputs using an outside check, or from a prompt template that happens to match the evaluation format. Each of those has a different implication for the data wall. Facts the generator knew are bounded by what the generator was trained on. Re-representation is bounded by what the seed contains. Selection is bounded by the quality of the selection signal. Design is bounded by the designer. The rest of this paper is an attempt to assign the reported gains to those sources using the controls the studies actually ran.

3. A ledger for synthetic pretraining data

3.1 Three contributions

The author distinguishes three things a synthetic corpus can contribute to a model trained on it, defined relative to the organic corpus the same model would otherwise see.

Information is content - propositions, solutions, programs, proofs - that is correct, useful to the target model, and not present in the organic corpus available to it. A synthetic textbook written by a stronger model on a topic the organic corpus covers thinly contributes information in this sense, whatever else it does. So does a verified proof that no document in the corpus contains.

Re-presentation is the same content as some part of the organic corpus, restated in a form the target model learns from more efficiently: paraphrased, reformatted as questions and answers, decomposed into steps, made explicit where the source left inference implicit, or diversified so that one fact appears in many surface forms. Re-presentation adds no propositions. Its value is in learnability.

Selection is a reweighting of the generator's own distribution: some generated samples are kept, upweighted or reinforced and others discarded, according to a signal. Selection adds no content the generator could not have produced, but it changes which of its productions the target model sees. Sharpening, in the sense of Huang et al. (2024), is selection by the model's own judgement.

The three are not exclusive within a pipeline, and the boundaries blur at the edges. A paraphrase can introduce an error, which is negative information; a question-and-answer reformatting can make an inference explicit that the source left to the reader, which is arguably a little information. But the distinction is sharp at the level that matters for the data wall. Re-presentation can multiply the usefulness of a fixed corpus by a bounded factor and cannot add to what it says. Information can add to what it says but is bounded by its producer. Selection can raise reliability within what the generator already covers and is bounded by the signal that does the selecting.

3.2 Four sources of signal

Separately from what a corpus contributes, there is the question of where the signal that shaped it came from. The author distinguishes four sources.

S1, the organic seed. Human-written documents that the generator conditions on: the page being rephrased, the book being expanded into an entity graph, the document whose neighbour is being synthesised.

S2, the generator prior. Whatever the generating model brings from its own training: world knowledge, fluency, instruction-following competence, and the alignment it received. When the generator is a stronger instruction-tuned model, S2 includes everything its developers' human annotators supplied.

S3, the selection signal. Whatever decides which generated samples survive or how they are weighted: a symbolic deduction engine, a code executor, a ground-truth answer, a reward model, a language-model judge, a majority vote over samples, a quality classifier, or the likelihood of held-out human text.

S4, design. Human choices encoded in the pipeline: the prompts and output formats, the topic lists, the curriculum, the formal grammar of procedurally generated data, the mixture ratio of synthetic to organic tokens.

The imported-signal reading, in its conservation form, says that every successful pipeline has at least one of S1 to S4 with content from outside the generating model, and that S2 itself is learned from people. That is true of every result reviewed here. Its strong form, "the verifier is doing the work", says something more specific: that S3, and in particular a verifier, accounts for the gain. That is a claim about a particular cell, and it can be checked.

3.3 Controls that isolate the cells

A cell is attributable only if the study ran the control that holds everything else fixed. Six controls recur in the literature, each isolating a different source.

The **repetition baseline** trains on the organic seed alone, repeated to match the token or compute budget of the synthetic run. A synthetic corpus that beats it has contributed something beyond S1 as presented. Repetition is a strong baseline: up to about four epochs, repeated data gives nearly the same loss as unique data at fixed compute (Muennighoff et al., 2023).

The **unique-data oracle** trains on more organic data of the same kind instead of the synthetic corpus. It bounds what the synthetic corpus could have contributed had it been as good as fresh human text, and lets a gain be expressed as a share of that bound.

The **generator-scale ablation** varies the size or capability of the generator while holding seed, prompt and selection fixed. If gains rise with generator capability, S2 is doing work; if they saturate early, the generator is contributing competence at the transformation rather than knowledge.

The **seed ablation** varies the quality or presence of S1: rephrasing high-quality versus low-quality pages, or generating with and without conditioning documents.

The **signal ablation** removes or randomises S3 while holding the rest fixed. If gains survive a randomised signal, the signal was not doing the work.

The **teacher ceiling** compares the trained model with the generator. A trained model that exceeds its own teacher on a measured task cannot be explained by distillation of that teacher alone.

Sections 4 to 7 place the results by which of these controls they ran. Section 9 audits the coverage.

4. Re-presentation: the bulk of the pretraining gains

4.1 The rephrasing line

The modern rephrasing line begins with Web Rephrase Augmented Pretraining (WRAP; Maini et al., 2024), which prompts an off-the-shelf instruction-tuned model, a frozen Mistral-7B in the main experiments, to restate noisy web documents in styles such as encyclopaedic prose or question-and-answer form, and trains on the

restated text jointly with the original. On C4 the authors report pretraining roughly three times faster, more than 10 percent lower perplexity on average across subsets of the Pile, and an improvement of more than two points in zero-shot question-answering accuracy over 13 tasks. The paper is explicit that the rephraser is not used as a store of knowledge; it frames the method as preserving the information of the source while changing its style.

Two of WRAP's ablations matter for the ledger. Its comparison across rephrasing models found that pretraining on paraphrases from small models (Qwen-1.8B and Mistral-7B) reached lower perplexity than paraphrases from a 13-billion-parameter model, while a fine-tuned T5-base rephraser was clearly worse than all of them. That is the first published sign of a pattern that recurs below: the generator must clear a competence floor, and above it, more generator does not buy more. Its comparison against classical augmentations found that WRAP beat synonym replacement and random deletion, which rules out the deflationary account that any perturbation of the seed would do.

The WRAP authors attribute their gains to two properties: style diversity that resembles the style of downstream evaluations, and higher quality than raw web text. The first of these is the benchmark-proximity confound in its plainest form. A rephrased corpus written partly as question-answer pairs is closer in format to question-answering benchmarks than the web text it came from. That is a gain, but it is a gain whose generality is not established by the benchmarks that measured it.

4.2 Systematic studies: the generator is not the source

The largest controlled study of synthetic pretraining data located for this paper trained more than a thousand models under a common protocol (Kang et al., 2025). Its findings bear on the ledger directly. Pretraining on rephrased synthetic text by itself was not faster than pretraining on natural web text. Mixing roughly one third rephrased text with two thirds natural text reached the same validation loss five to ten times faster at larger data budgets. Pretraining on synthetic textbooks alone produced notably higher loss on many downstream domains, especially at small budgets. The best mixture ratios depended on model size and budget and converged near 30 percent for rephrased text. And generators larger than about eight billion parameters did not necessarily yield better pretraining data.

Each of those findings constrains the readings. That rephrased text alone is no faster than web text, while a mixture is much faster, says the synthetic text is valuable as a complement to its own source, not as a replacement for it - the signature of re-presentation rather than information. That textbook-only training degrades many domains says that teacher-written content, unanchored to a broad organic distribution, behaves like the narrowing the collapse literature predicts; Kang et al. say

as much, describing the textbook-style results, including mixtures, as showing patterns predicted by model collapse even in a single round of training. And that larger generators do not reliably help says the generator's knowledge is not the active ingredient.

BeyondWeb (DatologyAI et al., 2025) reaches the same conclusion by a different route. Its first research question asked whether generator-driven synthetic data, in which a model writes educational content from topic prompts with no conditioning document, can be approximated by simply summarising web pages. A summarisation prompt nearly matched the generator-driven Cosmopedia corpus; the paper's own heading for the result is that synthetic data "Is Not Just Knowledge Distillation". Its study of rephraser families found consistent improvements across four generators from four families and reported that a model's benchmark performance does not correlate positively with its value as a rephraser. Its size study found gains from one to three billion parameters, then saturation by eight. Its strongest configuration reported up to 7.7 times faster training than open web data and up to 2.7 times faster than the synthetic subset of Nemotron-CC, and a 3-billion-parameter model trained on its data outperforming an 8-billion-parameter model trained on Cosmopedia at the same token budget.

The 2026 FinePhrase study pushed the size question further, generating more than a trillion tokens across rephrasing strategies, generators and source data (Niklaus et al., 2026). It reports that structured output formats such as tables, mathematical problems, FAQs and tutorials outperformed both curated web baselines and earlier synthetic methods, that generator size beyond one billion parameters provided no further benefit, and that the choice of original data mixed in alongside the synthetic text materially changed results.

