ASKD: Reinforcement Learning-Style Knowledge Distillation with Quality-Adaptive Skewness

Mingjie Zhang[1*], **Xiaoling Zhou**[1*], **Yuxiao Luo**[1*], **Yiyu Liu**[1], **Shikun Zhang**[1], **Wei Ye**[1†]

[1]Peking University, Beijing, China

{mjzhang0621@stu.pku.edu.cn, xiaolingzhou@stu.pku.edu.cn, wye@pku.edu.cn}

Abstract

Knowledge distillation (KD) is a widely adopted technique for transferring the capabilities of large teacher models to smaller student models, thereby significantly reducing inference costs and memory consumption. However, existing KD methods are all constrained by an inherent greedy optimization objective, rooted in the assumption of teacher superiority: *"Trust all teacher-generated outputs (TGOs)"* and *"Distrust any student-generated outputs (SGOs) unsupported by the teacher"*. We propose **ASKD**, a novel KD method with adaptive skewness determined by sample quality, refining this objective to: *"Learn TGOs proportionally to their quality, and distrust only low-quality unsupported SGOs"*. **ASKD** comprises three key components: (1) A reinforcement learning-style optimization formulation to mitigate the inherent approximation bias in sample-based Kullback-Leibler (KL) divergence approximations used by previous KD methods; (2) Well-designed quality supervision signals to map and achieve adaptive skewness in skewed KL loss, pioneering the usage of sample quality to adjust learning magnitudes; (3) A gradient-clip function on high-quality SGOs for findings that high-quality SGOs in KL loss fail to yield positive updates and even cause adverse effects on some samples. Extensive experiments indicate that **ASKD** builds high-performance student models across various tasks, including instruction following, mathematical reasoning, and code generation, outperforming state-of-the-art methods comprehensively and surpassing GRPO-like approaches that use advantages as multiplicative factors. We also provide detailed mathematical proofs demonstrating properties such as Lipschitz continuity of the update coefficient and uniform convergence of the loss function, ensuring theoretical rigor for key components of **ASKD**.

1 Introduction

Large language models (LLMs) have achieved remarkable capabilities through scaling laws (Kaplan et al. 2020), where expanded model capacity and training data synergistically enhance performance (Zhou et al. 2024, 2025). However, escalating computational demands necessitate efficient compression techniques for broader deployment. Knowledge distillation(KD, Hinton, Vinyals, and Dean (2015)) addresses this by transferring knowledge from larger teacher to smaller student models, significantly advancing small language models (sLMs) as evidenced by Llama 3.2 (Meta AI 2024) and Gemma-2 (Deepmind Team et al. 2024). Current practice predominantly employs *black-box distillation* for proprietary systems (OpenAI et al. 2024; Anthropic Team 2024), where students learn from teacher-generated outputs (Kim and Rush 2016; Fu et al. 2023; Li et al. 2024). This approach remains constrained by limited supervision signals. With growing accessibility of open-source models (DeepSeek-AI et al. 2025; Qwen et al. 2025), *white-box distillation* has emerged as a promising alternative, leveraging full architectural access to teacher models (Zhang et al. 2024; Li, Zhou, and Song 2025) to develop theoretically grounded distillation frameworks (Fang et al. 2025).

Kullback-Leibler divergence (KLD)-based KD methods have shown significant success, primarily through refining loss functions and curating training data (Sun et al. 2019; Mirzadeh et al. 2019). From the loss perspective, standard KLD often fails to capture the teacher's complex generative behavior (Wen et al. 2023; Gu et al. 2024), prompting the proposal of alternatives like skew KL (SKL; Ko et al. 2024) to better guide student training. From the data perspective, prior work has focused on optimizing training data to enhance KD effectiveness. For instance, relying solely on offline data (e.g., fixed datasets Arora et al. 2023; teacher-generated outputs, TGOs; Sanh et al. 2020) is problematic when student outputs diverge significantly from fixed training samples (Agarwal et al. 2024). To mitigate this, some approaches incorporate student-generated outputs (SGOs) into training (Lin et al. 2020; Xu et al. 2025) and explore strategies for optimal use (Ko et al. 2025; Liu and Zhang 2025b).

However, existing methods often assume teacher superiority: that teachers consistently generate high-quality answers, support high-quality SGOs, and reject low-quality ones (Gu et al. 2024; Ko et al. 2024, 2025). This constrains the learning objective to suboptimal behavior: "*Trust all TGOs*" and "*Distrust any SGOs unsupported by the teacher*." This issue is especially prominent in long-chain reasoning tasks (Chen et al. 2025), where teachers may be reluctant to generate high-quality SGOs, leading to harmful updates that erode the student's performance on originally mastered tasks. Additionally, most prior methods approxi-

[*]Equal contribution.

[†]Corresponding authors.

mate KLD using Eq. (2), introducing inherent approximation bias (Ko et al. 2024; Li et al. 2025). To address these challenges, we propose ASKD, a novel KD framework that incorporates adaptive skewness through quality supervision in S(R)KLD, which consists of three main components:

- **Monte-Carlo Formulation of S(R)KL:** We introduce a reinforcement learning (RL; Czarnecki et al. 2019)-inspired loss function grounded in a Monte-Carlo formulation (Han et al. 2023) of SKLD. Unlike prior methods relying on biased sample-based KL approximations (Agarwal et al. 2024), our approach mitigates approximation errors, providing a more accurate measure of teacher-student divergence (§3.1).
- **Adaptive Skewness via Quality Supervision:** We pioneer the use of sample quality to dynamically adjust the skewness of the SKL loss. By mapping normalized quality advantages(Shao et al. 2024) to skewness parameters, ASKD amplifies learning from high-quality samples and diminishes the impact of low-quality ones, enabling fine-grained control over the distillation process (§3.2).
- **Gradient Clipping for High-Quality SGOs:** We incorporate a gradient-clipping mechanism that excludes high-quality SGOs from penalization. This prevents adverse updates when teacher model performs bad on student model's high-quality outputs, unlocking its potential to generalize beyond the teacher's capabilities (§3.3).

ASKD resolves fundamental limitations of conventional KD through a threefold contribution: **(1)** A quality-proportional optimization paradigm that supersedes greedy assumptions in prior work, implementing the principle *"Learn TGOs proportionally to their quality; distrust only low-quality unsupported SGOs"* via rigorous quality-adaptive mechanisms; **(2)** Theoretically grounded stability through provable properties: Lipschitz continuity of update coefficients and uniform convergence of the loss function, ensuring robust training dynamics; **(3)** Empirical validation across various tasks demonstrates consistent state-of-the-art performance, with significant gains in long-CoT domains (Wei et al. 2023) where conventional teacher-superiority assumptions fail.

