LLM-Oriented Token-Adaptive Knowledge Distillation

Xurong Xie[1*], Zhucun Xue[1*], Jiafu Wu[2], Jian Li[2], Yabiao Wang[2], Xiaobin Hu[3], Yong Liu[1†], Jiangning Zhang[1,2†]

[1]Zhejiang University
[2]Tencent Youtu Lab
[3]National University of Singapore
{xurongxie, 12432038, 186368}@zju.edu.cn, yongliu@iipc.zju.edu.cn,

Abstract

Knowledge Distillation (KD) is a key technique for compressing Large-scale Language Models (LLMs), but prevailing logit-based methods employ static strategies misaligned with the student's dynamic learning process. By treating all tokens indiscriminately with a fixed temperature, these methods result in suboptimal knowledge transfer. To address this, we propose LLM-oriented token-**Ada**ptive **K**nowledge **D**istillation (**AdaKD**), a framework that adapts the distillation process to each token's real-time learning state. AdaKD consists of two synergistic modules driven by a unified token difficulty metric. First, the Loss-driven Adaptive Token Focusing (LATF) module dynamically concentrates distillation on valuable tokens by monitoring the student's learning stability. Second, Inverse Difficulty Temperature Scaling (IDTS) introduces a counter-intuitive token-level temperature: low for difficult tokens to target error correction, and high for easy tokens to learn the teacher's smooth output distribution for better generalization. As a plug-and-play framework, AdaKD consistently improves performance across diverse distillation methods, model architectures, and benchmarks.

Code — https://github.com/SassyRong/AdaKD
Extended version — https://arxiv.org/abs/2510.11615

Introduction

Large Language Models (LLMs) have made significant advancements in recent years. They perform excellently on many natural language processing tasks, such as text generation, comprehension, and reasoning (Achiam et al. 2023; Anil et al. 2023; Grattafiori et al. 2024). This success is mainly due to their *extensive parameter sizes* and the pre-training they undergo on *vast amounts of data* (Kaplan et al. 2020). However, this powerful capability comes at the cost of *enormous computational and storage resources*. These requirements create significant barriers to ***deployment on edge devices in low-latency scenarios and to achieving widespread accessibility***, limiting the practical reach of LLMs (Wan et al. 2023; Zheng et al. 2025; Bai et al. 2024).

To solve above challenges, Knowledge Distillation (KD) has emerged as a promising solution for model compression and acceleration. Our work focuses on logit-based distillation, a prevalent white-box approach that directly transfers knowledge by matching the output distributions of the teacher and student models. While conceptually simple and effective, we argue that current logit-based methods still face two key limitations in adapting to the dynamic learning process of the student model: ***1)*** **Indiscriminate token treatment.** Most methods treat all tokens indiscriminately, applying a uniform distillation objective across the entire sequence. This lack of differentiation is misaligned with the student's real-time learning progress, resulting in suboptimal knowledge transfer and potentially introducing noise from tokens that are already well-mastered. ***2)*** **Fixed global distillation temperature.** The distillation temperature is typically set as a fixed, global hyperparameter. This uniform temperature setting ignores the distinct learning needs of individual tokens, preventing a tailored knowledge transfer process where difficult tokens receive corrective focus and easy tokens contribute to learning the full output distribution for better generalization.

To better understand the consequences of this uniform treatment, we first investigate the learning dynamics of a instruction-following task at the token level (Fig. 1). As shown in the top of Fig. 1, the difficulty of tokens for the student model is not static but evolves throughout the training process. Some tokens are persistently challenging (*e.g.*, the token 2007, highlighted in the red box), requiring continuous focus. Others see their difficulty change dynamically (*e.g.*, Le and Bron, in the orange box), while many "easy" tokens are quickly mastered in the early training stages (*e.g.*, NBA and in, in the green box). This complex dynamic suggests that a static approach is suboptimal and motivates a token-wise, adaptive strategy.