Read together, these three studies are the strongest evidence in this paper. Across independent groups, protocols and scales, the knowledge in the generator is not what makes rephrased pretraining data valuable. What matters is the seed (DatologyAI et al., 2025), though even documents that quality filters discard can be rewritten into useful training text (Nguyen et al., 2025); the organic data mixed in alongside the synthetic text (Niklaus et al., 2026); the prompt and output format (Niklaus et al., 2026; Maini et al., 2024); diversity (DatologyAI et al., 2025); and a generator competent enough to transform text without wrecking it. There is no verifier anywhere in these pipelines. If the strong imported-signal claim, "the verifier is doing the work", were literally true, these results could not exist.

4.3 Re-presentation is not nothing: beating repetition

The imported-signal reading has a fallback here: perhaps re-presentation merely recovers what a model would have learned from the seed with enough repetition, so that the synthetic text is a convenience, not a contribution. The repetition baseline answers that, and it

answers against the fallback.

SynPro (Yu and Xiong, 2026) is the cleanest test, because it is designed to add no outside content. It applies two operations to a limited organic corpus - rephrasing and reformatting - with rewards that explicitly ground outputs in the source document, and works in a deliberately data-bound regime: models of 400 million, 1.1 billion and 2 billion parameters with unique organic data equal to 10 percent of their compute-optimal token counts. The authors report that SynPro yields 3.4 to 5.2 times the effective tokens of repetition, and at the two larger scales exceeds a non-data-bound reference that trains on the equivalent amount of unique organic data. If that holds up, it means re-presentation of a fixed corpus can be worth more than the same number of fresh organic tokens, which is exactly what the "merely recovers repetition" fallback denies.

But SynPro's design is an object lesson in how hard it is to remove S3. Its generators are initialised from an instruction-tuned 1-billion-parameter model. Its quality reward comes from a judge; its faithfulness reward for rephrasing combines semantic similarity with a judge's verdict on structure; and its influence reward scores each synthetic sample by how much the current model's loss on it falls after a training step on a held-out reference set - and that reference set is FLAN, a collection of human-annotated tasks. So the content is the seed's, but the choice of which restatements to produce is steered by human-built signals at three points. The study shows that re-presentation beats repetition, and its own ablation (Section 9.2) shows that about half of the recovered gain depends on those imported signals.

The knowledge-acquisition literature explains why re-presentation could beat repetition at all. In a controlled study on synthetic biographies, Allen-Zhu and Li (2023) found that whether a model could extract a fact by question answering depended on how diversely the fact was presented during pretraining. Without augmentation such as paraphrasing, sentence shuffling or translation, facts could be memorised and still be unextractable, with question-answering accuracy at zero regardless of later instruction tuning. Repetition presents a fact in one form many times; re-presentation presents it in many forms. On that account, the gain from rephrasing is a gain in how knowledge is stored, and it is available only for knowledge the seed already contains. Allen-Zhu and Li recommend rewriting pretraining data with small auxiliary models for exactly this reason - a recommendation that the generator-size results in Section 4.2 later bore out.

4.4 Re-presentation at small scale: knowledge injection

The same mechanism operates when a pretrained model must learn a small specialised corpus. Synthetic continued pretraining with EntiGraph (Yang et al., 2024) extracts entities from 265 books totalling 1.3 million

tokens, prompts a strong model to write text about relations among sampled entities, and continues pretraining an 8-billion-parameter model on the result. With 455 million synthetic tokens, the authors report accuracy on questions about the books that scales log-linearly in the synthetic token count and, at the endpoint, recovers 80 percent of the gain that having the source documents available at inference time provides. Rephrasing alone did worse.

The teacher here was gpt-4-turbo, so S2 is present and strong. But the authors also report that the teacher family itself knows the books poorly: GPT-4 answered 51.30 percent of the questions closed-book and 86.09 percent with the documents in context. A teacher that answers half the questions without the source cannot have supplied most of what the trained model learned about it. The documents supplied the content; the teacher supplied the restatement. The authors' own mathematical model describes the effect as rearranging knowledge to make learning more data-efficient, which is the re-presentation cell under another name.

Later knowledge-injection work points the same way. SPA (Tang et al., 2026) reports that a small set of carefully designed prompts, scaled up, is a tough baseline to beat, and that reinforcement-learned augmentation methods that improve token efficiency at small scale suffer diversity collapse as data grows. That is S4, design, competing successfully with S3, selection, inside the re-presentation cell.

4.5 Where re-presentation borrows structure

Two 2025-2026 methods go beyond single-document rephrasing by synthesising across documents, and both borrow the cross-document structure from outside the generator. WRAP++ (Zhou et al., 2026) uses Wikipedia hyperlinks to find related document pairs and synthesises questions that require both documents, amplifying about 8.4 billion tokens of Wikipedia into 80 billion tokens of cross-document question-answer data and reporting gains on SimpleQA at 7 and 32 billion parameters. The hyperlink graph is human-made. SynthLLM (Qin et al., 2025) extracts and recombines concepts across documents with a graph algorithm and reports that the resulting mathematical data follows a rectified scaling law, with gains plateauing near 300 billion tokens.

The plateau is informative. A re-presentation process over a fixed seed should eventually exhaust the useful restatements of what the seed says. A plateau at a finite synthetic token count is what that account predicts; it is not what an information-creating process predicts.

4.6 Re-presentation with an instruction-shaped prior

Two further lines restructure organic text into task form. Instruction Pre-Training (Cheng et al., 2024) augments raw corpora with instruction-response pairs produced by a synthesiser fine-tuned on a broad collection of existing

context-based task datasets, and reports that continued pretraining with it lets an 8-billion-parameter model match or exceed a 70-billion-parameter model of the same family. MIND (Akteer et al., 2024) rewrites mathematical web text as dialogues, finds that a knowledge gap between the conversing parties is essential to the value of the dialogue, and reports a 13.42-point gain on GSM8K over pretraining on the raw corpus alone. In both, the content is the seed's and the form comes from S2 and S4: a synthesiser trained on human task datasets in one case, and a human-designed conversational template in the other.

4.7 What this section establishes

The best-controlled pretraining gains from synthetic data are gains from re-presentation. They are anchored to an organic seed, they beat repetition of that seed, they do not depend on the generator's knowledge above a competence floor, and they need no verifier. In the ledger, the active sources are S1 and S4, with S2 contributing competence rather than content. This complicates both readings at once. The self-generation reading is wrong that these pipelines create information: nothing here adds propositions to the seed. The verifier reading is wrong that the gain needs an external check: none of these pipelines has one, and the one that adds selection signals (SynPro) does so to steer re-presentation, not to verify content.

5. Information: where a teacher supplies content

5.1 The teacher-written corpus

The line that made synthetic pretraining famous is the opposite case. TinyStories (Eldan and Li, 2023) is a corpus of short stories in the vocabulary of small children, written by GPT-3.5 and GPT-4, on which models below ten million parameters learned to produce fluent, consistent multi-paragraph stories. Phi-1 (Gunasekar et al., 2023) trained a 1.3-billion-parameter code model on about 6 billion tokens of filtered web data plus about 1 billion tokens of textbooks and exercises written by GPT-3.5, and reported 50.6 percent pass@1 on HumanEval. These are information results in the ledger's sense: the generator wrote content that the student had no organic access to. They are also, plainly, distillation results. S2 is the active source, and the student's ceiling is the teacher's.

Distillation's ceiling is now quantified. Busbridge et al. (2025) fit a scaling law for distilled students as a function of compute split between student and teacher, and find that distillation beats ordinary supervised training only up to a compute level that grows with student size, and that when the teacher must itself be trained for a single student, supervised learning is generally preferable. Gudibande et al. (2023) showed the qualitative version earlier: models fine-tuned to imitate a

proprietary system acquired its style well enough to persuade crowd raters but closed little of the factual and capability gap on tasks the imitation data did not heavily cover. Bansal et al. (2024) complicate the picture from the other side: when synthetic reasoning data is generated under a fixed compute budget, a weaker and cheaper generator produced data with more coverage and diversity, and models trained on it beat models trained on data from a stronger, more expensive one, despite a higher false-positive rate. Teacher quality is one input among several even where the teacher is the source.

5.2 Generator-driven data, measured against re-presentation

Where generator-driven and seed-anchored corpora have been compared directly, the generator-driven corpus has not won by enough to credit the generator's knowledge with much. BeyondWeb's summarisation control nearly matched Cosmopedia (DatologyAI et al., 2025), and Kang et al. (2025) found textbook-style generation alone harmful to many domains. The information cell exists, and for narrow domains with a strong teacher it can be large. But it is not where the scalable pretraining gains have come from, and it is bounded by construction.

