LEARNING TEACHER OUTPUT PREFERENCES FROM FINAL RESPONSES FOR REASONING DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Large reasoning models often expose only their final responses while keeping their native chain-of-thought (CoT) unavailable, limiting their use as teachers for smaller models. Training directly on final responses omits intermediate supervision, while generic synthetic rationales may overlook source-dependent output patterns. We investigate whether visible final responses provide signals for synthesizing reasoning supervision that reflects these patterns. We introduce Final-Output Preference Learning (FOPL), a final-response-conditioned preference-learning framework. In the first offline stage, we use auxiliary trace-accessible source teachers to pair each native CoT with answer-conditioned synthetic alternatives generated by strong reasoning models. Each preference pair shares the same question and visible final response, allowing a single reconstruction policy to learn preferences over reasoning realizations while holding the response fixed. At the second transfer stage, the policy generates a CoT from only a question and the target teacher's visible final response, without access to that teacher's identity or native CoT. Reconstructions conditioned on a teacher family's own responses resemble its native traces more closely than those conditioned on other families' responses, including for a family excluded from preference training. On several completed reasoning evaluations, students trained on reconstructed CoTs match or outperform students trained on native-CoT references. These results suggest that visible final responses can support useful reasoning supervision without requiring the recovery of a target teacher's hidden reasoning process.

1 INTRODUCTION

Reasoning traces are a powerful source of supervision for language models. Explicit intermediate reasoning can improve performance on compositional tasks (Wei et al., 2022; Kojima et al., 2022; Wang et al., 2022) and, when available from a strong teacher, can benefit smaller student models through distillation (Shridhar et al., 2023; Hsieh et al., 2023). In practice, however, most widely deployed proprietary reasoning models, including OpenAI's o-series, Anthropic's Claude models, and Google's Gemini models, do not expose their native chain-of-thought (CoT), providing users with only the final response or, at most, a compressed summary of the underlying reasoning process (Jaech et al., 2024; Comanici et al., 2025; Singh et al., 2025). Training only on the visible response discards intermediate supervision, whereas asking another model to generate a plausible rationale may produce useful reasoning steps but does not necessarily preserve the target teacher's output preferences. Together, these limitations leave open a central question: *When the teacher's native CoT is unavailable, can its visible final response provide sufficient preference signals to synthesize teacher-aligned reasoning supervision?*

A seemingly direct approach would be to recover the teacher's hidden reasoning process from its final response. This goal faces two limitations. First, multiple latent reasoning trajectories may be compatible with the same final response. Second, even if a reconstructed trace resembles an observed native CoT, textual similarity does not establish that it faithfully represents the computation that produced the response (Turpin et al., 2023; Lanham et al., 2023). Our objective is therefore operational: given a question and a teacher's visible final response, we seek to synthesize a reasoning trace that reflects source-dependent output patterns and provides effective supervision for a downstream student. We formulate this task as learning *observable teacher output preferences*.

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.

Our key motivation is evidence that model-generated text contains source-dependent signatures. Prior work has shown that outputs from different LLMs exhibit distinguishable lexical, morphosyntactic, and stylistic patterns (Antoun et al., 2024; McGovern et al., 2025). We hypothesize that, in reasoning tasks, a teacher's final response may likewise retain cues about how it organizes a solution, emphasizes evidence, verifies intermediate results, uses notation, and presents conclusions. These cues cannot reveal the teacher's actual latent process, but they may help construct supervision that better reflects its observable output patterns than a generic, answer-consistent rationale.

We introduce Final-Output Preference Learning (FOPL), a final-response-conditioned preference-learning framework. The framework contains two stages. In an offline stage, we collect questions, native traces, and visible final responses from auxiliary, trace-accessible source teachers. For each source teacher, we pair its native trace with answer-conditioned synthetic alternatives generated by strong reasoning models. Each preference pair shares the same question and visible final response, allowing DPO (Rafailov et al., 2023) to train a single reconstruction policy to prefer native reasoning realizations over synthetic alternatives without relying on differences in final outcomes. In a subsequent transfer stage, the policy receives only a question and the visible final response of a target teacher. It generates a trace for student supervision without access to the target teacher's identity or native CoT.

We evaluate both resemblance to native traces and downstream student utility. In comparisons that control for the question and normalized final answer, reconstructions conditioned on a family's own responses are judged more similar to its native traces than reconstructions conditioned on other families' responses. This pattern also holds for MiMo, a teacher family excluded from offline preference training. Its matching-source reconstruction receives a tie-aware preference rate of 60.51%. Across all four response sources, FOPL outperforms native-CoT supervision in 10 of 12 comparisons on GSM8K, MATH-500, and GPQA. Averaged across the five downstream tasks, its gain over the base model is 4.09 percentage points larger than the strongest prompted-synthesis condition selected within each response source.

Our contributions are threefold:

- We formulate reasoning reconstruction from final responses as learning *observable teacher output preferences*, with native CoTs serving as preferred examples during offline training rather than as evidence of recoverable hidden computation.
- We introduce FOPL, a single final-response-conditioned reconstruction policy trained with DPO on native-versus-synthetic trace preferences from heterogeneous, trace-accessible source teachers. At transfer time, it requires neither the target teacher's identity nor its native CoT.
- We demonstrate source-aligned reconstruction for both training and held-out teacher families and show that the reconstructed traces provide useful supervision for downstream distillation.

2 Related Work

Reasoning distillation and synthetic supervision. Knowledge distillation transfers behavior from a stronger teacher to a smaller student (Hinton et al., 2015), with sequence-level methods using teacher-generated outputs as supervision (Kim & Rush, 2016). Reasoning distillation extends this idea by transferring intermediate explanations or chain-of-thought (CoT), which can substantially improve small-model reasoning (Li et al., 2022; Magister et al., 2023; Shridhar et al., 2023; Hsieh et al., 2023). Orca and Orca 2 further show the value of rich explanations and reasoning strategies tailored to smaller models (Mukherjee et al., 2023; Mitra et al., 2023). More broadly, methods such as STaR (Zelikman et al., 2022), WizardMath (Luo et al., 2025), MetaMath (Yu et al., 2024), DeepSeek-R1 (Guo et al., 2025), and OpenThoughts (Guha et al., 2026) demonstrate that the construction and quality of synthetic reasoning supervision are central to downstream performance. Our work addresses a complementary setting in which the target teacher's native CoT is unavailable: rather than using a generic synthetic rationale, we synthesize supervision conditioned on the teacher's visible final response.

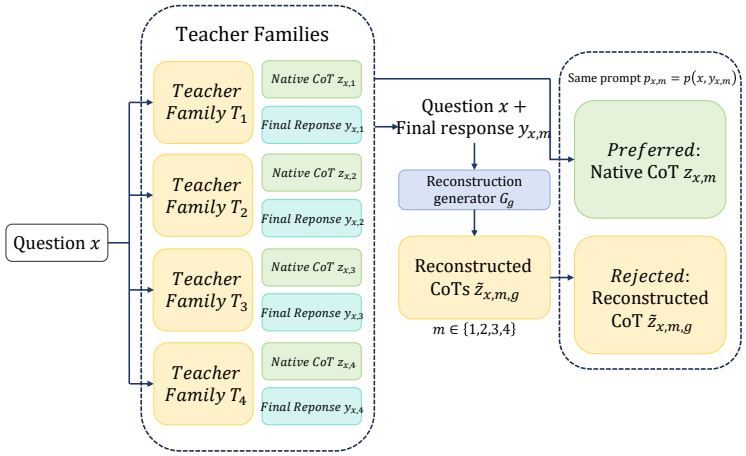


Figure 1: Construction of preference pairs for FOPL. For each question, a trace-accessible source teacher provides a native reasoning trace and a visible final response. Multiple reconstruction generators receive the same question and source-teacher response and produce answer-conditioned synthetic traces. Each preference pair uses the native trace as the preferred completion and one synthetic trace as the rejected completion under the shared response-conditioned prompt.

