BEST BEFORE POST-TRAINING? THE SHELF LIFE OF PRETRAINING MIXTURE RANKINGS

Anonymous authors
Paper under double-blind review

ABSTRACT

The pretraining data mixture strongly influences a language model's capabilities, yet prior research on data mixture optimization typically selects candidates before observing their post-training performance. This practice rests on the untested assumption that pretraining rankings survive post-training. In this paper, we conduct a thorough experimental study over 106 models of 1.5B parameters that differ only in their pretraining mixtures and share identical mid-training and supervised fine-tuning (SFT) data and configurations. Our results demonstrate that knowledge rankings change across these stages, while math rankings largely persist but are diluted by a fixed-weight composite of pretraining scores. Yet these scores retain useful information: pretraining GSM8K alone predicts post-SFT rankings better than the composite for knowledge, math, and code. We therefore introduce **MIXSELECTOR**, a ridge-regression method that uses a subset of post-trained candidates to learn how to weight the pretraining scores together with the mixture composition and then ranks the remaining candidates. On held-out candidates, its ranking agrees with the post-SFT ranking at a mean Spearman correlation of 0.72 across capability families, against 0.10 for the composite, while maintaining a smaller lead over the best single score. With only ten post-trained candidates, it retains most of this advantage. Fixed pretraining rankings thus have a limited shelf life, but recalibrating pretraining signals on a few post-trained models suffices to align mixture selection with post-training performance.

1 INTRODUCTION

The pretraining data mixture, the sampling proportion of each data domain, strongly shapes language model capabilities (Xie et al., 2023; Liu et al., 2025a; The Microsoft AI Team, 2026), yet candidate mixtures are typically selected using pretraining signals alone. Standard methods train small proxies on candidate mixtures, regress their pretraining losses or benchmark scores on the mixture weights, and select the mixture with the highest predicted score (Liu et al., 2025a; Ye et al., 2025; Chen et al., 2025; Shukor et al., 2025; Chen et al., 2026), while iterative searches use these scores to propose new mixtures (Diao et al., 2025). Open technical reports likewise choose mixtures from small-scale runs evaluated on base models (Grattafiori et al., 2024; Team OLMo et al., 2024; Liu et al., 2025b). Because post-training every candidate is prohibitively expensive (The Microsoft AI Team, 2026), typically only the top-ranked mixture proceeds to mid-training and supervised fine-tuning (SFT), where models are ultimately evaluated.

This workflow assumes that pretraining mixture rankings survive post-training. Even for base models, proxy rankings can depend on training hyperparameters (Wang et al., 2026) and reverse as a fixed-size model trains longer (The Microsoft AI Team, 2026), despite often transferring across scales (Magnusson et al., 2025; Chen et al., 2026). Mid-training and SFT further shift the data distribution, yet prior studies tracing pretraining choices through post-training vary model scale (Shen et al., 2026; Qi et al., 2025), compare few mixtures (Fan et al., 2026), alter other training settings (Maskey et al., 2026), or change more than the mixture (Zeng et al., 2025).

To isolate the effect of the pretraining mixture across post-training, we study 106 1.5B-parameter models that differ only in their pretraining mixtures and share identical mid-training (YaRN context extension; Peng et al., 2024) and SFT through the first SFT epoch (§3.1). We compare each candidate's pretraining evaluations with its post-SFT capability across four *capability families*: knowl-

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.

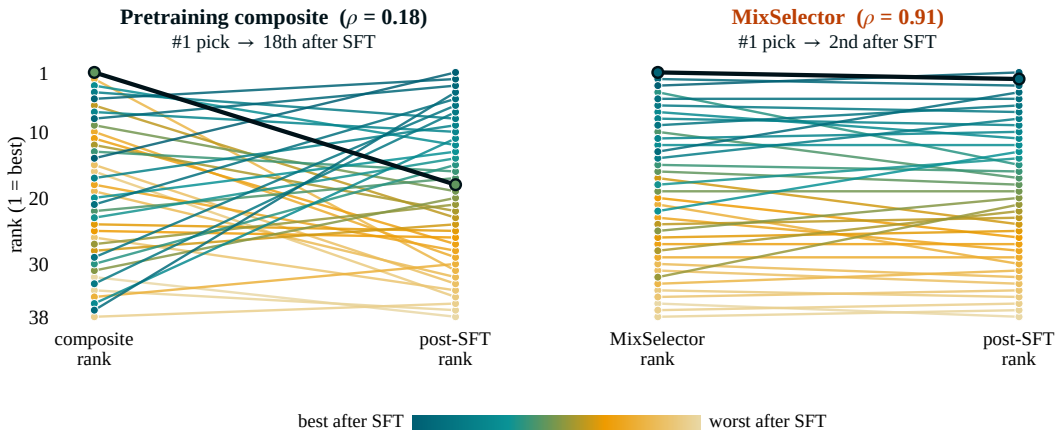


Figure 1: Selecting a pretraining mixture for post-SFT math capability on 38 held-out candidates. **Left:** the pretraining composite, a fixed-weight average of standardized pretraining benchmark scores, agrees with the post-SFT ranking at Spearman $\rho = 0.18$; its top pick ranks 18th after SFT. **Right:** MIXSELECTOR (ours), a ridge regression fit on a subset of 68 post-trained candidates, ranks the same candidates at $\rho = 0.91$; its top pick ranks 2nd. Colour indicates post-SFT rank (dark: best).

edge, math, code, and instruction following. To represent selection by average pretraining performance, we evaluate the *pretraining composite*, a fixed-weight average of standardized pretraining benchmark scores, alongside the individual pretraining metrics (Figure 1, left).

Our results show that pretraining rankings survive mid-training and SFT only in part, and committing to the composite's top candidate incurs substantial cost. Knowledge rankings shift across post-training even on the same benchmark ($\rho = 0.50$ on BBH), whereas math rankings largely persist ($\rho = 0.93$ on GSM8K) but are diluted by the fixed-weight composite (§3.2). Although the composite's top candidate beats random selection in expectation on every benchmark, on GSM8K it falls 19.0 points short of the best candidate, versus 20.6 points for a random choice. On code, even retaining its top ten candidates incurs a larger loss than retaining ten random candidates (§3.3).

Yet the signal needed to rank knowledge, math, and code capability is mis-weighted rather than missing (§3.4). Pretraining GSM8K alone predicts post-SFT rankings better than the composite across all three families, whereas the commonsense and open-QA groups receive half of the composite's weight despite correlating weakly with every post-SFT family and negatively with code. Beyond benchmark scores, the mixture composition—which the composite ignores—carries additional predictive signal for knowledge and code capability (§3.4).

We therefore introduce MIXSELECTOR, a ridge-regression method that fits family-specific weights over pretraining scores and domain shares on a small subset of post-trained candidates to rank the remaining candidates (§3.5; Figure 1, right). Obtaining each post-trained target requires only mid-training and one SFT epoch: in a 32-model cohort trained for five epochs, rankings after one epoch already closely track those after five on all benchmarks except HumanEval+ and IFEval (§3.2).

Fit on 68 post-trained candidates, the selector ranks 38 held-out candidates at a mean Spearman correlation of 0.72 across the four families, against 0.10 for the composite (0.80 versus 0.40 under cross-validation), with a smaller lead over the best single pretraining metric (0.59) and single-source ridges. With only ten post-trained candidates, its median correlation remains 0.63 and exceeds that of the composite in every draw, though no evaluated method ranks instruction following reliably on the held-out candidates. Recalibrating pretraining signals on a few post-trained models thus suffices to align mixture selection with post-training performance under a given SFT recipe.

Our contributions are threefold: (i) a controlled study over 106 1.5B-parameter models isolating the effect of the pretraining mixture across mid-training and SFT (§3.1); (ii) a selection-level analysis of ranking persistence, selection regret, and the predictive signal in pretraining scores and mixture composition (§3.2–§3.4); and (iii) MIXSELECTOR, evaluated on held-out candidates and across post-training budget sizes (§3.5).

2 Related Work

Mixture selection. Data-mixture methods use pretraining evidence to choose sampling proportions. They predict proxy-model performance from mixture weights (Liu et al., 2025a; Zhao et al., 2026; Diao et al., 2025; Chen et al., 2026), sometimes adding per-domain losses (Belenki et al., 2025); fit mixing laws across model size and token budget (Ye et al., 2025; Shukor et al., 2025); or learn domain weights through a proxy model or during training (Xie et al., 2023; Chen et al., 2025). Open technical reports likewise use small proxy, ablation, or annealing runs, evaluated primarily on base models (Grattafiori et al., 2024; Team OLMo et al., 2024; Allal et al., 2025; Yang et al., 2025; NVIDIA et al., 2025; Liu et al., 2025b; Ettinger et al., 2025). Some also instruction-tune selected mid-training mixtures (Ettinger et al., 2025), but doing so for every candidate is costly (The Microsoft AI Team, 2026). MixSelector follows the regression approach of RegMix while targeting post-SFT capability and using pretraining scores as well as domain shares. We evaluate this use of post-training feedback in the order candidates enter a search.

Transfer of mixture rankings. Small-model rankings often predict performance at larger scales (Magnusson et al., 2025; Khaddaj et al., 2025; Chen et al., 2026), but depend on proxy hyperparameters (Wang et al., 2026), token budget (Kang et al., 2025), and data repetition (Zhou et al., 2026). Two mixtures can also exchange order as a fixed-size model trains longer (The Microsoft AI Team, 2026). These studies evaluate base models. Our comparison holds model scale fixed and follows mixture rankings through a shared mid-training and SFT pipeline.

Pretraining quality and post-training outcomes. Improvements before fine-tuning need not persist afterwards. With pretraining data held fixed, weight decay (Han et al., 2026), sharpness (Watts et al., 2026), training duration (Springer et al., 2025), and learning-rate cooldown (Maskey et al., 2026) can affect the two stages differently. Data choices also matter: the preferred context-extension mixture can change after SFT (Gao et al., 2025), mid-training gains can emerge only after reinforcement learning (Wang et al., 2025), and pretraining data influence later preference and reinforcement learning (Zhao et al., 2025; Lu et al., 2026). Other studies compare nine mixtures (Fan et al., 2026), vary scale and stage budgets (Shen et al., 2026; Qi et al., 2025), or move reasoning data between stages (Akter et al., 2025). Our study isolates the mixture across 106 candidates of one search and measures the consequences for selection.

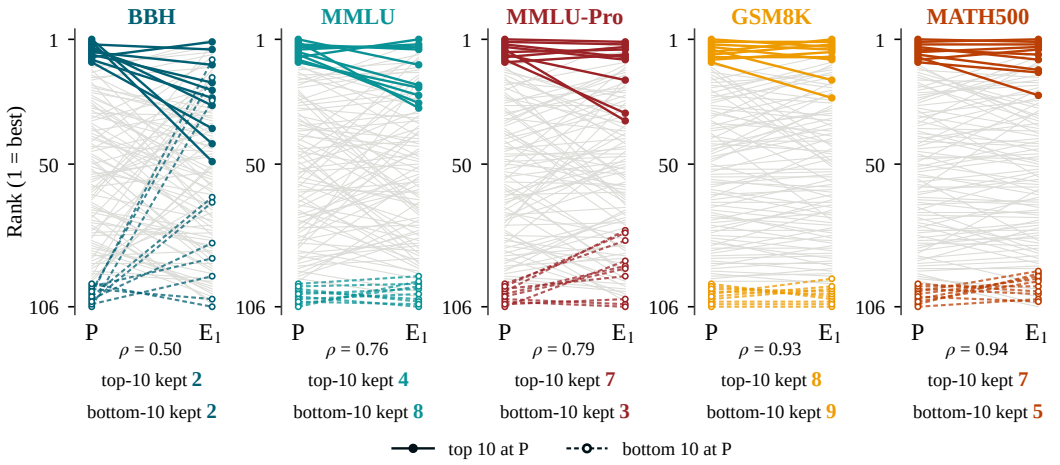
Predicting post-training performance. Model performance can be predicted from proxy metrics (Patel et al., 2026), benchmark subsets (Zeng & Papailiopoulos, 2026), smaller-model losses (Koh et al., 2026), or disclosed mixtures (Hamidieh et al., 2026). Studies of post-training examine ranking consistency under shared fine-tuning (Zhang et al., 2026), the limits of base-model knowledge scores (Li et al., 2026), and pretraining benchmarks whose rankings agree with post-training (Ling Team, 2025). Closest to our setting, Zeng et al. (2025) learn pairwise comparisons among fixed-size models with varied objectives or data, and aggregate them for model selection. We focus on mixture variation within an adaptive search, add composition as an input, and evaluate ranking agreement and selection regret on candidates proposed after those used for fitting.