5.3 The one teacher-ceiling breach

Phi-4 (Abdin et al., 2024) is the result that most directly tests whether synthetic pretraining can exceed its source. Its report states that the model substantially exceeds its teacher, GPT-4o, on STEM-focused question answering, and its benchmark table lists GPQA at 56.1 for phi-4 against 50.6 for GPT-4o. The authors take this as evidence that their methods go beyond distillation. On the teacher-ceiling control, that is a breach.

The ledger asks what else was in the pipeline. Phi-4's synthetic data - about 400 billion unweighted tokens across 50 dataset types - was seeded from curated organic excerpts of web pages, books and code repositories, and supplemented by filtered organic web data used directly. For some question sets, the authors generated several independent answers, discarded questions on which all answers agreed or none did, and used the plurality answer in place of ground truth in rejection sampling. Post-training added further data and a preference-optimisation scheme. So the breach arrived with S1 (organic seeds), S3 (majority-vote selection) and S4 (a long list of design choices) all present. The report's post-training ablation (Abdin et al., 2024, Table 9) places the breach more precisely. After supervised fine-tuning phi-4 scored 47.3 on GPQA, below its teacher's 50.6; it scored 53.6 after the first preference-optimisation stage, pivotal-token DPO, and 56.1 after both stages, and the authors find that "pivotal token DPO is most useful on reasoning-heavy tasks (GPQA, MATH)". So the margin over the teacher appears only in the preference-optimisation stages, which are selection, and not after training on the synthetic corpus and fine-tuning

alone. The breach shows that a pipeline can exceed one of its sources. What the report does not ablate is the pretraining side: how much of the score that preference optimisation starts from is owed to the organic seeds and to majority-vote selection.

6. Selection: where the verifier does the work, and where the corpus is the verifier

6.1 Checkable targets

The imported-signal reading in its verifier form is strongly supported in one place: problems whose answers can be checked.

AlphaGeometry (Trinh et al., 2024) is the cleanest instance, and it is a pretraining result. Its language model was trained from scratch on synthetic theorems and proofs, millions of them, produced by a symbolic deduction engine without human demonstrations, and the combined system solved 25 of 30 recent olympiad geometry problems. The symbolic engine is S3 and S4 together: it guarantees that every synthetic proof is valid, and its rules encode human geometry. The model contributes search guidance - which auxiliary constructions to try - learned entirely from verified synthetic data. Here the verifier really is doing the work.

Absolute Zero (Zhao et al., 2025) carries the same structure into reasoning post-training. A single model proposes coding tasks and solves them, and a code executor both validates the proposed tasks and checks the answers. The authors report state-of-the-art results among zero-setting reasoners on coding and mathematics without any external task data. The "zero" in the title refers to curated human examples. It does not refer to signal: the executor is the signal, and the model being trained began as a pretrained base model.

ReST-EM (Singh et al., 2023) filters self-generated solutions with binary correctness feedback and reports that this self-training surpasses fine-tuning on human data for advanced mathematics and code. STaR (Zelikman et al., 2022) keeps rationales that reach correct answers, and when the model fails it regenerates the rationale with the correct answer in hand. Both are selection by ground truth. Feng et al. (2024) give a theoretical account of when a verifier selecting synthesised data avoids collapse, and confirm it on eigenvalue computation and news summarisation; Yi et al. (2025) add the limit, that the loop then converges toward what the verifier knows. Amin et al. (2025) analyse curation as boosting, with weak-learner guarantees, and show that concentrating labelling effort on the hardest examples improves performance. In every case, the gain is explained by the selection signal, and its ceiling is the signal's reliability.

6.2 When the signal is the model's own

The interesting cases are those where S3 comes from the model itself. Huang et al. (2022) fine-tuned a 540-billion-parameter model on its own high-confidence chain-of-thought answers to unlabelled questions, selected by self-consistency, and reported gains such as 74.4 to 82.1 percent on GSM8K with no ground-truth labels. Test-time reinforcement learning (Zuo et al., 2025) used majority vote as the reward on unlabelled test problems and reported pass@1 on AIME 2024 rising by about 211 percent for Qwen2.5-Math-7B, with results exceeding the initial model's majority-vote accuracy. Lee et al. (2025) showed small transformers extending arithmetic and string tasks far beyond their training lengths - from 10-digit to 100-digit addition, for instance - by training on their own outputs over a curriculum of gradually harder problems, with filtering by correctness in some settings and by unsupervised length and majority-vote filters in others.

These are the results most often cited for endogenous improvement, and the sharpening account covers them without strain. Majority voting is a way of selecting the mode of the model's own distribution; training on the mode moves mass toward it. That raises single-sample accuracy toward the model's own ensemble accuracy and, through iteration, can push somewhat beyond it, as TTRL reports. But it adds nothing the model's distribution did not contain, and it fails when the mode is wrong. Shafayat et al. (2025) ran self-rewarding reinforcement learning with majority voting for long enough to see the failure: performance first improved, along with the quality of the self-feedback, and then prolonged training led to reward hacking and a sudden, complete collapse. Lee et al. (2025) needed a human-designed easy-to-hard curriculum, which is S4.

The spurious-reward results sharpen the point from the other side. Shao et al. (2025) reported that reinforcement learning with random rewards improved Qwen2.5-Math-7B on MATH-500 by 21.4 points, against 29.1 points with ground-truth rewards. A random signal cannot be doing the work. Their explanation is that the optimiser amplifies a behaviour already favoured in pretraining - reasoning in code without executing it - and they report that the same spurious rewards often fail for other model families. In ledger terms: most of the gain was attributable to S2, the prior, and S3 was nearly irrelevant for that model. That is a signal-ablation control, and it is the control most of the selection literature does not run. The 2026 "Phantom Gains" audit adds a measurement caution: run against a frozen control through the same pipeline, three forms of self-training on Qwen3-8B did not improve problems the base model rarely solved, while external distillation did (Xu et al., 2026).

Yue et al. (2025) offer the boundary result in reinforcement learning: across model families,

algorithms and benchmarks, models trained with verifiable rewards beat their bases at small k but the bases had higher $\text{pass}@k$ at large k , and the reasoning observed was bounded by the base model, whereas distillation from a teacher introduced patterns the base lacked. Whatever the resolution of that debate, the reading relevant here is narrow: in the located evidence, selection redistributes what a model can already produce, and information from a teacher extends it.

6.3 The corpus as verifier

The pretraining results that come closest to genuine bootstrapping share a feature the verifier reading does not name: the selection signal is the organic corpus itself.

Quiet-STaR (Zelikman et al., 2024) trains a model to generate a rationale before each token of ordinary web text and rewards rationales that improve the prediction of the text that follows. After continued pretraining on internet text, the authors report zero-shot improvements from 5.9 to 10.9 percent on GSM8K and from 36.3 to 47.2 percent on CommonsenseQA, with no task-specific fine-tuning. The rationales are self-generated. The scorer is the human text they must predict.

Reinforcement Pre-Training (Dong et al., 2025) recasts next-token prediction as a reasoning task with a verifiable reward for predicting the next token correctly. The verifier is the corpus.

Bootstrapping Latent Thoughts (Ruan et al., 2025) treats web text as the compressed output of an unobserved reasoning process and trains a model on documents augmented with inferred latent thoughts. With latent thoughts written by GPT-4o-mini, a model reached 25.4 percent on MATH against 5.74 percent for a model trained on the raw data. The bootstrapped variant then replaces the teacher with the model itself in an expectation-maximisation loop: it samples candidate latent thoughts, weights each by how well it explains the observed text under the model, resamples, and retrains. The authors report that a 1-billion-parameter model improved across at least three iterations and outperformed raw-data baselines. Two imported signals sit inside this loop. The model is first warm-started on a small corpus of teacher-written latents, and every subsequent iteration scores its own latents by the likelihood of human text.

Synthetic Bootstrapped Pretraining (SBP; Yang et al., 2025b) is the most carefully controlled of the four, and it is the case shown in Figure 1. SBP pairs related documents within the pretraining corpus by nearest-neighbour search, fine-tunes a synthesiser to produce one document of a pair conditioned on the other, and then generates a large synthetic corpus for joint training. The authors stress that this avoids an external teacher model, so that any improvement comes from using the same corpus better. Under a compute-matched design with 3-billion and 6-billion-parameter models trained on

up to a trillion tokens, they compare SBP against a strong repetition baseline and against an oracle with access to 20 times more unique data. In their Table 1, the average question-answering gain over repetition is 2.32 points for SBP against 5.54 for the oracle at the 200-billion-token scale, 0.84 against 1.73 at the trillion-token scale with the 3-billion model, and 1.32 against 2.26 with the 6-billion model. The abstract summarises this as delivering up to 60 percent of the oracle's improvement.