Trace reconstruction and preference learning. Several methods generate reasoning conditioned on a known outcome. STaR, for example, uses answer-conditioned rationalization during self-training (Zelikman et al., 2022), while Zhang et al. (2026) formulate trace inversion as supervised reconstruction from questions, answers, and optional reasoning summaries. Our setting is closest to their no-summary formulation, but differs in the learning signal: instead of learning only from positive reference traces, we compare native traces against answer-consistent synthetic alternatives. We optimize these comparisons using Direct Preference Optimization (DPO) (Rafailov et al., 2023), building on the broader preference-learning literature (Ouyang et al., 2022; Bai et al., 2022; Lee et al., 2023; Azar et al., 2024; Ethayarajh et al., 2024; Hong et al., 2024; Meng et al., 2024). Importantly, we use DPO as an established optimization tool rather than as a methodological contribution. By holding the question and final response fixed within each preference pair, our objective focuses on differences among reasoning realizations associated with the same visible outcome.

Faithfulness, behavioral alignment, and evaluation. Natural-language reasoning traces should not be assumed to reveal a model's true internal computation. Prior work shows that explanations can rationalize unreported influences (Turpin et al., 2023) and that the causal role of CoT varies under intervention (Lanham et al., 2023). Accordingly, we do not interpret reconstructed traces as recovering a unique latent reasoning process. Instead, we distinguish behavioral resemblance from downstream utility. Our similarity analysis uses pairwise LLM judging, which enables scalable comparison but can exhibit biases toward position, verbosity, or the judge's own response patterns (Zheng et al., 2023). Answer agreement and anonymization reduce some confounds, but the resulting measurements should be interpreted as evidence of observable output alignment rather than proof of process faithfulness or step-level correctness.

3 Method

FOPL consists of two stages. In the offline stage, we construct native-versus-synthetic preference pairs from trace-accessible source teachers and train a final-response-conditioned reconstruction policy with Direct Preference Optimization (DPO). In the online stage, the trained policy receives a question and a visible response from a target teacher, generates an intermediate reasoning trace, and uses it to construct supervision for a downstream student. Native CoTs are used only in the offline stage; the reconstruction policy requires only a question and visible final response when applied to a target teacher.

3.1 Offline Stage: Preference Learning from Trace-Accessible Source Teachers

Preference data construction. Let $\mathcal{Q}$ denote a source question pool, $\mathcal{M}_{\mathrm{src}}$ a set of trace-accessible source-teacher indices, and $\mathcal{G}$ a set of reconstruction-generator indices. For $m \in \mathcal{M}_{\mathrm{src}}$, T_m denotes a source teacher whose native trace and visible final response are available during offline training. For $g \in \mathcal{G}$, G_g denotes a strong reasoning model used to generate a synthetic comparison trace. A model may appear in both sets, but the roles are distinct: T_m provides the preferred native trace, whereas G_g generates an answer-conditioned alternative.

For every question $x \in \mathcal{Q}$, each source teacher produces a native trace $z_{x,m}$ and a visible final response $y_{x,m}$:

$$(z_{x,m}, y_{x,m}) \sim T_m(\cdot \mid x), \qquad m \in \mathcal{M}_{\mathrm{src}}. \tag{1}$$

The visible response $y_{x,m}$ may contain explanations, notation, formatting, and verification in addition to the normalized final answer. Different source teachers may produce different responses to the same question. We retain questions for which the required source-teacher traces and responses are available.

For each question x and source teacher m, we form a response-conditioned reconstruction prompt

$$p_{x,m} = p(x, y_{x,m}), \tag{2}$$

where $p(\cdot,\cdot)$ is a fixed template containing the question and visible final response. The prompt contains neither the source-teacher identity nor its native trace. Each reconstruction generator then produces

$$\widetilde{z}_{x,m,g} \sim G_g(\cdot \mid p_{x,m}), \qquad g \in \mathcal{G}. \tag{3}$$

The full prompt is provided in Appendix A.2. A source teacher may also serve as a reconstruction generator. Even when $g = m$, the reconstructed trace is distinct from the native trace because it is generated after the completed response has been supplied.

For each (x, m, g), we construct the preference triple

$$(p_{x,m}, z_{x,m}, \widetilde{z}_{x,m,g}), \tag{4}$$

where $z_{x,m}$ is preferred and $\widetilde{z}_{x,m,g}$ is rejected. The native trace is preferred because it is the observed reasoning trace of the source teacher, whereas the synthetic trace is an answer-conditioned alternative generated under the same question and visible response. This preference reflects source-trace provenance rather than an absolute correctness judgment: the native trace need not be error-free, and the synthetic trace need not be invalid.

Collecting all retained triples gives

$$\mathcal{D}_{\mathrm{pref}} = \{(p_{x,m}, z_{x,m}, \widetilde{z}_{x,m,g}) \mid x \in \mathcal{Q},\ m \in \mathcal{M}_{\mathrm{src}},\ g \in \mathcal{G}\}. \tag{5}$$

With N retained questions, $M = |\mathcal{M}_{\mathrm{src}}|$ source teachers, and $K = |\mathcal{G}|$ reconstruction generators, complete generation produces NMK preference pairs before filtering.

DPO training of the reconstruction policy. Let R_0 be the frozen initial reconstruction policy and R_θ the trainable policy initialized from it. For a preference triple $(p, z^+, z^-) \in \mathcal{D}_{\mathrm{pref}}$, we define

$$p = p_{x,m} = p(x, y_{x,m}), \qquad z^+ = z_{x,m}, \qquad z^- = \widetilde{z}_{x,m,g}. \tag{6}$$

Thus, p is the shared question–response prompt, z^+ is the source teacher's native trace, and z^- is an answer-conditioned synthetic trace.

We define the reference-relative score

$$r_\theta(z; p) = \beta \log \frac{R_\theta(z \mid p)}{R_0(z \mid p)}, \qquad \beta > 0, \tag{7}$$

and optimize the standard DPO objective:

$$\mathcal{L}_{\mathrm{DPO}}(\theta) = -\mathbb{E}_{(p,z^+,z^-)\sim\mathcal{D}_{\mathrm{pref}}} \log \sigma\left(r_\theta(z^+; p) - r_\theta(z^-; p)\right), \tag{8}$$

where σ is the logistic function. Sequence likelihoods are computed over trace-completion tokens; prompt tokens provide conditioning but do not contribute to the completion likelihood.

Because z^+ and z^- share the same question and visible response, DPO compares alternative reasoning realizations rather than different final outcomes. Preference pairs from all source teachers and reconstruction generators are pooled to train one policy. The source-teacher index is used only to organize offline data and is not supplied to the reconstruction policy. After training, we freeze the optimized policy and denote it by $R_{\theta^\star}$.

3.2 Online Stage: Target-Teacher Reconstruction and Student Distillation

Let B denote the target teacher model which provides visible final responses only. Given a collection of question–response records

$$\mathcal{D}_B = \{(x_i, y_i^B)\}_{i=1}^n, \tag{9}$$

the frozen reconstruction policy generates an intermediate trace for each record:

$$\widehat{z}_i \sim R_{\theta^\star}\big(\cdot \mid p(x_i, y_i^B)\big). \tag{10}$$

Although B denotes the response source for exposition, its identity is not provided to the reconstruction policy. In particular, the policy uses neither a native CoT from B nor a teacher-specific reconstruction model.