3 Experiments

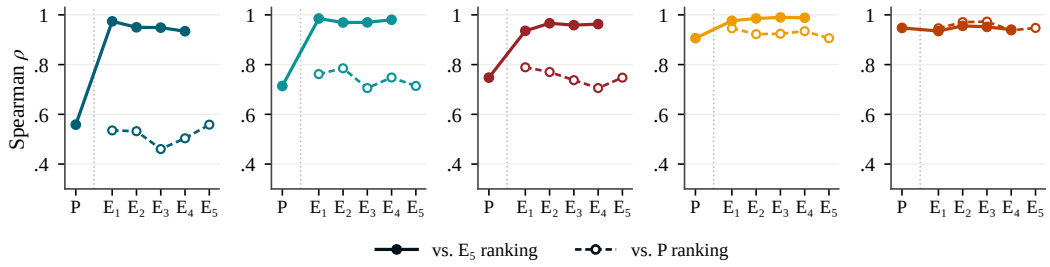
We first define our experimental setup and evaluation metrics (§3.1), then examine ranking persistence across post-training (§3.2), selection regret under the pretraining composite (§3.3), predictive signal in individual metrics and domain shares (§3.4), and learned selection with MixSelector (§3.5).

3.1 Experimental setup

Candidate mixtures. The 106 candidates are 1.5B-parameter Llama-architecture models from a RegMix search (Liu et al., 2025a) that share an identical architecture, tokenizer, learning-rate schedule, token budget, and final optimizer step (Figure 1), differing only in their sampling weights across 587 provenance bins aggregated into 8 data domains: web, code, math corpus, math reasoning traces, academic, synthetic QA, synthetic reasoning, and synthetic general (Appendix A). Following



(a) Rank of all 106 candidates at P and E_1



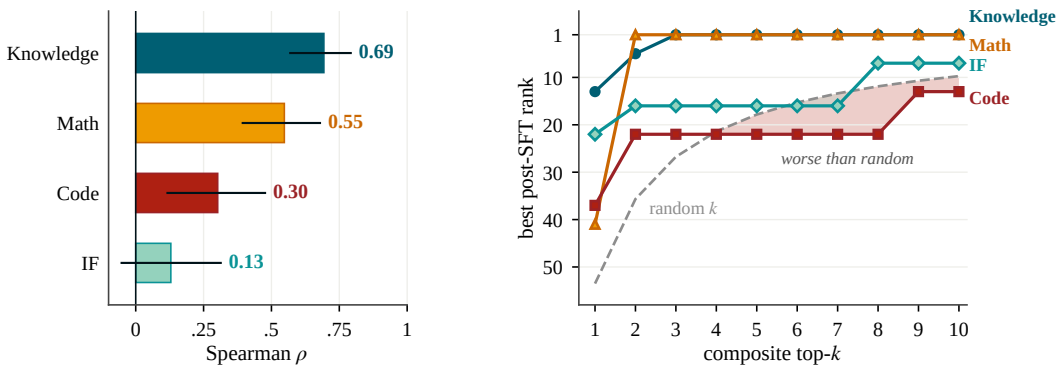
(b) Agreement of each stage's ranking with E_5 and with P (32-model cohort)

Figure 2: Same-benchmark rankings are more stable after the first SFT epoch. (a) Rankings of all 106 candidates at pretraining (P) and after one SFT epoch (E_1). Solid lines follow the pretraining top ten; dashed lines follow the bottom ten; colors identify benchmarks. Below each panel: Spearman ρ and top- and bottom-ten overlap. (b) In the 32-model cohort, agreement of each stage's ranking (E_t: after t SFT epochs) with the five-epoch ranking (E_5; solid) and the pretraining ranking (dashed). On all five benchmarks shown, rankings from E_1 onward agree with E_5 at $\rho \geq 0.93$.

the order in which candidates were proposed during the search, the population is partitioned into 68 training candidates and 38 held-out candidates (§3.5).

Shared post-training. Every candidate undergoes identical mid-training via YaRN context extension (Peng et al., 2024) from 8,192 to 40,960 tokens over 10,000 steps, followed by SFT on 347,360 examples from Mixture-of-Thoughts (Open-R1 Team, 2025) at a constant learning rate of 6×10^{-6} after 500 warmup steps (Appendix J). We denote the pretraining checkpoint by P and the checkpoint after t SFT epochs by E_t. All 106 candidates are trained to E_1, and a stratified cohort of 32 models selected using pretraining information alone continues through five epochs to E_5 (Appendix C). Because the learning rate is constant after warmup, the first epoch follows an identical trajectory across run lengths; throughout, *post-SFT* refers to E_1 (§3.2 verifies that E_1 rankings closely track E_5).

Evaluation. We evaluate candidates at each stage under its standard protocol (Appendix B). At P, base-model evaluation comprises 17 metrics partitioned into four *ability groups*: commonsense and multiple choice (10 likelihood-based metrics, including HellaSwag and PIQA; Zellers et al., 2019; Bisk et al., 2020), math (GSM8K, MATH500; Cobbe et al., 2021; Hendrycks et al., 2021b; Lightman et al., 2023), knowledge (MMLU, MMLU-Pro, BBH; Hendrycks et al., 2021a; Wang et al., 2024; Suzgun et al., 2023), and open QA (Natural Questions, TriviaQA; Kwiatkowski et al., 2019; Joshi et al., 2017). After SFT, checkpoints are evaluated zero-shot with the SFT chat template across eight benchmarks grouped into four *capability families*: knowledge (MMLU, MMLU-Pro, BBH), math (GSM8K, MATH500), code (HumanEval+, MBPP+; Chen et al., 2021; Austin et al.,



(a) Composite vs. SFT capability

(b) Best of the composite's top k

Figure 3: The pretraining composite as a ranking and a shortlist, over all 106 candidates at E_1 (after one SFT epoch; IF: instruction following). (a) Spearman correlation with each family score; intervals: 95% candidate bootstrap. (b) Post-SFT rank of the best candidate among the composite's top k (1: best). Dashed lines give the expected best rank of k uniformly random candidates, $(106+1)/(k+1)$; shading marks performance below this reference. The best of the top ten ranks 13th on code.

2021; Liu et al., 2023), and instruction following (IFEval strict prompt-level accuracy; Zhou et al., 2023).

Scores and measures. To represent selection by average pretraining performance, the *pretraining composite* Z standardizes each pretraining metric, averages within each ability group, and weights the four groups equally (Eq. (1)). Similarly, each post-SFT *family score* $F_{i,f}$ averages the standardized post-SFT benchmark scores of candidate i within family f. Standardization uses all 106 candidates in descriptive analyses (§3.2–§3.4) and only the 68 training candidates in held-out selection (§3.5). Given a selection signal σ with top-k set $L_k(\sigma)$ (ties broken by candidate name), we evaluate Spearman correlation ρ (average ranks for ties), top-k overlap, and selection regret on target t over candidate set $\mathcal{C}$, $\mathrm{Regret@}k(\sigma;t) = \max_{i\in\mathcal{C}} t_i - \max_{i\in L_k(\sigma)} t_i$ (in percentage points for a benchmark and z units for a family), compared against the exact expectation for k uniformly random candidates (Appendix F; cf. Wang et al., 2026). Unless stated otherwise, results cover all 106 candidates at E_1, with 95% percentile intervals from 2,000 shared candidate-bootstrap resamples (computed before rounding).

3.2 Persistence of Pretraining Rankings

On individual benchmarks evaluated before and after post-training, candidate rankings persist strongly in math but reorder substantially in knowledge (Figure 2a; Appendix E). Same-benchmark Spearman correlations from P to E_1 reach 0.93 on GSM8K and 0.94 on MATH500, whereas they drop to 0.76 on MMLU, 0.79 on MMLU-Pro, and 0.50 on BBH, yielding a mean correlation for knowledge that is 0.25 [0.18, 0.34] below that for math. This reordering directly impacts top-end selection: on BBH, only two of the pretraining top-ten candidates remain in the post-SFT top ten—compared with eight on GSM8K and seven on MATH500—and the pretraining BBH leader falls to 42nd after SFT.

The composite ranks math and code poorly. Even when rankings persist on individual benchmarks, aggregating pretraining scores into the fixed-weight composite Z substantially degrades ranking quality (Figure 3a; Table 8). Across all 106 candidates, Z correlates with post-SFT knowledge, math, code, and instruction-following capability at 0.69, 0.55, 0.30, and 0.13, respectively, and its top ten captures six, two, zero, and one of the post-SFT top ten, against a random expectation of 0.94. On the 38 held-out candidates, its correlations with math and code fall further to 0.18 and -0.07. A strong signal on an individual benchmark can therefore be diluted when averaged with other pretraining scores.

Agreement after the first SFT epoch. Within the 32-model cohort trained for five epochs, E_1 and E_5 rankings agree at $\rho \geq 0.93$ on six of the eight benchmarks, with lower agreement only on HumanEval+ (0.83) and IFEval (0.61; Figure 2b; Table 9). A single SFT epoch thus suffices to establish the multi-epoch ranking on most evaluated benchmarks, making E_1 an efficient target for mixture selection.

3.3 Selection costs of the pretraining composite

Committing to the single top candidate of the pretraining composite ($k = 1$) achieves lower regret than random selection in expectation on every benchmark, yet still incurs large losses on math and code (Table 10). Across the eight benchmarks, the composite's top pick finishes between 11th and 49th out of 106 models; on GSM8K, it trails the best candidate by 19.0 points, barely improving over the 20.6-point expected loss of a random choice. Expanding the selection budget to a top-k shortlist eliminates this loss on math at $k = 2$, but fails to recover code performance even at $k = 10$.

A narrow composite lead hides a large math gap. Comparing the composite's leader ($Z = 1.28$) and runner-up ($Z = 1.17$) illustrates how equal group weights induce large top-1 regret: after SFT, the leader scores 18.0 on GSM8K (rank 49), whereas the runner-up scores 36.4 (rank 2). During pretraining, the runner-up already leads the math group by $0.94\,z$ (2.22 versus 1.28), but trails by $1.17\,z$ in open QA (0.14 versus 1.31); equal group weighting allows the open-QA difference to outweigh the math signal and determine the overall winner. Retaining both candidates ($k = 2$) reduces GSM8K regret to 0.6 points and math-family regret to zero, while resampling pretraining benchmark items selects the same leader in 92.5% of replicates (Appendix F).

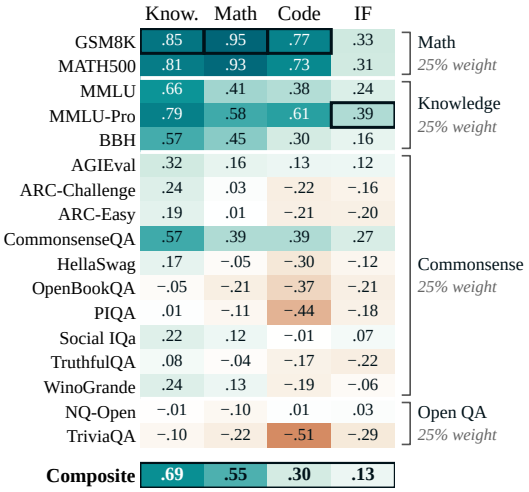


Figure 4: Individual pretraining metrics retain predictive information that the composite dilutes. Spearman correlation with each E_1 family (after one SFT epoch) over 106 candidates; boxes mark the best metric per family. Brackets show ability groups, each assigned 25% of the composite's weight. Know.: knowledge; IF: instruction following.

A larger shortlist does not resolve code selection. For post-SFT code capability, the best candidate among the composite's top two through eight ranks only 22nd of 106 (Figure 3b). From $k = 4$ onward, shortlisting by the composite underperforms uniform random selection, which achieves an expected best rank of 21.4 at $k = 4$. Even at $k = 10$, the best retained candidate ranks 13th (compared with 9.7 for random selection) and incurs a family regret of $0.60\,z$ versus $0.44\,z$. Increasing the shortlist to ten therefore cannot compensate for the composite's misaligned code ranking.

3.4 Predictive information before post-training

Evaluating the 17 pretraining metrics individually reveals that the signal needed to rank post-SFT knowledge, math, and code capability is mis-weighted by the composite rather than absent before post-training (Figure 4). Pretraining GSM8K alone correlates with post-SFT knowledge, math, and code capability at 0.85, 0.95, and 0.77, substantially outperforming the composite's 0.69, 0.55, and 0.30. Across all 22 pretraining signals—the 17 metrics, four group scores, and Z—the composite ranks only fifth to ninth on the four families, trailing the best single signal in every family by a paired difference whose 95% interval excludes zero (Table 11).