Furthermore, we question the utility of continuing to train on "easy" tokens. We categorize tokens into "hard", "mid", and "easy" groups based on their difficulty and analyze their gradients. As shown in Fig. 1b, easy tokens contribute negligibly to the parameter update, with their gradient magnitude being very small and their direction being nearly orthogonal to the overall batch gradient. More critically, Fig. 1a reveals that the gradients of these easy tokens are unstable and poorly aligned with the supervised fine-tuning (SFT) direction, sometimes even moving in the opposite direction (negative cosine similarity). This evidence suggests that easy tokens provide limited learning value post-initial learning

[*]These authors contributed equally.
[†]Corresponding Authors.

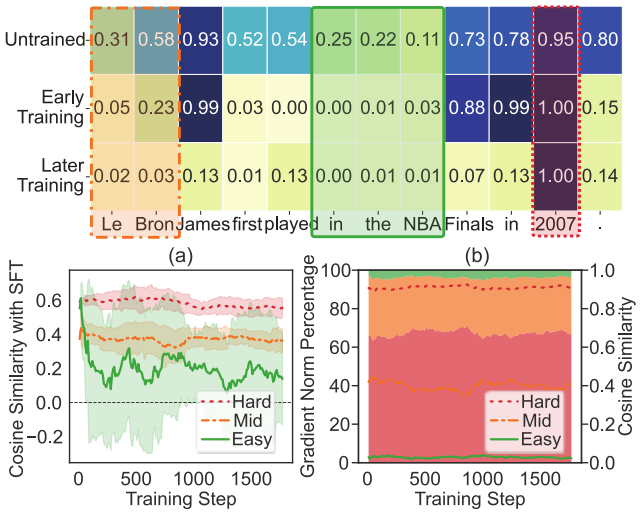

Figure 1: **Top:** Evolution of token difficulty (Hellinger distance between teacher and student models) across training stages. **Bottom:** Tokens are grouped into *Hard*, *Mid*, and *Easy* by difficulty. Sub-figure (a) shows cosine similarity of token group gradients with the SFT gradient, while (b) Group's gradient norm percentage and cosine similarity with the total batch gradient.

and may hinder knowledge transfer efficiency and stability via small, unstable gradients introducing conflicting signals.

To address above two limitations, we propose a novel Token-**Ada**ptive **K**nowledge **D**istillation (AdaKD) framework, which introduces a unified token difficulty metric driving two adaptive modules: ***1)*** Loss-driven Adaptive Token Focusing (LATF) module dynamically selects the most valuable tokens for training at each stage, ***2)*** while the Inverse Difficulty Temperature Scaling (IDTS) module taps temperature scaling's potential by assigning individual temperatures to tokens according to their learning difficulty.

In summary, our contributions are threefold.

- We introduce a novel adaptive token selection mechanism that improves distillation efficiency by dynamically adjusting its focus based on the student model's learning stability.
- A novel token-level temperature scaling strategy that inversely correlates temperature with token difficulty to achieve both targeted error correction and enhanced generalization.
- Extensive empirical validation of AdaKD as a versatile and plug-and-play enhancement that consistently improves a variety of distillation baselines and architectures.

Related Work

Knowledge Distillation for LLMs. Knowledge Distillation (KD) transfers knowledge from a large teacher to a smaller student. Methods are broadly divided into black-box and white-box distillation. Black-box approaches (Yu et al. 2024; Hsieh et al. 2023; Ho, Schmid, and Yun 2022) use only the teacher's final outputs, making them suitable for closed-source models (Achiam et al. 2023; Team et al. 2023; Anthropic 2023) with less practical utility. Our work is in white-box distillation, which accesses teacher internals. Within this setting, feature-based distillation aligns intermediate hidden states, but this often requires complex, architecture-specific layer matching (Sun et al. 2019; Wang et al. 2020; Liang et al. 2023). Logit-based distillation, in contrast, offers a simpler approach by matching the final output distributions using a divergence measure. Beyond the foundational Forward KL Divergence (FKD)(Hinton, Vinyals, and Dean 2015) and Reverse KL Divergence (RKD)(Gu et al. 2024), much recent work has focused on developing more advanced objective functions (Wu et al. 2024; Ko et al. 2024; Wang et al. 2025). Our approach is *a plug-and-play framework that can be flexibly combined with these different objective functions*.