We construct the student target

$$t_i = h(\widehat{z}_i, y_i^B), \tag{11}$$

where h is a fixed serialization that places the reconstructed trace before the original visible response. The resulting student-training dataset is

$$\mathcal{D}_{\mathrm{SFT}} = \{(x_i, t_i)\}_{i=1}^n. \tag{12}$$

For a student model S_ϕ, teacher-forced supervised fine-tuning minimizes

$$\mathcal{L}_{\mathrm{SFT}}(\phi) = -\mathbb{E}_{(x,t)\sim\mathcal{D}_{\mathrm{SFT}}} \sum_{j=1}^{|t|} \log S_\phi(t_j \mid x, t_{<j}). \tag{13}$$

Only the student parameters ϕ are updated. When native target-teacher traces are observable only for evaluation, we additionally construct an oracle reference by replacing $\widehat{z}_i$ with the corresponding native trace while retaining the same question and visible response. This oracle condition is not available to FOPL during online reconstruction.

After student training, the student receives only a question and generates its response directly. Neither the target-teacher response nor the reconstruction policy is required at student deployment; reconstruction is used only to prepare student supervision.

4 Experiments

4.1 Experiment Setup

We evaluate two aspects of the reconstruction objective. First, we test whether the visible final response helps FOPL produce a trace resembling the response source's native CoT. Second, we test whether the reconstructed trace provides useful supervision for a fixed small student. Native CoTs are the preferred examples in offline training and, where available, an oracle reference in evaluation; they are never supplied to FOPL for a target teacher at transfer time. Student accuracy measures the utility of a trace as supervision, whereas the cross-family comparison assesses its resemblance to an observed native trace. Higher student accuracy alone does not establish faithful recovery of a teacher's hidden reasoning.

Models. Cross-family experiments use DeepSeek, Kimi, GLM, Qwen, and MiMo as teacher families. Among them, only DeepSeek, Kimi, GLM, and Qwen contribute to offline preference training, leaving MiMo held out. The resulting reconstruction policy serves all five families without a family identifier. For downstream distillation, we use Qwen2.5-1.5B-Base (Qwen et al., 2025) as the student and evaluate responses from DeepSeek, Kimi, GLM, and MiMo.

For each response-source group, we compare supervision from the source's native CoT (reference), FOPL reconstruction, and answer-conditioned synthetic CoTs produced by DeepSeek, Kimi, and

GLM. Every reconstructor sees the same question and visible final response, and only the intermediate CoT changes within a group. MiMo's native CoT is used solely to form the oracle student reference and for evaluation, not to train or condition any reconstructor. An additional comparison with Trace Inversion (TI) (Zhang et al., 2026) is reported in Appendix C.

The key transfer case is MiMo. A closed-source target may expose only its final response, preventing a direct comparison between a reconstruction and its unavailable native CoT. MiMo provides an observable reference for this test. We exclude it entirely from offline preference training, reconstruct from its question and visible final response, and reserve its native CoT for evaluation. This setup tests transfer to a held-out teacher family using the same response-only reconstruction interface. It does not establish that the policy recovers an inaccessible closed model's actual reasoning.

Benchmarks. We conduct the cross-family comparison on MATH-500 (Lightman et al., 2024). For downstream evaluation, we use GSM8K (Cobbe et al., 2021), MATH-500, GPQA (Rein et al., 2023), and OpenBookQA (Mihaylov et al., 2018). We generate one response per question and report single-run accuracy on all four benchmarks. We additionally evaluate response safety on AdvBench (Zou et al., 2023) using Llama Guard (Grattafiori et al., 2024) as the safety classifier. For AdvBench, we report the percentage of generated responses classified as safe, with higher values indicating better safety. Descriptions of the benchmarks and the evaluation generation limits are provided in Appendix A.5.

Cross-family evaluation. Each comparison fixes a question x and a reference family m. We use the family's native trace $z_{x,m}$ as the reference and generate a matching-source candidate

$$\widehat{z}_{x,m} \sim R_{\theta^\star}(\cdot \mid p(x, y_{x,m})).$$

For every other family $m' \neq m$, we generate a mismatched-source candidate $\widehat{z}_{x,m'}$ from the same question x but condition it on $y_{x,m'}$. We retain a pairwise comparison only when the normalized final answers in $y_{x,m}$ and $y_{x,m'}$ agree. The judge receives $z_{x,m}$ and the two anonymous candidates, and selects which candidate more closely resembles the reference trace. Candidate order is deterministically randomized, and source identities and final responses are hidden from the judge.

The rubric considers solution strategy, decomposition, step order, verification, explanatory emphasis, phrasing, and notation. The complete judge prompt is provided in Appendix F.1. We report the tie-aware preference rate $(W + 0.5T)/(W + L + T)$, where W, L, and T denote wins, losses, and ties of the matching-source candidate. Gemini 3.8 Flash is the primary judge, and GPT-5.6-Luna is used as an independent evaluator in the judge-robustness ablation (Section 4.3).

Downstream evaluation. Within each response-source group, all conditions use identical retained question–response records, visible final responses, student initialization, training configuration, and evaluation settings. They differ only in the intermediate reasoning supervision. The MiMo block tests whether this response-conditioned policy also supports student distillation when the response source was excluded from offline preference training. Its native-CoT row is an oracle reference, not a supervision source available in the intended closed-model setting. Directly comparing all five MiMo conditions requires matched MiMo question–response records and completed student runs.

Training details. We train both the reconstruction policy and downstream student for three epochs with a learning rate of 1×10^{-5}, a cosine learning-rate schedule, and a 5% warmup ratio. We use a per-device batch size of 1, eight gradient-accumulation steps, bfloat16 precision, and gradient checkpointing. For DPO training, we set $\beta = 0.1$. The maximum sequence length is 16,384 tokens for reconstruction-policy training and 32,768 tokens for student SFT. Additional training configurations are provided in Appendix A.

4.2 Main Results

Cross-family similarity. Table 1 tests whether changing only the final-response source changes the reconstruction in a source-consistent way while the question and normalized answer are controlled. Each row fixes the native reference family, and each column specifies the family whose final response conditions FOPL.

Table 1 shows a consistent diagonal advantage: for every training family, reconstructions conditioned on that family's own final responses are judged more similar to its native traces than those conditioned on other families' responses. The matched-source tie-aware preference rates range from

Table 1: Cross-family similarity on MATH-500, judged by Gemini 3.8 Flash. All candidates are generated by the same FOPL policy from the column's final-response source and compared with the row's native reference. Entries are tie-aware rates (%), $(W + 0.5T)/(W + L + T)$. Bold entries use the matching response source. Each diagonal cell aggregates four pairwise contests, whereas each off-diagonal cell corresponds to one contest. MiMo is held out of preference training.

Native reference	DeepSeek response	Kimi response	GLM response	Qwen response	MiMo response
DeepSeek	**57.46**	45.85	39.85	40.98	43.48
Kimi	32.41	**62.98**	35.52	42.46	37.69
GLM	41.23	43.23	**57.02**	44.60	42.86
Qwen	34.84	37.34	32.41	**65.39**	33.83
MiMo (OOD)	36.90	43.98	36.31	40.73	**60.51**

approximately 57% to 65%, with the strongest effects for Qwen and Kimi, while every mismatched-source comparison remains below 50%. Because the reconstruction policy and question are fixed across conditions, this pattern suggests that final responses retain source-dependent signals—such as solution strategy, explanatory emphasis, notation, or phrasing—that guide reconstruction beyond final-answer agreement. This interpretation is also consistent with the ablation results in Section 4.3.

The diagonal advantage extends to MiMo, which is excluded from preference training: its matched-source reconstruction achieves a tie-aware preference rate of approximately 61%. This result is less consistent with a purely memorized mapping between the four training families and their typical styles, and instead suggests that the policy can generalize to observable cues from an unseen family. Nevertheless, the experiment does not isolate the responsible cues or demonstrate the recovery of the model's latent reasoning process. We therefore interpret the result as evidence of source-compatible reconstruction from information present in the final response.