2 Preliminary

Loss Function of KD in LLMs Given a prompt and response sequence pair denoted as (x, y), KD aims to minimize the divergence D between the distributions of a teacher model $p(y|x)$ and a student model $q_\theta(y|x)$ parameterized by θ. Conventionally, KLD, denoted as D_{KL}, is the most widely adopted loss function in KD owing to its simplicity and tractability. Specifically, sequence-level distillation using KL divergence can be precisely decomposed into the sum of token-wise distillation processes (Ko et al. 2024):

$$D_{KL}(p, q_{\boldsymbol{\theta}}) = \mathbb{E}_x \mathbb{E}_{y\sim p(\cdot|x)} \left[\log \frac{p(y|x)}{q_0(y|x)}\right] \tag{1}$$

$$\approx \frac{1}{|D|} \sum_{(x,y)\in D} p(y|x) \log \frac{p(y|x)}{q_{\boldsymbol{\theta}}(y|x)} \tag{2}$$

The reverse KL can also be denoted as $D_{RKL}(p\|q_\theta) = D_{KL}(q_\theta\|p)$. Despite its tractability, such KL has limitations of either mode-averaging (Holtzman et al. 2020) or mode-collapsing (Agarwal et al. 2024) for forward and reverse versions. To address this issue, Ko et al.(2024) proposed skew KL (SKL) and skew RKL (SRKL), which have been proved effective from both empirical and theoretical perspectives:

$$D^{(\alpha)}_{SKL}(p\|q_\theta) = D_{KL}(p\|(1-\alpha)p + \alpha q_\theta), \tag{3}$$

$$D^{(\alpha)}_{SRKL}(p\|q_\theta) = D_{KL}(q_\theta\|(1-\alpha)p + \alpha q_\theta). \tag{4}$$

Most prior works (Kim and Rush 2016; Ko et al. 2024; Wu et al. 2024) approximate KL divergence using Eq. (2) under the assumption that training data aligns with distribution D, introducing systematic bias compared to the exact Monte Carlo formulation (Eq. 1) (Han et al. 2023). While Gu et al. (2024) adopts the unbiased Monte Carlo approach, it is limited to vanilla KL divergence and does not address optimal TGO/SGO utilization. Our method instead implements the unbiased Monte Carlo formulation for S(R)KL, where D_{SKL} and D_{SRKL} in our loss function correspond to Eq. (1) rather than Eq. (2), first realizing unbiased estimation of S(R)KLD.

RL-Style Comprehension of KLD From MiniLLM's derivation (Gu et al. 2024), the gradient of (R)KL is:

$$\nabla D^{(\alpha)}_{(R)KL}(p, q_\theta) = -\mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim D \\ \boldsymbol{y}\sim p(q_\theta)(\cdot|\boldsymbol{x})}} \left[\sum_{t=1}^{T} R_t \nabla \log q_\theta(y_t|y_{<t}, x)\right] \tag{5}$$

where the gradient loss is specifically the same type as **Reinforce** algorithm (Williams 1992) when we consider R, function of p and q, as Q-function of current action and state, which is generating y_t following $(x, y_{<t})$, specifically. DistiLLM-2 (Ko et al. 2025) formulates KLD using a contrastive approach, combining the SKL loss on TGOs and the SRKL loss on SGOs. This formulation effectively results in a **DPO-based** loss function (Rafailov et al. 2024):

$$-\mathbb{E}_{y_t\sim p(\cdot|x), y_s\sim q_0(\cdot|x)} \left[\frac{1}{\lambda} \cdot \left(\lambda \log \frac{\tilde{q}_0(y_t|x)}{p(y_t|x)} - \lambda \log \frac{q_0(y_s|x)}{\tilde{p}(y_s|x)}\right)\right], \tag{6}$$

where $\tilde{q}_0(\cdot|x) = \alpha p(\cdot|x) + (1-\alpha)q_0(\cdot|x)$ and $\tilde{p}(\cdot|x) = \alpha q_0(\cdot|x) + (1-\alpha)p(\cdot|x)$.

Utilization of TGOs and SGOs The approximation of Eq.(2) assumes that the training distribution aligns with the training set D, meaning that the distribution p corresponds to the training data in KL divergence, while the student distribution q_θ aligns with the training data in RKL (Ko et al. 2024). This directly demonstrates the correct use of TGOs and SGOs: TGOs should be employed in the forward KL, while SGOs in the reverse KL during training with Eq.(2). Most prior works overlook this fundamental assumption and neglect the differential usage of TGOs and SGOs, relying instead on a single output type in (R)KL or simple KL-RKL combinations, with results only valid under the aforementioned assumption (Agarwal et al. 2024; Ko et al. 2024). While Li et al. (2025) pioneered the correct dual-output framework, Ko et al. (2025) demonstrates its susceptibility to reward hacking from ill-posed D_{SKL} composition. Their contrastive loss mechanism achieves correct and effective TGO/SGO assignment but lacks rigorous theoretical justification and constrained by approximation error under Eq. (2).

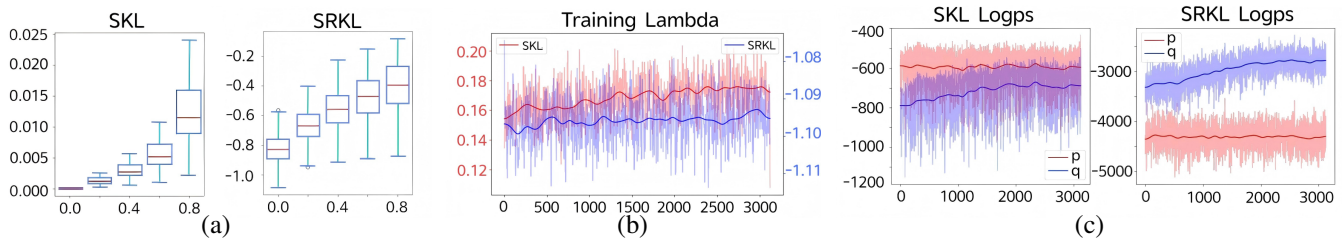

Figure 1: (a) The influence of skewness α to update coefficient λ within S(R)KL, illustrating the monotonicity of skewness to gradient update, which serves as the **core motivation** of quality-adaptive skewness. (b) Distribution of the update coefficient λ of negative log-likelihood(NLL) in the gradient policy loss of S(R)KL, where positive λ_{SKL} and negative λ_{SRKL} verify the so-called **"pulling-up"** effect in SKL and **"pushing-down"** effect in SRKL. (c) Cumulative probabilities of p and q for teacher-generated outputs (TGOs) and student-generated outputs (SGOs), clearly demonstrating $p > q$ for TGOs and $p < q$ for SGOs, helping validate the monotonic relation of skewness to λ theoretically. All results are conducted on code generation tasks.

3 Methodology

We propose **ASKD**, a knowledge distillation framework that intrinsically integrates output quality assessment. Unlike state-of-the-art methods relying on sample-based Kullback-Leibler divergence (KLD) approximations—which introduce inherent approximation bias—our approach employs an unbiased RL-style Monte Carlo formulation(§3.1) of symmetric SKL divergence. Furthermore, to address the suboptimal optimization objective in prior work, as is stated in Section 1, we introduce a quality-adaptive skewness mechanism that modulates the skewness in SKL divergence through normalized advantage, thereby implementing the main rule of our optimization objective: *Learn TGOs and distrust SGOs proportionally to their quality*(§3.2). Complementing this, we strategically discard high-quality SGOs during training to realize the second-half of optimized objective: *Distrust only low-quality unsupported SGOs*(§3.3), establishing a theoretically grounded framework for quality-aware distillation. The loss function in ASKD is as follows:

$$L(\boldsymbol{\theta}) = \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim\mathcal{D}\\ \boldsymbol{y}\sim p(\cdot|\boldsymbol{x})}} \mathcal{D}_{SKL}^{A(r)}(p, q_\theta; r) + \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim\mathcal{D}\\ \boldsymbol{y}\sim q_\theta(\cdot|\boldsymbol{x})}} \kappa(y|x)\mathcal{D}_{SRKL}^{A(r)}(p, q_\theta; r) \tag{7}$$

where $\mathcal{D}_{SKL}^{A(r)}$ and $\mathcal{D}_{SRKL}^{A(r)}$ correspond to the *gradient losses* of SKL and SRKL for TGOs and SGOs, respectively; $A(r)$ denotes the well-designed projection function mapping rewards to skewness, and $\kappa(y|x)$ is determined by the quality of SGOs, with the loss associated with high-quality SGOs directly clipped to zero. In the following subsections, we provide a detailed description of the motivation and advantages of ASKD, as stated in Algorithm 1 and Figure 3.