Selective Token Distillation. Traditional KD treats all tokens equally, though not all tokens are equally informative (Piantadosi 2014). Consequently, many selective strategies have been proposed. One major direction is to focus the distillation loss on a subset of "important" tokens, identified based on metrics like difficulty or contribution to the teacher's prediction (Wang et al. 2021; Wu et al. 2023; Zhou et al. 2023). Another line of work operates at the vocabulary level, for instance by preserving the relative order of top predictions (Zhang et al. 2023; Peng and Zhang 2025) or distilling only the top-k logits for efficiency (Raman et al. 2023; Liu et al. 2024). A common limitation in these approaches is the reliance on static or scheduled criteria. In contrast, *our framework adjusts its learning focus based on the dynamic evolution of the training loss*.

Adaptive Temperature Scaling. The distillation temperature is a key hyperparameter that modulates knowledge transfer by smoothing logits. Most methods employ a fixed temperature, which struggles to adapt to the student's evolving learning state. Consequently, dynamic temperature scaling has been well-explored, particularly in computer vision. Some strategies involve using different temperatures to normalize teacher and student logits (Guo 2020; Chi et al. 2023), while others adopt curriculum-based approaches that adjust the temperature to create an easy-to-difficult learning path (Li et al. 2023). However, such adaptive strategies are less common in LLM distillation. A representative work is Annealing KD (Jafari et al. 2021), which lowers the temperature according to a predefined schedule. Such scheduled approaches are not adaptive to the model's real-time needs. In contrast to the global, fixed temperature prevalent in existing work, we introduce *a novel token-level adaptive temperature scaling*.

Methodology

Preliminary of Knowledge Distillation in LLM

Inference in Large Language Models (LLMs) is a sequential vocabulary classification task. Given a pair of prompt and target response, denoted as $(\mathbf{x}, \mathbf{y})$, where $\mathbf{y} = (y_1, \ldots, y_L)$ is the target output sequence of length L, LLMs aim to predict the conditional probability distribution $p(\cdot|\mathbf{x}, y_{<i})$ over the vocabulary $\mathcal{V}$ for each token $y_i \sim p(\cdot|\mathbf{x}, y_{<i})$. KD minimizes the difference between the distributions predicted by the

teacher p and the student q_θ (parameterized by θ). These distributions are obtained by applying a softmax function to the model output logits z, scaled by a distillation temperature τ: $P(\cdot|\mathbf{x}, y_{<i}; \tau) = \text{softmax}(z_P(\cdot|\mathbf{x}, y_{<i})/\tau)$, where $P \in \{p, q_\theta\}$. The distillation loss is typically the average of token-level divergences computed using these temperature-scaled distributions, for each token y_i in the ground-truth response $\mathbf{y}$. Thus, the classic FKD distillation loss (Hinton, Vinyals, and Dean 2015) is defined as:

$$\mathcal{L}_{\text{FKD}} = \frac{1}{L}\sum_{i=1}^{L} D_{KL}(p(\cdot|\mathbf{x}, y_{<i}; \tau) \parallel q_\theta(\cdot|\mathbf{x}, y_{<i}; \tau)). \quad (1)$$

The KL divergence is computed over the vocabulary $\mathcal{V}$. Notably, the inclusion of temperature τ in the softmax function leads to a τ^2 scaling factor in the final loss computation:

$$D_{KL}(p \parallel q_\theta) = \tau^2 \sum_{y_i \in \mathcal{V}} p(y_i|\mathbf{x}, y_{<i}; \tau) \log \frac{p(y_i|\mathbf{x}, y_{<i}; \tau)}{q_\theta(y_i|\mathbf{x}, y_{<i}; \tau)}. \quad (2)$$

Conversely, RKD loss (Gu et al. 2024) swaps the order of the distributions in the KL divergence, focusing on matching the modes of the teacher's distribution. These divergence measures form the basis of the distillation loss.

AdaKD: Token-Adaptive Knowledge Distillation

Building upon the insights from our analysis of token-level learning dynamics (Figure 1), we introduce Token-Adaptive Knowledge Distillation (AdaKD). Instead of a static approach, AdaKD is designed to dynamically tailor the distillation process—both its focus and intensity—to the real-time learning difficulty of each individual token. A detailed comparison of our framework with other relative methods is deferred to the extended version.