Equal group weights dilute useful signals. The composite's weak alignment stems from assigning half of its weight to the commonsense and open-QA groups, neither of which correlates above 0.29 with any post-SFT family and both of which correlate negatively with code (including -0.51 on

Table 1: Ranking the 38 held-out candidates with methods fit on the 68 training candidates: mean over the four E_1 families (after one SFT epoch). Features: D, the 8 domain shares; P, the 17 pretraining metrics. CV ρ: mean over 50 repeats of 5-fold cross-validation on all 106 candidates. Bold: best per column.

Method	Features	$\rho \uparrow$	Top-10↑	Regret@1 ↓	Regret@3 ↓	CV $\rho \uparrow$
Random	–	0.00	2.63	1.61	0.89	0.00
Pretraining composite	P, fixed weights	0.10	3.00	1.43	1.17	0.40
Best single metric	one of P	0.59	5.25	1.14	0.34	0.72
Ridge, D only	D	0.70	5.75	**0.26**	**0.21**	0.77
Ridge, P only	P	0.68	6.00	0.52	0.32	0.76
MIXSELECTOR	D + P	**0.72**	**6.25**	0.77	0.36	**0.80**

TriviaQA and -0.44 on PIQA). Task names also need not identify the most informative pretraining proxy: post-SFT BBH correlates much more strongly with the pretraining math group (0.91) than with pretraining BBH itself (0.50), whereas for post-SFT MMLU, pretraining MMLU-Pro (0.81) and MMLU (0.76) perform similarly (Appendix G).

Composition adds information for knowledge and code capability. Beyond benchmark scores, the mixture's domain shares (D) carry predictive information that pretraining metrics (P) do not capture (Figure 5; Table 14). In cross-validated ridge regressions fit on P, D, or both, domain shares alone predict post-SFT code capability more accurately than the 17 pretraining metrics ($R^2 = 0.81$ versus 0.73), and combining both sources raises R^2 over P alone by $R^2_{\text{P+D}} - R^2_{\text{P}} = 0.11$ for code (with a 95% pooled out-of-bag lower bound of 0.01 across 2,000 draws over ten seeds; Appendix H) and 0.09 for knowledge. For math, most explained variance is shared between P and D ($R^2 = 0.87$), leaving a small unique increment from D (0.02 [-0.02, 0.06]), while for instruction following neither source exceeds $R^2 = 0.18$.

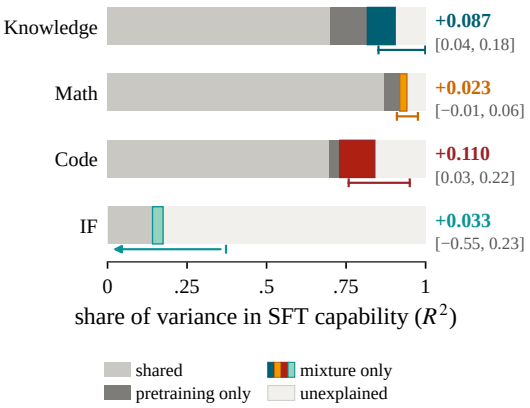
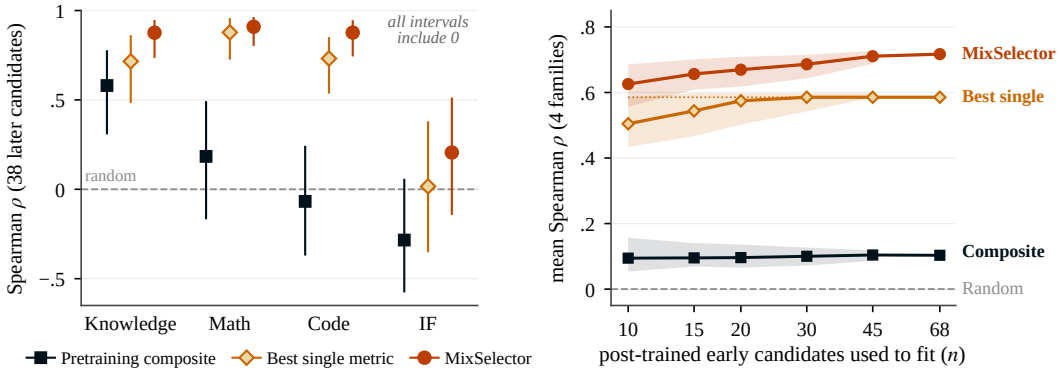


Figure 5: Predictive information in pretraining metrics and domain shares. Cross-validated R^2 for each E_1 family (after one SFT epoch) is decomposed into shared, source-specific, and unexplained components. Right: the increment from domain shares with the original single-seed 95% bootstrap interval; pooled intervals appear in Table 14. IF: instruction following.

Domain associations conditional on pretraining scores. Correlating domain shares with the residuals of a cross-fitted ridge model on the 17 pretraining metrics reveals why composition adds predictive value (Table 15). Higher code share and lower web share are associated with higher residual post-SFT code capability (Spearman 0.42 [0.25, 0.56] and -0.31 [-0.48, -0.13]). Conversely, math-corpus share correlates positively with pretraining MMLU (0.44) and the composite (0.39), but negatively with residual post-SFT knowledge (-0.37 [-0.54, -0.19]) and math capability (-0.32 [-0.49, -0.13]). All four residual associations pass Benjamini–Hochberg correction (Benjamini & Hochberg, 1995) at $q < 0.05$ across the 32 domain–family pairs. Together, these findings motivate learning a selector that combines pretraining metrics and mixture composition using post-SFT supervision.

3.5 LEARNING TO RANK WITH MIXSELECTOR

To align mixture selection with post-training outcomes, MIXSELECTOR fits one ridge regression per capability family on a subset of post-trained candidates, combining domain shares and pretraining metrics to rank the remaining candidates in the search pool. Fit on the 68 training candidates, the selector ranks the 38 held-out candidates at a mean Spearman correlation of 0.72 across the four



(a) Ranking the 38 later candidates

(b) How many post-trained candidates it needs

Figure 6: Ranking the 38 held-out candidates using pretraining evidence and post-SFT outcomes from training candidates (IF: instruction following). (a) Spearman correlation per family, with 95% test-bootstrap intervals. (b) Mean correlation across families versus the number n of post-trained training candidates used for fitting. Lines and bands show medians and interquartile ranges over 200 subset draws; each learned method, including the choice of a single metric, is refit per draw. At $n = 68$, results are from the single main fit. The dotted line fixes the single metric chosen on all 68 candidates. The composite uses no post-SFT labels; its band reflects changes in pretraining z-scaling.

capability families, compared with 0.10 for the pretraining composite (paired difference 0.61 [0.40, 0.82]; Table 1).

Fitting and evaluation. For capability family f, the selector predicts the post-SFT family score $F_{i,f}$ from all 25 standardized pretraining features via a linear predictor $\hat{F}_{i,f} = w_f^\top [d_i; x_i]$, where d_i contains the 8 domain shares and x_i contains the 17 pretraining benchmark metrics. Target and feature standardization, leave-one-out generalized cross-validation for the ℓ_2 ridge penalty, and weight estimation use only the 68 training candidates (Appendix I.1). Unlike the pretraining composite, which applies a single fixed-weight ranking across all downstream evaluations, the selector learns a distinct weight vector w_f tailored to each target capability family. We compare the selector against uniform random selection, the pretraining composite, the single pretraining metric with the highest training-set correlation per family (GSM8K for knowledge, math, and code; MMLU-Pro for instruction following), and ridge regression ablations fit using domain shares (D) or pretraining metrics (P) alone. Because the 38 held-out candidates contain eight of the ten best post-SFT knowledge models and seven of the ten best code models across the entire population, this evaluation split directly tests whether post-training feedback from earlier candidates can identify the strongest mixtures among subsequent candidates.

Ranking agreement and selection regret. On the 38 held-out candidates, the selector improves every aggregate ranking and selection measure in Table 1 over the pretraining composite. Mean top-pick selection regret ($\mathrm{Regret@1}$) across the four families falls from 1.43 to 0.77 z (paired difference -0.67 [-1.24, -0.48]; Appendix I.3), while top-10 overlap more than doubles from 3.00 to 6.25 candidates. Measured in benchmark percentage points, selecting the top candidate of MIXSELECTOR reduces the average post-SFT performance loss from 11.3 to 2.0 points across GSM8K and MATH500 (compared with 12.1 points for random selection) and from 6.5 to 2.4 points across HumanEval+ and MBPP+ (compared with 7.7 points for random selection). Evaluated per capability family, the selector ranks held-out knowledge, math, and code capability at Spearman correlations of 0.88, 0.91, and 0.88, respectively (Figure 6a). Relative to selecting by the best single pretraining metric per family, combining all features in MIXSELECTOR improves mean held-out correlation from 0.59 to 0.72 (paired difference 0.13 [0.05, 0.23]) and reduces $\mathrm{Regret@1}$ from 1.14 to 0.77 z, while achieving comparable shortlist regret at $k = 3$ (0.36 versus 0.34 z).

Both input sources are informative. Ridge regressions fit on domain shares alone (D only) or pretraining metrics alone (P only) already recover strong held-out rankings, reaching mean Spearman correlations of 0.70 and 0.68, respectively, while combining both feature sources in MIXSELECTOR reaches 0.72. Across 50 repeats of five-fold cross-validation over all 106 candidates, the combined model achieves a higher mean correlation than either single-source ablation in all 50 repeats (0.80, versus 0.77 for D only and 0.76 for P only; Appendix I.3). On the held-out split, the D-only ridge attains a lower mean $\mathrm{Regret@1}$ ($0.26\,z$ versus $0.77\,z$), a difference driven entirely by the noisy instruction-following family ($0.55\,z$ versus $2.58\,z$).

Learning from fewer post-trained candidates. To quantify how many post-trained models are needed to calibrate selection in practice, we refit every method on random subsets of $n \in \{10, 15, 20, 30, 45\}$ training candidates (200 random draws per budget n) and evaluate on the same 38 held-out candidates (Figure 6b; Appendix I.5). With a budget of only $n = 10$ post-trained candidates, the selector already reaches a median family-mean correlation of 0.63 (interquartile range 0.56–0.69), rising smoothly to 0.71 at $n = 45$. At $n = 10$, MIXSELECTOR outperforms the pretraining composite in 100% of subset draws, matches or exceeds the refitted single-metric baseline in 87.0% of draws, and improves median correlation over the D-only and P-only ridge models by 0.06 and 0.08, respectively. Because obtaining each post-trained target requires only mid-training and one SFT epoch (roughly 32 H100 GPU-hours per model for SFT; Appendix J), post-training just ten candidates already suffices to calibrate mixture selection across the remaining pool.

Instruction following remains unresolved. On the 38 held-out candidates, no evaluated method ranks instruction following with a 95% confidence interval that excludes zero: the selector reaches a Spearman correlation of 0.21 and the highest point estimate among all methods, D only, reaches 0.32 (Table 16). Instruction following also accounts for nearly all of the selector's mean top-pick regret ($2.58\,z$, compared with 0.02–$0.31\,z$ across knowledge, math, and code, and $2.07\,z$ for uniform random selection). Collecting post-training feedback on a small subset of candidates thus reliably aligns mixture selection with post-SFT knowledge, math, and code capability, while predicting instruction-following rankings remains an open challenge.

4 CONCLUSION

Pretraining mixture rankings are useful for model development only insofar as they predict capabilities after post-training. Across 106 1.5B-parameter models sharing an identical mid-training and SFT pipeline, we find that pretraining benchmark rankings are capability-specific and cannot be summarized by a fixed heuristic composite: knowledge rankings invert across post-training, code rankings are near noise at the pretraining stage, and the math signal that persists on individual benchmarks is diluted when aggregated into a uniform pretraining score. Because complementary predictive signal remains in individual pretraining benchmark scores and in the domain composition itself, MIXSELECTOR fits family-specific ridge regressions over both feature blocks on a small subset of post-trained candidates. On 38 held-out candidates, MIXSELECTOR raises mean Spearman correlation across the four capability families from 0.10 for the pretraining composite to 0.72, reduces top-1 selection regret across knowledge, math, and code, and reaches a median correlation of 0.63 when fit on just ten candidates post-trained for a single SFT epoch. Fixed pretraining rankings thus have a limited shelf life, but recalibrating pretraining signals on a small pilot of post-trained models suffices to align mixture selection with downstream post-SFT performance.

