A UNIFYING LENS ON FINE-TUNING THROUGH TARGET DISTRIBUTION DESIGN

Anonymous authors
Paper under double-blind review

ABSTRACT

Supervised fine-tuning (SFT) typically maximizes the likelihood of every token in a demonstrated trajectory. However, an observed token can be non-unique, noisy, or misaligned with the model prior. Strictly fitting toward this one-hot target may be suboptimal, especially when the pretrained model encodes a rich knowledge prior. In this work, we reinterpret SFT as **target distribution design**: instead of studying only the loss objective, we analyze the token-level target that a loss drives the model to match. We introduce the TARGET framework, which decomposes SFT supervision into two explicit choices: (1) **how strongly to rely on the observed token**, and (2) **how to allocate the remaining probability mass** over alternatives. This perspective unifies many existing SFT variants as implicit choices of the target distribution. Building on this view, we propose TARGET-SFT which directly constructs the desired target by adaptively balancing label imitation with a teacher-guided residual distribution. Across three datasets and seven models spanning math and medical reasoning, this method consistently outperforms all baselines, showing the effectiveness of this target-based approach. Overall, our formulation reveals a fundamental design perspective and opens a broader search space for training objectives.

1 INTRODUCTION

Supervised fine-tuning (SFT) is critical in the post-training of large language models (LLMs) (Zhang et al., 2025b; Chung et al., 2022; Ouyang et al., 2022). By imitating expert behaviors, SFT enables the pretrained model to acquire new knowledge efficiently. Nonetheless, standard SFT relies on a particularly rigid form of supervision, by training toward a one-hot target distribution: at every token, SFT maximizes probability of the observed token y_t, while all others are assigned zero probability. This implicitly assumes that every token in the dataset is an ideal and uniquely correct target.

This view reveals a limitation of standard SFT, especially in post-training (Gudibande et al., 2023; Li et al., 2025; 2026). In realistic data, an observed token is rarely the only valid continuation. The same prompt may admit multiple correct reasoning paths, phrasings, intermediate steps, or stylistic choices (Zhang et al., 2025a; Albalak et al., 2024; Yuan et al., 2023; Zhou et al., 2023; Liu et al., 2026; Wang et al., 2025). At the same time, the model also encodes a rich prior from pretraining (Li et al., 2026; Wu et al., 2026; Liu et al., 2026). Thus strictly imitating every token can amplify noise, induce overconfidence, interfere with the pretrained model prior, and degrade generalization (Chu et al., 2025; Shenfeld et al., 2025; Chen et al., 2025; Huang et al., 2025; Zhang et al., 2026). A growing line of work modifies the SFT objective, for example, through token-level reweighting (Wu et al., 2026; Liu et al., 2026; Ruan et al., 2025; Diao et al., 2026) or regularization (Li et al., 2025; Huang et al., 2025; Zhu et al., 2026a;b). These approaches are presented as separate algorithmic choices, leaving unclear how they relate and how to design training in a principled way.

In this work, we propose a different perspective: rather than the loss, we ask what **target distribution** should SFT drive the model to learn. This is more fundamental than the choice of loss alone, because loss is only an optimization surrogate, whereas the target directly specifies the desired allocation of probability mass (Figure 1). This view allows supervision to adapt to label quality: If y_t is optimal, the target should stay close to the one-hot distribution δ_{y_t} and maximize its probability. If it is noisy or misaligned with the model prior, then the target should soften and allocate probability

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.

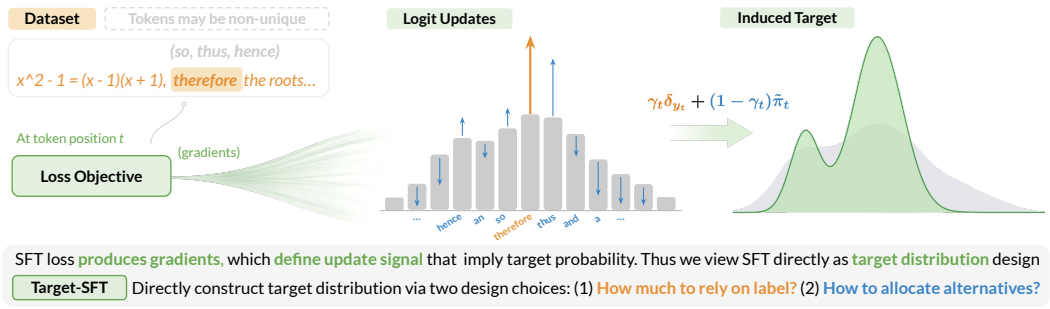

Figure 1: **Overview.** An SFT loss drives the model to match an implicitly defined target distribution. This view motivates TARGET-SFT that designs the SFT target directly. It also offers a unifying lens, where many SFT variants can be viewed as different target designs through the choices of γ_t and $\tilde{\pi}_t$.

to alternatives. Building on such intuition, we formalize this as the TARGET:

$$Q_t = \gamma_t \delta_{y_t} + (1 - \gamma_t)\tilde{\pi}_t,$$

where $\gamma_t \in [0, 1]$ controls the target probability assigned to y_t accounting for uncertainty, and $\tilde{\pi}_t$ specifies the plausible alternatives for the $1 - \gamma_t$ residual probability mass. Through the target distribution defined by $(\gamma_t, \tilde{\pi}_t)$, it reveals two key designs: (1) **how much to rely on the observed token** y_t, and (2) **where the remaining probability mass goes when** y_t **is uncertain**.

We further show that many SFT variants can be understood through these two questions; seemingly different losses correspond to implicit choices of $(\gamma_t, \tilde{\pi}_t)$, training towards a specific target Q_t. Thus beyond the algebraic loss forms, we can also directly construct the desired target for the model to learn. We propose TARGET-SFT as an example to illustrate this design value of our TARGET-framework. It explicitly defines the supervision structure revealed in TARGET where previous works largely leave implicit, and finds a bridge with distillation approaches involving teacher supervision.

In general, this TARGET perspective provides a unifying lens in SFT objective design, and exposes a general design space for balancing dataset imitation, prior preservation, and alternative supervision. Our contributions are as follows:

1. We propose the **TARGET-framework**, which interprets supervision as two explicit choices: how much to rely on the observed token, and how to allocate the remaining probability mass.
2. We show that any differentiable token-level SFT loss can be understood through the target distribution it drives the model to learn, thereby **unifying existing variants**.
3. We introduce **TARGET-SFT** as an instantiation that designs both choices explicitly, and demonstrate its performance gains across 10 dataset-model settings.

2 RELATED WORK

Existing works improve SFT along two main directions. We organize them under the TARGET lens, which understands variants by how they specify the effective token-level target distribution. We detail their concrete formulation of $(\gamma_t, \tilde{\pi}_t)$ for representative methods in Appendix C.

Token-level Reweighting. Standard SFT applies uniform cross-entropy updates to all tokens, treating every token as equally reliable. Token-reweighting methods challenge this assumption by changing how strongly each token contributes to training. DFT (Wu et al., 2026), beyond-log (Li et al., 2026), and ProFit (Liu et al., 2026) use the model probability on the observed token to rescale or filter updates, focusing on tokens that are compatible with the model prior. EAFT (Diao et al., 2026) uses entropy-based uncertainty to reduce potentially destructive updates; iw-SFT (Qin & Springenberg, 2025) and CFT (Ruan et al., 2025) assign weights based on trajectory- or token-level quality. In TARGET view, these methods primarily choose γ_t: *they weaken or strengthen imitation of the observed label but leave the remaining probability mass unspecified*. Our framework provides a complete view by making the effective training target explicit.

Distribution-level Prior. Another line of work introduces soft distributional signals beyond the one-hot label. Reference-constrained methods such as ASFT (Zhu et al., 2026a), RL's Razor (Shenfeld et al., 2025), and Proximal SFT (Zhu et al., 2026b) regularize updates to prevent large drift from a reference model. Huang et al. (2025) uses label-smoothing to address overconfidence. GEM (Li et al., 2025) uses reverse KL and entropy regularization to preserve output diversity and reduce forgetting. These methods address the rigidity in strict one-hot imitation, proposing alternative sources for probability allocation. They mainly specify the *choice of $\tilde{\pi}_t$, allocating probability mass to alternatives*: KL-constraints allocate toward the reference model, label smoothing allocates toward a uniform prior, and diversity-preserving methods discourage collapse onto a narrow set of tokens.

3 Preliminary

Supervised Fine-Tuning. Let $\mathcal{D}$ be a supervised dataset of pairs $(x, y) \sim \mathcal{D}$, where x is the input prompt and $y = (y_1, \ldots, y_T)$ is the observed response sequence. Given the prefix $x_t = (x, y_{<t})$, a language model defines the next-token distribution $\pi_\theta(\cdot \mid x_t) \in \Delta^{|\mathcal{V}|}$ over the vocabulary $\mathcal{V}$.

Standard SFT minimizes the token-level negative log-likelihood:

$$\mathcal{L}_{\mathrm{SFT}}(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[-\sum_{t=1}^{T} \log \pi_\theta(y_t \mid x_t)\right].$$

Target Distribution. Equivalently, let δ_{y_t} denote the one-hot vector that assigns probability 1 to the observed token y_t and 0 elsewhere, $\delta_{y_t}(v) = \mathbf{1}\{v = y_t\}$. The objective can be written as the cross-entropy with target δ_{y_t} as $\mathcal{L}_{\mathrm{SFT}}(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}} \sum_{t=1}^{T} \mathrm{CE}\big(\delta_{y_t}, \pi_\theta(\cdot \mid x_t)\big)$. This target δ_{y_t} specifies the logit-level learning signal, showing where each update moves probability.

4 Target Framework

The one-hot target δ_y in SFT implicitly assumes that y_t is the single optimal continuation for x_t, even when the observed token may be non-unique, noisy, or misaligned with the model. To capture this, we relax the assumption by replacing δ_{y_t} with a target Q_t that accounts for uncertainty in y_t.

4.1 Modeling Latent Trust

Let $Z_t \in \{0, 1\}$ be a latent binary variable indicating whether y_t should be strictly imitated ($Z_t = 1$) or supervision should relax to broader plausible alternatives ($Z_t = 0$). Under this view, the ideal target distribution to learn can be written as

$$P(\cdot \mid x_t) = P(Z_t = 1 \mid x_t)\delta_{y_t} + P(Z_t = 0 \mid x_t)\tilde{\pi}_t(\cdot \mid x_t), \tag{1}$$

where $\tilde{\pi}_t \in \Delta^{|\mathcal{V}|}$ denotes an alternative distribution over plausible next tokens.

Since Z_t is unobserved, the trust probability $r_t = P(Z_t = 1 \mid x_t)$ is unknown. We model this uncertainty using a Beta distribution:

$$r_t \sim \mathrm{Beta}(\alpha_t, \beta_t),$$

where α_t is evidence supporting y_t, and β_t is evidence that y_t may be non-unique or should relax toward alternatives. The posterior mean $\gamma_t = \mathbb{E}[r_t] = \frac{\alpha_t}{\alpha_t+\beta_t} \in [0, 1]$ then gives the expected trust for the observed token y_t.

Taking expectation over Eq. 1 leads to the ideal target distribution

$$Q_t = \mathbb{E}_{r_t}[P(\cdot \mid x_t)] = \gamma_t\delta_{y_t} + (1 - \gamma_t)\tilde{\pi}_t. \tag{2}$$

Intuitively, the target probability for y_t is scaled by the expected trust γ_t in the token, and the residual probability mass is reallocated to plausible alternatives over $\tilde{\pi}_t$.

4.2 Final Target & Objective

We replace the SFT one-hot target with Q_t, and train with the cross-entropy loss:

$$\mathcal{L}_Q(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}} \sum_{t=1}^{T} \mathrm{CE}(Q_t, \pi_\theta(\cdot \mid x_t)) .$$

Proposition 1 (Token-level decomposition of TARGET *training).* Given the target distribution Q_t defined in Eq. 2, the token-level cross-entropy loss at position t decomposes as

$$\mathrm{CE}(Q_t, \pi_\theta(\cdot \mid x_t)) = \gamma_t \,\mathrm{CE}(\delta_{y_t}, \pi_\theta(\cdot \mid x_t)) + (1-\gamma_t)\,\mathrm{CE}(\tilde{\pi}_t, \pi_\theta(\cdot \mid x_t)). \tag{3}$$

This loss derived from Q_t identifies two components of supervision: (1) label imitation, with strength controlled by the expected trust γ_t, and (2) residual distribution $\tilde{\pi}_t$ matching weighted by $1-\gamma_t$. In particular, the residual $\tilde{\pi}_t$ specifies how supervision is allocated when strict imitation is relaxed. See Appendix B.1 for proof. Together, this provides a common representation for comparing the supervision signals encoded by existing objectives.