The entire AdaKD procedure is described in Fig. 2. Our framework is driven by two synergistic modules: Loss-driven Adaptive Token Focusing (LATF), which selects the most valuable tokens for training at each phase, and Inverse Difficulty Temperature Scaling (IDTS), which assigns a tailored temperature to each selected token. The foundation for both modules is a robust token difficulty indicator, which we will describe first.

Choice of Difficulty Indicator

The effectiveness of AdaKD depends on a metric that accurately quantifies token-level learning difficulty. We define this difficulty using the Hellinger distance (Hellinger 1909), which measures the divergence between the teacher's and student's output probability distributions. For the i-th output token y_i, its difficulty score s_i is calculated as:

$$s_i = \frac{1}{\sqrt{2}}\sqrt{\sum_{y_i \in \mathcal{V}} \left(\sqrt{p(y_i|\mathbf{x}, y_{<i})} - \sqrt{q_\theta(y_i|\mathbf{x}, y_{<i})}\right)^2}. \quad (3)$$

The resulting score s_i is bounded within the range of $[0, 1]$. This indicator is chosen for its advantageous properties. First, its symmetry provides an unbiased measure of discrepancy, avoiding the inherent mode- or mean-seeking tendencies of asymmetric metrics like FKD and RKD. Second, its square-root operation compares the entire output distributions and

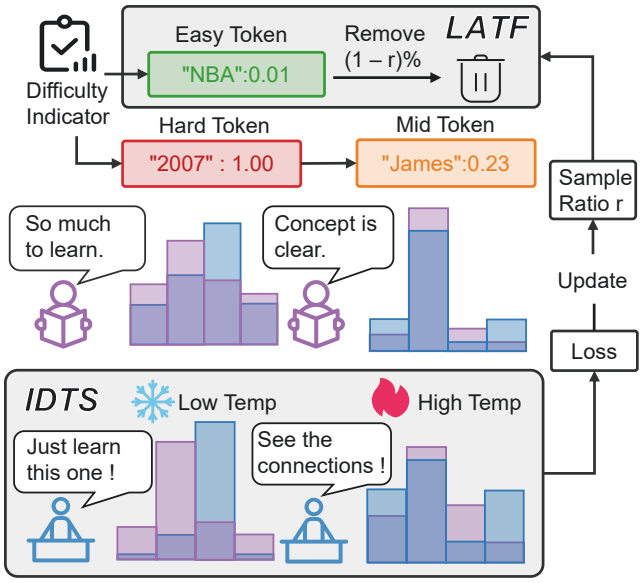

Figure 2: **Illustration of the AdaKD framework**. The bar charts visualize simplified teacher (blue) and student (purple) probability distributions. The top charts depict the initial learning gaps for "hard" and "mid-difficulty" tokens. After the LATF module filters tokens based on difficulty calculated via indicator, the IDTS module (bottom) applies low temperature to hard tokens for a sharp, corrective signal, and high temperature to easier tokens for a smoother distribution that enhances generalization.

is particularly sensitive to disagreements on low-probability candidates, thus providing a more comprehensive difficulty signal that captures subtle deviations in the student's replication of the teacher's full output distribution. *This difficulty indicator* $\mathbf{s} = (s_1, \ldots, s_L)$ *then serves as the sole driving signal to synergistically guide the following two innovative modules that we designed.*

Loss-driven Adaptive Token Focusing (LATF)

The gradient analysis in Fig. 1a and Fig. 1b reveals that training on "easy" tokens becomes inefficient and potentially unstable as training progresses. This strongly suggests that selectively focusing the distillation loss on a more valuable subset of tokens is beneficial. We implement this by applying the loss only to the top-$r\%$ of tokens with the highest difficulty scores:

$$\mathcal{L}_{\text{distill}} = \frac{1}{L * r\%}\sum_{i=1}^{L} I_{r\%}(y_i) \cdot D_{KL}(q_\theta \parallel p), \quad (4)$$

where $L * r\%$ is the number of tokens that fall within the top-$r\%$ of the difficulty metric. The indicator function $I_{r\%}(y_i)$ is defined as:

$$I_{r\%}(y_i) = \begin{cases} 1 & \text{if } s_i \text{ ranks in the top } r\% \text{ of } \mathbf{s} \\ 0 & \text{otherwise} \end{cases}. \quad (5)$$