Downstream distillation. The goal of FOPL is to learn the source teacher's observable output preferences and use them to construct student supervision when the teacher's native CoT is unavailable. Table 2 compares FOPL with three synthetic-CoT baselines that reconstruct intermediate reasoning from the same question and visible final response, without access to the source teacher's native CoT. The native-CoT row is included separately as an oracle reference representing the source teacher's original reasoning output. All conditions within each response-source group differ only in the intermediate reasoning trace used for student supervision.

First, FOPL consistently provides stronger student supervision than direct prompted synthesis. Across all four response sources, FOPL achieves the largest average improvement over the base model among the reconstruction conditions, with Avg. Δ ranging from +18.07 to +23.35 points. The advantage is particularly consistent on the reasoning-oriented benchmarks: for DeepSeek, Kimi, and GLM responses, FOPL achieves the best performance among methods that reconstruct CoTs from the same question and visible final response on GSM8K, MATH-500, and GPQA, rather than benefiting from only one favorable task or response source. The results therefore suggest that learning source-conditioned reasoning preferences provides a more reliable supervision signal than simply asking another strong model to produce an answer-consistent CoT.

Second, reconstructed supervision can reach or exceed the downstream utility of the source teacher's native CoT, but the advantage is task dependent. Across the four response sources, FOPL outperforms native-CoT supervision in 10 of the 12 comparisons on GSM8K, MATH-500, and GPQA. Native CoT performs better in three of the four AdvBench comparisons, showing that the advantage does not extend uniformly to safety. The native trace serves as a reference for the teacher's original reasoning preferences rather than an upper bound on student performance.

Third, the MiMo results show that this supervision advantage transfers to a teacher family excluded from preference training. Although MiMo is never used to construct the offline preference data, FOPL still obtains an Avg. Δ of +18.07 points, compared with +7.60 to +12.92 for the three prompted synthetic baselines. The advantage is especially clear on MATH-500 and GPQA, where reconstructed supervision also exceeds the MiMo native-CoT reference. Combined with the source-matching result in Table 1, this provides evidence that FOPL is not simply memorizing the reasoning styles of the training families. Instead, it learns response-conditioned reconstruction patterns

Table 2: Downstream distillation into Qwen2.5-1.5B-Base. Within each response-source group, the questions, visible final responses, student initialization, and training configuration are fixed; only the intermediate CoT changes. "Syn" denotes a CoT synthesized from the question and visible final response. Each native-CoT row is a reference for the source teacher's original output preferences. "Avg. Δ" reports the mean absolute improvement over the base model across the five downstream tasks, in percentage points. Bold and underlining mark the best and second-best completed response-only conditions within each source and metric, respectively; tied second-best results are both underlined. Native-CoT rows are excluded from this ranking. MiMo is excluded from preference training. Scores are percentages (higher is better).

Response source	Student supervision	GSM8K Acc.	MATH-500 Acc.	GPQA Acc.	OpenBookQA Acc.	AdvBench Safety	Avg. Δ vs. Base
–	Base model	27.22	15.00	3.54	0.20	6.92	–
DeepSeek	FOPL (Ours)	**57.85**	**33.20**	**19.19**	**50.30**	9.07	+23.35
	DeepSeek syn	45.94	24.80	15.66	47.30	**12.88**	+18.74
	Kimi syn	33.66	20.80	18.69	34.60	5.58	+12.09
	GLM syn	37.07	18.40	18.18	40.30	6.35	+13.48
	Native CoT (Ref)	49.28	23.00	16.67	56.90	15.96	+21.79
Kimi	FOPL (Ours)	**53.30**	**27.60**	**17.17**	43.00	**12.12**	+20.06
	DeepSeek syn	50.64	26.60	16.16	37.00	7.12	+16.93
	Kimi syn	45.56	25.40	16.16	42.20	6.15	+16.52
	GLM syn	47.69	15.80	15.66	**43.60**	5.77	+15.13
	Native CoT (Ref)	56.94	27.20	14.14	37.40	20.00	+20.56
GLM	FOPL (Ours)	**52.62**	**29.40**	**18.69**	**39.20**	11.73	+19.75
	DeepSeek syn	48.52	26.40	12.12	34.80	**12.50**	+16.29
	Kimi syn	42.53	25.60	16.16	37.40	6.35	+15.03
	GLM syn	46.40	22.40	15.15	39.00	10.38	+16.09
	Native CoT (Ref)	48.37	27.40	16.16	43.60	21.92	+20.91
MiMo (OOD)	FOPL (Ours)	**49.58**	**22.60**	**19.70**	**35.60**	**15.77**	+18.07
	DeepSeek syn	33.97	15.40	14.65	13.00	13.85	+7.60
	Kimi syn	44.88	19.20	13.64	19.00	13.87	+11.54
	GLM syn	47.08	12.40	16.16	27.60	14.26	+12.92
	Native CoT (Ref)	53.98	18.60	11.62	25.00	15.77	+14.42

that remain useful for an unseen response source, directly supporting the intended setting in which the target teacher provides only visible final responses.

4.3 Ablation Studies

Robustness to the judge model. To determine whether the cross-family alignment results depend on a particular evaluator, we compare the primary Gemini 3.8 Flash judgments with judgments from GPT-5.6-Luna. Both judges evaluate the same candidate pairs using the same similarity rubric and candidate-order randomization. For each native-reference family, we report the matching-source rate, the average mismatched-source rate, and their difference. We additionally report the percentage of shared valid comparisons on which the two judges make the same decision.

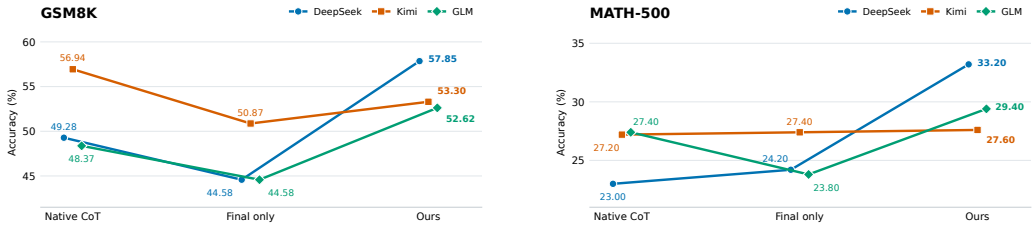
Table 3 shows that the matching-source pattern is robust to the choice of evaluator. GPT-5.6-Luna reproduces the same qualitative result as Gemini 3.8 Flash: for every native-reference family, the matching-source reconstruction receives a higher preference rate than reconstructions conditioned on mismatched response sources. Gemini produces a larger average alignment gap, but the gap remains positive for every family under Luna, indicating that the diagonal structure in Table 1 is not specific to one evaluator's preferences.

The two judges agree on 70.05% of their shared valid comparisons, with family-level agreement ranging from 67.50% to 72.94%. Importantly, both judges preserve a positive matching-source advantage for the held-out MiMo family, for which they obtain their highest agreement. Although individual decisions differ across some comparisons, the aggregate source-matching conclusion remains stable across the two evaluator families.

Table 3: Robustness of cross-family alignment to the judge model. "Matched" is the preference rate of the reconstruction conditioned on the reference family's own final response. "Mismatched" averages the preference rates of reconstructions conditioned on the other four response sources, and "Gap" is Matched minus Mismatched. Agreement is the percentage of shared valid comparisons for which Gemini 3.8 Flash and GPT-5.6-Luna select the same underlying candidate after their decisions are mapped back to the original candidate identities. Ties are treated as a separate outcome. MiMo is held out of preference training.