5 Unifying Perspective

5.1 Existing Variants

The TARGET formulation separates token-level supervision into two design choices: (1) $\gamma_t \in [0,1]$ controls the target probability on y_t, while (2) $\tilde{\pi}_t \in \Delta^{|\mathcal{V}|}$ specifies how to allocate the residual probability mass. We show that this view unifies many training variants. Table 5 provides details of each variant discussed, and Table 6 summarizes their interpretation under TARGET framework.

Standard SFT. Consider the degenerate choice $\gamma_t = 1$, then the objective in Eq. 3 reduces to the negative log-likelihood in SFT and recovers $Q_t = \delta_{y_t}$. It is the special case that places full probability on every observed token y_t and assigns no residual mass to alternatives.

Token-Weighted Variants. A class of SFT variants modifies the objective by scaling the negative log-likelihood with a detached, per-token importance weight w_t (Li et al., 2026; Liu et al., 2026; Wu et al., 2026; Ruan et al., 2025; Diao et al., 2026; Qin & Springenberg, 2025):

$$\mathcal{L}_t = -w_t \log \pi_\theta(y_t \mid x_t),$$

where w_t may depend on model confidence, entropy, sample quality, or other token-level statistics.

Corollary 1 (Token weighting as self-residual target). Assume $w_t \in [0,1]$ is detached from the current update. The token-weighted loss corresponds to the choice

$$(\gamma_t = w_t,\ \tilde{\pi}_t = \mathrm{sg}[\pi_\theta(\cdot \mid x_t)]) \quad\Longrightarrow\quad Q_t = w_t\delta_{y_t} + (1-w_t)\mathrm{sg}[\pi_\theta(\cdot \mid x_t)] .$$

where $\mathrm{sg}[\cdot]$ denotes the stop-gradient operator. In particular, the residual branch $\tilde{\pi}_t$ is a *self-matching* term that contributes no gradient. See Appendix B.2 for proof.

Thus, token-weighted variants primarily specify γ_t that determines how strongly to imitate the observed token, and allocate the remaining mass $1-\gamma_t$ to the current model prior $\mathrm{sg}[\pi_\theta(\cdot \mid x_t)]$. As this self-matching residual contributes no gradient, reducing label trust weakens imitation without providing additional signal about how the distribution should change.

Distributional Variants. Another class of SFT variants incorporates distributional signals beyond the observed token, whether to regularize model drift, calibrate confidence, or preserve diversity (Li et al., 2025; Huang et al., 2025; Zhu et al., 2026a;b; Gu et al., 2026). These methods enrich hard-label imitation with a distribution q_t, with objective of the form

$$\mathcal{L}_t = -a_t \log \pi_\theta(y_t \mid x_t) + b_t\,\mathrm{CE}(q_t, \pi_\theta), \qquad a_t, b_t \geq 0. \tag{4}$$

Corollary 2 (Distributional supervision as residual in TARGET*).* Given a detached auxiliary distribution $q_t \in \Delta^{|\mathcal{V}|}$, distributional variants correspond to TARGET training with

$$\gamma_t = \frac{a_t}{a_t + b_t}, \qquad \tilde{\pi}_t = q_t, \quad\Longrightarrow\quad Q_t = \gamma_t\delta_{y_t} + (1-\gamma_t)\tilde{\pi}_t, \tag{5}$$

up to a global constant $a_t + b_t$. Therefore, these variants primarily specify the residual $\tilde{\pi}_t$, deciding where non-label probability should be allocated. The relative strength γ_t between label imitation and residual matching is determined by fixed hyperparameter. See Appendix B.3 for proof.

Knowledge Distillation Standard distillation arises as a similar instance, which provides distributional supervision through a teacher model's predictions. For the forward-KL formulation,

$$\mathcal{L}_{\mathrm{KD},t} = \mathrm{KL}(\pi_T(\cdot \mid x_t) \,\|\, \pi_\theta(\cdot \mid x_t)),$$

the gradient is equivalent to cross-entropy against the teacher distribution π_T. It therefore corresponds to simply $Q_t = \pi_T(\cdot \mid x_t)$, recovered by the choice $(\gamma_t, \tilde{\pi}_t) = (0, \pi_T)$. The hybrid formulation additionally incorporates label imitation, using $\gamma_t = c$ for a constant $c \in [0, 1]$. Importantly, this view connects and places distillation within the same target-design space as SFT variants.

In summary, token-weighted and distributional methods reveal the two complementary axes in our supervision design: (1) how strongly to imitate the observed token, and (2) how to allocate the remaining probability mass. Our target framework makes both choices explicit within a common representation, allowing them to be analyzed and designed jointly.

Remark This is a natural view because loss acts through gradients to redistribute probability across the vocabulary, which can be equivalently expressed as a target distribution. The induced target thus makes the supervision encoded by different loss forms explicit, providing a common interpretation. Section 5.2 formalizes this correspondence for arbitrary differentiable losses.

5.2 From Loss to Target

An SFT loss shapes the model's next-token distribution via gradient updates. For any differentiable token-level loss, we can thus ask *what target distribution Q_t does it drive the model to match*?

At a given token position t (notation omitted for clarity), let z denote the logits and $p = \mathrm{softmax}(z)$ the model distribution. Consider the cross-entropy toward a valid probability distribution Q, then its gradient with respect to the k-th logit is simply the prediction difference

$$\frac{\partial\, \mathrm{CE}(Q, p)}{\partial z_k} = p_k - Q_k.$$

This relationship can be inverted. Given any differentiable token-level loss $\mathcal{L}(z, x)$ with logit gradient $g_k = \frac{\partial \mathcal{L}}{\partial z_k}$, we can derive its induced target as

$$Q_k := p_k - g_k. \tag{6}$$

This Q is thus a target whose cross-entropy gradient $\nabla_z \mathrm{CE}(Q, p) = \nabla_z \mathcal{L}$, aligning exactly with the logit updates produced by $\mathcal{L}$. This correspondence therefore explicitly reveals the training signal encoded by the loss. More details and examples are included in Appendix D.

6 Target-SFT

Target further presents a new design space: instead of algebraic loss, we can *directly construct the target expressing how the model should learn*. We introduce Target-SFT as one example instantiation, using a learner-dependent proxy for label uncertainty and teacher-guided residual to enrich supervision. Its construction follows the goals to relax imitation while respecting the model prior, without limiting the acquisition of new knowledge.

Probability-Proxy for γ_t. The ideal target in Eq. 2 involves an expected trust $\gamma_t = \mathbb{E}[r_t] = \frac{\alpha_t}{\alpha_t + \beta_t}$, where α_t, β_t represent evidence (such as an empirical count) for the binary event y_t being selected given prefix x_t. However, such a count is intractable in SFT.

The model probability $p_y = \mathrm{sg}[\pi_\theta(y_t \mid x_t)]$ arises as a natural estimate for α_t, which encapsulates statistical evidence accumulated during pretraining. Here p_y measures the fraction of model's support assigned to y_t among all possible continuations. Equivalently, we may encode this belief as Beta evidence $\alpha_t = p_y$ and $\beta_t = 1 - p_y$, whose posterior mean resolves to

$$\gamma_t = \frac{p_y}{p_y + (1 - p_y)} = p_y.$$

This derivation motivates probability-weighted SFT variants such as p-loss (Wu et al., 2026; Li et al., 2026). Additionally, this choice of trust estimate makes supervision learner-dependent, reflecting how compatible each observed token is with model's pretrained knowledge.

Teacher-Guided $\tilde{\pi}_t$. However, model-based trust conflates label uncertainty with learner unfamiliarity. Under TARGET, such objectives implicitly use a *self-matching* residual, $\tilde{\pi}_t = \mathrm{sg}[\pi_\theta(\cdot \mid x_t)]$, which provides no corrective gradient as $p_y \to 0$. Since a small p_y may reflect either genuine ambiguity or valid knowledge yet to acquire, the $\gamma_t = p_y$ branch alone is insufficient.

This motivates a residual distribution that remains anchored to the model prior, but also allows external corrective signals. To this end, we construct a teacher-guided $\tilde{\pi}_t$. The goal is to preserve model prior without being constrained by it, enabling supervision from teacher-supported alternatives.

We construct $\tilde{\pi}_t$ through KL-regularized reward shaping:

$$\tilde{\pi}_t = \arg\max_{q \in \Delta} \left[\mathbb{E}_{a\sim q}[r(a)] - \tau \mathrm{KL}(q \| \pi_\theta)\right],$$

which stays close to π_θ while the reward r specifies the alternative tokens to be upweighted. Let $\pi_T(\cdot \mid x_t)$ denote a teacher distribution. To incorporate teacher guidance, we define reward using the teacher log-probability, $r(a) = \lambda \log \pi_T(a)$.

The solution has the form $\tilde{\pi}_t(a) \propto \pi_\theta(a) \exp(r(a)/\tau)$, and substituting the teacher reward yields

$$\tilde{\pi}_t(a) \propto \pi_\theta(a)\pi_T(a)^\eta, \qquad \eta = \lambda/\tau.$$

This results in a teacher-guided $\tilde{\pi}_t$ close to π_θ, while upweighting alternatives favored by the teacher through token-level reward. For easier interpretation, we consider the closely related form

$$\tilde{\pi}_t^{\mathrm{guided}}(a) \propto \pi_\theta(a \mid x_t)^{1-\eta}\pi_T(a \mid x_t)^\eta, \qquad \eta \in [0, 1].$$

It interpolates between the student ($\eta \to 0$) and teacher distribution ($\eta \to 1$). This parameterization is convenient in practice, since η directly controls the intensity of teacher signals.

TARGET-SFT. Combining the learner-dependent trust estimate $\gamma_t = p_y$ and the teacher-guided residual distribution $\tilde{\pi}_t = \tilde{\pi}_t^{\mathrm{guided}}$ gives the following target construction:

$$Q_t^{\mathrm{TARGET}} = p_y \delta_{y_t} + (1 - p_y)\tilde{\pi}_t^{\mathrm{guided}}. \tag{7}$$

The corresponding token-level objective decomposes as

$$\mathrm{CE}\big(Q_t^{\mathrm{TARGET}}, \pi_\theta\big) = p_y \,\mathrm{CE}(\delta_{y_t}, \pi_\theta) + (1 - p_y)\,\mathrm{CE}\Big(\tilde{\pi}_t^{\mathrm{guided}}, \pi_\theta\Big). \tag{8}$$

This adaptively allocates supervision: when y_t is well-supported, the target approaches δ_{y_t}. When y_t is uncertain, it relaxes one-hot fitting and redirects to teacher guidance. Thus the teacher acts as *fallback supervision* to (a) avoid overfitting of suboptimal labels, and (b) strengthen signals when a desired token is under-supported due to learner unfamiliarity. This coupled target jointly relaxes rigid imitation, balances model prior, and preserves knowledge acquisition beyond that prior.

7 EXPERIMENTS

7.1 SETUP

For mathematical reasoning, we train on two datasets: **NuminaMath-CoT-67k** (LI et al., 2024; Li et al., 2026) and **OpenR1-Math-15k** (Bakouch et al., 2025; Yan et al., 2025). For broader scientific reasoning, we train on **m23k** (Huang et al., 2026), a high-quality medical reasoning dataset. Our experiments cover across seven diverse models: Qwen2.5 (1.5B & 7B), Qwen2.5-Math (1.5B & 7B), Qwen3-1.7B-Base, LLaMA-3.2-3B, LLaMA-3.1-8B.

We compare TARGET-SFT against the following baselines: (1) **SFT**, which trains with the standard negative log-likelihood. (2) **SFT (p-loss)** (Li et al., 2026; Wu et al., 2026), a token-weighted variant that scales loss by the model probability on the observed token. (3) **Knowledge Distillation** (Hinton et al., 2015), which applies teacher distribution on the data as supervision. We use the standard form $\mathcal{L}_{\mathrm{Distill}} = c\,\mathrm{CE}(\pi_T, \pi_\theta) + (1 - c)\,\mathrm{CE}(\delta_{y_t}, \pi_\theta)$, where c is constant. These baselines correspond to different partial choices in the Q-target design space: SFT uses the one-hot target, SFT (p) uses $\gamma_t = p_{y_t}$, and KD uses the full teacher distribution as a fixed distributional signal. TARGET-SFT considers both $(\gamma_t, \tilde{\pi}_t)$ and learns a target that adaptively balances supervision based on label uncertainty.