Limitations and future work. Our controlled population holds model scale (1.5B parameters), architecture, token horizon (30B tokens), and the subsequent mid-training and SFT recipe fixed in order to isolate the causal effect of the pretraining mixture; testing how learned selectors transfer across model scales, alternative SFT datasets and hyperparameter recipes, and downstream reinforcement-learning stages (Lu et al., 2026) is an important next step. Moreover, while post-training supervision reliably calibrates mixture selection for knowledge, math, and code, no evaluated method ranks instruction following reliably on held-out candidates, motivating richer pretraining probes and mixture descriptors for alignment-sensitive capabilities. Finally, moving from scoring a fixed pool of candidate mixtures to integrating MIXSELECTOR directly into an active mixture-design loop is a promising direction for future work.

AI USE STATEMENT

We used large-language-model assistants, primarily Claude (Anthropic), slightly in this work for checking related work and to aid or polish writing.

Checking and Writing. The assistants polished the writing of the text, including the current version of the related work section; searched for and summarized related literature; suggested the structure of the paper; and formatted references.

We reviewed and verified all AI-assisted work and take full responsibility for the content of this paper, including text, code, figures and results produced with the aid of AI tools.

ETHICS STATEMENT

This study does not recruit human participants, collect new personal data, or commission annotations. It evaluates models pretrained on mixtures of 587 provenance bins grouped into 8 domains (Appendix A), fine-tuned on a public SFT corpus (Open-R1 Team, 2025), and tested on public benchmarks used under their licenses. The contribution concerns mixture selection and comparative evaluation. Better selection accuracy does not by itself establish the safety, fairness, or suitability of the resulting models for deployment. Its main cost is compute: SFT took about 32.4 H100 GPU-hours per model to E_1 and 162.3 per model to E_5 (median training time on eight GPUs), about 7,600 GPU-hours for the 106-model population. Pretraining, mid-training and evaluation compute are not included in these figures. Fitting MIXSELECTOR itself costs a negligible amount of compute. We report this so that the cost of the measurement can be weighed against the selection decisions it informs.

REPRODUCIBILITY STATEMENT

Section 3.1 defines the candidates, training pipeline, evaluation, and selection measures. Appendices A–D detail the domain mapping, evaluation protocols, cohort selection, and measurement reliability. Appendices E–I provide the full results and selector implementation, and Appendix J describes the training pipeline and compute settings. We will release the analysis code and protocol, the frozen snapshot of per-item evaluations, the exact mixture weights, and the scripts that produce the reported tables and statistics upon publication.

REFERENCES

Syeda Nahida Akter, Shrimai Prabhumoye, Eric Nyberg, Mostofa Patwary, Mohammad Shoeybi, Yejin Choi, and Bryan Catanzaro. Front-loading reasoning: The synergy between pretraining and post-training data. *arXiv preprint arXiv:2510.03264*, 2025.

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Agustín Piqueres Lajárín, Hynek Kydlíček, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan-Son Nguyen, Ben Burtenshaw, Clémentine Fourrier, Haojun Zhao, Hugo Larcher, Mathieu Morlon, Cyril Zakka, Colin Raffel, Leandro von Werra, and Thomas Wolf. SmolLM2: When smol goes big – data-centric training of a fully open small language model. In *COLM*, 2025. arXiv:2502.02737.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

Lior Belenki, Alekh Agarwal, Tianze Shi, and Kristina Toutanova. Optimizing pre-training data mixtures with mixtures of data expert models. In *ACL*, pp. 32570–32587, 2025. arXiv:2502.15950.

Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(1):289–300, 1995. doi: 10.1111/j.2517-6161.1995.tb02031.x.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about physical commonsense in natural language. In *AAAI*, 2020. arXiv:1911.11641.

William Brown. Some experimental results in the correlation of mental abilities. *British Journal of Psychology*, 3(3):296–322, 1910. doi: 10.1111/j.2044-8295.1910.tb00207.x.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Mayee F. Chen, Michael Y. Hu, Nicholas Lourie, Kyunghyun Cho, and Christopher Ré. Aioli: A unified optimization framework for language model data mixing. In *ICLR*, 2025. arXiv:2411.05735.

Mayee F. Chen, Tyler Murray, David Heineman, Matt Jordan, Hannaneh Hajishirzi, Christopher Ré, Luca Soldaini, and Kyle Lo. Olmix: A framework for data mixing throughout LM development. *arXiv preprint arXiv:2602.12237*, 2026.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Shizhe Diao, Yu Yang, Yonggan Fu, Xin Dong, Dan Su, Markus Kliegl, Zijia Chen, Peter Belcak, Yoshi Suhara, Hongxu Yin, Mostofa Patwary, Yingyan Celine Lin, Jan Kautz, and Pavlo Molchanov. Nemotron-CLIMB: Clustering-based iterative data mixture bootstrapping for language model pre-training. In *NeurIPS*, 2025. arXiv:2504.13161.

Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, Jacob Morrison, Jake Poznanski, Kyle Lo, Luca Soldaini, Matt Jordan, Mayee Chen, Michael Noukhovitch, Nathan Lambert, Pete Walsh, Pradeep Dasigi, Robert Berry, Saumya Malik, Saurabh Shah, Scott Geng, Shane Arora, Shashank Gupta, Taira Anderson, Teng Xiao, Tyler Murray, Tyler Romero, Victoria Graf, Akari Asai, Akshita Bhagia, Alexander Wettig, Alisa Liu, Aman Rangapur, Chloe Anastasiades, Costa Huang, Dustin Schwenk, Harsh Trivedi, Ian Magnusson, Jaron Lochner, Jiacheng Liu, Lester James V. Miranda, Maarten Sap, Malia Morgan, Michael Schmitz, Michal Guerquin, Michael Wilson, Regan Huff, Ronan Le Bras, Rui Xin, Rulin Shao, Sam Skjonsberg, Shannon Zejiang Shen, Shuyue Stella Li, Tucker Wilde, Valentina Pyatkin, Will Merrill, Yapei Chang, Yuling Gu, Zhiyuan Zeng, Ashish Sabharwal, Luke Zettlemoyer, Pang Wei Koh, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. Olmo 3. *arXiv preprint arXiv:2512.13961*, 2025.

Simin Fan, Dimitris Paparas, Natasha Noy, Binbin Xiong, Noveen Sachdeva, and Berivan Isik. The magic correlations: Understanding knowledge transfer from pretraining to supervised fine-tuning. *arXiv preprint arXiv:2602.11217*, 2026.

Tianyu Gao, Alexander Wettig, Howard Yen, and Danqi Chen. How to train long-context language models (effectively). In *ACL*, 2025. arXiv:2410.02660.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Kimia Hamidieh, Lester Mackey, and David Alvarez-Melis. Domain-aware scaling laws uncover data synergy. *arXiv preprint arXiv:2607.11052*, 2026.

Tessa Han, Sebastian Bordt, Hanlin Zhang, and Sham Kakade. Weight decay improves language model plasticity. In *ICML*, 2026. arXiv:2602.11137.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *ICLR*, 2021a. arXiv:2009.03300.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021b.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *ACL*, 2017. arXiv:1705.03551.

Feiyang Kang, Yifan Sun, Bingbing Wen, Si Chen, Dawn Song, Rafid Mahmood, and Ruoxi Jia. AutoScale: Scale-aware data mixing for pre-training LLMs. In *COLM*, 2025. arXiv:2407.20177.

Alaa Khaddaj, Logan Engstrom, and Aleksander Madry. Small-to-large generalization: Training data influences models consistently across scale. In *ICLR*, 2025. arXiv:2505.16260.

Woosung Koh, Juyoung Suk, Sungjun Han, Se-Young Yun, and Jamin Shin. Predicting llm reasoning performance with small proxy model. In *ICLR*, 2026. arXiv:2509.21013.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019.

Xiaoyuan Li, Yubo Ma, Kexin Yang, Moxin Li, Keqin Bao, Wenjie Wang, Fuli Feng, and Dayiheng Liu. On predicting the post-training potential of pre-trained LLMs. *arXiv preprint arXiv:2605.11978*, 2026.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023.

Ling Team. Every activation boosted: Scaling general reasoner to 1 trillion open language foundation. *arXiv preprint arXiv:2510.22115*, 2025.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. In *NeurIPS*, 2023. arXiv:2305.01210.

Qian Liu, Xiaosen Zheng, Niklas Muennighoff, Guangtao Zeng, Longxu Dou, Tianyu Pang, Jing Jiang, and Min Lin. Regmix: Data mixture as regression for language model pre-training. In *ICLR*, 2025a. arXiv:2407.01492.

Zhengzhong Liu, Liping Tang, Linghao Jin, Haonan Li, Nikhil Ranjan, Desai Fan, Shaurya Rohatgi, Richard Fan, Omkar Pangarkar, Huijuan Wang, Zhoujun Cheng, Suqi Sun, Seungwook Han, Bowen Tan, Gurpreet Gosal, Xudong Han, Varad Pimpalkhute, Shibo Hao, Ming Shan Hee, Joel Hestness, Haolong Jia, Liqun Ma, Aaryamonvikram Singh, Daria Soboleva, Natalia Vassilieva, Renxi Wang, Yingquan Wu, Yuekai Sun, Taylor Killian, Alexander Moreno, John Maggs, Hector Ren, Guowei He, Hongyi Wang, Xuezhe Ma, Yuqi Wang, Mikhail Yurochkin, and Eric P. Xing. K2-v2: A 360-open, reasoning-enhanced llm. *arXiv preprint arXiv:2512.06201*, 2025b.

Cen Lu, Yung-Chen Tang, and Andrea Cavallaro. Similar models learn differently: Final-window pretraining shapes post-training beyond sft. *arXiv preprint arXiv:2607.25063*, 2026.

Ian Magnusson, Nguyen Tai, Ben Bogin, David Heineman, Jena D. Hwang, Luca Soldaini, Akshita Bhagia, Jiacheng Liu, Dirk Groeneveld, Oyvind Tafjord, Noah A. Smith, Pang Wei Koh, and Jesse Dodge. Datadecide: How to predict best pretraining data with small experiments. In *ICML*, 2025. arXiv:2504.11393.

Sohir Maskey, Philipp Scholl, Jonas Knupp, Pit Neitemeier, and Sascha Wirges. Good pretraining, bad SFT: Checkpoint quality across the training stack. *arXiv preprint arXiv:2609.08966*, 2026.

Kim F. Nimon and Frederick L. Oswald. Understanding the results of multiple linear regression: Beyond standardized regression coefficients. *Organizational Research Methods*, 16(4):650–674, 2013. doi: 10.1177/1094428113493929.

NVIDIA, Aarti Basant, Abhijit Khairnar, Abhijit Paithankar, Abhinav Khattar, Adithya Renduchintala, Aditya Malte, Akhiad Bercovich, et al. NVIDIA Nemotron Nano 2: An accurate and efficient hybrid Mamba-transformer reasoning model. *arXiv preprint arXiv:2508.14444*, 2025.

Open-R1 Team. Mixture-of-thoughts. `https://huggingface.co/datasets/open-r1/Mixture-of-Thoughts`, 2025. Dataset.

Arkil Patel, Siva Reddy, Marius Mosbach, and Dzmitry Bahdanau. Forecasting downstream performance of llms with proxy metrics. *arXiv preprint arXiv:2605.18607*, 2026.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. Yarn: Efficient context window extension of large language models. In *ICLR*, 2024. arXiv:2309.00071.

Zhenting Qi, Fan Nie, Alexandre Alahi, James Zou, Himabindu Lakkaraju, Yilun Du, Eric Xing, Sham Kakade, and Hanlin Zhang. Evolm: In search of lost language model training dynamics. In *NeurIPS*, 2025. arXiv:2506.16029.

Jingyan Shen, Ang Li, Salman Rahman, Yifan Sun, Micah Goldblum, Matus Telgarsky, and Pavel Izmailov. Understanding reasoning from pretraining to post-training. *arXiv preprint arXiv:2607.16097*, 2026.

Mustafa Shukor, Louis Béthune, Dan Busbridge, David Grangier, Enrico Fini, Alaaeldin El-Nouby, and Pierre Ablin. Scaling laws for optimal data mixtures. In *NeurIPS*, 2025. arXiv:2507.09404.

Charles Spearman. Correlation calculated from faulty data. *British Journal of Psychology*, 3(3): 271–295, 1910. doi: 10.1111/j.2044-8295.1910.tb00206.x.

Jacob Mitchell Springer, Sachin Goyal, Kaiyue Wen, Tanishq Kumar, Xiang Yue, Sadhika Malladi, Graham Neubig, and Aditi Raghunathan. Overtrained language models are harder to fine-tune. In *ICML*, 2025. arXiv:2503.19206.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. Challenging big-bench tasks and whether chain-of-thought can solve them. In *ACL Findings*, 2023. arXiv:2210.09261.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, Allyson Ettinger, Michal Guerquin, David Heineman, Hamish Ivison, Pang Wei Koh, Jiacheng Liu, Saumya Malik, William Merrill, Lester James V. Miranda, Jacob Morrison, Tyler Murray, Crystal Nam, Jake Poznanski, Valentina Pyatkin, Aman Rangapur, Michael Schmitz, Sam Skjonsberg, David Wadden, Christopher Wilhelm, Michael Wilson, Luke Zettlemoyer, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. 2 OLMo 2 furious. *arXiv preprint arXiv:2501.00656*, 2024.