3.1 Monte-Carlo Formulation of S(R)KL

Motivation Current state-of-the-art approaches(Ko et al. 2024, 2025; Li et al. 2025) rely on computationally tractable batch approximations of KL divergence. While (Ko et al. 2024) demonstrated bounded estimation error under specific α constraints, these methods fundamentally overlook the Monte-Carlo nature of gradient estimation, yielding suboptimal optimization. We resolve this limitation by establishing the first unbiased RL-style Monte Carlo formulation for S(R)KL, rigorously formalizing SKL/SRKL optimization as stochastic estimation processes. This framework parallels vanilla (R)KL treatment in (Eq. 5;Gu et al. (2024)) while conducting gradient decomposition on S(R)KL with fully different gradient coefficient from vanilla ones.

Updating Direction of S(R)KL The gradient of S(R)KL can be decomposed by the policy gradient theorem (Sutton et al. 1999), with detailed deduction in Appendix B.1:

$$\nabla D_{SKL}^{(\alpha)}(p, q_\theta) = -\mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim D\\ \boldsymbol{y}\sim p(\cdot|\boldsymbol{x})}} \left[\frac{\alpha \boldsymbol{q_\theta}}{\boldsymbol{m}} \nabla_\theta \log \boldsymbol{q_\theta}(\boldsymbol{y}|\boldsymbol{x})\right] \tag{8}$$

$$\nabla D_{SRKL}^{(\alpha)}(p, q_\theta) = -\mathop{\mathbb{E}}_{\boldsymbol{x}\sim D\boldsymbol{y}\sim q_\theta(\cdot|\boldsymbol{x})} \left[(\log \frac{\boldsymbol{m}}{\boldsymbol{q_\theta}} + \frac{\alpha \boldsymbol{q_\theta}}{\boldsymbol{m}} - 1)\nabla_\theta \log \boldsymbol{q_\theta}\right] \tag{9}$$

where $\boldsymbol{m} = \alpha q_\theta(y|x) + (1-\alpha)p(y|x)$ denotes the token-wise mixture distribution. We define the coefficient of the negative log likelihood (NLL) term $\frac{\alpha \boldsymbol{q_\theta}}{\boldsymbol{m}}$ in SKL as λ_{SKL}, and the term $\left(\log \frac{\boldsymbol{m}}{\boldsymbol{q_\theta}} + \frac{\alpha \boldsymbol{q_\theta}}{\boldsymbol{m}} - 1\right)$ in SRKL as λ_{srkl}, respectively. As shown in Figure 1(b), $\lambda_{\text{SKL}} > 0$ and $\lambda_{\text{srkl}} < 0$, yielding SKL's **pulling-up** effect (amplifying student confidence on selected samples) and SRKL's **pushing-down** effect (suppressing confidence), with formal proof in Appendix B.2. The Monte-Carlo formulation establishes both the inherent directional dichotomy of S(R)KL and the monotonic relationship between skewness α and gradient coefficients(§3.2)—forming the theoretical cornerstone for adaptive skewness and novel optimization objective in ASKD.

3.2 Adaptive Skewness via Quality Supervision

To evaluate the quality of generated answers, we introduce an external supervised signal in the form of a rule-based reward (Mu et al. 2024). We then design a mapping function that transforms the normalized reward—using group normalization, consistent with the setting in the advantage estimation of GRPO—into a skewness value. This mapping function serves to amplify the gradient ("pulling-up") for high-quality TGOs while attenuating the gradient ("pushing-down") for high-quality SGOs. By explicitly differentiating sample quality during the learning process, this approach has been empirically shown to yield improved performance.

Contribution of Skewness in S(R)KL We begin by analyzing the contribution of skewness in gradient computation. By randomly selecting a pair of samples from the TGOs and

Algorithm 1: Training algorithm of **ASKD**.

Input: Training epochs E, iterations T, quality supervision function R, skewness projection function A, gradient clipping function κ, teacher p, student q_{θ_0} with parameter θ_0, prompt set, learning rate η, and scaling factor β.
Output: Student model q_{θ_e} with trained parameter θ_e.

for $e = 1$ to E **do**
  Sample responses y_t, y_s from teacher $p(\cdot|x)$ and student $q_{\theta_{e-1}}(\cdot|x)$ for the given prompt x
  Compute $r_t = R(x, y_t), r_s = R(x, y_s)$ and normalized reward in each batch
  Compute sample-wise skewness α_t, α_s using $\alpha = A(r, \beta)$
  Construct $\mathcal{D}_t = \{(x, y_t, y_s, \alpha_t, \alpha_s)\}$ for training dataset for training epoch e
  Initialize $\theta_e \leftarrow \theta_{e-1}$
  for $\tau = 1$ to T **do**
    Sample $\{(x^{(i)}, y_t^{(i)}, y_s^{(i)}, \alpha_t^{(i)}, \alpha_s^{(i)})\}_{i=1}^{|\mathcal{B}|}$ from $\mathcal{D}_t$
    Compute ∇D_{SKL} by Eq. (8) on y_t, α_t
    Compute ∇D_{SRKL} by Eq. (9) on y_s, α_s
    Compute $\nabla L = \nabla D_{SKL} + \kappa \nabla D_{SRKL}$
    Update θ_e by $\theta_e \leftarrow \theta_e - \eta \nabla L$
  end for
end for

SGOs, we observe from Figure 1(a) that a higher value of α leads to an increase in gradient magnitude for both SKL and SRKL, which corresponds to a larger pulling-up effect in SKL and a smaller pushing-down effect in SRKL, respectively. This observation can be further substantiated through a detailed gradient analysis, where the coefficients of NLL in SKL and SRKL are as follows:

$$\lambda_{SKL} = \frac{\alpha q_\theta}{m} = \prod_{t=1}^{T} \frac{1}{1 + (\frac{1}{\alpha} - 1)c_t}, \tag{10}$$

$$\lambda_{SRKL} = \sum_{t'=t}^{T} \log\left((1 - c_t)\alpha + 1\right) + \prod_{t=t'}^{T} \frac{1}{1 + (\frac{1}{\alpha} - 1)c_t} - 1, \tag{11}$$

where c_t denotes the ratio $\frac{p(y_{t'}|y_{<t'}, x)}{q_\theta(y_{t'}|y_{<t'}, x)}$. For λ_{SKL}, since $\alpha \in (0, 1)$ and $\frac{1}{\alpha} > 1$, λ_{SKL} increases monotonically with α over $(0, 1)$: zero gradient at $\alpha = 0$, and degenerating to vanilla KL loss ($\lambda = 1$) at $\alpha = 1$. For λ_{srkl}, Figure 1(c) and sampling assumptions (for TGOs: $\boldsymbol{p} > \boldsymbol{q}$; for SGOs: $\boldsymbol{p} < \boldsymbol{q}$) imply $c_t < 1$, so λ_{srkl} also increases monotonically with α over $(-1, 0)$: degenerating to vanilla RKL loss at $\alpha = 0$, and zero gradient at $\alpha = 1$. To ensure that the resulting skewness parameter fully satisfies the specified requirements, the proposed mapping function should be positively correlated with the reward and constrained within the interval $(0, 1)$.