However, using a fixed sample ratio r is suboptimal. A static ratio cannot adapt to the model's changing learning state. To address this, we introduce LATF to adjust the focusing ratio r_t dynamically. LATF operates via a simple feedback loop that monitors learning stability through the distillation loss. To obtain a stable signal, we first compute the exponential moving average (EMA) of the loss, denoted as $\bar{\mathcal{L}}_t$:

$$\bar{\mathcal{L}}_t = \beta \cdot \bar{\mathcal{L}}_{t-1} + (1-\beta) \cdot \mathcal{L}_{\text{distill},t}, \tag{6}$$

where β is the decay rate of EMA and $\bar{\mathcal{L}}_0$ is the distillation loss when the student model is untrained.

After an warm-up phase (where $r_t = 1.0$), we set a loss reference point $\mathcal{L}_{\text{ref}}$, which is initialized with the current EMA loss $\bar{\mathcal{L}}_t$. At each subsequent training step, LATF dynamically adjusts r_t by comparing the latest $\bar{\mathcal{L}}_t$ to $\mathcal{L}_{\text{ref}}$ within a tolerance ϵ. Specifically, the update rules be described as:

$$r_t = \begin{cases} r_{t-1} \cdot (1-\delta) & \text{if } \bar{\mathcal{L}}_{t-1} < \mathcal{L}_{\text{ref}} \cdot (1-\epsilon) \\ \min(1.0, r_{t-1} \cdot (1+\delta)) & \text{if } \bar{\mathcal{L}}_{t-1} > \mathcal{L}_{\text{ref}} \cdot (1+\epsilon) \\ r_{t-1} & \text{otherwise,} \end{cases} \tag{7}$$

where δ is a small step size that controls the magnitude of adjustment. This rule creates an intuitive feedback loop. We decrease the selection ratio r_t to focus on more challenging tokens when the learning state is stable ($\bar{\mathcal{L}}_t$ drops below the lower bound). Conversely, we increase r_t to incorporate simpler tokens for stabilization when the model struggles. The ratio remains unchanged within the tolerance zone to prevent over-reaction to normal training oscillations. After any adjustment to r_t, the reference point $\mathcal{L}_{\text{ref}}$ is reset to the current $\bar{\mathcal{L}}_t$, keeping the performance baseline adaptive.

Inverse Difficulty Temperature Scaling (IDTS)

Once LATF selects the tokens, IDTS determines the optimal temperature for distilling each one. Contrary to the conventional approach of using a high temperature to soften the teacher's distribution (Jafari et al. 2021), we propose an inverse strategy: applying low temperatures to difficult tokens and high temperatures to easier ones.

Consider the information entropy (Shannon 1948) of a probability distribution $\mathbf{p} = (p_1, \cdots, p_{\mathcal{V}})$, defined as $H(\mathbf{p}) = -\sum_i p_i \ln(p_i)$, which quantifies the uncertainty of the distribution. The relationship between entropy and temperature can be precisely described by the derivative:

$$dH/d\tau = \text{Var}_{p(\tau)}(z)/\tau^3, \tag{8}$$

where $\text{Var}_{p(\tau)}(z)$ denotes the variance of logit z under the distribution p generated with temperature τ. As variance is non-negative and $\tau > 0$, the derivative in Equation (8) is always non-negative, indicating that entropy is a monotonically increasing function of temperature. The proof is in extended version.

Our IDTS module leverages this mathematical principle. For difficult tokens (high s_i), a low τ_i reduces the entropy, simplifying the learning objective into a sharp, corrective signal that focuses the student on matching the teacher's single best prediction. For easy tokens (low s_i), a high τ_i increases entropy, changing the objective to be more extractive

Algorithm 1 Training Procedure of AdaKD.