The Microsoft AI Team. MAI-Thinking-1: Building a hill-climbing machine. Technical report, Microsoft AI, 2026. URL `https://microsoft.ai/pdf/mai-thinking-1.pdf`.

Jiachen T. Wang, Tong Wu, Kaifeng Lyu, James Zou, Dawn Song, Ruoxi Jia, and Prateek Mittal. Can small training runs reliably guide data curation? rethinking proxy-model practice. In *ICLR*, 2026. arXiv:2512.24503.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. In *NeurIPS Datasets and Benchmarks*, 2024. arXiv:2406.01574.

Zengzhi Wang, Fan Zhou, Xuefeng Li, and Pengfei Liu. Octothinker: Mid-training incentivizes reinforcement learning scaling. *arXiv preprint arXiv:2506.20512*, 2025.

Ishaan Watts, Catherine Li, Sachin Goyal, Jacob Mitchell Springer, and Aditi Raghunathan. Sharpness-aware pretraining mitigates catastrophic forgetting. In *ICML*, 2026. arXiv:2605.02105.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V. Le, Tengyu Ma, and Adams Wei Yu. Doremi: Optimizing data mixtures speeds up language model pretraining. In *NeurIPS*, 2023. arXiv:2305.10429.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Jiasheng Ye, Peiju Liu, Tianxiang Sun, Jun Zhan, Yunhua Zhou, and Xipeng Qiu. Data mixing laws: Optimizing data mixtures by predicting language modeling performance. In *ICLR*, 2025. arXiv:2403.16952.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *ACL*, 2019. arXiv:1905.07830.

Hansi Zeng, Kai Hui, Honglei Zhuang, Zhen Qin, Zhenrui Yue, Hamed Zamani, and Dana Alon. Can pre-training indicators reliably predict fine-tuning outcomes of llms? *arXiv preprint arXiv:2504.12491*, 2025.

Yuchen Zeng and Dimitris Papailiopoulos. You don't need to run every eval. *arXiv preprint arXiv:2606.24020*, 2026.

Guanhua Zhang, Ricardo Dominguez-Olmedo, and Moritz Hardt. Train-before-test harmonizes language model rankings. In *ICLR*, 2026. arXiv:2507.05195.

Kaiyan Zhao, Zhongtao Miao, Akiko Aizawa, and Yoshimasa Tsuruoka. Regmix-d: Dynamic data mixing via proxy training trajectories. *arXiv preprint arXiv:2606.18663*, 2026.

Rosie Zhao, Alexandru Meterez, Sham Kakade, Cengiz Pehlevan, Samy Jelassi, and Eran Malach. Echo chamber: Rl post-training amplifies behaviors learned in pretraining. In *COLM*, 2025. arXiv:2504.07912.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

Kevin Zhou, Lisa Alazraki, Kris Cao, and Marek Rei. Repetition mismatch: Why data mixture experiments don't scale and how to fix them. In *EMNLP*, 2026. arXiv:2606.07597.

A CANDIDATE MIXTURES

The 106 candidates are the checkpoints of one multi-round RegMix search (Liu et al., 2025a). All are 1.5B-parameter Llama-architecture decoders with an 8,192-token context, trained with the same tokenizer, schedule and token budget and taken at the same optimizer step (71,525). The search proposed 68 candidates in its first round, 34 in its second and 4 in a final round; the first 68 are the *early* and the other 38 the *later* candidates of §3.1. Each round built on the evaluations of the previous ones to propose new candidate mixtures.

Each candidate has recorded sampling weights over 587 provenance bins: 576 web bins defined by 24 topics and 24 formats, and 11 additional source bins. We aggregate these into the 8 domains in Table 2. Because the shares sum to one, increasing one requires decreasing at least one other. Variation is largest in web, code, and synthetic QA, and most domains are absent from at least some mixtures. Later candidates allocate less weight to web data and more to synthetic general data, math reasoning traces, and synthetic reasoning. The search varies shares jointly: web and academic shares correlate at 0.62, while web and code shares correlate at -0.45.

Table 2: **The 8 data domains.** Bins: provenance bins summed into the domain (web: 24 topics $\times$ 24 formats; academic: papers, legal text and long documents; synthetic reasoning: reasoning and planning data; synthetic general: general and AI-topic data). Share statistics are over the 106 candidates, in percent of the mixture; $<0.1\%$: number of candidates whose share is below 0.1%. Early and later: mean share over the 68 early and the 38 later candidates.

Domain	Bins	Mean	SD	Min	Max	$<0.1\%$	Early	Later
Web	576	26.9	13.7	10.0	66.4	0	29.4	22.6
Code	1	17.9	16.1	0.0	72.3	10	18.1	17.5
Math corpus	1	3.7	3.4	0.0	8.4	27	3.7	3.7
Math reasoning traces	1	9.6	8.4	0.0	22.4	17	8.2	11.9
Academic	3	3.5	3.8	0.0	10.4	21	3.9	2.8
Synthetic QA	1	18.0	17.8	0.0	65.4	18	18.6	16.9
Synthetic reasoning	2	8.2	5.8	0.0	18.8	7	7.3	9.8
Synthetic general	2	12.2	7.9	0.0	26.8	6	10.7	14.9

B EVALUATION PROTOCOLS

Pretraining metrics. We evaluate base models at P across the 17 metrics in Table 3, computing each metric from the frozen per-item results. Metrics average over items (500 problems with 64 samples per problem for MATH500; 6,511 items across subtasks for BBH); AGIEval first averages within subtasks and then across subtasks. The five chain-of-thought tasks use few-shot prompts and sampled responses, with pass@1 estimated by mean correctness across samples. The remaining tasks use likelihood-based scoring or greedy exact match, as listed in the table.

Two additional base-model evaluations are excluded from the 17-metric suite. DROP F1 averages 1.7%, with between-candidate standard deviation 0.16 pp; under the same zero-shot greedy protocol, reference 1B and 4B base models score 1.9% and 13.5%, respectively, placing the candidate scores near the protocol floor. Pretraining IFEval, evaluated with one sampled response per prompt, has split-half reliability 0.32 (Appendix D).

Composite and family scores. With $x_{i,m}$ the score of candidate i on pretraining metric m, $\mathcal{M}_g$ the metrics of ability group g, $Y_{i,b}$ its score (%) on post-SFT benchmark b, and $\mathcal{B}_f$ the benchmarks of capability family f, the pretraining composite and the family scores of §3.1 are

$$Z_i = \frac{1}{4}\sum_{g=1}^{4}\frac{1}{|\mathcal{M}_g|}\sum_{m\in\mathcal{M}_g}\frac{x_{i,m}-\bar{x}_m}{s_m}, \qquad F_{i,f} = \frac{1}{|\mathcal{B}_f|}\sum_{b\in\mathcal{B}_f}\frac{Y_{i,b}-\bar{Y}_b}{s_b}, \tag{1}$$

where bars and s denote means and population standard deviations over the 106 candidates (over the training candidates in §3.5). The composite weights the four ability groups equally, so a single metric has weight $1/40$ in the commonsense and multiple-choice group, $1/8$ in the math and open-QA groups and $1/12$ in the knowledge group. The commonsense and open-QA groups together carry half of the composite.

Table 3: **The 17 pretraining metrics**, grouped into the four ability groups of Z. Samples: generations per item (pass@1 is the mean correctness over them); LL: likelihood-based scoring (MC2 for TruthfulQA); EM: greedy exact match. Mean and SD: over the 106 candidates (%, population SD). Rel.: item split-half reliability (Appendix D).

Metric	Protocol	Samples	Items	Mean (SD)	Rel.
Commonsense and multiple choice					
AGIEval	0-shot LL, macro-average over subtasks	1	7,150	26.5 (0.6)	0.46
ARC-Challenge	25-shot LL	1	1,172	37.3 (2.7)	0.92
ARC-Easy	0-shot LL	1	2,376	61.3 (3.5)	0.97
CommonsenseQA	0-shot LL	1	1,221	24.1 (3.4)	0.84
HellaSwag	0-shot LL	1	10,042	56.0 (2.6)	0.99
OpenBookQA	0-shot LL	1	500	35.0 (1.6)	0.45
PIQA	0-shot LL	1	1,838	73.0 (1.2)	0.81
Social IQa	0-shot LL	1	1,954	44.7 (0.7)	0.45
TruthfulQA (MC2)	0-shot LL	1	817	26.3 (2.4)	0.89
WinoGrande	5-shot LL	1	1,267	57.0 (1.4)	0.55
Math					
GSM8K	few-shot CoT, pass@1	16	1,319	14.8 (6.9)	1.00
MATH500	4-shot CoT, pass@1	64	500	6.3 (2.0)	0.99
Knowledge					
MMLU	4-shot CoT, pass@1	16	14,042	28.4 (2.2)	1.00
MMLU-Pro	5-shot CoT, pass@1	16	12,032	12.6 (1.0)	0.99
BBH	3-shot CoT, pass@1	16	6,511	27.1 (2.7)	0.99
Open QA					
Natural Questions	0-shot EM	1	3,610	3.4 (0.8)	0.92
TriviaQA	5-shot EM	1	17,944	24.4 (3.5)	1.00

Post-SFT benchmarks. Every SFT checkpoint is scored zero-shot through the chat template of the SFT corpus with one generation per item (Table 4), at temperature 0.6 and top-p 0.95 (top-k 20 except for IFEval). The thinking format is enabled on every benchmark except IFEval. MMLU, BBH and GSM8K use the zero-shot chain-of-thought prompts of the LM Evaluation Harness.

Table 4: **The eight post-SFT benchmarks** (Hendrycks et al., 2021a; Wang et al., 2024; Suzgun et al., 2023; Cobbe et al., 2021; Hendrycks et al., 2021b; Lightman et al., 2023; Chen et al., 2021; Austin et al., 2021; Liu et al., 2023; Zhou et al., 2023). Max tok.: generation budget; Think: thinking format enabled. E_1 mean (SD): over the 106 candidates (%, population SD).

Benchmark	Family	Items	Scoring	Max tok.	Think	E_1 mean (SD)
MMLU	knowledge	14,042	choice match	8,192	yes	34.8 (2.4)
MMLU-Pro	knowledge	12,032	choice match	8,192	yes	14.2 (1.3)
BBH	knowledge	6,511	answer match	8,192	yes	29.8 (2.1)
GSM8K	math	1,319	math verification	16,384	yes	16.4 (8.8)
MATH500	math	500	verification + LLM judge	8,192	yes	11.8 (5.3)
HumanEval+	code	164	EvalPlus pass@1	32,000	yes	5.9 (3.2)
MBPP+	code	378	EvalPlus pass@1	32,000	yes	8.9 (4.7)
IFEval	instruction	541	strict prompt-level	4,096	no	18.0 (2.4)

C THE 32-MODEL COHORT

The 32 candidates trained to E_5 were selected prior to the population SFT runs using only pretraining information. The cohort comprises an initial ten-model panel, the remaining final-round candidates so that all four final-round proposals are included, and 20 additional candidates stratified by decile of pretraining score via farthest-point sampling in the joint space of pretraining capability scores and mixture weights.

The cohort spans the full range of Z. Its members hold ranks 1 to 106, with at least one member in each consecutive group of 11 ranks (4, 2, 4, 3, 5, 2, 1, 3, 4 and 4 members from the highest-ranked group down), and include the two leaders of Z. It holds 23 early and 9 later candidates. Its mean E_1

scores lie within 0.2 pp of the population means on every benchmark. Table 9 gives the agreement of every epoch's ranking with the E_5 ranking in the cohort for all eight benchmarks.

D MEASUREMENT NOISE

To verify that observed ranking shifts reflect model differences rather than finite-benchmark sampling noise, we quantify the item-level split-half reliability of each pretraining metric and post-SFT benchmark. Main-text analyses report observed rankings directly without disattenuation.

Reliability. We estimate ranking reliability by repeatedly splitting benchmark items into two random halves, ranking candidates on each half, correlating the rankings, and applying the Spearman–Brown correction (Spearman, 1910; Brown, 1910). The estimates summarize 200 random splits. All five chain-of-thought pretraining metrics have reliability at least 0.98 (Table 3). Four commonsense metrics—AGIEval, OpenBookQA, Social IQa, and WinoGrande—have estimates of 0.45–0.55 and remain part of the composite. After SFT, six benchmarks have reliability at least 0.93 at every epoch (Table 5); HumanEval+ and IFEval have reliabilities of 0.83 and 0.76 over all candidates at E_1.