Skewness Mapping Function We project skewness α in S(R)KL using normalized rule-based rewards. Following the SOTA reinforcement learning framework (Shao et al. 2024), we sample responses based on question groups and compute group-wise normalized rewards. The mapping function

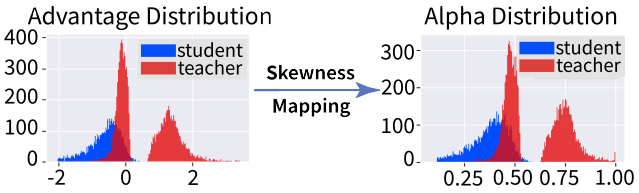

Figure 2: Distribution of advantage and according skewness projected by the skewness mapping function(SMF), which depicts the distribution transform from $\mathcal{N}(0, 1)$ to $\mathcal{U}(0, 1)$, with monotonic relationship and boundaries guaranteed.

is formally defined as follows:

$$\alpha(r; x, y) = A(r) = \Phi(r; \beta) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\beta r} e^{-t^2/2} dt. \tag{12}$$

where $\Phi(r)$ is the cumulative distribution function (CDF) of the standard normal distribution and β is the scaling constant. The mapping function owns the following key properties, which rigorously satisfy all the prior assumptions:

- **Monotonicity**: The function exhibits a strict monotonic increase, as evidenced by its derivative: $\frac{dA}{dr} = \phi(r) = \frac{1}{\sqrt{2\pi}} e^{-\beta^2 r^2/2} > 0 \quad \forall r \in \mathbb{R}$.
- **Range and Boundaries**: The function maps real numbers to the unit interval, with well-defined limits: $\lim_{r \to -\infty} A(r)$=0, $\lim_{r \to \infty} A(r)$=1, and $A(r) \in (0, 1)$ for all r.
- **Probability Integral Transform**: Under the standard normal distribution assumption, the function satisfies: If $r \sim \mathcal{N}(0, 1)$, then $A(r) \sim \text{Uniform}(0, 1)$.

Its monotonicity ensures that higher reward signals (indicating higher-quality responses) correspond to larger α values, thereby enhancing the positive gradient update in SKL and mitigating the negative gradient in SRKL. The strict boundedness within $[0, 1]$ aligns perfectly with the valid parameter range of α. And transformation of reward contributions from a normalized to uniform distribution is theoretically beneficial in subsequent analysis. Figure 2 presents a sample of the projection process with $\beta = 1$, where Skewness is projected by normalized reward. Section 4 establishes rigorous theoretical guarantees for ASKD's superiority, demonstrating uniform convergence property and enhanced training stability. Further analysis rigorously presents the strengths of our advantage projection mechanism over conventional multiplicative factor approaches in GRPO, providing formal justification for the proposed quality-adaptive framework.

3.3 Gradient Clipping for High-Quality SGOs

As discussed in Section 1, prior methods rely on the greedy assumption of teacher superiority, which can have detrimental effects on high-quality SGOs, particularly when the teacher is unlikely to generate them (Agarwal et al. 2024). This adverse effect is prevalent in long-chain reasoning tasks (Chen et al. 2025). To optimize training, we propose utilizing quality supervision, implemented through the adaptive skewness in §3.2, to differentiate the magnitudes of

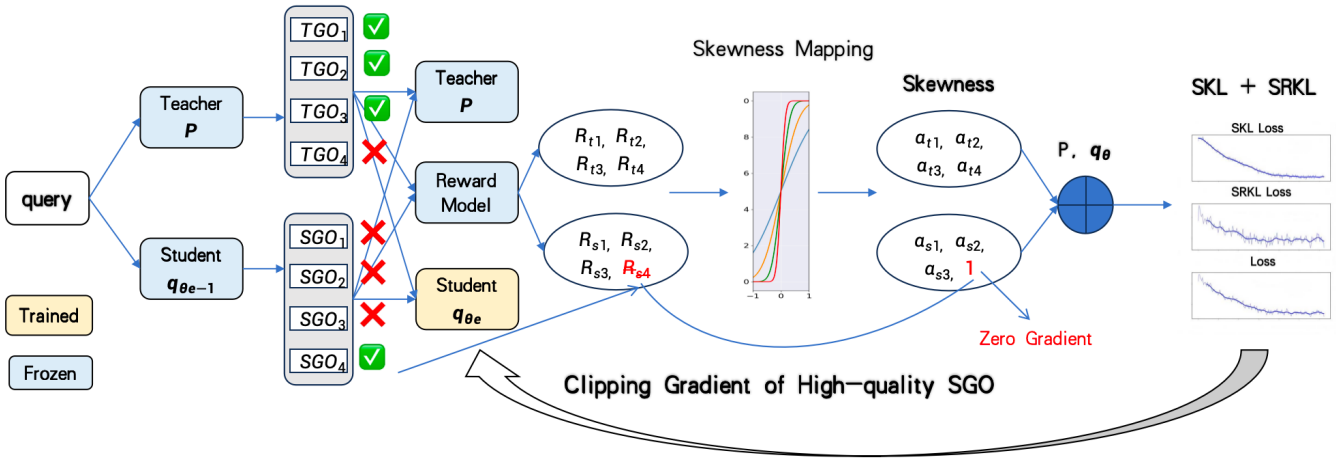

Figure 3: Diagram of ASKD framework for mathematical reasoning tasks (group=4), illustrating three core components: (1) Monte-Carlo S(R)KL formulation (§3.1), (2) quality-adaptive skewness (§3.2), and (3) high-quality SGO gradient clipping (§3.3). Reward group normalization and importance sampling (practically applied) are omitted for visual clarity.

sample-level updates. Surprisingly, merely clipping the gradients of high-quality SGOs in the SRKL training loss leads to significant improvements. This underscores the advantage of preventing the "pushing-down effect" on samples that the student model already handles correctly:

$$\kappa(y|x) = 0 \text{ if answer } (y|x) \text{ is correct else } 1. \tag{13}$$

4 Theoretical Analysis

4.1 Uniform Convergence of Adaptive Skewness

KD aims to minimize the distributional discrepancy between student and teacher models. We establish that our adaptive skewness framework achieves **uniform convergence** (Weierstrass 1894) (Theorem 1), guaranteeing convergence to the optimal parameter θ^* where q_θ fully aligns with teacher distribution p ($\theta_q = \theta_p$). While our loss formulation permits negative values ($D_{\text{S(R)KL}} < 0$), indicating student superiority on specific samples, the global optimum remains perfect alignment with p due to scaling law—unlike fixed-α approaches that converge to mixture distribution $m \neq p$, or vanilla KL losses that exhibit high variance and gradient instability (Ko et al. 2024).