Input: Teacher p, student q_{θ_0}, dataset $\mathcal{D}$, total iterations T, temperature scale c, EMA decay rate β, tolerance ϵ, step size δ, warm-up steps T_{warmup}
Output: Trained student model q_{θ_T}.
Initialize $t = 1$, sample ratio $r_0 = 1.0$, compute initial loss $\bar{\mathcal{L}}_0$ with q_{θ_0}, $\mathcal{L}_{\text{ref}} = \infty$.
while $t < T$ **do**
  Sample batch $(\mathbf{x}, \mathbf{y}) \sim \mathcal{D}$; Compute logits $\mathbf{z}_p, \mathbf{z}_{q_{\theta_t}}$
  Compute per-token difficulty scores $\mathbf{s}$ using Eq. 3
  Update focusing ratio r_t using Eq. 7
  if $t > T_{\text{warmup}}$ and $r_t \neq r_{t-1}$ **then**
    $\mathcal{L}_{\text{ref}} \leftarrow \bar{\mathcal{L}}_{t-1}$.
  end if
  Compute per-token temperatures τ using Eq. 10
  $p \leftarrow \text{softmax}(\mathbf{z}_p/\tau)$
  $q_\theta \leftarrow \text{softmax}(\mathbf{z}_{q_{\theta_t}}/\tau)$.
  Compute $\mathcal{L}_{\text{AdaKD}}$ using Eq. 4
  Update $\theta : \theta_t \leftarrow \theta_{t-1} - \eta \cdot \nabla_{\theta_t} \mathcal{L}_{\text{AdaKD}}$
  $\bar{\mathcal{L}}_t = \beta \cdot \bar{\mathcal{L}}_{t-1} + (1-\beta) \cdot \mathcal{L}_{\text{AdaKD}}$
  $t \leftarrow t + 1$
end while

. This encourages the student to learn the broader shape of the teacher's distribution, thereby enhancing generalization. This approach is consistent with recent findings in adaptive temperature methods (Yang et al. 2025; Long et al. 2024).

The implementation begins by converting the raw difficulty score s_i into a normalized learning state $\hat{s}_i \in [-1, 1]$. This process is designed for robustness and stability: we first compute the ratio of s_i to the batch median, chosen for its robustness to outliers. We then apply a $\log$ function to compress the long-tail distribution of these ratios, followed by a $\tanh$ function to smoothly map the result into the bounded range:

$$\hat{s}_i = \tanh\left(\log\left(s_i / \text{median}(\mathbf{s})\right)\right). \tag{9}$$

Subsequently, this learning state $\hat{s}_i$ dynamically modulates a base temperature τ_{base} via an exponential function:

$$\tau_i = \tau_{\text{base}} \cdot \exp(-c \cdot \hat{s}_i.\text{detach}()). \tag{10}$$

Here, the negative sign enacts our inverse difficulty principle, with the hyperparameter c controlling the modulation intensity. We chose this multiplicative approach because it makes the scaling effect robust to the specific value of τ_{base} and naturally constrains the final temperature τ_i to the predictable range of $[\tau_{base} \cdot e^{-c}, \tau_{base} \cdot e^{c}]$. The entire calculation is detached from the computation graph, treating the resulting temperatures as fixed supervisory signals. The full AdaKD procedure, combining LATF and IDTS, is detailed in Alg. 1.

Gradient Analysis of IDTS

The loss function activates only high-difficulty tokens (where $I_{r\%}(y_i) = 1$). For these tokens, we compute the temperature τ_i scaling gradient of the KL divergence $D_{\text{KL}}(q_\theta \parallel p)$ with respect to student logits z_q:

$$\frac{\partial D_{\text{KL}}^{(\tau_i)}}{\partial z_q(y_j)} = \frac{1}{\tau_i}\left(q_\theta^{(\tau_i)}(y_j) - p^{(\tau_i)}(y_j)\right). \tag{11}$$