SBP shows that a model can extract usable signal from its own corpus that next-token training on that corpus misses - the relations among documents. In ledger terms that is re-presentation with an unusual structure: the synthesiser learns how documents relate, and writes new documents that make those relations explicit. It is also selection by the corpus, since the synthesiser is trained only on real document pairs. But it is not free of imported signal. The nearest-neighbour pairing uses a separately pretrained embedding model, Qwen3-0.6B-Embedding, which the authors acknowledge an ideal implementation would replace with the model's own representations. And the factuality cost is measurable. The same paper reports that 15.1 percent of synthesised documents at the 200-billion-token scale contained verifiable factual errors as judged by automated fact-checking, falling to 8.7 percent and then 6.5 percent with more data and a larger model, against 1.8 percent for real data. The synthesiser approaches the corpus's factuality as it gets better, but even at its best it writes errors at more than three times the real-data rate.

Figure 1 makes the conservation point and the credit-assignment point together. Panel (a) shows that a synthesiser learned from the corpus beats repetition of the same corpus at every scale: something is being recovered from the corpus that training on it directly misses. Panel (b) shows that the recovered signal arrives with errors the corpus did not contain. The synthesiser is a source of re-presentation and a source of noise; it is not a source of facts.

6.4 What this section establishes

The verifier reading is right where the target is correctness: geometry, code, and arithmetic with checkable answers. There the selection signal is the active source, and its reliability is the ceiling. Where the signal comes from the model's own votes, the gains fit the sharpening account and fail when the vote is wrong. And the closest thing to endogenous bootstrapping at pretraining scale replaces an external verifier with the organic corpus. That is still an imported signal - the human text - but it is not a verifier in any sense the claim "the verifier is doing the work" intends, and it supports a much more specific claim: the organic corpus holds more learnable structure than next-token training extracts, and self-generated text can recover some of it.

7. When nothing is imported: collapse, accumulation and continuation

7.1 Replacement and accumulation

The collapse literature is where the imported-signal reading is strongest, and where its scope is most often overstated. Shumailov et al. (2024) showed that training each generation on the previous generation's output erodes the tails of the original distribution, in language models as well as in simpler generative models; Alemohammad et al. (2023) showed the image-model analogue, in which loops without enough fresh real data lose either quality or diversity. Theoretical accounts make the mechanisms precise: Dohmatob et al. (2024a) describe collapse as a change in the scaling law, including the unlearning of skills; Seddik et al. (2024) show that training on synthetic data alone cannot avoid collapse and give a bound on how much synthetic data can be mixed with real data before it occurs; Wang et al. (2024) prove that once a fixed corpus begins to absorb generated text and no new real text is added, collapse eventually follows however little each model contributes; and Dohmatob et al. (2024b) report a strong form in a regression setting, in which a synthetic fraction as small as 1 percent can stop larger training sets from helping.

The scope condition is the replacement assumption. Gerstgrasser et al. (2024) pretrained sequences of language models and confirmed collapse when each generation's synthetic data replaced the real data, then showed that accumulating synthetic data alongside the original real data avoided it, and proved a finite error bound under accumulation in a linear setting. Kazdan et al. (2024) extended this across three task settings and added a middle case: when the accumulated pool is subsampled to a fixed size each generation, degradation is gradual rather than explosive. Schaeffer et al. (2025) argued in a position paper that the collapse literature uses eight distinct and sometimes conflicting definitions, and that several prominent collapse scenarios depend on conditions that do not match how real training corpora evolve.

For the ledger, the important feature of the accumulation results is that accumulation keeps the organic corpus in the training set, so S1 is never removed. Mostly what accumulation shows is that collapse is avoided, and avoiding degradation is a different finding from producing a gain. Kazdan et al. (2024) do report a gain: in kernel density estimation and in pretraining on TinyStories, real and synthetic data accumulated over several iterations reached lower loss on real test data than real data alone. The gain is conditional. The authors find that adding small amounts of synthetic data improves test loss when real data is scarce, and that when real data is abundant it only raises test loss. And it arrives with S1 present throughout.

7.2 Synthetic data in the pretraining mix

Zhu et al. (2024) pretrained models across different proportions of synthetic data and found that performance fell as the synthetic proportion rose, along with a shift in the token distribution and an over-concentration of n-gram features. Their remedy, token-level editing, keeps human text and resamples only the tokens to which a prior model assigns high confidence. The result is semi-synthetic data that the authors prove, under their model, has bounded test error. In ledger terms, this is a method for keeping S1 almost entirely and limiting S2's contribution to a narrow band of tokens.

7.3 The continuation control

The plainest form of generation tested at pretraining scale is BeyondWeb's continuation run (DatologyAI et al., 2025). Within a 20-billion-token budget, it compared three corpora: the full organic data as an upper bound; half of the organic data repeated twice as a lower bound; and that half plus 10 billion tokens that an outside model, Llama-3.1-8B, wrote by continuing partial organic documents in the same style. Averaged over the benchmark suite, repetition scored 45.5 percent, the continuation corpus 46.2 percent, and the full organic data 46.2 percent. The continuation corpus gained 0.7 points over repetition, which the authors call small, but it matched the full-data upper bound, so it recovered all of the gain that fresh organic data gave. The authors name a confounder themselves: the generator had seen far more than the 20 billion tokens of the experiment, and may have used its parametric knowledge to add knowledge the organic data lacked. BeyondWeb's own corpus, built to fill distributional gaps, scored 50.4 percent, above the upper bound.

Continuation is the operation the self-generation reading has in mind: a model producing more text like the text it has seen. But this run is not self-generation in the ledger's sense. The generator conditioned on organic documents (S1) and was an outside model trained on far more text than the experiment contained (S2). Its match to the upper bound is what either re-presentation of the seed or information from the generator would predict, and the run does not separate the two. The pipeline that went beyond the upper bound added design (S4) to anchoring in a seed restated in a new form (S1 with re-presentation).

7.4 Procedural data: design as the whole signal

At the other end of the scale is synthetic data with no generator model at all. Pre-pretraining on formal languages - strings generated from grammars, not from a language model - can make subsequent natural-language pretraining more token-efficient. Hu et al. (2025) found that formal languages with hierarchical dependencies gave lower natural-language loss and better linguistic generalisation, and that for a 1-billion-parameter model

trained on about 1.6 billion natural tokens, pretraining reached the same loss with a 33 percent smaller token budget. Yamaguchi et al. (2026) tested the idea at up to 7 billion parameters and 100 billion pretraining tokens, found that the gains persist at scale (saving at least 21 billion tokens at the 3-billion-parameter scale), and found that the gains do not track grammatical acceptability but come instead from tasks that train long-range retrieval. The gains shrank only when web text was absent from the pretraining mixture.

Here the entire signal is S4: a human choice of formal structure. The result is evidence that synthetic data can contribute an inductive bias that organic text supplies inefficiently. It is not evidence for self-generation, since no model generated anything.

8. The consolidated evidence

Table 1 brings together the measured values on which the argument rests, one row per cited result. The ledger column records the cell the author assigns the result to on the evidence the source reports; it is the paper's reading, not the source's.

Three patterns stand out. First, the rows assigned to representation are the ones with the strongest controls - repetition baselines, generator-scale ablations and unique-data oracles - and they are the rows with the largest reported exchange rates. Second, the rows where a verifier is the active source are the rows with checkable targets. Third, the rows closest to self-generation either fall back on the corpus as scorer or, like BeyondWeb's continuation run, keep an organic seed and an outside generator. The reinforcement-learning row that randomised its signal (Shao et al., 2025) found most of the gain present without it, while the one pretraining signal ablation located, SynPro's (Section 9.2), found that removing the rewards halved the share of the unique-data gain recovered.

9. Assigning the credit

9.1 A protocol

The ledger turns into a procedure for reading a synthetic-data result and deciding what it is evidence of. Algorithm 1 states it. Its first part assigns a result to a cell using the controls the study ran; its second part says what a claim of endogenous improvement would have to show.