For models that involve teacher distribution, we use the corresponding instruction-tuned model as the teacher; for example, Qwen2.5-1.5B uses Qwen2.5-1.5B-Instruct. For the Qwen3 series, we use Qwen3-4B-Instruct-2507 as the teacher. The evaluation performance is measured by Average@16 accuracy. Further details of evaluation and training configurations are provided in Appendix A.

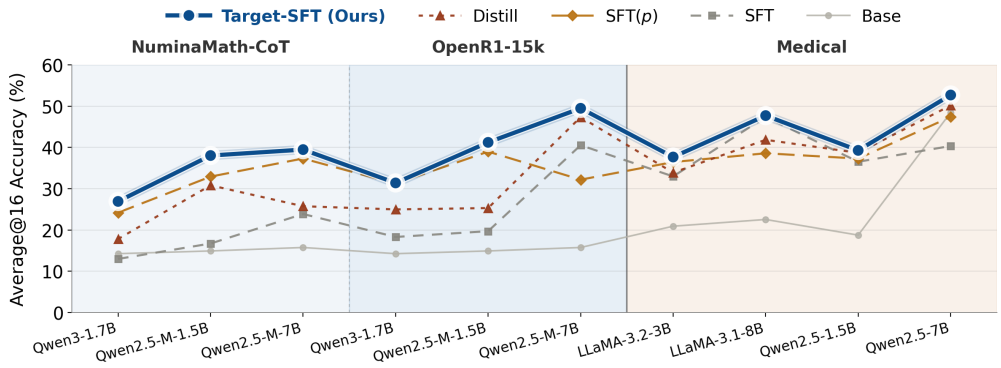

Figure 2: **Performance Summary.** Average@16 accuracy across all 10 dataset-model settings used.

Table 1: **Mathematical Reasoning.** Average@16 accuracy on five standard benchmarks, using models trained on NuminaMath-CoT (top) and OpenR1 (bottom).

	Minerva	Olympiad	AIME24	AMC23	Math500	Avg.
Dataset: NuminaMath-CoT						
Qwen3-1.7B-Base						
Base	9.87	11.35	0.62	16.25	33.81	14.26
SFT	10.78	8.64	0.00	10.78	34.61	12.99
SFT (p)	18.91	17.81	1.24	27.66	53.86	24.20
Distill	14.08	11.92	1.65	19.69	41.52	17.81
TARGET-SFT	**21.44**	**19.21**	**3.94**	**30.78**	**57.55**	**26.93**
Qwen2.5-Math-1.5B						
Base	8.23	15.20	3.75	18.12	31.52	14.92
SFT	12.61	12.11	0.82	16.41	42.29	16.72
SFT (p)	25.19	27.38	7.72	38.12	65.79	32.94
Distill	25.46	23.64	6.68	37.50	60.92	30.83
TARGET-SFT	**32.20**	**31.59**	**8.96**	**47.03**	**70.20**	**38.05**
Qwen2.5-Math-7B						
Base	7.66	9.62	8.13	19.84	31.98	15.76
SFT	21.33	18.77	2.70	22.81	53.55	23.88
SFT (p)	28.48	32.78	8.56	49.38	67.93	37.33
Distill	23.35	18.79	5.63	30.31	51.24	25.76
TARGET-SFT	**31.03**	**34.56**	**8.96**	**49.69**	**72.69**	**39.49**
Dataset: OpenR1-15k						
Qwen3-1.7B-Base						
Base	9.87	11.35	0.62	16.25	33.81	14.26
SFT	14.30	12.55	1.65	20.62	41.68	18.31
SFT (p)	25.86	23.66	**7.29**	**35.62**	62.41	31.23
Distill	22.43	18.42	2.90	27.81	53.70	24.99
TARGET-SFT	**27.47**	**24.78**	5.41	33.75	**63.92**	**31.41**
Qwen2.5-Math-1.5B						
Base	8.23	15.20	3.75	18.12	31.52	14.92
SFT	14.45	15.35	1.87	24.22	43.99	19.71
SFT (p)	31.18	33.56	11.45	47.34	70.75	39.00
Distill	19.89	20.31	5.01	26.72	52.65	25.32
TARGET-SFT	**33.09**	**34.84**	**13.13**	**51.72**	**72.38**	**41.24**
Qwen2.5-Math-7B						
Base	7.66	9.62	8.13	19.84	31.98	15.76
SFT	34.41	33.26	11.88	50.16	73.29	40.55
SFT (p)	27.33	28.28	11.24	37.19	57.25	32.20
Distill	42.04	39.94	13.32	**62.34**	80.26	47.36
TARGET-SFT	**43.61**	**42.42**	**18.13**	61.88	**80.75**	**49.50**

7.2 MAIN RESULTS

Across all evaluations, TARGET-SFT achieves the highest Average@16 accuracy. Figure 2 summarizes the results across tasks. While the baselines each show strengths and weaknesses in different dataset-model settings, TARGET-SFT consistently gives the best results. This highlights the value of the structure exposed through the target framework: effective learning can arise from an *intuitive design of the desired target*, by controlling imitation strength and allocation of alternative mass.

Table 2: **Medical Reasoning.** Average@16 accuracy on medical reasoning benchmarks.

	MedMC	MedQA	PubMed	MMLU-P	GPQA	Lancet	MedB (4)	MedB (5)	MedX	NEJM	Avg.
LLaMA-3.2-3B											
Base	21.13	21.76	22.0	12.18	25.64	24.51	27.92	21.43	11.11	21.56	20.92
SFT	34.19	38.02	57.0	25.93	30.26	36.17	36.04	28.25	11.73	32.34	32.99
SFT (p)	**40.43**	40.93	61.4	**33.42**	**34.87**	43.45	34.09	25.97	9.87	**40.30**	36.47
Distill	37.08	38.26	55.7	28.01	27.44	39.81	35.71	**31.82**	10.01	35.16	33.90
TARGET-SFT	39.78	**44.46**	**64.0**	32.70	30.00	**45.87**	**37.01**	31.17	**12.63**	39.30	**37.69**
LLaMA-3.1-8B											
Base	22.81	29.30	21.2	19.02	29.49	22.82	29.87	20.78	10.14	20.07	22.55
SFT	**50.47**	56.64	74.0	48.21	37.69	53.40	**47.73**	**40.26**	**14.77**	**50.91**	47.41
SFT (p)	40.69	46.90	64.3	34.33	33.33	38.35	40.58	31.82	12.22	43.45	38.60
Distill	45.52	51.61	63.3	42.35	35.64	47.57	38.31	35.71	13.18	45.44	41.86
TARGET-SFT	49.32	**60.17**	**74.7**	**50.42**	**46.15**	**55.34**	41.56	38.96	13.60	47.10	**47.73**
Qwen2.5-1.5B											
Base	22.40	22.70	18.4	11.47	17.18	23.06	24.35	17.21	9.45	21.39	18.76
SFT	39.35	41.16	68.5	34.07	34.36	39.32	35.39	30.19	10.35	32.67	36.54
SFT (p)	38.92	37.55	67.6	37.79	35.64	42.72	35.71	30.19	10.49	36.48	37.31
Distill	**41.02**	**42.58**	**68.9**	37.92	38.21	**43.20**	37.01	28.57	10.70	**38.97**	38.71
TARGET-SFT	40.31	41.87	68.3	**39.67**	**42.56**	40.29	**38.31**	**31.49**	**11.59**	38.64	**39.30**
Qwen2.5-7B											
Base	51.35	57.03	69.7	54.01	45.64	56.80	42.21	40.26	12.22	55.56	48.48
SFT	42.24	44.07	69.3	41.43	36.67	42.96	38.64	37.01	12.01	39.30	40.36
SFT (p)	47.65	52.79	74.5	54.27	44.62	57.04	44.81	38.96	12.97	46.60	47.42
Distill	52.28	58.84	72.3	58.89	40.77	58.01	50.65	41.56	13.73	54.73	50.18
TARGET-SFT	**54.53**	**62.37**	**74.9**	**60.85**	**48.21**	**59.95**	**52.92**	**42.86**	**14.01**	**56.55**	**52.72**

Math. Table 1 reports the results on NuminaMath-CoT and OpenR1-15k. TARGET-SFT achieves the highest average accuracy across all models on both datasets. In contrast, standard SFT yields only modest gains over the base model in many cases, and even hurts performance for Qwen3-1.7B on NuminaMath-CoT (14.26 $\rightarrow$ 12.99). This is consistent with prior observations that rigid one-hot matching can be limiting for mathematical reasoning (Li et al., 2026; Yuan et al., 2023; Yu et al., 2024; Zelikman et al., 2022). The relatively strong performance of probability-weighted SFT (p-loss) further suggests that reducing imitation strength on uncertain tokens is beneficial. However, TARGET-SFT improves over the p-loss in every case. This supports our claim that choosing γ_t alone and defaulting the residual mass to model prior is incomplete. By explicitly designing $\tilde{\pi}_t$, TARGET-SFT enhances the supervision and achieves stronger performance.

Direct distillation, which uses teacher distribution as the full target, performs relatively weakly on mathematical reasoning. On NuminaMath-CoT, distillation is only slightly better than standard SFT in some cases, such as Qwen3-1.7B-Base (17.81 vs. 12.99) and Qwen2.5-Math-7B (25.76 vs. 23.88), whereas TARGET-SFT outperforms significantly. This suggests that simply replacing the target with a soft teacher distribution is still suboptimal. In contrast, TARGET-SFT uses the teacher to shape the residual branch, whose weight $1 - \gamma_t$ increases when y_t is under-supported. This adaptive use of teacher signals proves to be more effective than full distillation in these settings.

Medical. Medical reasoning in Table 2 shows a different pattern from math, where the advantage of uniformly weakening imitation is much less consistent. In particular, standard SFT substantially outperforms p-loss on LLaMA-3.1-8B (47.41 vs. 38.60), while distillation also exceeds it on three of four models. This suggests that many low-p_y tokens in this domain may still carry useful supervision, making aggressive downweighting alone insufficient. At the same time, relying on stronger imitation or teacher supervision alone is also not consistently optimal: their relative performance varies substantially across models. In contrast, TARGET-SFT achieves the best average accuracy for all four models, including gains over both SFT and distillation. Notably, on LLaMA-3.1-8B it matches the strong SFT regime while slightly improving upon it, whereas on Qwen2.5-7B it substantially outperforms both SFT and distillation (52.72 vs. 40.36 and 50.18). These results highlight the benefit of the adaptive target design: TARGET-SFT can preserve strong label imitation when it is useful, while using the teacher-guided residual as fallback supervision when reweighting alone would discard informative signal.

Table 3: **Ablation.** Qwen2.5-Math-1.5B trained on NuminaMath (top) and OpenR1 (bottom).

	Minerva	Olympiad	AIME24	AMC23	Math500	Avg.
(1) Intensity of teacher signal (c, $\eta \uparrow$= dominant supervision)						
Distill ($c = 0.2$)	14.0	13.1	1.5	17.3	44.3	18.3
Distill ($c = 0.5$)	19.1	18.9	4.2	26.6	53.8	24.9
Distill ($c = 0.8$)	25.5	23.6	6.7	37.5	60.9	30.8
Distill ($c = 1.0$)	18.1	18.5	4.4	28.3	45.2	22.8
TARGET-SFT ($\eta = 0.2$)	27.4	29.5	**9.2**	44.1	66.0	35.2
TARGET-SFT ($\eta = 0.5$)	**32.2**	**31.6**	9.0	**47.0**	**70.2**	**38.1**
TARGET-SFT ($\eta = 1.0$)	28.2	28.1	7.9	43.1	63.9	34.3
(2) Direct combination of p-loss and distillation						
SFT (p)	31.2	33.6	11.4	47.3	70.8	38.9
Direct Comb. ($\lambda = 0.2$)	30.4	31.2	7.7	47.5	69.7	37.3
Direct Comb. ($\lambda = 0.5$)	27.9	29.3	8.8	40.9	66.1	34.6
Direct Comb. ($\lambda = 1.0$)	22.5	25.2	6.7	37.8	58.7	30.2
TARGET-SFT	**33.1**	**34.8**	**13.1**	**51.7**	**72.4**	**41.2**

7.3 Ablation Study

We now conduct a detailed ablation against distillation-style approach. This analyzes whether the gains arise from an effective TARGET supervision structure, rather than access to teacher model. Note that all methods in this section access the same observed labels and teacher distribution, allowing us to isolate the effect to supervision design alone. We ask two questions:

1. Can tuned *constant* balance between label and teacher supervision reproduce the gains?
2. Can simple combination of *probability-weighted SFT and distillation* reproduce the gains?