Theorem 1 *Under the following conditions:*

1. *Bounded reward:* $\exists\, B < \infty$ *such that* $|r| \leq B$;
2. *Bounded probability ratio:* $\exists\, c_{\min} > 0, c_{\max} < \infty$ *with* $c_t = \frac{p(y_t|y_{<t},x)}{q_\theta(y_t|y_{<t},x)} \in [c_{\min}, c_{\max}]$;
3. *Compact parameter space: bounded gradients:* $\exists\, M_s < \infty$ *such that* $\|\nabla_{\boldsymbol{\theta}} \log q_{\boldsymbol{\theta}}(y \mid x)\| \leq M_s \quad \forall\, (x, y, \boldsymbol{\theta})$.

The parameter update rule $\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_k)$ *with* $\mathcal{L}(\boldsymbol{\theta}) = D^{(\alpha)}_{SRKL}(p \| q_{\boldsymbol{\theta}})$ *satisfies:* $\|\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}^*\| \leq \kappa \|\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\|$, *where* $\kappa = 1 - 2\eta\mu + \eta^2\beta^2 \in (0,1)$ *for some* $\mu > 0$, $\beta > 0$, *and step size* $\eta < \frac{2\mu}{\beta^2}$. *The contraction rate* κ *is independent of* $(r, \{c_t\})$.

To prove Theorem 1, we first establish three lemmas:

Lemma 1 *The gradient coefficient* $\lambda_{(\Phi(r);\{c_t\})}$ *is jointly Lipschitz continuous in* $(r, \{c_t\})$ *with respect to the Euclidean norm (Rudin 1976). Formally, there exists a constant* $L_{\text{joint}} > 0$ *such that:*

$$|\lambda(r_1, c_1) - \lambda(r_2, c_2)| \leq L_{\text{joint}} \sqrt{(r_1 - r_2)^2 + \|c_1 - c_2\|^2}$$

Lemma 2 *At the optimal point* $\boldsymbol{\theta}^*$, *the Hessian of the loss function (Eq 7) is positive definite:*

$$\mathcal{L}(\theta_k) - \mathcal{L}(\theta^*) \geq \mu \|\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\|^2$$

Lemma 3 *The gradient of our loss function (SKL) is Lipschitz continuous (Asadi, Misra, and Littman 2018):*

$$\|\nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) - \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}')\| \leq \beta \|\boldsymbol{\theta} - \boldsymbol{\theta}'\|$$

With the three lemmas, we can conclude that:

$$\|\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}^*\|^2 \leq (1 - 2\eta\mu + \eta^2\beta^2) \|\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\|^2$$

Thus, for $\eta < \frac{2\mu}{\beta^2}$, we obtain $\kappa = 1 - 2\eta\mu + \eta^2\beta^2 \in (0,1)$. This implies that with detailed adaptation of the learning rate, our method guarantees uniform convergence. Detailed proof is presented in Appendix B.3.

4.2 Skewness Projection vs. Utilizing Reward as Multiplicative Factor

This subsection addresses a critical design choice:

> **"Why project advantage onto skewness rather than adopt GRPO's multiplicative factor approach regarding advantage usage?"**

Our skewness projection mechanism offers two theoretical advantages over GRPO-based multiplicative scaling when integrating rewards into SKL:

- **Training Stability**: GRPO suffers from sparse-reward degeneracy (Rengarajan et al. 2022), where sample-deficient groups yield zero normalized rewards and vanishing gradients. ASKD circumvents this by mapping

Method	Qwen2-7B-Inst→Qwen2-1.5B				Mistral-7B-Inst→Danube2-1.8B				Gemma2-9B→Gemma2-2B			
	AlpacaEval WR(%)	EvolInst WR(%)	UltraFeed WR(%)	AVG. WR(%)	AlpacaEval WR(%)	EvolInst WR(%)	UltraFeed WR(%)	AVG. WR(%)	AlpacaEval WR(%)	EvolInst WR(%)	UltraFeed WR(%)	AVG. WR(%)
M_T(Teacher Model)	88.4	70.7	69.3	76.1	91.9	73.5	83.6	83.0	95.8	88.8	85.9	90.2
M_S(Student Model)	51.1	18.0	21.9	30.3	48.2	12.8	20.1	27.0	42.5	16.7	26.6	28.6
SFT	58.2	29.5	39.3	42.3	55.7	16.3	40.2	37.4	61.4	32.4	52.9	48.9
KD(Hinton, Vinyals, and Dean 2015)	57.5	28.2	37.9	41.2	60.2	18.2	41.6	40.0	61.8	32.5	54.4	49.5
SeqKD(Kim and Rush 2016)	58.0	29.1	38.4	41.8	59.8	18.5	42.1	40.1	62.4	33.2	55.2	50.3
ImitKD(Lin et al. 2020)	59.4	30.6	39.9	43.3	58.3	17.9	40.9	39.0	63.1	31.9	53.9	49.6
GKD(Agarwal et al. 2024)	66.1	44.6	57.7	56.1	69.8	24.5	57.7	50.7	81.4	50.6	77.2	69.7
AKL(Wu et al. 2024)	67.2	44.6	57.9	56.6	70.2	26.4	56.0	50.9	82.6	53.7	77.4	71.2
MiniLLM(Gu et al. 2024)	62.0	42.7	48.2	50.9	59.7	25.6	48.9	44.7	69.8	41.3	65.3	58.8
Speculative KD(Xu et al. 2025)	61.5	45.0	56.8	54.4	64.6	**38.9**	60.0	54.5	78.5	57.1	72.2	69.3
DistiLLM-2(Ko et al. 2025)	69.9	47.1	59.1	58.7	74.0	32.8	62.5	56.5	**86.0**	59.5	79.0	74.8
ASKD	**71.7**	**48.6**	**59.9**	**60.0**	**74.8**	35.5	**65.4**	**58.6**	84.2	**61.1**	**79.9**	**75.1**

Table 1: Comparison of winning rates (WR%) on three instruction-following benchmarks. The baselines are *text-davinci-003* (AlpacaEval) and *gpt-3.5-turbo* (Evol-Instruct/UltraFeedback), and judges are GPT-4o (AlpacaEval/Evol-Instruct) and GPT-4o-mini (UltraFeedback). Best and second-best rates in bold and underline.

Method	Qwen2-Math-7B-Inst → Qwen2-Math-1.5B			Qwen2.5-Math-7B-Inst → Qwen2.5-Math-1.5B		
	GSM8K Pass@1	MATH Pass@1	AVG. Pass@1	GSM8K Pass@1	MATH Pass@1	AVG. Pass@1
M_T	83.24	64.00	73.62	88.78	66.12	77.45
M_S	75.21	41.60	58.41	78.78	44.82	61.80
GKD	75.89	42.27	59.08	80.56	45.16	62.86
MiniLLM	74.75	41.98	58.37	78.92	44.98	61.95
AKL	76.24	42.98	59.61	81.92	45.98	63.95
DistiLLM-2	77.52	43.93	60.73	81.58	46.14	63.86
ASKD	**78.92**	**45.20**	**62.06**	**82.72**	**46.88**	**64.80**

Table 2: Comparison results on the GSM8K and MATH benchmarks. The best pass@1 score is highlighted in **bold**.