Part B is deliberately demanding, and no located study satisfies all of it. The closest is SBP, which conditions on organic documents at generation time (it fails step 15 by design, since that conditioning is the method), uses a synthesiser fine-tuned from the trained model's own architecture on its own corpus (it passes step 16 up to the separate embedding model), and runs both controls in step 19. BoLT passes step 16 in its later iterations but fails it in the warm start. SynPro's self-generation variant

(Yu and Xiong, 2026, Appendix E.4) comes close to step 16, since its generator starts from the repetition-trained 400-million-parameter model itself, but that model is first instruction-tuned on FLAN and then trained with the same judge and influence rewards, so it fails step 17; it still substantially outperformed repetition (0.4979 against 0.4675 average accuracy). BeyondWeb's continuation test satisfies step 17 only: it conditions on organic documents and uses an outside generator, so it fails steps 15 and 16, and it matched its full-data upper bound.

9.2 Which controls the located studies run

Auditing the studies in Table 1 against Part A shows where the field's evidence is thick and where it is thin. Repetition baselines are now standard in the data-constrained line (Yang et al., 2025b; Yu and Xiong, 2026; DatologyAI et al., 2025) and implicit in the scaling-law work (Muennighoff et al., 2023). Generator-scale ablations are present in all three systematic rephrasing studies (Kang et al., 2025; DatologyAI et al., 2025; Niklaus et al., 2026) and in WRAP. Unique-data oracles appear in SBP, in BeyondWeb's full-data upper bound, and as a reference in SynPro.

Signal ablations are rare. The one pretraining signal ablation located is SynPro's, run at the 400-million-parameter scale only (Yu and Xiong, 2026, Table 3). Measured as the share of the gap between repetition and unique data that each variant recovers, the full method recovered 79 percent; prompting the base generator with the same two operations and no RL training, and so with no rewards, recovered 39 percent; and removing one reward at a time left 52 percent without the influence reward, 39 percent without the faithfulness reward and 26 percent without the quality reward. In that setting the selection signals roughly double the share recovered, while the half that remains without them is representation by an instruction-tuned generator. The ablation removes signals rather than randomising them, and it was not repeated at the larger scales. SBP's embedding-based pairing is examined in an appendix on random pairs and embeddings, but the main results do not separate the contribution of the external embedding model. The one randomised-signal control in the reviewed evidence comes from reinforcement learning, not pretraining (Shao et al., 2025), and it found the signal mostly idle for one model family. The teacher-ceiling control is likewise rare; phi-4 reports one breach and traces it to its preference-optimisation stages, but does not ablate the sources of its synthetic pretraining data (Abdin et al., 2024).

This asymmetry matters for the title question. The verifier reading cannot be confirmed for pretraining from the located evidence, because the control that would confirm it - removing the verifier and watching the gain vanish - is a control the pretraining literature has run once, at small scale, on judge and influence

Table 1. Reported effects of synthetic data, one row per cited result, with the cell of the ledger the author assigns each to. "Rep." is representation, "Info." is teacher information, "Sel." is selection. Values are as each source reports them; units differ across rows and are not comparable across them.

Source	Setting	Reported value	Control	Ledger cell
Maini et al. (2024)	Rephrased C4	~3x faster; > 2 pt zero-shot QA	Real data	Rep.
Kang et al. (2025)	1/3 rephrased mix	5-10x faster to equal loss	Web text	Rep.
Kang et al. (2025)	Rephrased alone	Not faster than web	Web text	Rep.
Kang et al. (2025)	Generator size	No reliable gain beyond ~8B	Generator scale	Rep.
DatologyAI et al. (2025)	Best mix	Up to 7.7x faster	Open web	Rep.
DatologyAI et al. (2025)	Generator size	Saturates by 8B	Generator scale	Rep.
DatologyAI et al. (2025)	Continuation	46.2% vs 45.5% repeat, 46.2% full data	Repetition, oracle	S1 + S2 (generator information possible)
Niklaus et al. (2026)	Generator size	No benefit beyond 1B	Generator scale	Rep.
Yu and Xiong (2026)	Rephrase + reformat	3.4-5.2x effective tokens	Repetition	Rep. + Sel.
Yang et al. (2024)	EntiGraph, 455M tokens	80% of open-book gain	Open-book	Rep.
Yang et al. (2025b)	SBP, 200B scale	+ 2.32 vs oracle + 5.54 QA	Repetition, oracle	Rep. + Sel. (corpus)
Yang et al. (2025b)	SBP, 1T, 6B model	+ 1.32 vs oracle + 2.26 QA	Repetition, oracle	Rep. + Sel. (corpus)
Yang et al. (2025b)	SBP factuality	6.5-15.1% vs 1.8% real	Real data	(noise)
Ruan et al. (2025)	Teacher latents	MATH 25.4% vs 5.74%	Raw data	Info. + Rep.
Zelikman et al. (2024)	Quiet-STaR	GSM8K 5.9% to 10.9%	Base model	Sel. (corpus)
Gunasekar et al. (2023)	phi-1	HumanEval 50.6%	Other models	Info.
Abdin et al. (2024)	phi-4 vs teacher	GPQA 56.1 vs 50.6	Teacher ceiling	Info. + Sel. + S1
Trinh et al. (2024)	AlphaGeometry	25 of 30 problems	Prior systems	Sel. (verifier)
Shao et al. (2025)	Random reward	+ 21.4 vs + 29.1 MATH-500	Signal ablation	Prior (S2)
Huang et al. (2022)	Self-consistency SFT	GSM8K 74.4% to 82.1%	Base model	Sel. (own vote)
Hu et al. (2025)	Formal pre-pretraining	33% fewer tokens to equal loss	Natural tokens	Design (S4)
Yamaguchi et al. (2026)	Pre-pretraining at 3B	≥ 21B tokens saved	No pre-pretraining	Design (S4)
Dohmatob et al. (2024b)	Regression setting	1% synthetic stalls scaling	Real only	none

Algorithm 1. Assigning credit for a synthetic-data gain.

Input: a reported gain G of a synthetic pipeline P over a baseline B Part A – which source is active? 1 if B is not a repetition of P 's organic seed: 2 G may be re-representation, information or design; report "unattributed beyond the weak form" 3 if a generator-scale ablation exists: 4 if G saturates at small generators: $S2$ supplies competence, not content → candidate: re-representation 5 else: candidate: information from $S2$ 6 if a seed ablation exists and G tracks seed quality: 7 $S1$ active → re-representation 8 if a signal ablation exists ($S3$ removed or randomised): 9 if G survives: $S3$ not doing the work; G is prior ($S2$) 10 else: $S3$ active → selection, ceiling = $S3$ reliability 11 if P 's target is checkable and $S3$ is a sound checker: 12 verifier claim licensed for that target only 13 if the trained model exceeds its teacher on task T : 14 distillation alone does not explain T ; check $S1$, $S3$ Part B – what an endogenous claim must show 15 remove $S1$ at generation time (no conditioning documents) 16 use the trained model itself as generator (no outside $S2$) 17 use no outside $S3$ (no judge, executor, label, reference set) 18 hold $S4$ to a fixed, published prompt set 19 compare against repetition AND a unique-data oracle 20 report factuality and diversity of the synthetic corpus 21 claim endogenous gain only if $G > 0$ against repetition with steps 15-20 satisfied
--

rewards rather than on a verifier. In the pipelines where pretraining gains are largest, there is no verifier to remove.

10. Measurements that would settle the open part

Five measurements would move the open questions to

settled ones. Each uses methods the cited studies already use.

M1. A full Part-B run at pretraining scale. Train a model on an organic corpus, then use that model, with no outside generator, judge, reference set or executor, to generate unconditioned text, and pretrain a fresh model on the organic corpus plus that text. Compare against repetition and against a unique-data oracle at matched compute, at two or more scales, and report the factuality and diversity of the synthetic corpus. A gain over repetition here would be the first located evidence for the strong self-generation reading at pretraining scale. BeyondWeb's continuation control does not predict its size: that run matched its full-data upper bound, but with an outside generator conditioned on organic documents, whose own knowledge its authors flag as a possible source of the match.

M2. Signal ablations for the selection-steered pipelines. For SynPro-style generators, repeat the reward ablations that Yu and Xiong (2026) ran at 400 million parameters at larger scales, and add arms in which each reward is randomised rather than removed, at fixed generation budget, reporting effective tokens against repetition. For SBP-style synthesis, replace the external embedding model with representations from the model being trained, as its authors propose, and with random pairing. These ablations would say how much of the re-presentation gain needs imported selection.

M3. Information accounting for teacher-written corpora. For a generator-driven corpus, measure the share of its verifiable claims that do not appear in the organic training data of the student, and the share that are false, and relate both to the student's gain on held-out questions about those claims. This would separate the information cell from the re-presentation cell for teacher-written data, which currently share a single benchmark number.