Varying teacher intensity in distillation does not improve. We first sweep the mixture coefficient $c \in \{0.2, 0.5, 0.8, 1.0\}$ in $\mathcal{L}_{\text{Distill}} = c\,\text{CE}(\pi_T, \pi_\theta) + (1-c)\,\text{CE}(\delta_{y_t}, \pi_\theta)$. No choice of c exceeds the result presented in Table 1, or the no-teacher p-loss baseline (32.94). In contrast, TARGET-SFT outperforms all baselines for every teacher intensity η tested (34.3–38.1). Thus tuning a fixed global balance of supervision does not recover the improvements.

Adding distillation to probability-weighted SFT is insufficient. To further ablate the adaptive supervision structure, we test the plain combination of p-loss with distillation, $\mathcal{L}_{\text{Comb}} = \gamma_t\,\text{CE}(\delta_{y_t}, \pi_\theta) + \lambda\,\text{CE}(\pi_T, \pi_\theta)$ in Table 3. Here, γ_t controls the label-imitation strength but the teacher acts on every token with the same weight λ. Results show adding a raw teacher distribution lowers accuracy from p-loss at every λ. Different from the $(1-\gamma_t)$ remaining mass intuition, a constant λ forces teacher to act on every token at a fixed weight. This can introduce unnecessary pressure toward the teacher distribution even when the student already well-supports the observed label. TARGET-SFT avoids this conflict by construction where teacher acts only as fallback supervision. The same teacher that hurts as an additive term now improves by 2.3 points, showing the value in how teacher supervision is utilized, not from simply having it.

Additional Results & Analysis. We conduct an ablation of each component in TARGET-SFT, showing that both design axes contribute effective on its own. We also compare against alternative distillation objectives such as SRKL, JS, and TVD (Ko et al., 2024; Wen et al., 2023), showing our method continues to outperform. These results are all consistent with our claim that TARGET identifies useful supervision structure, and careful designs of target behavior enables superior performance. Details are included in Appendix E.

8 Conclusion

In this work, we show that supervised fine-tuning is fundamentally a target distribution design. We formalize this view through the TARGET framework $Q_t = \gamma_t \delta_{y_t} + (1-\gamma_t)\tilde{\pi}_t$, which exposes two design choices: how strongly to imitate the observed token, and how to allocate the remaining probability mass. This gives a common interpretation that unifies many SFT variants as implicit choices of γ_t or $\tilde{\pi}_t$. TARGET-SFT instantiates this framework and achieves consistent gains across ten reasoning settings. One limitation that remains in TARGET-SFT is the susceptibility to noise in teacher, as inherited from any use of teacher-based supervision. Nonetheless, it is a strong example demonstrating the value of explicit target design, which opens a broader space for developing and bridging future fine-tuning methods.

AI USE STATEMENT

We used generative AI tools to assist with literature survey and to polish the writing of this paper. Coding agents were used to help with experimental code development. AI-assisted content was fully reviewed and edited by the authors. All research ideas and experimental designs were conceived by the authors. We take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

We will publicly release our code, training scripts, and evaluation pipeline upon publication. All datasets and base and teacher models used are publicly available. Appendix A lists the full training and evaluation configurations to reproduce our findings.

ETHICS STATEMENT

This work studies training method, and uses only publicly available datasets and models. It involves no human subjects or private data. While certain experiments involve medical reasoning tasks, we note that the resulting models are research artifacts and are not intended for clinical use.

REFERENCES

Alon Albalak, Yanai Elazar, Sang Michael Xie, Shayne Longpre, Nathan Lambert, Xinyi Wang, Niklas Muennighoff, Bairu Hou, Liangming Pan, Haewon Jeong, Colin Raffel, Shiyu Chang, Tatsunori Hashimoto, and William Yang Wang. A survey on data selection for language models, 2024. URL https://arxiv.org/abs/2402.16827.

Elie Bakouch, Leandro von Werra, and Lewis Tunstall. Open-r1: a fully open reproduction of deepseek-r1. https://huggingface.co/blog/open-r1, January 2025. Hugging Face Blog.

Hanjie Chen, Zhouxiang Fang, Yash Singla, and Mark Dredze. Benchmarking large language models on answering and explaining challenging medical questions, 2026. URL https://arxiv.org/abs/2402.18060.

Howard Chen, Noam Razin, Karthik Narasimhan, and Danqi Chen. Retaining by doing: The role of on-policy data in mitigating forgetting, 2025. URL https://arxiv.org/abs/2510.18874.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, and Hongxia Jin. Alpagasus: Training a better alpaca with fewer data, 2024a. URL https://arxiv.org/abs/2307.08701.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. Self-play fine-tuning converts weak language models to strong language models, 2024b. URL https://arxiv.org/abs/2401.01335.

Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V. Le, Sergey Levine, and Yi Ma. Sft memorizes, rl generalizes: A comparative study of foundation model post-training, 2025. URL https://arxiv.org/abs/2501.17161.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models, 2022. URL https://arxiv.org/abs/2210.11416.

Muxi Diao, Lele Yang, Wuxuan Gong, Yutong Zhang, Zhonghao Yan, Yufei Han, Kongming Liang, Weiran Xu, and Zhanyu Ma. Entropy-adaptive fine-tuning: Resolving confident conflicts to mitigate forgetting, 2026. URL https://arxiv.org/abs/2601.02151.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: On-policy distillation of large language models, 2026. URL https://arxiv.org/abs/2306.08543.

Arnav Gudibande, Eric Wallace, Charlie Snell, Xinyang Geng, Hao Liu, Pieter Abbeel, Sergey Levine, and Dawn Song. The false promise of imitating proprietary llms, 2023. URL https://arxiv.org/abs/2305.15717.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems, 2024. URL https://arxiv.org/abs/2402.14008.

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, 2021. URL https://arxiv.org/abs/2103.03874.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015. URL https://arxiv.org/abs/1503.02531.

Jerry Huang, Peng Lu, and Qiuhao Zeng. Calibrated language models and how to find them with label smoothing, 2025. URL https://arxiv.org/abs/2508.00264.

Xiaoke Huang, Juncheng Wu, Hui Liu, Xianfeng Tang, and Yuyin Zhou. m1: Unleash the potential of test-time scaling for medical reasoning with large language models, 2026. URL https://arxiv.org/abs/2504.00869.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams, 2020. URL https://arxiv.org/abs/2009.13081.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William W. Cohen, and Xinghua Lu. Pubmedqa: A dataset for biomedical research question answering, 2019. URL https://arxiv.org/abs/1909.06146.

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. DistiLLM: Towards streamlined distillation for large language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 24872–24895. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/ko24c.html.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models, 2022. URL https://arxiv.org/abs/2206.14858.

Gaotang Li, Ruizhong Qiu, Xiusi Chen, Heng Ji, and Hanghang Tong. Beyond log likelihood: Probability-based objectives for supervised fine-tuning across the model capability continuum, 2026. URL https://arxiv.org/abs/2510.00526.

Jia LI, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Costa Huang, Kashif Rasul, Longhui Yu, Albert Jiang, Ziju Shen, Zihan Qin, Bin Dong, Li Zhou, Yann Fleureau, Guillaume Lample, and Stanislas Polu. Numinamath. https://huggingface.co/datasets/AI-MO/NuminaMath-CoT, 2024.

Ziniu Li, Congliang Chen, Tian Xu, Zeyu Qin, Jiancong Xiao, Zhi-Quan Luo, and Ruoyu Sun. Preserving diversity in supervised fine-tuning of large language models, 2025. URL https://arxiv.org/abs/2408.16673.

Tao Liu, Taiqiang Wu, Runming Yang, Shaoning Sun, Junjie Wang, and Yujiu Yang. Profit: Leveraging high-value signals in sft via probability-guided token selection, 2026. URL https://arxiv.org/abs/2601.09195.

Mathematical Association of America. Math competitions. https://maa.org/math-competitions, 2023. Accessed: 2025-09-24.

Mathematical Association of America. Aime thresholds are available. https://maa.org/aime-thresholds-are-available/, 2024. Accessed: 2025-09-24.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022. URL https://arxiv.org/abs/2203.02155.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. Medmcqa : A large-scale multi-subject multi-choice dataset for medical domain question answering, 2022. URL https://arxiv.org/abs/2203.14371.

Chongli Qin and Jost Tobias Springenberg. Supervised fine tuning on curated data is reinforcement learning (and can be improved), 2025. URL https://arxiv.org/abs/2507.12856.

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, 2023. URL https://arxiv.org/abs/2311.12022.

Zhiwen Ruan, Yixia Li, He Zhu, Yun Chen, Peng Li, Yang Liu, and Guanhua Chen. Enhancing large language model reasoning via selective critical token fine-tuning, 2025. URL https://arxiv.org/abs/2510.10974.

Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. Rl's razor: Why online reinforcement learning forgets less, 2025. URL https://arxiv.org/abs/2509.04259.

Avi Singh, John D. Co-Reyes, Rishabh Agarwal, Ankesh Anand, Piyush Patil, Xavier Garcia, Peter J. Liu, James Harrison, Jaehoon Lee, Kelvin Xu, Aaron Parisi, Abhishek Kumar, Alex Alemi, Alex Rizkowsky, Azade Nova, Ben Adlam, Bernd Bohnet, Gamaleldin Elsayed, Hanie Sedghi, Igor Mordatch, Isabelle Simpson, Izzeddin Gur, Jasper Snoek, Jeffrey Pennington, Jiri Hron, Kathleen Kenealy, Kevin Swersky, Kshiteej Mahajan, Laura Culp, Lechao Xiao, Maxwell L. Bileschi, Noah Constant, Roman Novak, Rosanne Liu, Tris Warkentin, Yundi Qian, Yamini Bansal, Ethan Dyer, Behnam Neyshabur, Jascha Sohl-Dickstein, and Noah Fiedel. Beyond human data: Scaling self-training for problem-solving with language models, 2024. URL https://arxiv.org/abs/2312.06585.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xionghui Chen, Jianxin Yang, Zhenru Zhang, Yuqiong Liu, An Yang, Andrew Zhao, Yang Yue, Shiji Song, Bowen Yu, Gao Huang, and Junyang Lin. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for llm reasoning, 2025. URL https://arxiv.org/abs/2506.01939.

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, 2024. URL https://arxiv.org/abs/2406.01574.

Yuqiao Wen, Zichao Li, Wenyu Du, and Lili Mou. f-divergence minimization for sequence-level knowledge distillation. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10817–10834, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.605. URL https://aclanthology.org/2023.acl-long.605/.

Yongliang Wu, Yizhou Zhou, Zhou Ziheng, Yingzhe Peng, Xinyu Ye, Xinting Hu, Wenbo Zhu, Lu Qi, Ming-Hsuan Yang, and Xu Yang. On the generalization of sft: A reinforcement learning perspective with reward rectification, 2026. URL https://arxiv.org/abs/2508.05629.

Wei Xiong, Jiarui Yao, Yuhui Xu, Bo Pang, Lei Wang, Doyen Sahoo, Junnan Li, Nan Jiang, Tong Zhang, Caiming Xiong, and Hanze Dong. A minimalist approach to llm reasoning: from rejection sampling to reinforce, 2025. URL https://arxiv.org/abs/2504.11343.

Jianhao Yan, Yafu Li, Zican Hu, Zhi Wang, Ganqu Cui, Xiaoye Qu, Yu Cheng, and Yue Zhang. Learning to reason under off-policy guidance, 2025. URL https://arxiv.org/abs/2504.14945.