Table 5: **Reliability of the post-SFT rankings.** Item split-half reliability with Spearman–Brown correction, rank version, median over 200 random splits, for all 106 candidates at E_1 and for the 32-model cohort at every epoch.

	All, $n = 106$	Cohort, $n = 32$				
Benchmark	E_1	E_1	E_2	E_3	E_4	E_5
MMLU	0.97	0.97	0.97	0.97	0.98	0.98
MMLU-Pro	0.95	0.96	0.96	0.97	0.97	0.97
BBH	0.94	0.95	0.94	0.95	0.96	0.94
GSM8K	0.99	0.99	0.99	0.99	0.99	0.99
MATH500	0.95	0.96	0.97	0.96	0.97	0.96
HumanEval+	0.83	0.84	0.86	0.79	0.82	0.77
MBPP+	0.94	0.95	0.95	0.94	0.95	0.95
IFEval	0.76	0.78	0.73	0.70	0.74	0.72

Reliability-adjusted agreement across stages. We use disattenuation to assess how item sampling noise affects observed rank correlations across stages (Table 6). The geometric mean of the two reliability estimates provides a reliability-based reference ceiling. The adjusted agreement R correlates disjoint item halves across stages and normalizes by within-stage half correlations. We average over 200 splits and obtain a 95% interval from 1,000 candidate-bootstrap draws, each using a fresh split, with items aligned by content across stages.

An adjusted value near one indicates rank agreement up to item-sampling noise. From P to E_1, R is 0.77 for MMLU, 0.81 for MMLU-Pro, and 0.53 for BBH, with intervals well below one. This pattern confirms the distinction between the substantial reordering of knowledge rankings and the high stability of math rankings, whose adjusted values are 0.94 for GSM8K and 0.98 for MATH500.

Rankings after the first epoch. Observed agreement from E_1 to E_5 is close to the reliability-based reference ceilings across benchmarks (Table 6, right). HumanEval+ and IFEval have lower observed agreement (0.83 and 0.61) alongside lower reference ceilings (0.80 and 0.75), with adjusted agreement $R = 0.98$ for HumanEval+ and $R = 0.80$ [0.46, 1.26] for IFEval.

Scope for predicting instruction following. IFEval's E_1 reliability of 0.76 corresponds to a reference correlation of about 0.87 with a noise-free signal under the split-half model, whereas the best single pretraining signal reaches 0.39 over all candidates and MIXSELECTOR reaches 0.21 on later candidates. This gap indicates that the difficulty of ranking instruction following is not merely an artifact of benchmark item noise.

Table 6: **Observed and reliability-adjusted agreement between stages.** ρ: Spearman correlation of the two rankings. Ceiling: reliability-based reference, the geometric mean of the two reliability estimates. R: adjusted agreement with a 95% candidate-bootstrap interval; sampling variability can produce $R > 1$.

	$\mathsf{P} \to \mathsf{E}_1$, $n = 106$			$\mathsf{E}_1 \to \mathsf{E}_5$, cohort, $n = 32$		
Benchmark	ρ	Ceiling	R [95%]	ρ	Ceiling	R [95%]
MMLU	0.76	0.98	0.77 [0.66, 0.86]	0.99	0.98	1.00 [0.98, 1.05]
MMLU-Pro	0.79	0.97	0.81 [0.73, 0.87]	0.94	0.96	0.98 [0.90, 1.03]
BBH	0.50	0.96	0.53 [0.37, 0.69]	0.97	0.95	1.02 [0.98, 1.11]
GSM8K	0.93	0.99	0.94 [0.89, 0.97]	0.98	0.99	0.99 [0.96, 1.01]
MATH500	0.94	0.97	0.98 [0.96, 0.99]	0.93	0.96	0.98 [0.92, 1.04]
HumanEval+	–	–	–	0.83	0.80	0.98 [0.74, 1.32]
MBPP+	–	–	–	0.95	0.95	1.00 [0.95, 1.08]
IFEval	–	–	–	0.61	0.75	0.80 [0.46, 1.26]

E RANKINGS AFTER MID-TRAINING AND SFT

Same-benchmark rankings. Table 7 extends the same-benchmark comparison of §3.2 with Kendall's τ_b, the fate of each pretraining leader and the comparison with E_5 in the cohort. The pattern at E_5 matches that at E_1: the mean Spearman correlation of the three knowledge benchmarks is lower than that of the two math benchmarks by 0.25 [0.18, 0.34] at E_1 and by 0.25 [0.11, 0.44] at E_5. The reordering also reaches the tail of the ranking: on BBH, two of the pretraining bottom ten reach the post-SFT top 20, one of them rank 9 (Figure 2, dashed lines).

Table 7: **Same benchmark before and after post-training.** ρ: Spearman correlation between the P and post-SFT scores with 95% model-bootstrap interval; τ_b: Kendall's τ_b. Top-10: candidates in both top-10 sets (random expectation 0.94 for $n = 106$, 3.13 for $n = 32$). Leader: post-SFT rank of the P leader and its shortfall from the post-SFT best (pp). Top-5: post-SFT ranks of the P top five, in P order.

Benchmark	ρ [95%]	τ_b	Top-10	Leader rank	Leader gap	Top-5
$\mathsf{P} \to \mathsf{E}_1$, *all 106 candidates*						
MMLU	0.76 [0.65, 0.84]	0.57	4	11	1.9	11, 28, 5, 4, 3
MMLU-Pro	0.79 [0.70, 0.85]	0.59	7	2	0.5	2, 3, 8, 9, 33
BBH	0.50 [0.32, 0.65]	0.36	2	42	2.4	42, 49, 5, 26, 21
GSM8K	0.93 [0.88, 0.96]	0.79	8	4	3.9	4, 2, 5, 9, 17
MATH500	0.94 [0.90, 0.96]	0.79	7	2	0.2	2, 1, 3, 9, 7
$\mathsf{P} \to \mathsf{E}_5$, *32-model cohort*						
MMLU	0.71 [0.43, 0.87]	0.54	7	3	1.0	3, 14, 5, 1, 10
MMLU-Pro	0.75 [0.51, 0.88]	0.57	7	3	0.3	3, 11, 1, 10, 5
BBH	0.56 [0.24, 0.82]	0.42	6	16	2.1	16, 2, 6, 4, 1
GSM8K	0.91 [0.76, 0.96]	0.75	7	3	2.0	3, 2, 1, 4, 13
MATH500	0.95 [0.87, 0.98]	0.83	8	1	0.0	1, 2, 2, 5, 6

The composite against the capability families. Kendall's τ_b between Z and the E_1 families is 0.51 (knowledge), 0.38 (math), 0.21 (code) and 0.09 (instruction following) over all candidates, and 0.40, 0.12, -0.06 and -0.19 over the 38 later ones, the same ordering as the Spearman correlations of Table 8. The leader of Z, `abstract_determinant_16`, ranks 13th, 41st, 37th and 22nd of 106 on the four families (on instruction following, tied with six others). Among the 38 later candidates, the leader of Z ranks first on knowledge capability but 18th, 15th and 31st on math, code and instruction following.

Agreement across later SFT epochs. Table 9 extends Figure 7 to the three post-SFT benchmarks without a corresponding metric in the 17-metric pretraining set. MBPP+ agrees with its E_5 ranking at 0.94 or more from E_1 on; HumanEval+ (0.77–0.83) and IFEval (0.61–0.78) agree less, in line with their lower reliability (Appendix D).

Table 8: The pretraining composite Z against each E_1 capability family: Spearman ρ and top-10 overlap (random: 0.94) over all candidates, and ρ over the 38 later candidates. Same-benchmark rankings are in Figure 2 and Table 7.

	All 106		Later 38
Family	ρ	Top-10	ρ
Knowledge	0.69	6	0.57
Math	0.55	2	0.18
Code	0.30	0	−0.07
Instr. fol.	0.13	1	−0.27

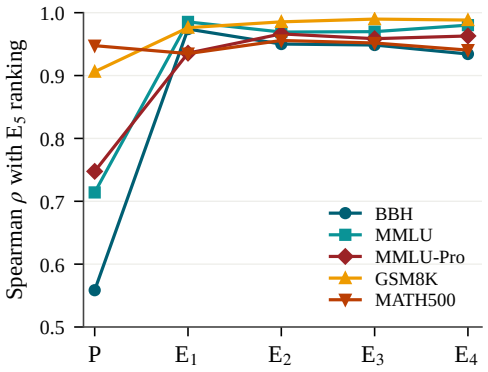


Figure 7: Rankings on the five benchmarks shown are largely stable after the first SFT epoch. 32-model cohort: Spearman ρ between the ranking at each stage (pretraining P, then SFT epochs 1–4, E_1–E_4) and the ranking after five SFT epochs (E_5), for the five benchmarks that have a pretraining score. From E_1 on, $\rho \geq 0.93$; from P, 0.56 (BBH) to 0.95 (MATH500).

F Cost of Committing to the Composite

Regret for every target. Post-training the top k candidates of a selection signal σ and keeping the best of them loses, on target t over the candidate set $\mathcal{C}$ (§3.1),

$$\mathrm{Regret@}k(\sigma; t) \;=\; \max_{i \in \mathcal{C}} t_i \;-\; \max_{i \in L_k(\sigma)} t_i, \tag{2}$$

in percentage points for a benchmark and in z units for a family. Regret@1 measures the cost of selecting one candidate, while larger k evaluates a shortlist; the random reference is the exact expected regret of a uniformly drawn k-subset of $\mathcal{C}$. Table 10 gives Regret@k of the top k candidates of Z for all twelve E_1 targets. Within the budgets tested, shortlists substantially reduce regret on knowledge and math: keeping the second candidate removes almost all of it on BBH, GSM8K and MATH500, and the top three and top five remove it on MMLU and MMLU-Pro. On code a shortlist does not: Regret@k stays above the random expectation on HumanEval+ from $k = 3$, on MBPP+ from $k = 5$ and on the code family from $k = 4$: the candidates that Z ranks third to eighth add nothing on code.

Loss under resampling of the pretraining items. The composite's top choice is stable under resampling of the evaluated pretraining items. We resample the items of all 17 pretraining metrics 1,000 times (stratified by MMLU subject, MMLU-Pro category, BBH subtask and AGIEval subtask, keeping AGIEval's macro-average), recompute the metrics, the z-scores and Z, and take its leader. The leader remains `abstract_determinant_16` in 92.5% of replicates and becomes `breezy_beer_3` in the other 7.5%; no third candidate ever leads. The expected loss over replicates (Table 10, Potential) is therefore close to the observed one, for instance 17.6 pp against 19.0 pp on GSM8K. Deeper in the ranking the resampling matters more: the top three stay the same set in 46.2% of replicates and the top ten in 10.4%.

G Pretraining Signals

All pretraining signals. Table 12 gives the Spearman correlation of every pretraining metric, group score and Z with every E_1 target. The two math metrics and the math group correlate at 0.73 or more with knowledge, math and code capability, whereas Z ranks only 6th, 5th, 8th and

Table 9: **Agreement with the E_5 ranking** in the 32-model cohort: Spearman ρ between each stage's ranking and the E_5 ranking; E_1 with 95% model-bootstrap interval. –: no pretraining evaluation.

Benchmark	P	E_1 [95%]	E_2	E_3	E_4
MMLU	0.71	0.99 [0.95, 0.99]	0.97	0.97	0.98
MMLU-Pro	0.75	0.94 [0.83, 0.98]	0.97	0.96	0.96
BBH	0.56	0.97 [0.92, 0.99]	0.95	0.95	0.93
GSM8K	0.91	0.98 [0.93, 0.99]	0.99	0.99	0.99
MATH500	0.95	0.93 [0.84, 0.97]	0.96	0.95	0.94
HumanEval+	–	0.83 [0.67, 0.90]	0.77	0.78	0.81
MBPP+	–	0.95 [0.86, 0.97]	0.94	0.97	0.95
IFEval	–	0.61 [0.33, 0.80]	0.78	0.70	0.72

Table 10: **Regret of committing to the top k of Z at E_1** ($n = 106$), in pp for benchmarks and z units for families. Random: exact expected regret of k uniformly drawn candidates. Potential: mean $\mathrm{Regret@1}$ over 1,000 item-bootstrap replicates of the pretraining evaluations. Rank: the leader's E_1 rank on the target.