M4. A teacher-ceiling ablation for the one breach. For a phi-4-style pipeline, report the student's score against its teacher on the breached tasks with (a) organic seeds replaced by teacher-generated seeds and (b) majority-vote selection removed, each followed by the same post-training. A third arm, post-training removed, is already partly answered: phi-4's own ablation puts it below its teacher on GPQA after supervised fine-tuning and above it only after preference optimisation (Abdin et al., 2024). Which of the two remaining ablations narrows the breach would identify how much the pretraining sources contribute.

M5. Proximity-controlled evaluation of re-presentation. Evaluate rephrased-data gains on tasks whose format differs from every rephrasing template used, and on held-out knowledge probes built from the seed, so that a format-matching gain can be told apart from a knowledge-storage gain of the Allen-Zhu and Li (2023) kind. Without this, part of the re-presentation cell

may be benchmark proximity.

11. Objections

"The conservation point makes the whole question trivial." It makes one version of it trivial: whether any signal is imported. It does not make trivial which signal accounts for a gain, and that is the question with consequences. A data wall for knowledge is not breached by re-presentation; a data wall for checkable skills can be extended by a verifier; a data wall for inductive bias can be moved by design. These are different forecasts, and they follow from different cells of the ledger.

"Re-presentation is just a kind of information, since a paraphrase changes what the model can extract." It changes what the model can extract from content the seed already holds. If a paraphrase makes a fact extractable that was memorised but unusable, the model gains a capability without the corpus gaining a proposition. The distinction matters because re-presentation's ceiling is set by the seed: no amount of rephrasing will teach a model a fact its seed does not contain. The plateaus reported for re-presentation pipelines (Qin et al., 2025) are what that ceiling predicts.

"SBP and BoLT are endogenous; the external embedding model and the warm start are implementation details." They may be. The SBP authors say an ideal implementation would drop the external embedder, and BoLT's later iterations do not use the teacher. But "may be" is the point: neither study reports the run without them, and the ledger records what was run. The nearest precedent is SynPro's, where a generator started from the trained model's own checkpoint, though instruction-tuned on FLAN, recovered 68 percent of the unique-data gain against 79 percent for the outside generator (Yu and Xiong, 2026). If M2 shows that SBP's gains survive with the model's own representations, the paper's assignment of SBP should be updated toward the self-generation reading, with the factuality cost in Figure 1 still attached.

"Phi-4 already settles that synthetic data can exceed its teacher." Phi-4 shows that a pipeline with organic seeds, majority-vote selection and post-training can exceed one teacher on some benchmarks. That rules out pure distillation from that teacher as the explanation. It does not establish that synthesis created the information. The report's own post-training ablation places the GPQA margin in the preference-optimisation stages, a selection step, since the model scored below its teacher after supervised fine-tuning; and the organic seeds and majority-vote selection in its pretraining data are not ablated.

"Verifier-based synthesis will dominate once pretraining moves to reasoning-heavy data." That may be true, and nothing here argues against it. It would make the verifier reading more relevant in future, but

only for targets with sound checkers. The re-presentation results concern general web knowledge, where no comparable checker exists; the collapse results concern the general text distribution. A shift of pretraining toward checkable domains would shift the share of the gain the verifier explains, not the ledger.

"The benchmark-proximity confound undermines the re-presentation results." It qualifies them. WRAP's own authors attribute part of their gain to style matching evaluation, and the structured formats that FinePhrase found best are, plausibly, closer to benchmark formats than raw web text. But the strongest re-presentation evidence includes loss-based measures (Kang et al., 2025, measure speed to a validation loss) and knowledge probes on held-out documents (Yang et al., 2024), which format matching does not explain. M5 would bound the confound.

12. What is not known

No study located for this paper has measured a pretraining gain from self-generated text with no organic conditioning, no outside generator, no outside selection signal and a fixed published prompt set, against both repetition and a unique-data oracle.

Only one pretraining study located has ablated its selection signals while holding the generator, seed and budget fixed (Yu and Xiong, 2026), at a single 400-million-parameter scale and by removal rather than randomisation. Whether its result holds at larger scales is not known.

It is not known how much of the re-presentation gain reported on downstream benchmarks is format proximity rather than improved knowledge storage.

Phi-4's margin over its teacher on GPQA appears only after preference optimisation (Abdin et al., 2024). It is not known how much the organic seeds and majority-vote selection in its synthetic pretraining data contribute to the score that optimisation starts from.

It is not known whether the gains from corpus-bootstrapped synthesis persist, grow or reverse over multiple generations of synthesis on a fixed corpus; SBP reports one round, and BoLT reports three iterations at 1 billion parameters on mathematics.

It is not known how the factuality cost of synthesis (Figure 1b) affects downstream factual reliability at scale. SBP reports the error rate of the synthetic documents, not the error rate those documents induce in the trained model.

13. Related work and how this paper differs

Three bodies of work cover parts of this ground. Surveys of synthetic data organise methods by workflow and stress factuality and fidelity (Liu et al., 2024; Long et al., 2024); they catalogue pipelines but do not assign their gains to sources. BeyondWeb (DatologyAI et al., 2025)

draws the closest published distinction, between a generator-driven paradigm that creates knowledge from the model and a source-rephrasing paradigm that enhances existing knowledge, and its evidence is central to Section 4. This paper adds the selection and design sources, the corpus-as-verifier class, the control-by-cell protocol, and the audit of which controls are missing. Chen et al. (2026) order feedback signals into a verification hierarchy for recursive self-improvement and observe that demonstrated improvement follows that hierarchy; this paper tests the corresponding claim for pretraining corpora and finds that the hierarchy, which ranks signals by how strongly they can verify, does not predict where the pretraining gains are, because the largest of them involve no verification. Yi et al. (2025) and Feng et al. (2024) give the theory of verifier-guided synthesis; this paper uses their results where verifiers are present and argues that most pretraining gains arise where they are not.

14. Conclusion

The question this paper set out to answer was whether synthetic pretraining data delivers capability the generating model did not have, or whether every success is an imported signal in disguise. The answer depends on what "imported" is allowed to mean. If it means that some signal from outside the generating model is present, then every success located is imported, and the claim cannot be wrong. If it means that a verifier carries the gain, the claim is right where the target is checkable and wrong for the larger and better-controlled body of pretraining results, which come from restating organic text in more learnable forms with no verifier involved.

The useful finding is the split itself. Re-presentation can make a fixed corpus worth several times its size, and is bounded by what the corpus says. Teacher information can add to what the corpus says, and is bounded by the teacher. Selection can raise reliability within what the generator covers, and is bounded by the signal. Self-generation with none of these present has, in the located evidence, produced degradation. The organic corpus turns out to be the most important imported signal of all - not because synthetic text cannot help, but because the forms of synthesis that help most are ways of reading the human corpus more thoroughly. The data wall, on this evidence, is lower for skills that can be checked and for knowledge that is stored inefficiently, and no lower at all for knowledge that is not in the corpus.

References

- Abdin, M., Aneja, J., Behl, H., Bubeck, S., Eldan, R., Gunasekar, S., Harrison, M., Hewett, R. J., Javaheripi, M., Kauffmann, P., et al. (2024). Phi-4 Technical Report. arXiv:2412.08905. doi:10.48550/arXiv.2412.08905. [TECHNICAL REPORT]
- Akter, S. N., Prabhumoye, S., Kamalu, J., Satheesh, S., Nyberg, E., Patwary, M., Shoeybi, M., & Catanzaro, B. (2024). MIND: Math Informed syNthetic Dialogues for Pretraining LLMs. arXiv:2410.12881. doi:10.48550/arXiv.2410.12881. [PREPRINT]