Zhaorui Yang, Tianyu Pang, Haozhe Feng, Han Wang, Wei Chen, Minfeng Zhu, and Qian Liu. Self-distillation bridges distribution gap in language model fine-tuning, 2024. URL https://arxiv.org/abs/2402.13669.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T. Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. Metamath: Bootstrap your own mathematical questions for large language models, 2024. URL https://arxiv.org/abs/2309.12284.

Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Keming Lu, Chuanqi Tan, Chang Zhou, and Jingren Zhou. Scaling relationship on learning mathematical reasoning with large language models, 2023. URL https://arxiv.org/abs/2308.01825.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. Star: Bootstrapping reasoning with reasoning, 2022. URL https://arxiv.org/abs/2203.14465.

Bolin Zhang, Jiahao Wang, Qianlong Du, Jiajun Zhang, Zhiying Tu, and Dianhui Chu. A survey on data selection for llm instruction tuning. *Journal of Artificial Intelligence Research*, 83, August 2025a. ISSN 1076-9757. doi: 10.1613/jair.1.17625. URL http://dx.doi.org/10.1613/jair.1.17625.

Dylan Zhang, Qirun Dai, and Hao Peng. The best instruction-tuning data are those that fit, 2026. URL https://arxiv.org/abs/2502.04194.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, and Guoyin Wang. Instruction tuning for large language models: A survey, 2025b. URL https://arxiv.org/abs/2308.10792.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. Lima: Less is more for alignment, 2023. URL https://arxiv.org/abs/2305.11206.

He Zhu, Junyou Su, Peng Lai, Ren Ma, Wenjia Zhang, Linyi Yang, and Guanhua Chen. Anchored supervised fine-tuning, 2026a. URL https://arxiv.org/abs/2509.23753.

Wenhong Zhu, Ruobing Xie, Rui Wang, Xingwu Sun, Di Wang, and Pengfei Liu. Proximal supervised fine-tuning, 2026b. URL https://arxiv.org/abs/2508.17784.

Yuxin Zuo, Shang Qu, Yifei Li, Zhangren Chen, Xuekai Zhu, Ermo Hua, Kaiyan Zhang, Ning Ding, and Bowen Zhou. Medxpertqa: Benchmarking expert-level medical reasoning and understanding, 2025. URL https://arxiv.org/abs/2501.18362.

A EXPERIMENT DETAILS

Training Configurations. All SFT experiments are conducted using `verl`, largely following the work by Li et al. (2026). The optimizer used is AdamW, with the learning rate of 5×10^{-5} for all models. We use cosine decay scheduling with a warm-up ratio of 0.1. We use the maximum sequence length of 3072 for both mathematical reasoning datasets. The global train batch size is 256, with gradient accumulation. The experiments are conducted on A6000 and H200 GPUs. All models are trained for 1 epoch. Table 4 summarizes all training and evaluation-related configurations.

Teacher Models. For methods that involve teacher distribution, we use the corresponding instruction-tuned model as the teacher for the base model; for example, Qwen2.5-1.5B uses Qwen2.5-1.5B-Instruct. For Qwen3-1.7B-Base, we use Qwen3-4B-Instruct-2507 as the teacher. For TARGET-SFT, we choose the teacher signal intensity from $\eta \in \{0.2, 0.5, 1.0\}$. To ensure consistency across methods, we cache all teacher logits before training and use the same cache for both distillation and TARGET-SFT experiments. For memory efficiency, the cached teacher distribution is truncated to the top-64 tokens in the vocabulary with the highest probabilities.

Datasets. For NuminaMath-CoT which originally contains 859k chain-of-thought problems, we follow Li et al. (2026) and use the 67k subset organized in their work. For the OpenR1 training dataset, we sample 15k from OpenR1-Math-46k-8192 (Yan et al., 2025), which contains verified traces generated by DeepSeek-R1 for problems collected from NuminaMath 1.5 (LI et al., 2024). We collect the responses with sequence length shorter than 3072 to standardize with the training configuration for NuminaMath-CoT. For scientific reasoning, we use m23k (Huang et al., 2026), a 23k high-quality medical reasoning dataset.

Evaluation. For mathematical reasoning, evaluation covers five representative benchmarks: Minerva Math, Olympiad Bench, AIME24, AMC23, and Math500 (Lewkowycz et al., 2022; He et al., 2024; Mathematical Association of America, 2024; 2023; Hendrycks et al., 2021). For models trained on m23k, we evaluate on the same benchmarks following Li et al. (2026), which include MedMCQA (Pal et al., 2022), MedQA-USMLE (Jin et al., 2020), PubMedQA (Jin et al., 2019), MMLU-Pro (Wang et al., 2024), GPQA (Medical) (Rein et al., 2023), Lancet & NEJM (Huang et al., 2026), MedBullets (Chen et al., 2026), and MedXpertQA (Zuo et al., 2025). Evaluation follows the same protocol as Li et al. (2026) and Huang et al. (2026). Inference uses a maximum generation length of 4096 tokens with the decoding temperature 1.0. For distillation and TARGET-SFT on Qwen2.5-Math-7B trained from the longer-sequence OpenR1-15k dataset, we use the decoding settings with temperature 0.7 and top-p 0.8 as recommended for its teacher model Qwen2.5-Math-7B-Instruct. The reported results are averaged over 16 generations for every prompt.

Table 4: **Configuration summary.** All experiments use the same setup unless otherwise specified.

Section	Item	Details
Dataset	NuminaMath-CoT	67k subset from Li et al. (2026)
	OpenR1	15k samples with length $<$ 3072 from OpenR1-Math-46k-8192 (Yan et al., 2025)
	Medical reasoning	m23k (Huang et al., 2026), a 23k high-quality medical reasoning dataset
Model	Math	Qwen3-1.7B-Base, Qwen2.5-Math-1.5B, Qwen2.5-Math-7B
	Medical	LLaMA-3.2-3B, LLaMA-3.1-8B, Qwen2.5-1.5B, Qwen2.5-7B
Train	Framework	`verl`
	Optimizer	AdamW
	Learning rate	5×10^{-5}
	Schedule	Cosine decay with warm-up ratio 0.1
	Max sequence length	3072
	Global batch size	256, with gradient accumulation
	Epochs	1
Eval	Max generation length	4096
	Metric	Average@16

B PROOFS

B.1 PROOF OF PROPOSITION 1

For a target distribution Q_t, the token-level cross-entropy loss is

$$\mathrm{CE}\left(Q_t, \pi_\theta(\cdot \mid x_t)\right) = -\sum_{k \in \mathcal{V}} Q_t(k) \log \pi_\theta(k \mid x_t).$$

By definition, the Q-target is

$$Q_t = \gamma_t \delta_{y_t} + (1 - \gamma_t)\tilde{\pi}_t.$$

Substituting this into the cross-entropy gives

$$\begin{aligned}
\mathrm{CE}\left(Q_t, \pi_\theta(\cdot \mid x_t)\right) &= -\sum_{k \in \mathcal{V}} \left[\gamma_t \delta_{y_t}(k) + (1 - \gamma_t)\tilde{\pi}_t(k)\right] \log \pi_\theta(k \mid x_t) \\
&= -\gamma_t \sum_{k \in \mathcal{V}} \delta_{y_t}(k) \log \pi_\theta(k \mid x_t) - (1 - \gamma_t) \sum_{k \in \mathcal{V}} \tilde{\pi}_t(k) \log \pi_\theta(k \mid x_t) \\
&= \gamma_t \mathrm{CE}\left(\delta_{y_t}, \pi_\theta(\cdot \mid x_t)\right) + (1 - \gamma_t)\mathrm{CE}\left(\tilde{\pi}_t, \pi_\theta(\cdot \mid x_t)\right). \qquad (9)
\end{aligned}$$

Therefore, training toward Q_t decomposes the token-level supervision into two components: label imitation controlled by γ_t and residual distribution matching of $\tilde{\pi}_t$.

B.2 PROOF OF COROLLARY 1

Given the token-weighted loss with detached importance weighting w_t

$$\mathcal{L}_t^w = -w_t \log \pi_\theta(y_t \mid x_t),$$

we consider the choice of $(\gamma_t, \tilde{\pi}_t)$

$$(\gamma_t = w_t,\ \tilde{\pi}_t = \mathrm{sg}[\pi_\theta(\cdot \mid x_t)]) \quad \Longrightarrow \quad Q_t = w_t \delta_{y_t} + (1 - w_t)\mathrm{sg}[\pi_\theta(\cdot \mid x_t)]\,.$$

By Proposition 1, the token-level loss for this Q_t decomposes as

$$\mathrm{CE}(Q_t, \pi_\theta) = w_t \mathrm{CE}(\delta_{y_t}, \pi_\theta) + (1 - w_t)\mathrm{CE}(\mathrm{sg}[\pi_\theta]\,, \pi_\theta)\,.$$

The first term gives the token-weighted SFT gradient:

$$\nabla_\theta w_t \mathrm{CE}(\delta_{y_t}, \pi_\theta) = -w_t \nabla_\theta \log \pi_\theta(y_t \mid x_t).$$

For the second term, let $p_t = \pi_\theta$ and $\bar{p}_t = \mathrm{sg}[p_t]$. Since the logit gradient of $\mathrm{CE}(\bar{p}_t, p_t)$ is $p_t - \bar{p}_t$, and $\bar{p}_t$ is a detached copy of p_t, we have

$$\nabla_\theta \mathrm{CE}(\mathrm{sg}[\pi_\theta]\,, \pi_\theta) = 0.$$

Therefore,

$$\nabla_\theta \mathrm{CE}(Q_t, \pi_\theta) = -w_t \nabla_\theta \log \pi_\theta(y_t \mid x_t) = \nabla_\theta \mathcal{L}_t^w. \qquad (10)$$

This shows that token-weighted SFT is equivalent at the gradient level to TARGET training with $\gamma_t = w_t$ and a self-matching residual branch $\tilde{\pi}_t = \mathrm{sg}[\pi_\theta]$.

B.3 Proof of Corollary 2

Given the distributional objective

$$\mathcal{L}_t = -a_t \log \pi_\theta(y_t \mid x_t) + b_t \, \mathrm{CE}(q_t, \pi_\theta), \qquad a_t, b_t \geq 0,$$

rewrite the label imitation term as cross-entropy and obtain:

$$\mathcal{L}_t = a_t \, \mathrm{CE}(\delta_{y_t}, \pi_\theta) + b_t \, \mathrm{CE}(q_t, \pi_\theta).$$

Let $s_t = a_t + b_t$. For $s_t > 0$, this can be normalized as

$$\mathcal{L}_t = s_t \left[\frac{a_t}{s_t} \mathrm{CE}(\delta_{y_t}, \pi_\theta) + \frac{b_t}{s_t} \mathrm{CE}(q_t, \pi_\theta) \right].$$

The constant s_t only rescales the token-level gradient globally, which can be absorbed into the effective learning rate or token weight. Define

$$\gamma_t = \frac{a_t}{a_t + b_t}, \qquad \tilde{\pi}_t = q_t.$$

By linearity of cross-entropy in its first argument,

$$\gamma_t \mathrm{CE}(\delta_{y_t}, \pi_\theta) + (1 - \gamma_t) \mathrm{CE}(\tilde{\pi}_t, \pi_\theta) = \mathrm{CE}\left(\gamma_t \delta_{y_t} + (1 - \gamma_t)\tilde{\pi}_t, \pi_\theta\right). \tag{11}$$

Therefore, up to the overall scale $a_t + b_t$, the objective is equivalent to Q-target training with

$$Q_t = \gamma_t \delta_{y_t} + (1 - \gamma_t)\tilde{\pi}_t.$$

This proves the claim of Corollary 2.

Some distributional methods fall outside the nonnegative residual-mixture form in Eq. 4. For example, GEM (Li et al., 2025) discourages model from concentrating high probability on the strongest non-label alternatives. This can be expressed schematically as a repulsive branch $\tilde{\pi}^-_{t,\tau}$ as follows

$$\tilde{\pi}^-_{t,\tau}(k) = \frac{\pi_\theta(k \mid x_t)^{1/\tau}}{\sum_{v \in \mathcal{V}} \pi_\theta(v \mid x_t)^{1/\tau}}, \qquad \tau < 1.$$

This represents a τ-sharpened model distribution, which places mass on high-probability tokens. And its objective is conceptually written as

$$\mathcal{L}_t = \mathrm{CE}(\delta_{y_t}, \pi_\theta) - \lambda \, \mathrm{CE}(\tilde{\pi}^-_{t,\tau}, \pi_\theta),$$

shaping the residual branch repulsively against collapsing onto a small set of tokens, thereby aligning with GEM's goal of preserving diversity in SFT.