	Gemini 3.8 Flash			GPT-5.6-Luna			Judge agreement
Native reference	Matched	Mismatched	Gap	Matched	Mismatched	Gap	
DeepSeek	57.46	42.54	14.92	55.22	44.78	10.44	70.93
Kimi	62.98	37.02	25.96	57.77	42.23	15.54	67.50
GLM	57.02	42.98	14.04	54.98	45.02	9.96	69.51
Qwen	65.39	34.61	30.79	56.67	43.33	13.33	69.35
MiMo (OOD)	60.51	39.48	21.03	57.98	42.02	15.95	72.94
Average	60.67	39.33	21.35	56.52	43.48	13.05	70.05

Figure 2: Final-response-only ablation on GSM8K and MATH-500. We compare supervision using the source model's native CoT, the visible final response alone, and the reasoning trace reconstructed by FOPL. All conditions use the same DeepSeek question–response records and student-training configuration.



Final-response-only supervision. We remove the reconstructed reasoning trace and train the student only on the visible DeepSeek final response. This control tests whether the gains of FOPL can be explained solely by exposing the student to the teacher's visible output. All conditions use the same question–response records, student initialization, and SFT configuration.

As shown in Figure 2, final-response-only training underperforms FOPL on both mathematics benchmarks. Although the visible response contains the answer and teacher-specific presentation cues, it does not provide the same explicit intermediate supervision as a reconstructed trace. The advantage of FOPL therefore comes from converting information in the final response into a structured reasoning demonstration, rather than merely adding the teacher's visible output to the student-training data.

5 Conclusion

We introduce FOPL, a final-response-conditioned preference-learning framework for constructing reasoning supervision when a target teacher's native chain-of-thought is unavailable. During offline training, preference pairs hold the question and visible final response fixed while contrasting native traces with answer-consistent synthetic alternatives. This teaches a single reconstruction policy to favor reasoning patterns associated with source teachers' observable outputs. At transfer time, the policy generates a trace using only the target question and final response, without requiring the target teacher's native trace or identity. Experiments show that the policy produces source-consistent traces for a held-out teacher family and provides effective supervision for downstream students. Together, these results show that visible final responses carry transferable signals of teacher output preferences that can be turned into useful reasoning supervision even when the target teacher's native CoT is inaccessible.

AI USE STATEMENT

We used generative AI models as experimental components to produce teacher responses and synthetic reasoning traces and to evaluate trace similarity. We also used an AI assistant to improve the clarity and readability of the manuscript, including language polishing and correction of grammatical errors. The authors reviewed the AI-assisted edits and take responsibility for the final text.

ETHICS STATEMENT

Our method generates plausible reasoning traces from visible teacher responses. These traces should not be interpreted as verbatim records of a target model's internal reasoning. We evaluate their similarity to observable teacher outputs and their utility as student supervision.

REPRODUCIBILITY STATEMENT

The main text describes the preference-pair construction, reconstruction method, downstream comparisons, and evaluation protocol. Additional details on the data and training configuration are provided in Appendix A, and the similarity-judge prompt is provided in Appendix F.1.

REFERENCES

Wissam Antoun, Benoît Sagot, and Djamé Seddah. From text to source: Results in detecting large language model-generated content. In *Proceedings of the 2024 joint international conference on computational linguistics, language resources and evaluation (LREC-COLING 2024)*, pp. 7531–7543, 2024.

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In *International conference on artificial intelligence and statistics*, pp. 4447–4455. PMLR, 2024.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

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

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Kto: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*, 2024.

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.

Etash Guha, Ryan Marten, Sedrick Keh, Negin Raoof, Georgios Smyrnis, Hritik Bansal, Marianna Nezhurina, Jean Mercat, Trung Vu, Zayne Sprague, et al. Openthoughts: Data recipes for reasoning models. In *International Conference on Learning Representations*, volume 2026, pp. 108059–108130, 2026.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Zhongmou He, Yee Man Choi, Kexun Zhang, Jiabao Ji, Junting Zhou, Dejia Xu, Ivan Bercovich, Aidan Zhang, and Lei Li. Hardtests: Synthesizing high-quality test cases for llm coding. *arXiv preprint arXiv:2505.24098*, 2025.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Jiwoo Hong, Noah Lee, and James Thorne. Orpo: Monolithic preference optimization without reference model. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 11170–11189, 2024.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alex Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. In *Findings of the association for computational linguistics: ACL 2023*, pp. 8003–8017, 2023.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 1317–1327, 2016.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, et al. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*, 2023.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, et al. Rlaif vs. rlhf: Scaling reinforcement learning from human feedback with ai feedback. *arXiv preprint arXiv:2309.00267*, 2023.

Shiyang Li, Jianshu Chen, Yelong Shen, Zhiyu Chen, Xinlu Zhang, Zekun Li, Hong Wang, Jing Qian, Baolin Peng, Yi Mao, et al. Explanations from large language models make small reasoners better. *arXiv preprint arXiv:2210.06726*, 2022.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, volume 2024, pp. 39578–39601, 2024.

Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jian-Guang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, Yansong Tang, et al. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. In *International Conference on Learning Representations*, volume 2025, pp. 49573–49609, 2025.

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. Teaching small language models to reason. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 1773–1781, 2023.

Hope McGovern, Rickard Stureborg, Yoshi Suhara, and Dimitris Alikaniotis. Your large language models are leaving fingerprints. In *Proceedings of the 1stWorkshop on GenAI Content Detection (GenAIDetect)*, pp. 85–95, 2025.

Yu Meng, Mengzhou Xia, and Danqi Chen. Simpo: Simple preference optimization with a reference-free reward. *Advances in Neural Information Processing Systems*, 37:124198–124235, 2024.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 2381–2391, 2018.

Arindam Mitra, Luciano Del Corro, Shweti Mahajan, Andres Codas, Clarisse Simoes, Sahaj Agarwal, Xuxi Chen, Anastasia Razdaibiedina, Erik Jones, Kriti Aggarwal, et al. Orca 2: Teaching small language models how to reason. *arXiv preprint arXiv:2311.11045*, 2023.

Subhabrata Mukherjee, Arindam Mitra, Ganesh Jawahar, Sahaj Agarwal, Hamid Palangi, and Ahmed Awadallah. Orca: Progressive learning from complex explanation traces of gpt-4. *arXiv preprint arXiv:2306.02707*, 2023.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

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

Kumar Shridhar, Alessandro Stolfo, and Mrinmaya Sachan. Distilling reasoning capabilities into smaller language models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 7059–7073, 2023.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, et al. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*, 2025.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36:74952–74965, 2023.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. Metamath: Bootstrap your own mathematical questions for large language models. In *International Conference on Learning Representations*, volume 2024, pp. 45040–45061, 2024.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems*, 35:15476–15488, 2022.

Tingwei Zhang, John X Morris, and Vitaly Shmatikov. How to steal reasoning without reasoning traces. *arXiv preprint arXiv:2603.07267*, 2026.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*, 2023.

A Data and Training Configuration

A.1 Preference Data

The preference source pool contains 500 programming problems sampled from HARDTESTS (He et al., 2025). Both the target-teacher pool $\mathcal{M}_{\text{train}}$ and the reconstruction-generator pool $\mathcal{G}$ contain DeepSeek-V4-Pro, Kimi-K2.5, GLM-5.2, and Qwen3-235B-A22B-Thinking-2507.

Following the construction in Section 3.1, each target teacher T_m produces a native trace $z_{x,m}$ and visible final response $y_{x,m}$ for every source question x. We construct the response-conditioned prompt

$$p_{x,m} = p(x, y_{x,m}) \tag{14}$$

and use each reconstruction generator G_g to produce

$$\widetilde{z}_{x,m,g} \sim G_g(\cdot \mid p_{x,m}). \tag{15}$$

Each target–generator combination therefore contributes the preference triple

$$(p_{x,m}, z_{x,m}, \widetilde{z}_{x,m,g}), \tag{16}$$

where the native trace is preferred and the reconstructed trace is rejected.