	Z, $k =$					Random, $k =$				
Target	1	2	3	5	10	1	3	10	Potential	Rank
MMLU	1.9	1.0	0.0	0.0	0.0	4.7	2.8	1.5	1.8	11
MMLU-Pro	1.6	1.1	0.5	0.0	0.0	2.9	1.8	1.1	1.5	17
BBH	1.6	0.3	0.3	0.3	0.0	3.4	1.7	0.7	1.5	24
GSM8K	19.0	0.6	0.6	0.6	0.6	20.6	12.9	6.8	17.6	49
MATH500	7.6	0.2	0.2	0.2	0.2	11.2	6.7	3.7	7.0	35
HumanEval+	5.5	4.9	4.9	4.9	3.0	7.5	4.7	2.6	5.4	33
MBPP+	9.5	5.8	5.8	5.8	5.8	11.2	7.0	3.9	9.2	39
IFEval	4.1	3.1	3.1	3.1	1.8	5.9	3.8	2.0	4.0	22
Knowledge (z)	0.66	0.21	0.00	0.00	0.00	1.69	0.91	0.42	0.62	13
Math (z)	1.75	0.00	0.00	0.00	0.00	2.19	1.33	0.71	1.62	41
Code (z)	1.37	0.88	0.88	0.88	0.60	1.86	1.02	0.44	1.33	37
Instr. fol. (z)	1.67	1.29	1.29	1.29	0.76	2.42	1.55	0.82	1.64	22

9th among the 22 signals on the four families. Eight of the ten commonsense metrics and TriviaQA correlate negatively with code. On single benchmarks, post-SFT BBH is predicted better by the pretraining math group (0.91) than by pretraining BBH (0.50; difference 0.41 [0.25, 0.58]), whereas for MMLU the best signal, pretraining MMLU-Pro (0.81), is not distinguishable from pretraining MMLU (0.76; difference 0.05 $[-0.03, 0.15]$).

Stability of the best signal. Across bootstrap replicates of the 106 candidates, the top-ranked pretraining signal for knowledge, math, and code consistently comes from the math evaluations. In every family the runner-up is the math group score, and re-selecting the best signal in each bootstrap replicate recovers the full-sample choice in 83.3% (knowledge), 75.5% (math), 78.5% (code) and 65.2% (instruction following) of replicates; in most of the remaining replicates the math group score takes its place, ranking knowledge, math and code within 0.011 of GSM8K.

Early and later candidates. Between early and later candidates, the correlation profiles broadly agree (Table 13). Over the 88 family cells of the matrix, the early and later profiles correlate at 0.88 (0.90 over all 264 cells); 50 of the 264 cells change sign, mostly commonsense and open-QA signals that are near zero among early candidates and negative among later ones. Among the later candidates GSM8K remains within 0.03 of the top single signal on knowledge and within 0.02 on math, whereas the composite's correlation falls in every family among the later candidates. Referencing the family z-scores to each subset instead of all 106 candidates changes no correlation by more than 0.02 and no best signal.

Table 11: Best of 22 pretraining signals (17 metrics, 4 group scores, Z) per E_1 family, Spearman ρ; Z's rank in parentheses; Δ: best minus Z. Last columns: the two groups holding half of Z's weight.

	Best single signal		Composite Z		Ability group ρ	
Family	Signal	ρ	ρ (rank)	Δ [95% CI]	Commonsense	Open QA
Knowledge	GSM8K	0.85	0.69 (6)	+0.16 [0.05, 0.28]	0.29	−0.12
Math	GSM8K	0.95	0.55 (5)	+0.41 [0.28, 0.55]	0.06	−0.23
Code	GSM8K	0.77	0.30 (8)	+0.46 [0.30, 0.63]	−0.21	−0.33
Instr. fol.	MMLU-Pro	0.39	0.13 (9)	+0.26 [0.12, 0.42]	−0.11	−0.18

Table 12: **Every pretraining signal against every E_1 target** (Spearman ρ, $n = 106$). Bold: highest in the column before rounding. The IFEval column equals the instruction-following column and is omitted. Know.: knowledge; Instr.: instruction following; MMLU-P: MMLU-Pro; MATH: MATH500; HE+: HumanEval+.

	Capability family				Benchmark						
Pretraining signal	Know.	Math	Code	Instr.	MMLU	MMLU-P	BBH	GSM8K	MATH	HE+	MBPP+
AGIEval	0.32	0.16	0.13	0.12	0.38	0.28	0.22	0.19	0.12	0.13	0.12
ARC-Challenge	0.24	0.03	−0.22	−0.16	0.44	0.17	0.10	0.09	−0.05	−0.20	−0.23
ARC-Easy	0.19	0.01	−0.21	−0.20	0.38	0.13	0.06	0.07	−0.06	−0.18	−0.22
CommonsenseQA	0.57	0.39	0.39	0.27	0.65	0.56	0.43	0.41	0.33	0.35	0.40
HellaSwag	0.17	−0.05	−0.30	−0.12	0.41	0.11	0.01	0.01	−0.12	−0.28	−0.30
OpenBookQA	−0.05	−0.21	−0.37	−0.21	0.14	−0.10	−0.16	−0.16	−0.26	−0.32	−0.40
PIQA	0.01	−0.11	−0.44	−0.18	0.19	−0.05	−0.05	−0.08	−0.14	−0.42	−0.43
Social IQa	0.22	0.12	−0.01	0.07	0.28	0.18	0.15	0.14	0.09	0.00	0.00
TruthfulQA	0.08	−0.04	−0.17	−0.22	0.19	0.04	0.00	0.01	−0.07	−0.18	−0.16
WinoGrande	0.24	0.13	−0.19	−0.06	0.38	0.15	0.13	0.16	0.08	−0.19	−0.19
GSM8K	**0.85**	**0.95**	**0.77**	0.33	0.69	0.78	0.90	**0.93**	0.95	**0.68**	**0.78**
MATH500	0.81	0.93	0.73	0.31	0.63	0.73	0.89	0.90	0.94	0.64	0.76
MMLU	0.66	0.41	0.38	0.24	0.76	0.62	0.51	0.44	0.35	0.34	0.40
MMLU-Pro	0.79	0.58	0.61	**0.39**	**0.81**	**0.79**	0.61	0.60	0.51	0.55	0.64
BBH	0.57	0.45	0.30	0.16	0.56	0.52	0.50	0.47	0.42	0.27	0.31
Natural Questions	−0.01	−0.10	0.01	0.03	0.07	0.01	−0.08	−0.10	−0.10	0.05	0.01
TriviaQA	−0.10	−0.22	−0.51	−0.29	0.08	−0.16	−0.17	−0.18	−0.26	−0.47	−0.51
Group: commonsense & MC	0.29	0.06	−0.21	−0.11	0.49	0.22	0.13	0.12	−0.01	−0.20	−0.20
Group: math	0.84	0.95	0.76	0.34	0.67	0.76	**0.91**	0.92	**0.95**	0.67	0.78
Group: knowledge	0.76	0.53	0.48	0.29	0.80	0.74	0.60	0.56	0.47	0.43	0.50
Group: open QA	−0.12	−0.23	−0.33	−0.18	0.05	−0.14	−0.19	−0.20	−0.26	−0.27	−0.34
Composite Z	0.69	0.55	0.30	0.13	0.76	0.61	0.59	0.58	0.49	0.26	0.32

H Mixture Composition

Estimation. We use a commonality-style decomposition (Nimon & Oswald, 2013) of predictive R^2 from ridge regressions on pretraining metrics (P), domain shares (D), and both. Each fit standardizes features within its training fold and selects a penalty from 10^{-3} to 10^3 (25 log-spaced values) by generalized cross-validation. We pool out-of-fold predictions from ten-fold cross-validation and average R^2 over 20 repeats, using identical folds for all feature sets. Shared and source-specific components are defined as algebraic contrasts of these cross-validated predictive R^2 scores.

Intervals pool 200 out-of-bag bootstrap draws from each of ten seeds, for 2,000 draws in total. No selected penalty reaches the grid boundary. Instruction following selects penalties around 56–178, compared with 1–10 for the other families. A permutation test shuffles domain-share vectors across candidates while holding metrics and targets fixed. On one ten-fold split, the observed increments from D (0.08, 0.02, and 0.12 for knowledge, math, and code) exceed all 200 permuted increments ($p = 0.005$, the minimum attainable), and the instruction-following increment reaches $p = 0.045$.

All domain associations. Table 15 reports all 32 domain–family pairs, both raw and after removing predictions from pretraining metrics. Twenty raw and five residual correlations pass Benjamini–Hochberg correction at $q < 0.05$ (Benjamini & Hochberg, 1995). All five residual associations also pass in each of 20 alternative cross-fitting splits, with permutation p-values of at most 0.0015.

Composition and pretraining scores. Some domain associations are largely captured by pretraining scores. The share of math reasoning traces correlates at 0.83 with the pretraining math group; its

Table 13: **Best pretraining signal among early and later candidates.** Best: the signal with the highest Spearman ρ with the E_1 family among the 22 of Table 12; Z: the composite's ρ.

	Early candidates ($n = 68$)			Later candidates ($n = 38$)		
Family	Best	ρ	Z	Best	ρ	Z
Knowledge	GSM8K	0.85	0.70	MMLU-Pro	0.75	0.57
Math	GSM8K	0.97	0.59	Group: math	0.89	0.18
Code	GSM8K	0.71	0.37	GSM8K	0.73	−0.07
Instr. fol.	MMLU-Pro	0.34	0.15	Group: math	0.21	−0.27

Table 14: Cross-validated R^2 of ridge regressions predicting each E_1 family from pretraining metrics (P), domain shares (D) or both. Shared $= R^2_{\mathrm{P}} + R^2_{\mathrm{D}} - R^2_{\mathrm{P+D}}$; unique to D $= R^2_{\mathrm{P+D}} - R^2_{\mathrm{P}}$ (95% out-of-bag bootstrap CI).

Family	R^2_{P}	R^2_{D}	$R^2_{\mathrm{P+D}}$	Shared	Unique to D [95% CI]
Knowledge	0.82	0.79	0.90	0.70	0.09 [0.03, 0.18]
Math	0.92	0.89	0.94	0.87	0.02 [−0.02, 0.06]
Code	0.73	0.81	0.84	0.70	0.11 [0.01, 0.23]
Instr. fol.	0.12	0.17	0.15	0.14	0.03 [−0.54, 0.23]

association with post-SFT math falls from 0.88 in the raw scores to 0.22 in the residuals. Other associations help explain why a broad composite can misrank candidates. Synthetic QA share correlates positively with the commonsense group (0.81), open-QA group (0.66), and composite (0.52), but negatively with post-SFT code (-0.35). Math-corpus share is also positively associated with pretraining MMLU (0.45) and the composite (0.39), but negatively associated with residual post-SFT capability (§3.4).

I MIXSELECTOR: IMPLEMENTATION AND FURTHER RESULTS

I.1 IMPLEMENTATION

We summarize the implementation details of MIXSELECTOR and its comparators on the 68 training and 38 held-out candidates.

- *Split and targets.* The selector is fit on the 68 early candidates and ranks the 38 later ones. Its targets are the four E_1 family scores, with z-scores referenced to the training candidates only, and one model is fit per family.
- *Features.* The 25 features, 17 pretraining metrics and 8 domain shares, are standardized with the training mean and standard deviation; none is removed or selected.
- *Model.* Ridge regression with the penalty chosen from 10^{-3} to 10^{3} (25 log-spaced values) by efficient leave-one-out generalized cross-validation on the training candidates. The selected penalties are 3.2 for knowledge, math and code and 178 for instruction following; none lies at the edge of the grid.
- *Comparators.* A random ranking (exact expectations); Z with per-metric z-scores from the training candidates; the single pretraining metric with the highest training Spearman correlation with the family (GSM8K for knowledge, math and code; MMLU-Pro for instruction following); and ridges on the domain shares alone (D only) and on the pretraining metrics alone (P only).
- *Metrics.* On the test candidates: Spearman correlation of predicted and true scores, overlap of the predicted and true top-10 sets, and $\mathrm{Regret@1}$ and $\mathrm{Regret@3}$ in z units (Eq. (2)); ties in a predicted ranking are broken by candidate name.
- *Uncertainty.* A paired test bootstrap resamples the 38 test candidates 2,000 times with one index matrix shared by all methods to form 95% confidence intervals.

Table 15: **Domain shares and E_1 capability** ($n = 106$). Raw: Spearman ρ between the domain share and the family score. Resid.: Spearman ρ between the share and the part of the family score that a cross-fitted ridge on the 17 pretraining metrics does not predict. Bold: $q < 0.05$ after Benjamini–Hochberg control over the 32 correlations of the column type. Associations within the observed mixtures; shares sum to one.