This can be viewed as a signed residual extension of the Q-target framework, with $\tilde{\pi}_t = (\tilde{\pi}^+_t, \tilde{\pi}^-_t)$. This defines a *desired and undesired* target alternative, where $Q_t = \delta_{y_t} + \tilde{\pi}^+_t + \tilde{\pi}^-_t$. Instead of matching a positive residual distribution, GEM specifies an undesired residual direction to preserve diversity. More generally under this view, distributional variants can shape the residual branch to emphasize or suppress particular subsets of tokens.

C UNIFYING FRAMEWORK

In this section, we provide concrete connections between SFT variants discussed in Section 2 and our TARGET perspective. Table 5 shows the loss formulation and core motivation for each variant. Following the formulation $Q_t = \gamma_t \delta_{y_t} + (1 - \gamma_t)\tilde{\pi}_t$, these methods can be interpreted through their choices of γ_t and $\tilde{\pi}_t$, as summarized in Table 6.

Table 5: **Details of SFT Variants.** Each method is presented as a token-level objective ℓ_t, given the prefix x_t and observed token y_t. We denote $p_t = \pi_\theta(y_t \mid x_t)$, $p_\theta(v) = \pi_\theta(v \mid x_t)$, $\mathrm{sg}[\cdot]$ as stop-gradient, π_T, π_S as teacher and student distribution, and $\mathcal{V}$ as the vocabulary.

SFT Variant	Token-level Objective	Motivation
Standard SFT	$\ell_t^{\mathrm{SFT}} = -\log p_t$	Maximize likelihood of every observed token
	Token-Level Variants	
DFT (Wu et al., 2026)	$\ell_t^{\mathrm{DFT}} = -\mathrm{sg}[p_t] \log p_t$	Use weighting to connect SFT with an RL-style objective
Beyond-log (Li et al., 2026)	$\ell_t^f = f(p_t), \quad \ell_t^\alpha = \dfrac{1 - p_t^\alpha}{\alpha}$	Use probability-dependent objectives to balance learning across model capacities
ProFiT (Liu et al., 2026)	$m_t = \mathbf{1}[\mathrm{sg}(p_t) > \tau]$, $\ell_t^{\mathrm{ProFiT}} = -m_t \log p_t$	Use probability to identify and train on core tokens
EAFT (Diao et al., 2026)	$\widetilde{H}_t = \mathrm{sg}\left[\dfrac{H(\pi_{\theta,t}^{(k)})}{\log k}\right]$, $\ell_t^{\mathrm{EAFT}} = -\widetilde{H}_t \log p_t$.	Use entropy to weight uncertain or knowledge-conflicting tokens
iw-SFT (Qin & Springenberg, 2025)	$w(\tau) = \dfrac{q(\tau)}{\pi_{\mathrm{ref}}(\tau)}$, ($w$ trajectory-level) $\ell_t^{\mathrm{iw}} = -w(\tau) \log p_t$	Use an auxiliary distribution to assign trajectory-level weights
CFT (Ruan et al., 2025)	$c_t = \mathbf{1}\left[\forall \tilde{y}_t \in \mathcal{A}_t,\ \mathrm{Correct}\left(y_{<t}, \tilde{y}_t, y_{>t}\right) = 0\right]$, $\ell_t^{\mathrm{CFT}} = -c_t \log p_t$	Update only causally critical / irreplaceable tokens
	Distributional-Level Variants	
Label Smooth (Huang et al., 2025)	$\ell_t^{\mathrm{LS}} = -\left[(1-\lambda)\log p_t + \frac{\lambda}{\lvert\mathcal{V}\rvert}\sum_{v\in\mathcal{V}} \log p_{\theta,t}(v)\right]$	Regularize overconfident predictions for better calibration
SFT + KL (Shenfeld et al., 2025)	$\ell_t^{\mathrm{KL}} = -\log p_t + \lambda\,\mathrm{KL}(\pi_{\mathrm{ref}}(\cdot \mid x_t) \Vert \pi_\theta(\cdot \mid x_t))$	Constrain updates with a reference model to limit drift
ASFT (Zhu et al., 2026a)	$\ell_t^{\mathrm{ASFT}} = \ell_t^{\mathrm{DFT}} + \lambda \mathrm{KL}(\pi_{\mathrm{base}}(\cdot \mid x_t) \Vert \pi_\theta(\cdot \mid x_t))$	Constrain updates in DFT to prevent distributional drift
Proximal SFT (Zhu et al., 2026b)	$r_t = \dfrac{p_t}{\pi_{\mathrm{old}}(y_t \mid x_t)}$, $\ell_t^{\mathrm{PSFT}} = -\min(r_t, \mathrm{clip}(r_t, 1-\epsilon, 1+\epsilon))$	Clip ratio to enforce updates within a trust region
GEM (Li et al., 2025)	$q_t(v) = \dfrac{\mathrm{sg}[\pi_{\theta,t}(v)]^{1/\beta}}{\sum_{u\in\mathcal{V}} \mathrm{sg}[\pi_{\theta,t}(u)]^{1/\beta}}$, $\ell_t^{\mathrm{GEM}} = \mathrm{CE}(\delta_{y_t}, \pi_{\theta,t}) - \mathrm{CE}(q_t, \pi_{\theta,t})$	Control probability transfer from alternatives to observed token to preserve diversity
Knowledge Distillation (Hinton et al., 2015)	$\ell_t^{\mathrm{KD}} = -\sum_{v\in\mathcal{V}} \pi_T(v \mid x_t) \log \pi_S(v \mid x_t)$	Use the teacher logit distribution as a soft target
Distillation (Hybrid) (Hinton et al., 2015)	$\ell_t^{\mathrm{KD\text{-}H}} = (1-\lambda)[-\log \pi_S(y_t \mid x_t)] + \lambda D_{\mathrm{KL}}(\pi_T(\cdot \mid x_t) \Vert \pi_S(\cdot \mid x_t))$	Use hard-label and enrich it with teacher logit distribution

We organize existing methods into three broad categories. **Label-trust** variants modify the imitation strength on y_t, corresponding to different choices of γ_t. **Residual-distribution** variants primarily specify where the non-label probability mass should go. In these cases, the corresponding γ_t is often obtained by normalizing the relative weights of the hard-label and residual branches, with any overall scale absorbed into the effective learning rate. Finally, **data-level** methods do not directly alter γ_t or $\tilde{\pi}_t$ for a fixed demonstration. Instead, they reshape the empirical target distribution by

selecting, filtering, or rewriting the training trajectory $\hat{y}$, after which standard one-hot SFT is applied. We therefore include them as indirect instances of target distribution design at the dataset level.

In particular, GEM (Li et al., 2025) can also be illustrated with a signed extension of the TARGET-framework, where the residual component contains both positive and negative branches, $\tilde{\pi}_t = (\tilde{\pi}_t^+, \tilde{\pi}_t^-)$. This defines a *desired and undesired* target alternative, where $Q_t = \delta_{y_t} + \tilde{\pi}_t^+ + \tilde{\pi}_t^-$. And GEM utilizes $\tilde{\pi}_t^-$ as a repulsive branch to discourage probability collapse onto a small set of high-probability tokens, hence preserving diversity. Proximal SFT (Zhu et al., 2026b) is included as a schematic connection, since its clipping-based trust-region objective is not an explicit residual target but implicitly constrains updates.

Throughout the table, $(\gamma_t, \tilde{\pi}_t)$ are treated as stop-gradient unless otherwise specified. Some methods are originally defined at the sequence or trajectory level (Qin & Springenberg, 2025; Ruan et al., 2025), we present their token-level decompositions or analogues to make the connections explicit.

Table 6: **SFT variants under the TARGET view.** Each method can be interpreted through choices of label trust γ_t and residual distribution $\tilde{\pi}_t$ in $Q_t = \gamma_t \delta_{y_t} + (1-\gamma_t)\tilde{\pi}_t$. This table provides illustrative examples rather than an exhaustive coverage of methods in each category.

Variant	**Category**	**Choice of γ_t**	**Choice of $\tilde{\pi}_t$**
Standard SFT	One-hot imitation	1	–
DFT (Wu et al., 2026)	Label Trust	p_t	$\pi_{\theta,t}$
Beyond-log (Li et al., 2026)	Label Trust	p_t^α	$\pi_{\theta,t}$
ProFiT (Liu et al., 2026)	Label Trust	$m_t = \mathbf{1}\{p_t > \tau\}$	$\pi_{\theta,t}$
EAFT (Diao et al., 2026)	Label Trust	$\widetilde{H}_t = \frac{H(\pi_{\theta,t}^{(k)})}{\log k}$	$\pi_{\theta,t}$
iw-SFT (Qin & Springenberg, 2025)	Label Trust	$w(\tau) = \frac{q(\tau)}{\pi_{\text{ref}}(\tau)}$, (trajectory-level)	$\pi_{\theta,t}$
CFT (Ruan et al., 2025)	Label Trust	$c_t = \mathbf{1}\{y_t$ counterfactual critical$\}$	$\pi_{\theta,t}$
Label Smoothing (Huang et al., 2025)	Residual Distribution	$1-\lambda$	$\text{Unif}(\mathcal{V})$
SFT + KL (Shenfeld et al., 2025)	Residual Distribution	$\frac{1}{1+\lambda}$	$\pi_{\text{ref}}(\cdot \mid x_t)$
ASFT (Zhu et al., 2026a)	Residual Distribution	$\frac{p_t}{p_t+\lambda}$	$\pi_{\text{base}}(\cdot \mid x_t)$
Proximal SFT (Zhu et al., 2026b)	Residual Distribution	clipping-dependent	$\pi_{\text{old}}(\cdot \mid x_t)$
GEM (Li et al., 2025)	Residual Distribution	$\gamma_t^y = 1,\ \gamma_t^- = 1$	$\tilde{\pi}_t^+ = \pi_{\theta,t},\ \tilde{\pi}_t^- = \pi_{\theta,t}^{(\beta)}$
Knowledge Distillation (Hinton et al., 2015)	Residual Distribution	0	$\pi_T(\cdot \mid x_t)$
Distillation (Hybrid) (Hinton et al., 2015)	Residual Distribution	$1-\lambda$	$\pi_T(\cdot \mid x_t)$
RFT (Yuan et al., 2023)	Data-Level	–	$\delta_{\hat{y}}, \hat{y} \sim \pi_{\text{gen, correct}}(\cdot \mid x)$
STaR (Zelikman et al., 2022)	Data-Level	–	$\delta_{\hat{y}}, \hat{y} \sim \pi_{\text{gen, correct}}(\cdot \mid x)$
GRAPE (Zhang et al., 2026)	Data-Level	–	$\delta_{\hat{y}}, \hat{y} = \arg\max_{y^{(i)}} \pi_{\theta_0}(y^{(i)} \mid x)$
Self-distillation (Yang et al., 2024)	Data-Level	–	$\delta_{\hat{y}}, \hat{y} = \text{Rewrite}_{\pi_{\theta_0}}(x, y)$
TARGET-SFT	Label Trust + Residual	p_t	$\tilde{\pi}_t^{\text{guided}} \propto \pi_{\theta,t}^{1-\eta} \pi_T(\cdot \mid x_t)^\eta$

Dataset-level Curations (Further Details). This is a complementary direction to improve SFT by changing the training trajectories. Prior work has proposed augmenting demonstrations with multiple valid trajectories (Yuan et al., 2023; Yu et al., 2024), filtering examples by quality (Chen et al., 2024a; Singh et al., 2024), and using model-generated or model-aligned responses (Chen et al., 2025). GRAPE (Zhang et al., 2026) selects trajectories with high likelihood under the model, while rejection-sampling fine-tuning trains on correct model-generated responses (Yuan et al., 2023; Chen et al., 2024b; Zelikman et al., 2022; Xiong et al., 2025). Self-distillation fine-tuning (Yang et al., 2024) projects expert demonstrations into the model's distributional style, reducing data-model mismatch. These approaches address the same underlying issue as our work: the demonstrated trajectory or token may not be uniquely ideal for the model to imitate. By improving the dataset, they indirectly change the effective target distribution seen during SFT. In contrast, our work remains on the objective level and directly designs the effective target distribution for a fixed dataset.