Method	DS-Coder-6.7B-Inst → DS-Coder-1.3B			Qwen2.5-Coder-7B-Inst → Qwen2.5-Coder-1.5B		
	HEval Pass@1	MBPP Pass@1	AVG. Pass@1	HEval Pass@1	MBPP Pass@1	AVG. Pass@1
M_T	71.34	75.10	73.22	85.98	82.50	84.24
M_S	37.20	63.76	50.48	59.15	74.90	67.02
GKD	37.80	64.02	50.91	60.37	76.19	68.28
MiniLLM	38.41	65.61	52.01	60.98	75.93	68.45
AKL	37.80	65.34	51.57	62.20	75.93	69.46
DistiLLM-2	39.63	**66.14**	52.89	60.98	76.72	68.72
ASKD	**40.24**	**66.14**	**53.19**	**63.40**	**77.19**	**70.30**

Table 3: Comparison results on the HEval and MBPP benchmarks. The best pass@1 score is highlighted in **bold**.

zero-normalized rewards to $\alpha = 0.5$ via $\Phi(r)$, ensuring robust gradient propagation. Crucially, while λ_{SKL} maintains bounded updates satisfying Theorem 1, the GRPO coefficient $\lambda_{\text{GRPO}} = \frac{\alpha' r}{\alpha' + (1-\alpha')c}$ remains unbounded.

- **Directional Consistency**: Multiplicative application of negative rewards reverses SKL's inherent *pulling-up* mechanism. Our approach preserves S(R)KL's directional properties by scaling update magnitudes without altering gradient directionality.

For SRKL, multiplicative advantage integration is fundamentally incompatible: λ_{SRKL} is uniformly negative (with magnitude inversely proportional to teacher-student similarity). Positive rewards r multiplied by negative λ_{SRKL} reverses intended optimization directions (Padarha 2025). Introducing sign corrections might align reward direction but compromises the theoretical role of λ_{SRKL} in representing distributional differences. Consequently, SRKL either conflicts with reward semantics or misrepresents divergence properties with multiplicative advantage usage. To conclude, skewness projection provides a theoretically grounded and practically superior framework for reward integration, resolving the directional conflicts and stability limitations inherent in multiplicative scaling approaches.

5 Experiments

We evaluate ASKD on instruction-following, mathematical reasoning and code generation tasks. We adopt batch sampling (Rosset et al. 2024) and determine scaling factor β via ablation studies(Appendix D). For instruction following and code generation tasks, we employ single-response sampling with batch-computed advantages(Reinforce++ (Hu, Liu, and Shen 2025)), whereas mathematical reasoning utilizes group sampling with fixed size 4 and group-computed advantages(GRPO). Compared baselines are as follows: (1) supervised fine-tuning(SFT); (2) KD; (3) SeqKD, applying SFT to TGO; (4) ImitKD, employing KLD on SGO; (5) MiniLLM, utilizing a policy gradient approach on RKL and SGO; (6) GKD, using JSD on a mixture of SGOs and fixed datasets; (7) AKL, proposing adaptive fusion of KL and RKL; (8) Speculative KD, using speculative generation for vanilla KL; (9) DistiLLM-2, a contrastive method using SKL for TGOs and SRKL for SGOs. Further details of the experimental setup are in Appendix C.

5.1 General Instruction Following

Setup Following Ko et al. (2025), we construct training datasets by randomly sampling 50k prompts from UltraChat200k (Ding et al. 2023), with responses generated by

	(1)	(2)	(3)	Qwen2 (†)	Qwen2.5 (†)	AVG. (†)
ASKD (Full)				**79.58**	**84.22**	**81.90**
w/o (2,3)		✓	✓	78.25	83.06	80.66
w/o (1)	✓			79.24	83.75	81.50
w/o (2)		✓		77.80	82.28	80.04
w/o (1,2,3)	✓	✓	✓	77.52	81.58	79.55

Table 4: Ablation study validating the necessity of ASKD's three core components, where ✓denotes component removal. Consistent performance degradation across all removal variants confirms the critical contribution of each element to the framework's efficacy.

teacher and student models. We use LLM-as-a-Judge (Zheng et al. 2023) to rate the win rate between sampled answers and baseline (text-davinci-003), using this as supervised reward to correct skewness. We evaluate ASKD on general instruction-following via AlpacaEval Li et al. (2023), Evol-Instruct (Xu et al. 2024), and UltraFeedback (Cui et al. 2023), with LLM-as-a-Judge using GPT-4o or GPT-4o-mini. We use Qwen2-7B (Hui et al. 2024), Mistral-7B (Jiang et al. 2023), Gemma-2-9B (Deepmind Team et al. 2024)as teacher models and Qwen2-1.5B, Danube2-1.8B (Singer et al. 2024), Gemma-2-2B as student models respectively.

Results Experimental results are reported in Table 1. Comparisons between ASKD and other baselines highlight the superiority of our method, with the exception of Speculative KD on Daube2-1.8B for EvolInst and DistiLLM-2 on Gemma2-2B for AlpacaEval. Overall, our method outperforms all baselines by 1.3%, 2.9%, and 0.3% over the second-best methods on Qwen2-1.5B, Daube2-1.8B, and Gemma2-2B, respectively.

5.2 Mathematical Reasoning

Setup We conduct experiments on two standard mathematical reasoning benchmarks: GSM8K (Cobbe et al. 2021) and MATH (Hendrycks et al. 2021). For teacher-student pairs, Qwen2-Math-7B-Inst and Qwen2.5-Math-7B-Inst serve as teacher models, with Qwen2-Math-1.5B and Qwen2.5-Math-1.5B as student models. Student models are trained on 50k randomly selected samples from MetaMathQA (Yu et al. 2024), specifically via supervised fine-tuning on the full dataset for one epoch.

Results Table 2 summarizes the effectiveness of ASKD in comparison to recent competitive baselines, including GKD, MiniLLM, AKL and DistiLLM-2. In both the Qwen2 and Qwen2.5 experimental setups, ASKD demonstrates significantly higher performance than the other baselines on the GSM8K and MATH evaluations, highlighting the strong potential of our method for reasoning tasks.

5.3 Code Generation

Setup For the accessibility of supervised signal, we utilize code-r1 (Liu and Zhang 2025a) dataset for training, and use verified accuracy as reward. We apply Deepseek-Coder-6.7B-Inst and Qwen-2.5-Coder-7B-Inst as teacher models and Deepseek-Coder-1.3B and Qwen2.5-Coder-1.5B as student models, respectively. We evaluate performance on two Code benchmarks: HumanEval (Chen et al. 2021) and MBPP (Austin et al. 2021).

	Qwen2-Math-1.5B			Qwen2.5-Math-1.5B		
Method	GSM8K	MATH	AVG.	GSM8K	MATH	AVG.
GRPO	78.24	44.56	61.42	81.88	46.52	64.20
ASKD	**78.92**	**45.20**	**62.06**	**82.72**	**46.88**	**64.80**

Table 5: Comparison of advantage usage: skewness projection (ASKD) and multiplicative factor (GRPO) in SKL.