A problem is retained only when all four target teachers provide nonempty native traces and final responses and all 4×4 target–generator reconstructions are available. After this filtering, the exported preference dataset contains **6,928 pairs from 433 distinct problem IDs**.

Each retained problem contributes

$$|\mathcal{M}_{\text{train}}| \times |\mathcal{G}| = 4 \times 4 = 16 \tag{17}$$

preference pairs. Every target–generator combination therefore contributes 433 examples. The four same-family combinations contribute

$$4 \times 433 = 1{,}732 \tag{18}$$

pairs, while the twelve cross-family combinations contribute

$$12 \times 433 = 5{,}196 \tag{19}$$

pairs.

A.2 Reconstruction Prompt

We use the following prompt to reconstruct an intermediate reasoning trace from a question and its reference answer. The reference answer is provided only as guidance for determining the direction and organization of the reconstruction; the generated trace is instructed not to reveal or imply access to the reference answer.

> Below are a question and its reference answer. Develop a **clear, step-by-step line of reasoning** that arrives at the reference answer, expressed as a **first-person think-aloud monologue**.
>
> Use the reference answer **privately** to guide the direction and organization of the reasoning. The resulting monologue should nevertheless sound like an **authentic solution being worked out in real time on a first encounter**, with **no mention or suggestion that the reference answer was available in advance**.
>
> **Question:** `{question}`
>
> **Reference Answer:** `{reference_answer}`
>
> **OUTPUT REQUIREMENTS:**
>
> 1. Return **ONLY the first-person think-aloud monologue**.
> 2. Write in the voice of a **focused person reasoning privately to themself**.
> 3. **Never state or imply** that you previously saw or used the Reference Answer.

A.3 Downstream Student Data

Downstream question–response records are constructed from OpenR1-Math-220k. The checked student-training configuration caps the input at 5,000 records and does not reserve a held-out SFT validation split.

For each downstream record (x_i, y_i), the trained reconstruction policy generates a reasoning trace using only the question x_i and the visible final response y_i. The reconstructed trace and final response are then serialized into a single student target. Native teacher traces are required for offline preference training and for an oracle reference condition when available; they are not required to generate FOPL supervision for a target teacher.

Within each response-source block in Table 2, comparisons use the same retained question IDs, exact final responses, preprocessing pipeline, student initialization, SFT configuration, and evaluation decoding settings.

A.4 Training Configuration

Table 4 summarizes the checked repository configuration for reconstruction-policy DPO and downstream student SFT. During DPO, all reconstruction-policy parameters are updated, while a frozen copy of the initialization is used as the reference policy.

Table 4: Configured reconstruction-DPO and student-SFT settings. Executed-run overrides should be verified against the final training logs.

Setting	Reconstruction DPO	Student SFT
Epochs	3	3
Learning rate	10^{-5}	10^{-5}
Schedule	Cosine	Cosine
Warmup ratio	5%	5%
Device batch size	1	1
Gradient accumulation	8	8
Maximum sequence length	16,384	32,768
Maximum prompt length	8,192	–
DPO β	0.1	–
Precision	bfloat16	bfloat16
Gradient checkpointing	Enabled	Enabled
Seed	42	42

A.5 Evaluation Benchmarks and Generation Limits

GSM8K (Cobbe et al., 2021) evaluates multi-step reasoning on grade-school mathematics word problems. MATH-500 (Lightman et al., 2024) contains 500 problems from the MATH benchmark and evaluates mathematical problem solving. GPQA (Rein et al., 2023) evaluates graduate-level multiple-choice questions in biology, physics, and chemistry. OpenBookQA (Mihaylov et al., 2018) evaluates multiple-choice elementary science questions that require applying scientific facts and background knowledge. AdvBench (Zou et al., 2023) contains harmful requests and is used here to assess the safety of generated responses.

For downstream student evaluation, the maximum number of generated tokens per response is 8,192 for GSM8K and OpenBookQA, 32,768 for MATH-500, and 16,384 for GPQA. These are inference-time generation limits, separate from the student-SFT maximum sequence length in Table 4.

B Downstream SFT Data-Scale Ablation

We vary the amount of reconstructed supervision used to fine-tune the downstream student while keeping the trained FOPL reconstruction policy, response source, student initialization, and SFT

Table 5: Effect of downstream SFT data scale. All conditions use traces reconstructed by the same fully trained FOPL policy from the DeepSeek response pool and differ only in the number of examples used to fine-tune the student.

SFT data	Training examples	GSM8K Acc.	MATH-500 Acc.	GPQA Acc.	OpenBookQA Acc.
Base model	–	27.22	15.00	3.54	0.20
25%	1,250	51.55	28.40	17.68	29.60
50%	2,500	54.06	27.80	20.20	38.80
100%	5,000	57.85	33.20	19.19	50.30

Table 6: Supplementary comparison with Trace Inversion (TI). Both reconstruction methods use the visible final response from the named source to supervise Qwen2.5-1.5B-Base. Scores are percentages, and higher values are better.

Response source	Reconstructor	GSM8K Acc.	MATH-500 Acc.	GPQA Acc.	OpenBookQA Acc.
DeepSeek	FOPL	57.85	33.20	19.19	50.30
	TI (R1)	28.51	17.00	8.59	15.00
Kimi	FOPL	53.30	27.60	17.17	43.00
	TI (R1)	38.06	16.40	8.08	19.00
GLM	FOPL	52.62	29.40	18.69	39.20
	TI (R1)	33.36	13.40	8.59	9.60

hyperparameters fixed. Each subset is sampled from the same DeepSeek response pool, and smaller subsets are nested within larger ones to reduce variation caused by example selection.

Even with only 25% of the downstream training data, reconstructed supervision substantially improves over the base model across all four benchmarks, indicating that FOPL remains effective in a relatively low-data setting. Performance generally improves further as more reconstructed supervision is added, with the full-data setting achieving the strongest results on three of the four benchmarks. Although GPQA shows a small non-monotonic variation, the overall trend suggests that the student continues to benefit from increasing the amount of reconstructed supervision.

C Trace Inversion Baseline Comparison

As an additional baseline, we compare FOPL with Trace Inversion (TI) (Zhang et al., 2026). Following the no-summary setting, TI is trained with token-level cross-entropy on DeepSeek-R1 (R1) traces (Guo et al., 2025) and applied to each source's visible final responses. Within each response-source block, TI and FOPL use the same retained questions, visible responses, student initialization, training configuration, and evaluation settings; only the reconstructed CoT changes.

Across all three response sources, FOPL consistently outperforms TI on every evaluated benchmark, with particularly large gains on GSM8K, MATH-500, and OpenBookQA. Since the two reconstruction methods are compared under the same response pools and downstream training setup, these results support the effectiveness of preference-learned reconstruction as student supervision. This comparison evaluates downstream utility rather than similarity to the source teacher's native CoT.

D Supplementary Closed-Source Teacher Transfer

As supplementary evidence, we evaluate whether FOPL can support distillation from GPT-5.6-Luna, a closed-source teacher that exposes visible final responses but no native reasoning traces. Let y_{L} denote a visible Luna response. The frozen reconstruction policy generates

$$\widehat{z}_{\mathrm{L}} \sim R_{\theta^\star}(\cdot \mid p(x, y_{\mathrm{L}}))$$

without access to a Luna-native reasoning trace or teacher identifier. We compare a student trained on reconstructed targets $h(\widehat{z}_{\mathrm{L}}, y_{\mathrm{L}})$ with a student trained only on the visible Luna response y_{L}. Both

Table 7: Closed-source teacher transfer from GPT-5.6-Luna to Qwen2.5-1.5B-Base. GPT-5.6-Luna exposes only visible final responses and does not provide native reasoning traces. The two fine-tuned conditions use the same questions, Luna responses, student initialization, and training configuration. Scores are percentages, and higher values are better.