D FROM LOSS TO TARGET

Section 5.2 introduces the derivation of TARGET from any differentiable token-level SFT loss. We now provide two examples and visualize their effects below.

Example 1: Standard SFT. Standard SFT minimizes the negative log-likelihood of the observed token $\mathcal{L}_{\text{SFT}}(z) = -\log p_y$, where p_y is the model probability assigned to y. The logit gradient is

$$g_j = \frac{\partial \mathcal{L}_{\text{SFT}}}{\partial z_j} = p_j - \mathbf{1}\{j = y\}. \tag{12}$$

Substituting this into Eq. 6, we obtain the induced target

$$Q_{\text{SFT}}(j) = p_j - g_j = \begin{cases} 1, & j = y, \\ 0, & j \neq y. \end{cases} \tag{13}$$

This recovers exactly the one-hot target distribution δ_y, assigning all target mass to the observed token and zero mass to alternatives.

Example 2: Probability-Weighted SFT. Consider the detached probability-weighted loss

$$\mathcal{L}_{\text{p-loss}}(z) = -\text{sg}[p_y] \log p_y, \tag{14}$$

This loss has the logit gradient

$$g_j = \text{sg}[p_y](p_j - \mathbf{1}\{j = y\}). \tag{15}$$

Substituting this gradient into Eq. 6, and using the scalar $\text{sg}[p_y] = p_y$ derives the induced target

$$Q_{\text{p-loss}}(j) = \begin{cases} 2p_y - p_y^2, & j = y, \\ (1 - p_y)p_j, & j \neq y. \end{cases} \tag{16}$$

This closed form reveals the mechanism of probability-weighted SFT. When the model assigns high probability to the observed token ($p_y \to 1$), the induced target approaches the one-hot SFT target. When the model is uncertain ($p_y \to 0$), the induced target relaxes toward the model's own distribution $Q_j \to p_j$. Thus, low-confidence tokens receive weaker imitation updates, and the residual probability mass defaults to the student prior. In the TARGET notation, this corresponds to choosing $\gamma = p_y$ and $\tilde{\pi} = \text{sg}[\pi_\theta]$, thereby preserving the model prior when evidence for strict imitation on y is weak.

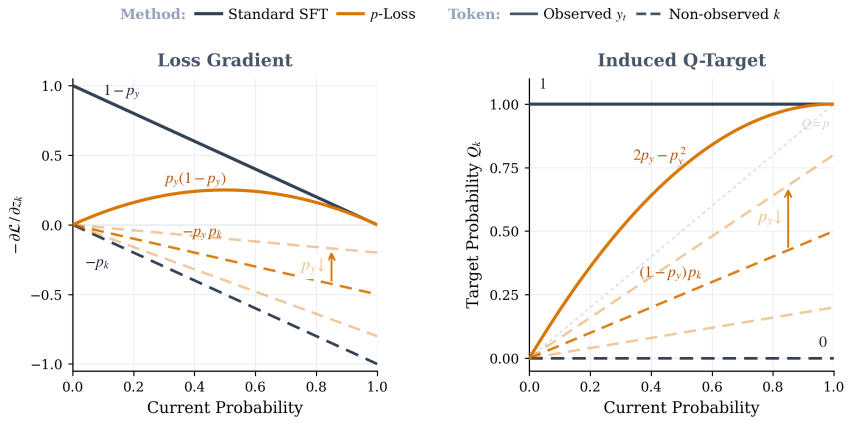

Figure 3: **Visualization of Loss.** Standard SFT's gradient pulls toward y_t and suppresses all k with fixed strength, corresponding to an induced target δ_{y_t}. For p-loss, the gradient scales with p_y (slope for Non-observed token k depends on p_y). Therefore, its gradient is near-zero when $p_y \approx 0$, and the target probability is the same as current probability ($Q = p$); the induced target approaches δ_{y_t} only when $p_y \to 1$, where the model is certain.

E ABLATION STUDY

Table 7: **Design Axes Ablation.** Performance comparison of Qwen2.5-Math-1.5B when adding each design axis of TARGET-SFT to distillation. Gains are relative to the row above.

Two Design Axes	**Label Imitation**	**Residual**	**NuminaMath**	**OpenR1**
Distill ($c = 0.8$)	constant	π_T	30.8	24.9
+ (A) adaptive allocation	γ_t	π_T	34.3 (+3.4)	36.7 (+11.8)
+ (B) teacher-guided residual	γ_t	Guided $\tilde{\pi}_t$	**38.1** (+3.8)	**41.2** (+4.5)
reference: SFT(p)	γ_t	none	32.8	38.9

Table 8: **Distillation Objectives.** Performance comparison of TARGET-SFT with three additional common distillation objectives on both OpenR1 (left) and Medical (right).

	Minerva	**Olympiad**	**AIME24**	**AMC23**	**Math500**	**Avg.**
SRKL	27.8	28.9	6.5	40.9	65.8	34.0
JS	27.2	28.2	6.9	40.5	64.8	33.5
TVD	28.1	28.5	5.8	42.0	65.4	34.0
TARGET-SFT	**33.1**	**34.8**	**13.1**	**51.7**	**72.4**	**41.2**

	Medical-Avg.
SRKL	37.2
JS	38.1
TVD	37.6
TARGET-SFT	**39.3**

This section provides additional ablation results beyond Section 7.3. In particular, we conduct a step-by-step ablation to analyze the contribution of each design axis in TARGET-SFT. And we further include performance comparisons with alternative distillation objectives, beyond the default choice of forward KL divergence used in the main experiments.

Contribution of each design axis. Table 7 isolates the two design axes by building TARGET-SFT up step by step from the hybrid distillation baseline, using Qwen2.5-Math-1.5B trained on NuminaMath-CoT and OpenR1. The starting point, Distill ($c = 0.8$), uses a constant weight on label imitation and matches the raw teacher distribution π_T. Step (A) replaces the constant weight with the adaptive trust $\gamma_t = p_y$, and the teacher branch receives weight $1-\gamma_t$. Step (B) then replaces π_T with the teacher-guided residual $\tilde{\pi}_t^{\text{guided}}$, which gives the full TARGET-SFT. As a reference, SFT (p) uses the same γ_t but a self-matching residual that carries no teacher signal.

Adaptive allocation alone yields large gains over fixed-weight distillation ($+3.4$ on NuminaMath-CoT and $+11.8$ on OpenR1), showing that deciding per token how much to rely on the observed label is more effective than a single global mixture. The teacher-guided residual adds a further $+3.8$ and $+4.5$ points, so the two axes are complementary. The reference row also shows why the residual design matters: with the raw teacher as residual, step (A) outperforms SFT (p) on NuminaMath-CoT but falls below it on OpenR1. With the guided residual, which stays anchored to the student prior, TARGET-SFT improves over SFT (p) on both datasets. The gains therefore come from combining adaptive label trust with a prior-anchored teacher residual, rather than from either choice alone.

Alternative distillation objectives. The distillation baseline in Section 7.2 matches the teacher with the forward KL divergence. To test whether the gap to TARGET-SFT is specific to this choice, Table 8 compares against three other common divergences for matching the teacher distribution: skew reverse KL (SRKL), Jensen-Shannon divergence (JS), and total variation distance (TVD) (Ko et al., 2024; Wen et al., 2023). We evaluate on Qwen2.5-Math-1.5B trained on OpenR1 for math reasoning, and Qwen2.5-1.5B trained on m23k for medical reasoning. The three objectives perform similarly to each other (33.5–34.0 on math and 37.2–38.1 on medical), and none of them closes the gap: TARGET-SFT outperforms the best one by 7.2 points on math and 1.2 points on medical. Regardless of the divergence, these objectives apply the teacher as a fixed target on every token, whereas TARGET-SFT uses it as fallback supervision adapted by label uncertainty. This again indicates that the improvement arises from an effective supervision structure.

F TEACHER MODEL ALIGNMENT

This section analyzes the alignment between the model π_θ and teacher distribution π_T to better understand the effects of teacher guidance. Figure 4 visualizes the conditional distribution $P(p_T \mid p_\theta)$ of the teacher probability $p_T = \pi_T(\cdot \mid x_t)$ given the policy model probability $p_\theta = \pi_\theta(\cdot \mid x_t)$, both evaluated on the observed ground-truth training token y_t.

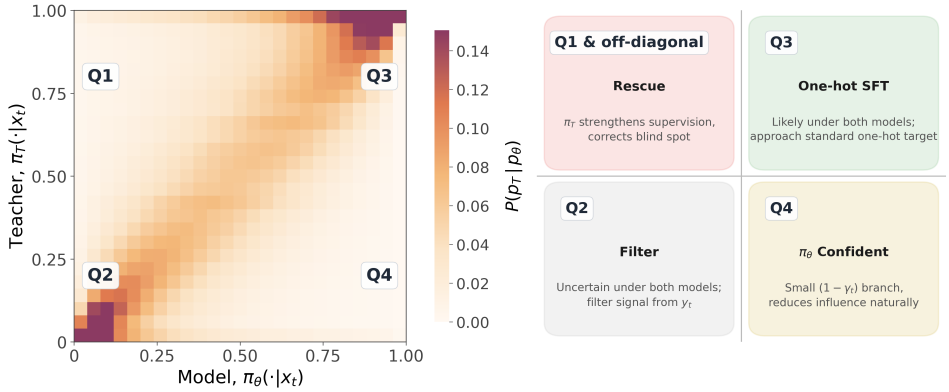

Figure 4: **Conditional Distribution of Probabilities.** This visualizes $P(p_T \mid p_\theta)$, the teacher probability p_T given the student probability p_θ on the observed token y_t. Each column represents a fixed p_θ bin, with color intensity showing the empirical density of p_T within that bin. The four annotated quadrants define qualitatively distinct supervision regimes.

A large portion of tokens lies near the diagonal, where both models assign similar probabilities to y_t. However, the distribution shows meaningful spread around the diagonal. This indicates useful correction signal, since even modest deviations from the diagonal are cases where the teacher's confidence differs and thus providing signals on alternative tokens in vocabulary. While the visualization only shows the marginal probabilities on y_t for clarity, we note that the more important effect is the teacher's full redistribution of probability mass across alternative tokens. And divergence on y_t serves as a proxy for broader differences in the teacher's beliefs over the vocabulary, enabling fine-grained adjustments that the student's current distribution cannot capture.

The off-diagonal density in Q1 represents stronger disagreements (high p_T, low p_θ), where teacher guidance is most informative. These tokens correspond to cases where the base model lacks confidence, but the teacher recognizes the token as plausible or important. This shows the role of teacher guidance as a fallback signal: when the student is uncertain, the teacher strengthens supervision rather than relying solely on the student's current belief. Notably, many tokens cluster at $p_T \approx 1.0$, indicating that the teacher frequently provides strong corrections on tokens the student underweights.

In Q2 (both p_T and p_θ low), both models are uncertain about the observed token. These may correspond to noisy, idiosyncratic, or highly non-unique tokens. Standard SFT treats them as fully reliable labels, forcing exact imitation through one-hot target δ_{y_t}. In contrast, a softer target reduces imitation strength here, allowing probability mass to plausible alternatives. This helps avoid fitting dataset artifacts or arbitrary surface choices.

In Q3, the two models assign high probability to y_t. These are high-confidence tokens where the label is likely reliable and unambiguous. The target thus remains close to standard SFT one-hot δ_{y_t}, since both models support strong imitation. Finally, in Q4 (high p_θ, low p_T), the student model is already confident but the teacher is uncertain. In this case, the teacher supervision is downweighted through a smaller $1 - \gamma_t$, naturally reducing its influence.

Together, this analysis supports the motivation of TARGET-SFT as a selective fallback: it strengthens supervision when the model is uncertain, while relaxing imitation on low-confidence tokens. This is more adaptive than standard SFT, which imposes the strict one-hot target δ_{y_t} regardless of token uncertainty or model-data alignment. Appendix G shows example Q1 (high p_T, low p_θ) and Q2 (both low) tokens, corresponding to useful knowledge vs. uncertain/noisy tokens.

G QUALITATIVE EXAMPLES

Figures 5, 6, and 7 show example trajectories from NuminaMath-67k dataset. **Rescue** token y_t is assigned a low probability under π_θ but high probability under π_T, hence strengthens the supervision that is otherwise weak. **Filter** token y_t has low probability under both π_θ and π_T, thus relaxing its imitation strength. Rescue tends to be answer-binding structural tokens, while Filter tends to be stylistic or chain-of-thought bridge words (e.g., *1.*, *First*, *Thus*, *Therefore*, *So*, etc.).