- Alemohammad, S., Casco-Rodriguez, J., Luzi, L., Humayun, A. I., Babaei, H., LeJeune, D., Siahkoohi, A., & Baraniuk, R. G. (2023). Self-Consuming Generative Models Go MAD. arXiv:2307.01850. doi:10.48550/arXiv.2307.01850. [PREPRINT]
- Allen-Zhu, Z., & Li, Y. (2023). Physics of Language Models: Part 3.1, Knowledge Storage and Extraction. arXiv:2309.14316. doi:10.48550/arXiv.2309.14316. [PREPRINT]
- Amin, K., Babakniya, S., Bie, A., Kong, W., Syed, U., & Vassilvitskii, S. (2025). Escaping Collapse: The Strength of Weak Data for Large Language Model Training. arXiv:2502.08924. doi:10.48550/arXiv.2502.08924. [PREPRINT]
- Bansal, H., Hosseini, A., Agarwal, R., Tran, V. Q., & Kazemi, M. (2024). Smaller, Weaker, Yet Better: Training LLM Reasoners via Compute-Optimal Sampling. arXiv:2408.16737. doi:10.48550/arXiv.2408.16737. [PREPRINT]
- Busbridge, D., Shidani, A., Weers, F., Ramapuram, J., Littwin, E., & Webb, R. (2025). Distillation Scaling Laws. ICML 2025. arXiv:2502.08606. doi:10.48550/arXiv.2502.08606.
- Chen, M., Wang, L., & Qu, B. (2026). Recursive Self-Improvement in AI: From Bounded Self-Refinement to Autonomous Research Loops. arXiv:2607.07663. doi:10.48550/arXiv.2607.07663. [PREPRINT]
- Cheng, D., Gu, Y., Huang, S., Bi, J., Huang, M., & Wei, F. (2024). Instruction Pre-Training: Language Models are Supervised Multitask Learners. EMNLP 2024. arXiv:2406.14491. doi:10.48550/arXiv.2406.14491.
- DatologyAI, Maini, P., Dorna, V., Doshi, P., Carranza, A., Pan, F., Urbanek, J., Burstein, P., Fang, A., Deng, A., et al. (2025). BeyondWeb: Lessons from Scaling Synthetic Data for Trillion-scale Pretraining. arXiv:2508.10975. doi:10.48550/arXiv.2508.10975. [PREPRINT]
- Dohmatob, E., Feng, Y., Yang, P., Charton, F., & Kempe, J. (2024a). A Tale of Tails: Model Collapse as a Change of Scaling Laws. arXiv:2402.07043. doi:10.48550/arXiv.2402.07043. [PREPRINT]
- Dohmatob, E., Feng, Y., Subramonian, A., & Kempe, J. (2024b). Strong Model Collapse. arXiv:2410.04840. doi:10.48550/arXiv.2410.04840. [PREPRINT]
- Dong, Q., Dong, L., Tang, Y., Ye, T., Sun, Y., Sui, Z., & Wei, F. (2025). Reinforcement Pre-Training. arXiv:2506.08007. doi:10.48550/arXiv.2506.08007. [PREPRINT]
- Eldan, R., & Li, Y. (2023). TinyStories: How Small Can Language Models Be and Still Speak Coherent English? arXiv:2305.07759. doi:10.48550/arXiv.2305.07759. [PREPRINT]
- Feng, Y., Dohmatob, E., Yang, P., Charton, F., & Kempe, J. (2024). Beyond Model Collapse: Scaling Up with Synthesized Data Requires Verification. arXiv:2406.07515. doi:10.48550/arXiv.2406.07515. [PREPRINT]
- Gan, Z., & Liu, Y. (2024). Towards a Theoretical Understanding of Synthetic Data in LLM Post-Training: A Reverse-Bottleneck Perspective. arXiv:2410.01720. doi:10.48550/arXiv.2410.01720. [PREPRINT]
- Gerstgrasser, M., Schaeffer, R., Dey, A., Rafailov, R., Sleight, H., Hughes, J., Korbak, T., Agrawal, R., Pai, D., Gromov, A., Roberts, D. A., Yang, D., Donoho, D. L., & Koyejo, S. (2024). Is Model Collapse Inevitable? Breaking the Curse of Recursion by Accumulating Real and Synthetic Data. arXiv:2404.01413. doi:10.48550/arXiv.2404.01413. [PREPRINT]
- Gillman, N., Freeman, M., Aggarwal, D., Hsu, C.-H., Luo, C., Tian, Y., & Sun, C. (2024). Self-Correcting Self-Consuming Loops for Generative Model Training. ICML 2024. arXiv:2402.07087. doi:10.48550/arXiv.2402.07087.
- Gudibande, A., Wallace, E., Snell, C., Geng, X., Liu, H., Abbeel, P., Levine, S., & Song, D. (2023). The False Promise of Imitating Proprietary LLMs. arXiv:2305.15717. doi:10.48550/arXiv.2305.15717. [PREPRINT]
- Gunasekar, S., Zhang, Y., Aneja, J., Mendes, C. C. T., Del Giorno, A., Gopi, S., Javaheripi, M., Kauffmann, P., de Rosa, G., Saarikivi, O., et al. (2023). Textbooks Are All You Need. arXiv:2306.11644. doi:10.48550/arXiv.2306.11644. [PREPRINT]
- Hu, M. Y., Petty, J., Shi, C., Merrill, W., & Linzen, T. (2025). Between Circuits and Chomsky: Pre-pretraining on Formal Languages Imparts Linguistic Biases. ACL 2025. arXiv:2502.19249. doi:10.48550/arXiv.2502.19249.
- Huang, A., Block, A., Foster, D. J., Rohatgi, D., Zhang, C., Simchowit, M., Ash, J. T., & Krishnamurthy, A. (2024). Self-Improvement in Language Models: The Sharpening Mechanism. arXiv:2412.01951. doi:10.48550/arXiv.2412.01951. [PREPRINT]
- Huang, J., Gu, S. S., Hou, L., Wu, Y., Wang, X., Yu, H., & Han, J. (2022). Large Language Models Can Self-Improve. arXiv:2210.11610. doi:10.48550/arXiv.2210.11610. [PREPRINT]
- Kang, F., Ardalani, N., Kuchnik, M., Emad, Y., Elhoushi, M., Sengupta, S., Li, S.-W., Raghavendra, R., Jia, R., & Wu, C.-J. (2025). Demystifying Synthetic Data in LLM Pre-training: A Systematic Study of Scaling Laws, Benefits, and Pitfalls. EMNLP 2025. arXiv:2510.01631. doi:10.48550/arXiv.2510.01631.
- Kazdan, J., Schaeffer, R., Dey, A., Gerstgrasser, M., Rafailov, R., Donoho, D. L., & Koyejo, S. (2024). Collapse or Thrive? Perils and Promises of Synthetic Data in a Self-Generating World. NeurIPS 2024 Workshops (M3L, ATTRIB). arXiv:2410.16713. doi:10.48550/arXiv.2410.16713.
- Lee, N., Cai, Z., Schwarzschild, A., Lee, K., & Papailiopoulos, D. (2025). Self-Improving Transformers Overcome Easy-to-Hard and Length Generalization Challenges. arXiv:2502.01612. doi:10.48550/arXiv.2502.01612. [PREPRINT]
- Liu, R., Wei, J., Liu, F., Si, C., Zhang, Y., Rao, J., Zheng, S., Peng, D., Yang, D., Zhou, D., & Dai, A. M. (2024). Best Practices and Lessons Learned on Synthetic Data. arXiv:2404.07503. doi:10.48550/arXiv.2404.07503. [PREPRINT]
- Long, L., Wang, R., Xiao, R., Zhao, J., Ding, X., Chen, G., & Wang, H. (2024). On LLMs-Driven Synthetic Data Generation, Curation, and Evaluation: A Survey. arXiv:2406.15126. doi:10.48550/arXiv.2406.15126. [PREPRINT]
- Maini, P., Seto, S., Bai, H., Grangier, D., Zhang, Y., & Jaitly, N. (2024). Rephrasing the Web: A Recipe for Compute and Data-Efficient Language Modeling. arXiv:2401.16380. doi:10.48550/arXiv.2401.16380. [PREPRINT]
- Muennighoff, N., Rush, A. M., Barak, B., Le Scao, T., Piktus, A., Tazi, N., Pyysalo, S., Wolf, T., & Raffel, C. (2023). Scaling Data-Constrained Language Models. arXiv:2305.16264. doi:10.48550/arXiv.2305.16264. [PREPRINT]
- Nguyen, T., Li, Y., Golovneva, O., Zettlemoyer, L., Oh, S., Schmidt, L., & Li, X. (2025). Recycling the Web: A Method to Enhance Pre-training Data Quality and Quantity for Language Models. COLM 2025. arXiv:2506.04689. doi:10.48550/arXiv.2506.04689.
- Niklaus, J., Yamaguchi, A., Stefanik, M., Penedo, G., Kydlicek, H., Bakouch, E., Tunstall, L., Beeching, E. E., Frere, T., Raffel, C., von Werra, L., & Wolf, T. (2026). How Can We Synthesize High-Quality Pretraining Data? A Systematic Study of Prompt Design, Generator Model, and Source Data. COLM 2026. arXiv:2604.13977. doi:10.48550/arXiv.2604.13977.
- Qin, Z., Dong, Q., Zhang, X., Dong, L., Huang, X., Yang, Z., Khademi, M., Zhang, D., Awadalla, H. H., Fung, Y. R., Chen, W., Cheng, M., & Wei, F. (2025). Scaling Laws of Synthetic Data for Language Models. COLM 2025. arXiv:2503.19551. doi:10.48550/arXiv.2503.19551.
- Ruan, Y., Band, N., Maddison, C. J., & Hashimoto, T. (2025). Reasoning to Learn from Latent Thoughts. arXiv:2503.18866. doi:10.48550/arXiv.2503.18866. [PREPRINT]
- Schaeffer, R., Kazdan, J., Arulandu, A. C., & Koyejo, S. (2025). Position: Model Collapse Does Not Mean What You Think. arXiv:2503.03150. doi:10.48550/arXiv.2503.03150. [PREPRINT]
- Seddik, M. E. A., Chen, S.-W., Hayou, S., Youssef, P., & Debbah, M. (2024). How Bad is Training on Synthetic Data? A Statistical Analysis of Language Model Collapse. arXiv:2404.05090. doi:10.48550/arXiv.2404.05090. [PREPRINT]
- Shafayat, S., Tajwar, F., Salakhutdinov, R., Schneider, J., & Zanette, A. (2025). Can Large Reasoning Models Self-Train? arXiv:2505.21444. doi:10.48550/arXiv.2505.21444. [PREPRINT]