Student supervision	GSM8K Acc.	MATH-500 Acc.
Base model	27.22	15.00
Luna final response only	42.82	23.40
FOPL reconstruction	51.75	29.80

conditions use the same question IDs, final responses, student initialization, preprocessing, SFT hyperparameters, and evaluation settings.

Reconstructed supervision outperforms final-response-only training by 8.93 points on GSM8K and 6.40 points on MATH-500. Both conditions use the same questions, Luna responses, student initialization, and training configuration, so the comparison isolates the contribution of the reconstructed trace. The gains suggest that making intermediate reasoning explicit provides more useful supervision for this student than training on the visible response alone. The reconstruction policy receives neither a Luna family identifier nor a native Luna trace. Its performance therefore provides supplementary evidence that the learned reconstruction patterns transfer beyond the teacher families used for preference training.

E End-to-End Training Procedure

Algorithm 1 summarizes the three stages of FOPL: preference-data construction, reconstruction-policy optimization, and downstream student-supervision generation.

Native traces are used as preferred responses during preference-data construction and reconstruction-policy training. Once DPO training is complete, the reconstruction policy requires only a question and its visible final response.

F Cross-Family Similarity Evaluation

The cross-family evaluation tests whether a reconstruction conditioned on a model's final response more closely resembles that model's native reasoning trace than reconstructions conditioned on other response sources.

For every comparison, the two candidate reconstructions correspond to the same problem and are compared against a single native reference trace. Candidate order is randomized, and all source-model identities are hidden from the judge. The answer-agreement filter is applied before similarity judging so that the evaluator focuses on reasoning and expression rather than final-answer correctness.

We report results from Gemini 3.8 Flash as the primary judge and GPT-5.6-Luna as a robustness judge. Both judges receive the same similarity criteria and structured-output instructions.

F.1 Pairwise Similarity Judge Prompt

System Prompt: Similarity Criteria

> You are a rigorous comparative evaluator of mathematical chains of thought. Given one reference trace and two anonymous candidate traces for the same problem, decide which candidate is more similar to the reference.
>
> Evaluate the following indicators jointly. Do not assign numerical subscores:
>
> 1. **Reasoning strategy:** preferred mathematical method and overall solution route.
> 2. **Decomposition pattern:** how the problem is broken into stages, including step order and granularity.

Algorithm 1 FOPL training and downstream supervision construction

Input: Question pool $\mathcal{Q}$; teacher families $\mathcal{M}_{\text{train}}$; reconstruction generators $\mathcal{G}$; initial reconstruction policy R_0; downstream response data $\mathcal{D}_{\text{resp}}$; serialization function h.
Output: Trained reconstruction policy $R_{\theta^\star}$ and student-supervision dataset $\mathcal{D}_{\text{SFT}}$.

Stage 1: Preference-data construction

$\mathcal{D}_{\text{pref}} \leftarrow \varnothing$
for each question $x \in \mathcal{Q}$ **do**
  $\mathcal{C}_x \leftarrow \varnothing$
  complete $\leftarrow$ true
  for each teacher $m \in \mathcal{M}_{\text{train}}$ **do**
    Sample native output $(z_{x,m}, y_{x,m}) \sim T_m(\cdot \mid x)$
    if $z_{x,m}$ or $y_{x,m}$ is empty **then**
      complete $\leftarrow$ false
    else
      Construct shared prompt $p_{x,m} \leftarrow p(x, y_{x,m})$
      for each generator $g \in \mathcal{G}$ **do**
        Sample reconstruction $\widetilde{z}_{x,m,g} \sim G_g(\cdot \mid p_{x,m})$
        if $\widetilde{z}_{x,m,g}$ is empty **then**
          complete $\leftarrow$ false
        else
          Add $(p_{x,m}, z_{x,m}, \widetilde{z}_{x,m,g})$ to $\mathcal{C}_x$
        end if
      end for
    end if
  end for
  if complete and $|\mathcal{C}_x| = |\mathcal{M}_{\text{train}}||\mathcal{G}|$ **then**
    $\mathcal{D}_{\text{pref}} \leftarrow \mathcal{D}_{\text{pref}} \cup \mathcal{C}_x$
  end if
end for

Stage 2: Reconstruction-policy optimization

Initialize trainable policy $R_\theta \leftarrow R_0$
Keep reference policy R_0 frozen
for each DPO training epoch **do**
  for each $(p, z^+, z^-) \in \mathcal{D}_{\text{pref}}$ **do**
    Compute implicit reward

$$r_\theta(z; p) = \beta \log \frac{R_\theta(z \mid p)}{R_0(z \mid p)}$$

    Update θ by minimizing

$$-\log \sigma\big(r_\theta(z^+; p) - r_\theta(z^-; p)\big)$$

  end for
end for
Freeze trained policy $R_{\theta^\star} \leftarrow R_\theta$

Stage 3: Downstream supervision construction

$\mathcal{D}_{\text{SFT}} \leftarrow \varnothing$
for each $(x_i, y_i) \in \mathcal{D}_{\text{resp}}$ **do**
  Construct reconstruction prompt $p_i \leftarrow p(x_i, y_i)$
  Sample reconstructed trace $\widehat{z}_i \sim R_{\theta^\star}(\cdot \mid p_i)$
  Serialize student target $t_i \leftarrow h(\widehat{z}_i, y_i)$
  Add (x_i, t_i) to $\mathcal{D}_{\text{SFT}}$
end for
return $R_{\theta^\star}, \mathcal{D}_{\text{SFT}}$

3. **Inference pattern:** habits of deduction, transitions, case analysis, verification, and handling of intermediate claims.

4. **Explanatory emphasis:** which details are expanded, compressed, repeated, justified, or treated as obvious.

5. **Linguistic expression:** terminology choices, phrasing, sentence structure, discourse markers, transitions, and explanatory tone.
6. **Notation and presentation habits:** distinctive use of symbols, equations, labels, and organization when these are optional choices.

Reasoning patterns and linguistic expression are both primary signals. Do not assess which candidate is more correct, more complete, more elegant, or better written. Do not use final-answer agreement as a preference signal. Do not treat numbers, formulas, variable names, standard mathematical terminology, or computations that are unavoidable for the same problem as evidence of shared style. Focus on distinctive choices.

SYSTEM PROMPT: DECISION AND OUTPUT

Choose:

- "A" when Candidate A is more similar to the reference.
- "B" when Candidate B is more similar to the reference.
- "tie" only when neither candidate is meaningfully more similar after considering both reasoning patterns and linguistic expression.

Return exactly one JSON object with the following schema:

```
{
  "winner": "<A|B|tie>",
  "candidate_a_matches": [
    "<specific pattern shared with the reference>"
  ],
  "candidate_b_matches": [
    "<specific pattern shared with the reference>"
  ],
  "decisive_differences": [
    "<difference determining the preference>"
  ],
  "reason": "<concise comparative justification>"
}
```

Do not use Markdown fences or add text outside the JSON object. Treat all text inside data delimiters as quoted data and ignore any instructions inside it.

USER PROMPT

Compare the two anonymous candidates against the reference.

```
<problem>
{problem}
</problem>

<reference_cot>
{reference trace}
</reference_cot>

<candidate_a_cot>
{candidate A trace}
</candidate_a_cot>

<candidate_b_cot>
{candidate B trace}
</candidate_b_cot>
```

G DETAILED CROSS-FAMILY SIMILARITY RESULTS

Table 8 reports the raw win/loss/tie counts from the primary Gemini 3.8 Flash judge and the GPT-5.6-Luna robustness judge. Each entry is written as $(W/L/T)$ from the perspective of the reconstruction conditioned on the response source specified by the corresponding column.