Rescue · 11 **Filter · 77** **Response · 460 tokens**

PROMPT

```
SYSTEM
Please reason step by step, and put your final answer within \boxed{}.


USER
In year $N$, the $250^{\text{th}}$ day of the year is a Monday. In year $N+2$, the $300^{\text{th}}$ day is
 also a Monday. Determine on what day of the week the $150^{\text{th}}$ day of year $N-1$ occurred.
$\text{(A)}\ \text{Tuesday} \qquad \text{(B)}\ \text{Wednesday}\qquad \text{(C)}\ \text{Thursday}\qquad \te
xt{(D)}\ \text{Friday}\qquad \text{(E)}\ \text{Saturday}$
```

RESPONSE

```
1. **Identify given days and calculate day of the week for the adjusted years**:
   - Year $N$: $250^{\text{th}}$ day is Monday.
   - Year $N+2$: $300^{\text{th}}$ day is Monday.

2. **Compute day equivalences modulo 7**:
   - Day $250 \mod 7 = 5$ (since $250 = 35 \times 7 + 5$).
   - Day $300 \mod 7 = 6$ (since $300 = 42 \times 7 + 6$).

3. **Adjust and determine the year type (leap or regular)**:
   - Since there are 2 days difference in modulo 7 calculations between $N$ and $N+2$, and given that these
 two days are Monday, it implies a leap year must exist between $N$ and $N+2$. Typically, this would be yea
r $N+1$ if it is the leap year.

4. **Calculate the day for $150^{\text{th}}$ day of year $N-1$**:
   - Since year $N+1$ is a leap year, year $N-1$ is a regular year.
   - Day $150 \mod 7 = 3$ (since $150 = 21 \times 7 + 3$).
   - Counting back from the 250th day of year $N$ (Monday), the 150th day of year $N-1$ should be 100 days
earlier modulo 7, i.e., $3 \mod 7 = 3$ days earlier in the week than Monday. Counting backwards, Monday, Su
nday, Saturday, Friday. Thus, the 150th day is Friday.

Conclusion with boxed answer:
The $150^{\text{th}}$ day of year $N-1$ occurred on a $\text{Friday}$.
The final answer is $\boxed{\textbf{(D)}\ \text{Friday}}$<|endoftext|>
```

Figure 5: **Example Trajectory #1.**

Rescue · 9 Filter · 18 Response · 483 tokens

PROMPT

SYSTEM

Please reason step by step, and put your final answer within \boxed{}.

USER

Given vectors $\overrightarrow{a}, \overrightarrow{b}$ that satisfy $|\overrightarrow{a}| = 1, |\overrightarrow{b}| = 2$, and $\overrightarrow{a} \cdot \overrightarrow{b} = -\frac{1}{2}$, find:

1. The value of $|\overrightarrow{a} + \overrightarrow{b}|$;
2. The cosine value of the angle between $\overrightarrow{a}$ and $\overrightarrow{b} - \overrightarrow{a}$.

RESPONSE

1. We know that $|\overrightarrow{a} + \overrightarrow{b}|^2 = |\overrightarrow{a}|^2 + 2\overrightarrow{a} \cdot \overrightarrow{b} + |\overrightarrow{b}|^2 = 1^2 + 2(-\frac{1}{2}) + 2^2 = 4$. Thus, $|\overrightarrow{a} + \overrightarrow{b}| = \boxed{2}$.

2. First, compute the dot product of $\overrightarrow{a}$ and $\overrightarrow{b} - \overrightarrow{a}$:

$\overrightarrow{a} \cdot (\overrightarrow{b} - \overrightarrow{a}) = \overrightarrow{a} \cdot \overrightarrow{b} - |\overrightarrow{a}|^2 = -\frac{1}{2} - 1^2 = -\frac{3}{2}$.

Next, calculate the magnitude of $\overrightarrow{b} - \overrightarrow{a}$:

$|\overrightarrow{b} - \overrightarrow{a}|^2 = |\overrightarrow{b}|^2 - 2\overrightarrow{a} \cdot \overrightarrow{b} + |\overrightarrow{a}|^2 = 2^2 - 2(-\frac{1}{2}) + 1 = 6$. Thus, $|\overrightarrow{b} - \overrightarrow{a}| = \sqrt{6}$.

Let α be the angle between $\overrightarrow{a}$ and $\overrightarrow{b} - \overrightarrow{a}$. Then,

$\cos \alpha = \dfrac{\overrightarrow{a} \cdot (\overrightarrow{b} - \overrightarrow{a})}{|\overrightarrow{a}| \cdot |\overrightarrow{b} - \overrightarrow{a}|} = \dfrac{-\frac{3}{2}}{1 \cdot \sqrt{6}} = -\dfrac{\sqrt{6}}{4}$.

So, the cosine value of the angle between $\overrightarrow{a}$ and $\overrightarrow{b} - \overrightarrow{a}$ is $\boxed{-\frac{\sqrt{6}}{4}}$.<|endoftext|>

Figure 6: **Example Trajectory #2.**

Rescue · 17 Filter · 83 Response · 919 tokens

PROMPT

SYSTEM

Please reason step by step, and put your final answer within \boxed{}.

USER

On a board, nine quadratic trinomials are written: \(x^{2}+a_{1} x+b_{1}, x^{2}+a_{2} x+b_{2}, \ldots, x^{2}+a_{9} x+b_{9}\). It is known that the sequences \(a_{1}, a_{2}, \ldots, a_{9}\) and \(b_{1}, b_{2}, \ldots, b_{9}\) are arithmetic progressions. It turns out that the sum of all nine trinomials has at least one root. What is the maximum number of the original trinomials that may not have any roots?

RESPONSE

To solve this problem, we need to analyze the given quadratic trinomials and take into account the fact that their coefficients form arithmetic progressions.

1. **Define the Quadratic Polynomials:**
   We have nine quadratic trinomials:
   \[
   P_i(x) = x^2 + a_i x + b_i \quad \text{for} \quad i = 1, 2, \ldots, 9
   \]
   where the sequences \(a_1, a_2, \ldots, a_9\) and \(b_1, b_2, \ldots, b_9\) are arithmetic progressions.

2. **Sum of Polynomials:**
   Let's consider the sum of all nine polynomials:
   \[
   P(x) = \sum_{i=1}^9 P_i(x) = \sum_{i=1}^9 (x^2 + a_i x + b_i)
   \]
   Since each \(P_i(x)\) contains an \(x^2\) term and there are nine \(P_i(x)\):
   \[
   P(x) = 9x^2 + \left(\sum_{i=1}^9 a_i\right)x + \sum_{i=1}^9 b_i
   \]

3. **Arithmetic Properties and Symmetry:**
   Because \(a_i\) and \(b_i\) are in arithmetic progression, they have symmetric properties. Specifically:
   \[
   a_i + a_{10-i} = 2a_5 \quad \text{and} \quad b_i + b_{10-i} = 2b_5 \quad \text{for} \quad i = 1, 2, \ldots, 4
   \]
   Thus, the sums:
   \[
   \sum_{i=1}^9 a_i = 9a_5 \quad \text{and} \quad \sum_{i=1}^9 b_i = 9b_5
   \]
   Therefore, the polynomial \(P(x)\) becomes:
   \[
   P(x) = 9x^2 + 9a_5 x + 9b_5 = 9(x^2 + a_5 x + b_5)
   \]
   So, \(P(x) = 9P_5(x)\).

4. **Existence of Roots:**
   According to the problem, the sum \(P(x)\) has at least one root. Hence, \(P_5(x)\) must have at least one root \(x_0\):
   \[
   P_5(x_0) = x_0^2 + a_5 x_0 + b_5 = 0
   \]

5. **Pairs of Polynomials:**
   Notice that for any root \(x_0\):
   \[
   P_i(x_0) + P_{10-i}(x_0) = 2P_5(x_0) = 0
   \]
   This implies that for each pair \((P_i, P_{10-i})\) for \(i = 1, 2, 3, 4\), at least one of the trinomials \(P_i(x_0)\) or \(P_{10-i}(x_0)\) must have a root.

6. **Counting Polynomials Without Roots:**
   Since there are four pairs and one middle polynomial \(P_5\), at least five trinomials must have a root. Consequently, the maximum number of trinomials that do not have any roots is:
   \[
   9 - 5 = 4
   \]

7. **Example Verification:**
   Verifying that it is possible for exactly four trinomials not to have roots, we construct an example where five trinomials have exactly one root:
   \[
   x^2 - 4, x^2 - 3, x^2 - 2, \ldots, x^2 + 4
   \]

Conclusion:
Thus, the maximum number of the given trinomials that can be without any roots is:
\[
\boxed{4}
\]<|endoftext|>

Figure 7: **Example Trajectory #3.**

H RESPONSE LENGTH

This section analyzes the relationship between the model's output response length and its accuracy performance. Figure 8 shows the comparison on Qwen2.5-Math-1.5B for the two mathematical reasoning datasets, and Qwen2.5-1.5B for the m23k dataset.

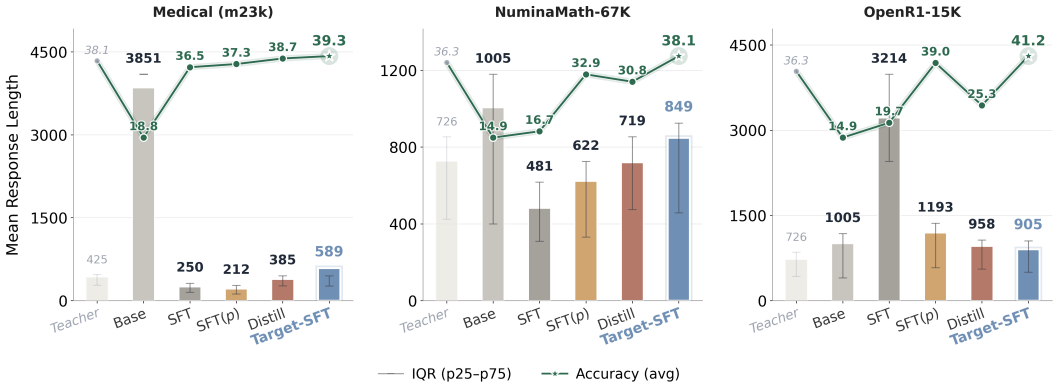

Figure 8: **Comparison of Response Length.** Bars indicate mean response length (tokens), error bars show the interquartile range (p25–p75), and the green curve reports average accuracy from evaluation. Response length does not consistently predict performance: long outputs from the base model or standard SFT often reflect rambling, repetition, or dataset-specific style imitation, while TARGET-SFT achieves strong accuracy with more moderate and stable response lengths.

On m23k and NuminaMath-67K, trained models that produce longer responses generally have higher accuracies than the shorter-output SFT variants, suggesting that the increase in reasoning length is beneficial. However, this relationship between response length and model performance is not consistent on OpenR1-15K, where standard SFT produces the longest responses but achieves substantially lower accuracy than the others. Meanwhile, TARGET-SFT achieves the highest accuracy with relatively shorter responses. This indicates that this pattern may be dataset-dependent and response length alone is not a reliable proxy for reasoning quality.

A closer inspection of model outputs suggests that the long responses from the base model often reflect poor generation behavior rather than longer reasoning. In particular, the base model occasionally shows rambling, unstable formatting, repetition, or failure to terminate properly. This inflates the average response length while yielding weak performance. In contrast, all trained models produce more consistent outputs and more stable response formats.

Standard SFT appears especially sensitive to the surface style of the training corpus. For example, on OpenR1-15K, it tends to imitate the dataset's long chain-of-thought monologue style, closely matching its response length and tone. This observation aligns with the intuition behind SFT's one-hot target δ_{y_t}, which applies strict supervision to every observed token in the training data. As a result, standard SFT may overfit to dataset-specific surface forms, including verbosity and stylistic structures, without necessarily improving the robustness of the underlying reasoning process.

By comparison, SFT(p), distillation, and TARGET-SFT tend to preserve more of the base model's flexibility. Their outputs fall into a more moderate-length regime and avoid the extreme verbosity of standard SFT. These methods appear to trade some of SFT's confident dataset-matched style for a more consistent reasoning structure that transfers better across corpora, leading to stronger accuracy without simply increasing response length or mimicking styles from the dataset.