Results Table 3 presents the pass@1 scores on HumanEval and MBPP. ASKD outperforms all baselines, including GKD, MiniLLM, AKL and DistiLLM-2, on both tasks. Notably, the RKL variant with SGOs (Gu et al. 2024) but without SFT performs worse than the base model, highlighting the superiority of ASKD's RL-style loss.

5.4 Ablation Study

Ablation studies are conducted on mathematical reasoning tasks, where reward accessibility is straightforward, including component ablation and a comparison of reward usage (skewness projection vs. GRPO-based multiplicative factor). Additional experiments are detailed in Appendix D.

Component Ablation ASKD consists of three core components: (1) an RL-type loss function based on the Monte-Carlo formulation (§3.1); (2) adaptive skewness projected by quality rewards (§3.2); (3) selection of low-quality SGOs during training (§3.3). Component ablation (Table 4) validates each ASKD element's necessity, with performance degrading when any component is removed from the full framework, proving every sub-component's effectiveness.

Usage of Advantages ASKD employs advantages via skewness projection. Through detailed projection function design, we theoretically and experimentally demonstrate the superiority of quality-supervised adaptive skewness. In §3.2, we theoretically analyze why advantages are not used as multiplicative factors in S(R)KL (like GRPO) and present an ablation study on usage variance in Table 5, indicating that this novel advantage usage is more effective in KD.

6 Conclusion

In this study, we address the limitations of existing KD methods, which are constrained by the greedy assumption of teacher superiority and suffer from approximation biases in KL divergence estimation. A novel KD framework, termed ASKD, is presented, which introduces three key innovations: a Monte Carlo formulation of S(R)KL, adaptive skewness through quality supervision, and gradient clipping for high-quality SGOs. We provide rigorous theoretical proofs for key properties, including the Lipschitz continuity of update coefficients and the uniform convergence of the loss, ensuring both stability and reliability. Empirically, ASKD consistently outperforms SOTA KD methods across various tasks, particularly excelling in reasoning-intensive domains where the assumption of teacher superiority often fails.

References

Agarwal, R.; Vieillard, N.; Zhou, Y.; Stanczyk, P.; Ramos, S.; Geist, M.; and Bachem, O. 2024. On-Policy Distillation of Language Models: Learning from Self-Generated Mistakes. arXiv:2306.13649.

Anthropic Team. 2024. Claude 4: Technical Report on Capabilities and Training Methods.

Arora, K.; Asri, L. E.; Bahuleyan, H.; and Cheung, J. C. K. 2023. Why Exposure Bias Matters: An Imitation Learning Perspective of Error Accumulation in Language Generation. arXiv:2204.01171.

Asadi, K.; Misra, D.; and Littman, M. L. 2018. Lipschitz Continuity in Model-based Reinforcement Learning. arXiv:1804.07193.

Austin, J.; Odena, A.; Nye, M.; Bosma, M.; et al. 2021. Program Synthesis with Large Language Models. arXiv:2108.07732.

Chen, M.; Tworek, J.; Jun, H.; Yuan, Q.; et al. 2021. Evaluating Large Language Models Trained on Code. arXiv:2107.03374.

Chen, Q.; Qin, L.; Liu, J.; Peng, D.; Guan, J.; Wang, P.; Hu, M.; Zhou, Y.; Gao, T.; and Che, W. 2025. Towards Reasoning Era: A Survey of Long Chain-of-Thought for Reasoning Large Language Models. arXiv:2503.09567.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; et al. 2021. Training Verifiers to Solve Math Word Problems. arXiv:2110.14168.

Cui, G.; Yuan, L.; Ding, N.; Yao, G.; He, B.; Zhu, W.; Ni, Y.; Xie, G.; Xie, R.; Lin, Y.; et al. 2023. Ultrafeedback: Boosting language models with scaled ai feedback. *arXiv preprint arXiv:2310.01377*.

Czarnecki, W. M.; Pascanu, R.; Osindero, S.; Jayakumar, S. M.; Swirszcz, G.; and Jaderberg, M. 2019. Distilling Policy Distillation. arXiv:1902.02186.

Deepmind Team; Riviere, M.; Pathak, S.; Sessa, P. G.; et al. 2024. Gemma 2: Improving Open Language Models at a Practical Size. arXiv:2408.00118.

DeepSeek-AI; Guo, D.; Yang, D.; Zhang, H.; Song, J.; et al. 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv:2501.12948.

Ding, N.; Chen, Y.; Xu, B.; Qin, Y.; Zheng, Z.; Hu, S.; Liu, Z.; Sun, M.; and Zhou, B. 2023. Enhancing Chat Language Models by Scaling High-quality Instructional Conversations. arXiv:2305.14233.

Fang, L.; Yu, X.; Cai, J.; Chen, Y.; Wu, S.; Liu, Z.; Yang, Z.; Lu, H.; Gong, X.; Liu, Y.; Ma, T.; Ruan, W.; Abbasi, A.; Zhang, J.; Wang, T.; Latif, E.; Liu, W.; Zhang, W.; Kolouri, S.; Zhai, X.; Zhu, D.; Zhong, W.; Liu, T.; and Ma, P. 2025. Knowledge Distillation and Dataset Distillation of Large Language Models: Emerging Trends, Challenges, and Future Directions. arXiv:2504.14772.

Fu, Y.; Peng, H.; Ou, L.; Sabharwal, A.; and Khot, T. 2023. Specializing Smaller Language Models towards Multi-Step Reasoning. arXiv:2301.12726.

Gu, Y.; Dong, L.; Wei, F.; and Huang, M. 2024. MiniLLM: Knowledge Distillation of Large Language Models. arXiv:2306.08543.

Han, I.; Oh, S.; Jung, H.; Chung, I.; and Kim, K.-J. 2023. Monte Carlo and Temporal Difference Methods in Reinforcement Learning [AI-eXplained]. *IEEE Computational Intelligence Magazine*, 18(4): 64–65.

Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; and Steinhardt, J. 2021. Measuring Mathematical Problem Solving With the MATH Dataset. arXiv:2103.03874.

Hinton, G. E.; Vinyals, O.; and Dean, J. 2015. Distilling the Knowledge in a Neural Network. *ArXiv*, abs/1503.02531.

Holtzman, A.; Buys, J.; Du, L.; Forbes, M.; and Choi, Y. 2020. The Curious Case of Neural Text Degeneration. arXiv:1904.09751.

Hu, J.; Liu, J. K.; and Shen, W. 2025. REINFORCE++: An Efficient RLHF Algorithm with Robustness to Both Prompt and Reward Models. arXiv:2501.03262.

Hui, B.; Yang, J.; Cui, Z.; Yang, J.; et al. 2024. Qwen2.5-Coder Technical Report. arXiv:2409.12186.

Jiang, A. Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; et al. 2023. Mistral 7B. arXiv:2310.06825.

Kaplan, J.; McCandlish, S.; Henighan, T.; Brown, T. B.; Chess, B.; Child, R.; Gray, S.; Radford, A.; Wu, J.; and Amodei, D. 2020. Scaling Laws for Neural Language Models. arXiv:2001.08361.

Kim, Y.; and Rush, A. M. 2016. Sequence-Level Knowledge Distillation. arXiv:1606.07947.