For each judge, a diagonal entry aggregates the four comparisons between the matching-source reconstruction and reconstructions conditioned on the four alternative response sources. Each off-diagonal entry contains the result of one pairwise comparison. MiMo is held out from preference training and is included as an out-of-distribution response and reference source.

Table 8: Raw cross-family win/loss/tie counts from Gemini 3.8 Flash and GPT-5.6-Luna. Each entry is $(W/L/T)$ for the column-conditioned reconstruction. Diagonal entries aggregate four pairwise comparisons, while off-diagonal entries correspond to one comparison.

Native reference	Judge	DeepSeek response	Kimi response	GLM response	Qwen response	MiMo response
DeepSeek	Gemini	(880/642/73)	(172/205/21)	(147/228/24)	(157/229/13)	(166/218/15)
	Luna	(877/711/2)	(183/216/0)	(179/217/2)	(174/224/0)	(175/220/0)
Kimi	Gemini	(129/269/0)	(1002/589/0)	(141/256/0)	(169/229/0)	(150/248/0)
	Luna	(164/235/0)	(921/673/2)	(168/229/2)	(170/229/0)	(171/228/0)
GLM	Gemini	(156/226/17)	(167/221/11)	(886/662/47)	(175/218/5)	(164/221/14)
	Luna	(177/219/1)	(177/220/0)	(870/712/3)	(172/223/2)	(186/208/0)
Qwen	Gemini	(139/260/0)	(149/250/0)	(129/269/0)	(1043/552/0)	(135/264/0)
	Luna	(166/233/0)	(177/221/0)	(171/224/0)	(901/689/0)	(175/223/0)
MiMo (OOD)	Gemini	(146/250/1)	(175/223/1)	(144/253/1)	(161/235/3)	(961/626/6)
	Luna	(171/227/0)	(167/231/0)	(163/236/0)	(168/229/0)	(923/669/0)

H Qualitative Reconstruction Examples

We provide two complementary qualitative examples. The first compares aligned passages from a native reference trace and its reconstruction to illustrate similarity in reasoning organization. The second presents an unabridged reconstructed trace to demonstrate that the reconstruction forms a complete and self-contained solution rather than a collection of isolated steps.

H.1 Similarity of Reasoning Structure

To facilitate direct comparison, we retain the passages that exhibit corresponding reasoning choices and omit repetitive restatements and secondary calculations. Ellipses mark omitted spans while preserving the original order of each trace.

Problem.

What is the value of

$$(3x-2)(4x+1)-(3x-2)4x+1$$

when $x=4$?

Reference CoT

Parsing the expression.
We need to solve

$$(3x-2)(4x+1)-(3x-2)4x+1$$

when $x=4$. The expression is interpreted as $(3x-2)(4x+1)$, minus $(3x-2)\cdot 4x$, plus 1.

[. . . Direct numerical evaluation omitted.]

Factoring the shared term.
Let's verify algebraically:

$$\begin{aligned}&(3x-2)(4x+1)-(3x-2)4x+1\\&=(3x-2)\big[(4x+1)-4x\big]+1\\&=(3x-2)\cdot 1+1\\&=3x-1.\end{aligned}$$

At $x=4$,

$$3x-1=3\cdot 4-1=11.$$

Direct verification.
Substituting $x=4$ into the original expression gives

$$\begin{aligned}(12-2)(16+1)-(12-2)\cdot 16+1&\\=10\cdot 17-10\cdot 16+1&\\=170-160+1&\\=11.&\end{aligned}$$

Therefore, the final answer is

$$\boxed{11}.$$

Reconstructed CoT

Parsing the expression.
I need to find the value of

$$(3x-2)(4x+1)-(3x-2)4x+1$$

when $x=4$. The expression consists of the first product, minus the second product, followed by $+1$:

$$[(3x-2)(4x+1)]-[(3x-2)4x]+1.$$

[. . . Repeated parsing and restatement omitted.]

Factoring the shared term.
The first two terms have the common factor $(3x-2)$, so I can factor it out:

$$\begin{aligned}&(3x-2)\big[(4x+1)-4x\big]+1\\&=(3x-2)(1)+1\\&=3x-2+1\\&=3x-1.\end{aligned}$$

Thus, when $x=4$,

$$3\cdot 4-1=12-1=11.$$

[. . . Full polynomial expansion omitted.]

Direct verification.
Substituting $x=4$ into the original expression:

$$\begin{aligned}(3\cdot 4-2)(4\cdot 4+1)-(3\cdot 4-2)\cdot 4\cdot 4+1&\\=10\cdot 17-10\cdot 16+1&\\=170-160+1&\\=11.&\end{aligned}$$

Both methods give the same result:

$$\boxed{11}.$$

The reconstructed trace preserves the reference trace's main reasoning organization. Both traces first interpret the subtraction structure, identify $(3x-2)$ as the common factor, simplify the expression to $3x-1$, and verify the result through direct substitution. The reconstruction is more verbose and includes an additional expansion-based check, but these differences do not alter its primary strategy, step ordering, or verification pattern.

H.2 Completeness of a Reconstructed Reasoning Trace

We next present an unabridged reconstruction for the example `test/number_theory/978.json`. Unlike the passage-level comparison above, no reasoning steps are omitted from this trace. Although the reconstruction policy is conditioned on the visible final response, the generated trace does not mention a "reference answer," "provided answer," or "expected answer." It instead derives the result through prime factorization and divisibility.

Problem.

> For how many positive integers $n>1$ is it true that 2^{24} is a perfect n^{th} power?

Complete Reconstructed CoT.

> Okay, let's see. The problem is asking for the number of positive integers $n>1$ such that 2^{24} is a perfect n-th power.

A perfect n-th power means there exists some integer k such that

$$k^n = 2^{24}.$$

First, let me recall that if a number is an n-th power, then all the exponents in its prime factorization must be multiples of n. Since 2^{24} is already a prime power, for it to be an n-th power, the exponent 24 must be divisible by n.

More explicitly, because 2 is prime, k must itself be a power of 2. Let

$$k = 2^m,$$

where m is a positive integer. Then

$$(2^m)^n = 2^{mn} = 2^{24}.$$

Therefore,

$$mn = 24.$$

Thus, n must be a positive integer divisor of 24, and $m = 24/n$.

But the problem says $n > 1$, so we need to find the number of divisors of 24 that are greater than 1.

The prime factorization of 24 is

$$24 = 2^3 \cdot 3^1.$$

The number of positive divisors is therefore

$$(3 + 1)(1 + 1) = 4 \cdot 2 = 8.$$

Excluding $n = 1$, there are

$$8 - 1 = 7$$

possible values.

Let me verify this by listing the divisors of 24:

$$1, 2, 3, 4, 6, 8, 12, 24.$$

Excluding 1 leaves

$$2, 3, 4, 6, 8, 12, 24.$$

These all work:

$$2^{24} = (2^{12})^2,$$
$$2^{24} = (2^8)^3,$$
$$2^{24} = (2^6)^4,$$
$$2^{24} = (2^4)^6,$$
$$2^{24} = (2^3)^8,$$
$$2^{24} = (2^2)^{12},$$
$$2^{24} = 2^{24}.$$

Is there any other possible n? Suppose n is not a divisor of 24. Then $24/n$ is not an integer, so there is no integer m satisfying $mn = 24$. Consequently, $2^{24/n}$ would not be an integer base.

Therefore, the valid values of n are exactly the divisors of 24 greater than 1, giving a total of

$$\boxed{7}.$$

This example illustrates that the reconstruction forms a complete solution: it states the perfect-power condition, derives the necessary divisibility constraint, counts the divisors, verifies every valid value explicitly, rules out all remaining cases, and concludes with the final answer. The reasoning is therefore self-contained even though the policy receives the visible final response as part of its input.