- Shao, R., Li, S. S., Xin, R., Geng, S., Wang, Y., Oh, S., Du, S. S., Lambert, N., Min, S., Krishna, R., Tsvetkov, Y., Hajishirzi, H., Koh, P. W., & Zettlemoyer, L. (2025). Spurious Rewards: Rethinking Training Signals in RLVR. arXiv:2506.10947. doi:10.48550/arXiv.2506.10947. [PREPRINT]
- Shumailov, I., Shumaylov, Z., Zhao, Y., Gal, Y., Papernot, N., & Anderson, R. (2023). The Curse of Recursion: Training on Generated Data Makes Models Forget. arXiv:2305.17493. doi:10.48550/arXiv.2305.17493. [PREPRINT]
- Shumailov, I., Shumaylov, Z., Zhao, Y., Papernot, N., Anderson, R., & Gal, Y. (2024). AI models collapse when trained on recursively generated data. *Nature*, 631, 755-759. doi:10.1038/s41586-024-07566-y.
- Singh, A., Co-Reyes, J. D., Agarwal, R., Anand, A., Patil, P., Garcia, X., Liu, P. J., Harrison, J., Lee, J., Xu, K., et al. (2023). Beyond Human Data: Scaling Self-Training for Problem-Solving with Language Models. *TMLR*. arXiv:2312.06585. doi:10.48550/arXiv.2312.06585.
- Song, Y., Zhang, H., Eisenach, C., Kakade, S., Foster, D., & Ghai, U. (2025). Mind the Gap: Examining the Self-Improvement Capabilities of Large Language Models. *ICLR 2025*. arXiv:2412.02674. doi:10.48550/arXiv.2412.02674.
- Su, D., Kong, K., Lin, Y., Jennings, J., Norick, B., Kliegl, M., Patwary, M., Shoenybi, M., & Catanzaro, B. (2024). Nemotron-CC: Transforming Common Crawl into a Refined Long-Horizon Pretraining Dataset. *ACL 2025*. arXiv:2412.02595. doi:10.48550/arXiv.2412.02595.
- Tang, K., Wang, J., Wang, S., & Lyu, K. (2026). SPA: A Simple but Tough-to-Beat Baseline for Knowledge Injection. arXiv:2603.22213. doi:10.48550/arXiv.2603.22213. [PREPRINT]
- Trinh, T. H., Wu, Y., Le, Q. V., He, H., & Luong, T. (2024). Solving olympiad geometry without human demonstrations. *Nature*, 625, 476-482. doi:10.1038/s41586-023-06747-5.
- Villalobos, P., Ho, A., Sevilla, J., Besiroglu, T., Heim, L., & Hobbhahn, M. (2022). Will we run out of data? Limits of LLM scaling based on human-generated data. arXiv:2211.04325. doi:10.48550/arXiv.2211.04325. [PREPRINT]
- Wang, L., Shi, X., Li, G., Li, J., Zhang, X., Dong, Y., Jiao, W., & Mei, H. (2024). Theoretical Proof that Auto-regressive Language Models Collapse when Real-world Data is a Finite Set. arXiv:2412.14872. doi:10.48550/arXiv.2412.14872. [PREPRINT]
- Xu, C., Yan, N., Chen, L., & Kechadi, M.-T. (2026). Phantom Gains: Auditing Self-Improvement Against a Measured Null. arXiv:2608.20290. doi:10.48550/arXiv.2608.20290. [PREPRINT]
- Yamaguchi, A., Inaba, T., Niklaus, J., Stefanik, M., Villavicencio, A., & Aletras, N. (2026). Synthetic Pre-pretraining Survives Scale, but Not as a Grammatical Prior. arXiv:2609.39827. doi:10.48550/arXiv.2609.39827. [PREPRINT]
- Yang, H., Xerri, M., Park, S., Zhang, H., Feng, Y., Kogilathota, S. A., & Zhou, J. (2026). Self-Improvement of Large Language Models: A Technical Overview and Future Outlook. *TMLR*. arXiv:2603.25681. doi:10.48550/arXiv.2603.25681.
- Yang, Z., Band, N., Li, S., Candes, E., & Hashimoto, T. (2024). Synthetic continued pretraining. arXiv:2409.07431. doi:10.48550/arXiv.2409.07431. [PREPRINT]
- Yang, Z., Zhang, A., Liu, H., Hashimoto, T., Candes, E., Wang, C., & Pang, R. (2025b). Synthetic bootstrapped pretraining. arXiv:2509.15248. doi:10.48550/arXiv.2509.15248. [PREPRINT]
- Yi, B., Liu, Q., Cheng, Y., & Xu, H. (2025). Escaping Model Collapse via Synthetic Data Verification: Near-term Improvements and Long-term Convergence. arXiv:2510.16657. doi:10.48550/arXiv.2510.16657. [PREPRINT]
- Yu, Z., & Xiong, C. (2026). Generating Pretraining Tokens from Organic Data for Data-Bound Scaling. *COLM 2026*. arXiv:2605.17849. doi:10.48550/arXiv.2605.17849.
- Yue, Y., Chen, Z., Lu, R., Zhao, A., Wang, Z., Yue, Y., Song, S., & Huang, G. (2025). Does Reinforcement Learning Really Incentivize Reasoning Capacity in LLMs Beyond the Base Model? *NeurIPS 2025*. arXiv:2504.13837. doi:10.48550/arXiv.2504.13837.
- Zelikman, E., Wu, Y., Mu, J., & Goodman, N. D. (2022). STaR: Bootstrapping Reasoning With Reasoning. arXiv:2203.14465. doi:10.48550/arXiv.2203.14465. [PREPRINT]
- Zelikman, E., Harik, G., Shao, Y., Jayasiri, V., Haber, N., & Goodman, N. D. (2024). Quiet-STaR: Language Models Can Teach Themselves to Think Before Speaking. arXiv:2403.09629. doi:10.48550/arXiv.2403.09629. [PREPRINT]
- Zhao, A., Wu, Y., Yue, Y., Wu, T., Xu, Q., Yue, Y., Lin, M., Wang, S., Wu, Q., Zheng, Z., & Huang, G. (2025). Absolute Zero: Reinforced Self-play Reasoning with Zero Data. arXiv:2505.03335. doi:10.48550/arXiv.2505.03335. [PREPRINT]
- Zhou, J., Wang, Y., Wu, X., Yu, T., & Zhang, F. (2026). WRAP + +: Web discoverY Amplified Pretraining. arXiv:2604.06829. doi:10.48550/arXiv.2604.06829. [PREPRINT]
- Zhu, X., Cheng, D., Li, H., Zhang, K., Hua, E., Lv, X., Ding, N., Lin, Z., Zheng, Z., & Zhou, B. (2024). How to Synthesize Text Data without Model Collapse? *ICML 2025*. arXiv:2412.14689. doi:10.48550/arXiv.2412.14689.
- Zuo, Y., Zhang, K., Sheng, L., Qu, S., Cui, G., Zhu, X., Li, H., Long, X., Zhang, X., Ma, Z., Hua, E., Qi, B., Sun, Y., Zhang, Y., Ding, N., & Zhou, B. (2025). TTRL: Test-Time Reinforcement Learning. arXiv:2504.16084. doi:10.48550/arXiv.2504.16084. [PREPRINT]