Ko, J.; Chen, T.; Kim, S.; Ding, T.; Liang, L.; Zharkov, I.; and Yun, S.-Y. 2025. DistiLLM-2: A Contrastive Approach Boosts the Distillation of LLMs. arXiv:2503.07067.

Ko, J.; Kim, S.; Chen, T.; and Yun, S.-Y. 2024. DistiLLM: Towards Streamlined Distillation for Large Language Models. arXiv:2402.03898.

Li, L. H.; Hessel, J.; Yu, Y.; Ren, X.; Chang, K.-W.; and Choi, Y. 2024. Symbolic Chain-of-Thought Distillation: Small Models Can Also "Think" Step-by-Step. arXiv:2306.14050.

Li, M.; Zhou, F.; and Song, X. 2025. BiLD: Bi-directional Logits Difference Loss for Large Language Model Distillation. arXiv:2406.13555.

Li, X.; Zhang, T.; Dubois, Y.; Taori, R.; Gulrajani, I.; Guestrin, C.; Liang, P.; and Hashimoto, T. B. 2023. Alpacaeval: An automatic evaluator of instruction-following models.

Li, Y.; Gu, Y.; Dong, L.; Wang, D.; Cheng, Y.; and Wei, F. 2025. Direct Preference Knowledge Distillation for Large Language Models. arXiv:2406.19774.

Lin, A.; Wohlwend, J.; Chen, H.; and Lei, T. 2020. Autoregressive Knowledge Distillation through Imitation Learning. arXiv:2009.07253.

Liu, J.; and Zhang, L. 2025a. Code-R1: Reproducing R1 for Code with Reliable Rewards. https://github.com/ganler/code-r1.

Liu, L.; and Zhang, M. 2025b. Being Strong Progressively! Enhancing Knowledge Distillation of Large Language Models through a Curriculum Learning Framework. arXiv:2506.05695.

Meta AI. 2024. Llama 3.2: Revolutionizing edge AI and vision with open, customizable models. Accessed: 2025-07-23.

Mirzadeh, S.-I.; Farajtabar, M.; Li, A.; Levine, N.; Matsukawa, A.; and Ghasemzadeh, H. 2019. Improved Knowledge Distillation via Teacher Assistant. arXiv:1902.03393.

Mu, T.; Helyar, A.; Heidecke, J.; Achiam, J.; Vallone, A.; Kivlichan, I.; Lin, M.; Beutel, A.; Schulman, J.; and Weng, L. 2024. Rule Based Rewards for Language Model Safety. arXiv:2411.01111.

OpenAI; Achiam, J.; Adler, S.; Agarwal, S.; and etc al., L. A. 2024. GPT-4 Technical Report. arXiv:2303.08774.

Padarha, S. 2025. Enhancing Reasoning Capabilities in SLMs with Reward Guided Dataset Distillation. *arXiv preprint arXiv:2507.00054*.

Qwen; Yang, A.; Li, A.; Yang, B.; Zhang, B.; Hui, B.; et al. 2025. Qwen3 Technical Report. arXiv:2505.09388.

Rafailov, R.; Sharma, A.; Mitchell, E.; Ermon, S.; Manning, C. D.; and Finn, C. 2024. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. arXiv:2305.18290.

Rengarajan, D.; Vaidya, G.; Sarvesh, A.; Kalathil, D.; and Shakkottai, S. 2022. Reinforcement Learning with Sparse Rewards using Guidance from Offline Demonstration. arXiv:2202.04628.

Rosset, C.; Cheng, C.-A.; Mitra, A.; Santacroce, M.; Awadallah, A.; and Xie, T. 2024. Direct Nash Optimization: Teaching Language Models to Self-Improve with General Preferences. arXiv:2404.03715.

Rudin, W. 1976. *Principles of Mathematical Analysis*. New York: McGraw-Hill, 3rd edition. ISBN 0-07-054235-X.

Sanh, V.; Debut, L.; Chaumond, J.; and Wolf, T. 2020. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv:1910.01108.

Shao, Z.; Wang, P.; Zhu, Q.; Xu, R.; Song, J.; Bi, X.; Zhang, H.; Zhang, M.; Li, Y. K.; Wu, Y.; and Guo, D. 2024. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. arXiv:2402.03300.

Singer, P.; Pfeiffer, P.; Babakhin, Y.; Jeblick, M.; Dhankhar, N.; Fodor, G.; and Ambati, S. S. 2024. H2O-Danube-1.8B Technical Report. arXiv:2401.16818.

Sun, S.; Cheng, Y.; Gan, Z.; and Liu, J. 2019. Patient Knowledge Distillation for BERT Model Compression. arXiv:1908.09355.

Sutton, R. S.; McAllester, D.; Singh, S.; and Mansour, Y. 1999. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E.; Le, Q.; and Zhou, D. 2023. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. arXiv:2201.11903.

Weierstrass, K. 1894. Zur Theorie der Potenzreihen. In *Mathematische Werke*, volume 2, 223–230. Berlin: Mayer Müller. Original work presented in 1841 lectures; first published in collected works.

Wen, Y.; Li, Z.; Du, W.; and Mou, L. 2023. f-Divergence Minimization for Sequence-Level Knowledge Distillation. arXiv:2307.15190.

Williams, R. J. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3): 229–256.

Wu, T.; Tao, C.; Wang, J.; Yang, R.; Zhao, Z.; and Wong, N. 2024. Rethinking Kullback-Leibler Divergence in Knowledge Distillation for Large Language Models. arXiv:2404.02657.

Xu, C.; Sun, Q.; Zheng, K.; Geng, X.; Zhao, P.; Feng, J.; Tao, C.; Lin, Q.; and Jiang, D. 2024. WizardLM: Empowering large pre-trained language models to follow complex instructions. In *The Twelfth International Conference on Learning Representations*.

Xu, W.; Han, R.; Wang, Z.; Le, L. T.; Madeka, D.; Li, L.; Wang, W. Y.; Agarwal, R.; Lee, C.-Y.; and Pfister, T. 2025. Speculative Knowledge Distillation: Bridging the Teacher-Student Gap Through Interleaved Sampling. arXiv:2410.11325.

Yu, L.; Jiang, W.; Shi, H.; Yu, J.; Liu, Z.; Zhang, Y.; Kwok, J. T.; Li, Z.; Weller, A.; and Liu, W. 2024. MetaMath: Bootstrap Your Own Mathematical Questions for Large Language Models. arXiv:2309.12284.

Zhang, S.; Zhang, X.; Sun, Z.; Chen, Y.; and Xu, J. 2024. Dual-Space Knowledge Distillation for Large Language Models. arXiv:2406.17328.

Zheng, L.; Chiang, W.-L.; Sheng, Y.; Zhuang, S.; Wu, Z.; Zhuang, Y.; Lin, Z.; Li, Z.; Li, D.; Xing, E.; et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36: 46595–46623.

Zhou, X.; Ye, W.; Wang, Y.; Jiang, C.; Lee, Z.; Xie, R.; and Zhang, S. 2024. Enhancing In-Context Learning via Implicit Demonstration Augmentation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2810–2828.

Zhou, X.; Zhang, M.; Lee, Z.; Ye, W.; and Zhang, S. 2025. Hademif: Hallucination detection and mitigation in large language models. In *The Thirteenth International Conference on Learning Representations*.