	Knowledge		Math		Code		Instr. fol.	
Domain	Raw	Resid.	Raw	Resid.	Raw	Resid.	Raw	Resid.
Web	**−0.56**	−0.20	**−0.42**	−0.21	**−0.61**	**−0.31**	**−0.26**	−0.11
Code	0.08	0.07	0.07	0.12	**0.58**	**0.42**	**0.26**	0.11
Math corpus	0.12	**−0.37**	0.13	**−0.32**	0.10	−0.20	0.01	−0.07
Math reasoning traces	**0.63**	0.13	**0.88**	0.22	**0.54**	0.06	0.21	0.06
Academic	**−0.35**	0.10	**−0.27**	−0.10	**−0.32**	−0.09	−0.18	−0.11
Synthetic QA	−0.02	−0.02	−0.14	0.09	**−0.35**	−0.04	**−0.25**	−0.08
Synthetic reasoning	**0.34**	0.18	0.16	−0.17	**0.22**	−0.10	**0.23**	0.11
Synthetic general	**0.31**	**0.30**	0.13	−0.03	**0.24**	0.07	**0.31**	0.20

I.2 Results per family and per benchmark

MixSelector ranks each of knowledge, math and code capability at 0.88 or more (Table 16), and its top-10 sets share 8, 7 and 7 candidates with the true top 10. Its gain over Z is largest where Z fails: the paired Spearman difference is 0.30 [0.07, 0.55] for knowledge, 0.73 [0.41, 1.05] for math, 0.94 [0.60, 1.24] for code and 0.49 [0.12, 0.81] for instruction following. Against the best single metric the difference is 0.16 [0.02, 0.34], 0.03 [−0.05, 0.15], 0.15 [0.03, 0.28] and 0.19 [−0.03, 0.44]. On instruction following every method ranks poorly (every 95% test interval includes zero; the D-only ridge, highest at 0.32, has [−0.002, 0.61]); the D-only ridge's top pick happens to lose least there (0.55 against 2.58 for the selector), which is why its family-mean $\mathrm{Regret@1}$ is the lowest in Table 1. In percentage points, the selector's top pick loses 2.0 pp on the math benchmarks against 11.3 pp for Z (random 12.1 pp) and 2.4 pp on the code benchmarks against 6.5 pp (random 7.7 pp). The selector ranks each family separately, so its top picks differ: `generative_icecream_20` for knowledge and math, `dreadful_os_14` for code and `usable_model_0` for instruction following. The composite commits `classical_eggs_32` to all four families; that mixture happens to be the best later candidate on knowledge capability.

Table 16: **Selection signals per capability family** on the 38 later candidates. ρ: Spearman correlation; $\mathrm{Regret@1}$ in z units; pp: mean loss of the top pick over the family's benchmarks, in pp. Random: exact expectation.

	Knowledge			Math			Code			Instr. fol.		
Method	ρ	Regret@1	pp	ρ	Regret@1	pp	ρ	Regret@1	pp	ρ	Regret@1	pp
Random	0.00	1.20	2.6	0.00	1.62	12.1	0.00	1.54	7.7	0.00	2.07	4.9
Composite Z	0.58	0.00	0.4	0.18	1.49	11.3	−0.07	1.20	6.5	−0.28	3.05	7.2
Best single metric	0.72	0.02	0.6	0.88	0.17	2.0	0.73	1.32	7.1	0.02	3.05	7.2
D only	0.73	0.02	0.6	0.90	0.17	2.0	0.84	0.31	2.4	0.32	0.55	1.3
P only	0.84	0.82	1.8	0.86	0.17	2.0	0.83	0.00	2.2	0.19	1.09	2.6
MixSelector (P + D)	0.88	0.02	0.6	0.91	0.17	2.0	0.88	0.31	2.4	0.21	2.58	6.1

Fitted on single benchmarks (in percent), the selector ranks every benchmark except IFEval (0.21) at 0.72 or more (Table 17). Z ranks MMLU well (0.72) but the math and code benchmarks at 0.21 or less.

I.3 Robustness

Cross-validation over all candidates. Repeated random 5-fold cross-validation over all 106 candidates (50 repeats) gives the same ordering by mean Spearman correlation as the main held-out split (Table 18). MixSelector achieves the highest mean Spearman correlation, 0.80, and exceeds every comparator in all 50 repeats.

Table 17: **Selection signals per benchmark** on the 38 later candidates, fit on benchmark scores in percent. Left: Spearman ρ. Right: $\mathrm{Regret@1}$ in pp.

	Spearman ρ					Regret@1 (pp)			
Benchmark	Z	Best single	D only	P only	MIXSELECTOR	Random	Z	Best single	MIXSELECTOR
MMLU	0.72	0.62	0.74	0.79	0.88	3.3	0.0	0.9	0.8
MMLU-Pro	0.47	0.63	0.61	0.75	0.73	2.1	0.5	0.0	0.0
BBH	0.38	0.75	0.69	0.71	0.73	2.4	0.8	1.0	1.0
GSM8K	0.21	0.84	0.88	0.84	0.86	16.4	16.1	3.9	7.0
MATH500	0.14	0.86	0.85	0.86	0.91	7.8	6.6	0.0	0.0
HumanEval+	−0.01	0.69	0.76	0.79	0.78	6.3	4.9	4.9	4.3
MBPP+	−0.18	0.70	0.86	0.80	0.85	9.1	8.2	9.3	4.2
IFEval	−0.28	0.02	0.32	0.19	0.21	4.9	7.2	7.2	6.1

Table 18: **Repeated 5-fold cross-validation** over all 106 candidates (50 repeats), mean over the four families. Metrics are computed within test folds of 21–22 candidates, so top-10 overlap is not comparable with the 38-candidate test split (random expectation 4.72). Better: share of repeats in which MIXSELECTOR's Spearman correlation exceeds the method's.

Method	Spearman ρ	Top-10	$\mathrm{Regret@1}$	$\mathrm{Regret@3}$	Better
Random	0.00	4.72	1.71	0.86	1.00
Composite Z	0.40	6.16	0.96	0.53	1.00
Best single metric	0.72	7.85	0.70	0.30	1.00
D only	0.77	8.03	0.56	0.22	1.00
P only	0.76	7.94	0.42	0.19	1.00
MIXSELECTOR (P + D)	0.80	8.22	0.41	0.19	–

Composite reference. The composite comparator uses z-scores from the training candidates, as all comparators use training information only. With z-scores over all 106 candidates instead, its family-mean Spearman correlation (0.10) and $\mathrm{Regret@1}$ (1.43) are unchanged, and its $\mathrm{Regret@3}$ falls from 1.17 to 0.77 because its third pick changes for math and instruction following.

I.4 FITTED WEIGHTS

The largest standardized coefficients of MIXSELECTOR fall on a small set of features. For knowledge capability the largest are pretraining MATH500 (0.33) and GSM8K (0.29), the synthetic general (0.17) and synthetic reasoning (0.15) shares and the web share (-0.19). For math they are GSM8K (0.34), MATH500 (0.33) and the share of math reasoning traces (0.28). For code they are GSM8K (0.37) and the code share (0.35), with negative weights on the web (-0.24) and synthetic QA (-0.20) shares. For instruction following the large penalty shrinks every coefficient to about 0.05 or less in magnitude, leaving predictions close to a constant.

I.5 NUMBER OF POST-TRAINED CANDIDATES

For each training budget $n \in \{10, 15, 20, 30, 45\}$ we draw 200 random subsets of n early candidates without replacement, fit every method exactly as in the main analysis with z-references and standardization from the subset, and test on the same 38 later candidates; to keep regret units comparable across n, test targets are referenced to all 68 early candidates. At $n = 68$ the procedure reproduces the main result exactly.

Table 19 gives the results behind Figure 6. MIXSELECTOR's median Spearman correlation rises from 0.63 at $n = 10$ to 0.71 at $n = 45$, beating Z in 100% of draws at every n and improving median correlation over both the D-only and P-only ridges across all budgets. Per family, the selector's median $\mathrm{Regret@1}$ already stabilizes at $n = 10$ at $0.02\,z$ for knowledge, $0.17\,z$ for math, and $0.31\,z$ for code across every n, matching the full $n = 68$ fit, while overall mean regret reflects instruction following ($2.19\,z$ up to $n = 45$ and $2.58\,z$ at $n = 68$).

At $n = 10$, selecting a single pretraining metric on ten candidates recovers GSM8K in only 47.5% (knowledge), 79.5% (math) and 40.5% (code) of draws, which explains why the single-metric base-

Table 19: **Ranking methods fit on n post-trained early candidates**, tested on the 38 later candidates; mean over the four families. Spearman ρ: median over 200 random draws of the n training candidates, with interquartile range for MIXSELECTOR; $n = 68$ is the single main fit. Share of draws in which the selector's ρ exceeds that of Z, is at least that of the best single metric, and exceeds that of the D-only and P-only ridges. Last column: the selector's median $\mathrm{Regret@1}$ (z).

	Median Spearman ρ					MIXSELECTOR better (% of draws)				
n	MIXSELECTOR [IQR]	D only	P only	Best single	Z	$>Z$	$\geq$Best	$>$D	$>$P	$\mathrm{Regret@1}$
10	0.63 [0.56, 0.69]	0.57	0.55	0.50	0.09	100	87.0	78.5	91.0	0.72
15	0.66 [0.61, 0.70]	0.63	0.60	0.54	0.10	100	89.5	62.5	84.5	0.66
20	0.67 [0.62, 0.71]	0.66	0.60	0.57	0.10	100	91.5	54.0	88.0	0.66
30	0.69 [0.64, 0.71]	0.68	0.64	0.59	0.10	100	96.5	55.0	87.5	0.63
45	0.71 [0.69, 0.73]	0.69	0.67	0.59	0.10	100	100	76.5	95.5	0.67
68	0.72	0.70	0.68	0.59	0.10	–	–	–	–	0.77

line degrades more sharply at small n than regularized regression over all features. Because the held-out test set is fixed across draws, the interquartile ranges in Table 19 reflect variation across the sampled training subsets.

J PIPELINE DETAILS AND REPRODUCIBILITY

Mid-training. Every candidate receives the same mid-training (YaRN context extension; Peng et al., 2024). The configuration sets the YaRN factor to 5 (from 8,192 to 40,960 positions), the RoPE base to 10^6 and the attention scaling to $1 + 0.1 \ln 5 \approx 1.161$, and trains with next-token prediction for 10,000 steps on 8 GPUs with two packed 40,960-token sequences per GPU (about 6.6B tokens) at a constant learning rate of 2×10^{-5} after 200 warmup steps, in bf16 with fully sharded data parallelism. The corpus is a long-context mix of web, code, mathematical, question–answer and non-English documents, sampled by document-length bucket from under 4k to over 32k tokens.

Supervised fine-tuning. All 106 candidates share one SFT recipe (Table 20). Because the learning rate is constant after warmup, the first epoch follows the same schedule whether a run stops after one epoch or continues to five, so the E_1 checkpoints of the 74 one-epoch runs and of the cohort are comparable.

Table 20: **SFT recipe**, identical for all 106 candidates; only the number of epochs differs.

Setting	Value
Framework	TRL `SFTTrainer`, DeepSpeed ZeRO-3, 8 H100 GPUs per run
Data	cleaned Mixture-of-Thoughts corpus (Open-R1 Team, 2025), 347,360 conversations
Chat template	shared tokenizer and chat template with a thinking format
Epochs	1 (74 candidates) or 5 (32-model cohort); 3,677 optimizer steps per epoch
Learning rate	6×10^{-6}, constant after 500 linear warmup steps
Optimizer	AdamW, $\beta = (0.9, 0.95)$, weight decay 0.1, gradient clipping at 1.0
Batch	16 sequences per step (1 per GPU, 2 accumulation steps)
Sequence length	32,768 tokens, packed
Loss	assistant tokens only
Precision	bf16, FlashAttention-2, gradient checkpointing
Checkpoints	end of each epoch ($\mathsf{E}_1, \ldots, \mathsf{E}_5$)
Seeds	one run per candidate (seed 42)

Compute. SFT took a median of 32.4 H100 GPU-hours per candidate to E_1 and 162.3 to E_5 (training time on eight GPUs), about 7,600 GPU-hours for the population. These figures exclude pretraining, mid-training, and evaluation. Fitting MIXSELECTOR takes seconds on a CPU.

Analysis. All analyses read one frozen snapshot of per-candidate, per-item evaluation results, from which a single script builds the shared inputs: the 17 pretraining metrics, Z, the post-SFT scores and the domain shares. Each result was verified from the frozen inputs with independently written code, and all point estimates agreed. Every analysis uses fixed random seeds and runs on CPU.

Artifacts. Upon publication we will release the frozen snapshot (per-item results at every stage, the pretraining evaluation table and the exact 587-bin mixture weights), the analysis protocol, the scripts that produce every table and number, and a manifest that traces each number in the paper to its output file.