Model	Parameters	Method	Dolly	Self-Inst	Vicuna Eval	S-NI	UnNI	Avg.	
Qwen2 (Yang et al. 2024)	7B	Teacher	29.29±0.56	24.01±0.63	20.18±0.72	40.74±0.63	37.21±0.76	30.29	
	1.5B	SFT (LoRA)	24.82±0.30	18.79±0.55	17.99±0.65	32.13±0.93	31.04±0.60	24.95	
		FKD	25.72±0.85	20.13±0.97	18.24±0.30	35.77±0.37	32.93±1.21	26.56	
		w/ AdaKD	25.94±0.31	19.75±0.24	18.36±0.18	35.88±0.53	33.21±0.51	26.63	(↑0.07)
		RKD	29.52±0.50	**24.92**±0.66	22.50±0.51	41.68±0.67	39.90±0.49	31.70	
		w/ AdaKD	30.03±0.40	24.88±0.71	22.97±0.58	43.82±0.78	**43.17**±0.32	**32.97**	(↑1.27)
		ABKD	29.43±0.60	23.45±0.63	22.72±0.60	41.60±0.79	40.34±0.47	31.51	
		w/ AdaKD	**30.44**±0.50	23.60±0.75	**23.40**±0.77	**44.23**±1.39	42.54±0.78	32.84	(↑1.33)
		GKD	27.13±0.47	20.89±0.90	19.41±0.39	38.25±0.84	35.01±0.59	28.14	
		w/ AdaKD	27.98±0.58	23.00±0.78	19.62±0.17	40.31±0.97	37.77±0.71	29.74	(↑1.60)
		Distillm	29.10±0.51	22.92±0.64	21.79±0.44	41.26±0.46	38.80±0.73	30.77	
		w/ AdaKD	29.69±0.40	23.55±0.96	22.11±0.55	42.91±0.80	40.73±0.82	31.80	(↑1.03)
OpenLLaMA2 (Geng and Liu 2023)	7B	Teacher	28.16±0.60	20.40±0.92	17.62±0.48	30.45±0.82	33.18±0.47	25.96	
	3B	SFT (LoRA)	26.54±0.13	17.45±0.42	16.87±0.27	31.64±0.88	30.64±0.49	24.63	
		FKD	26.56±0.38	18.11±0.60	16.78±0.40	31.94±0.79	30.97±0.52	24.87	
		w/ AdaKD	26.96±0.58	18.75±0.55	16.64±0.47	32.78±0.92	31.64±0.65	25.35	(↑0.48)
		RKD	29.13±0.34	20.08±0.66	19.49±0.28	35.20±0.60	37.60±0.62	28.30	
		w/ AdaKD	29.81±0.35	20.00±0.55	19.49±0.37	36.80±1.13	**40.26**±0.54	29.27	(↑0.97)
		ABKD	29.45±0.77	20.96±0.76	**19.78**±0.26	35.98±0.74	38.60±0.63	28.95	
		w/ AdaKD	**30.19**±0.50	20.65±0.32	19.55±0.28	36.38±0.30	39.82±0.56	29.32	(↑0.37)
		GKD	29.23±0.41	19.96±0.80	18.10±0.75	34.68±0.58	35.05±0.63	27.40	
		w/ AdaKD	29.48±0.15	20.96±0.56	19.07±0.32	**37.60**±0.43	39.31±0.27	29.28	(↑1.88)
		Distillm	29.50±0.56	20.67±0.86	19.09±0.44	35.58±0.66	37.39±1.13	28.45	
		w/ AdaKD	29.52±0.63	**22.13**±0.47	19.50±0.50	37.23±0.72	40.14±0.71	**29.70**	(↑1.25)

Table 1: **Comparison of ROUGE-L scores for various KD methods on five instruction-following benchmarks.** All experiments were conducted using five different random seeds, with results reported as 'mean ± standard deviation'. For each student model configuration, optimal and sub-optimal results are highlighted in **bold** and underline. 'w/ AdaKD' denotes our proposed plug-and-play enhancement, which **consistently improves performance across different base models**.

The update magnitude is governed by the gradient norm:

$$\left\|\nabla D_{\text{KL}}^{(\tau_i)}\right\|^2 = \sum_{y_j \in \mathcal{V}} \left(\frac{\partial D_{\text{KL}}^{(\tau_i)}}{\partial z_q(y_j)}\right)^2 = \frac{1}{\tau_i^2}\left\|q_\theta^{(\tau_i)} - p^{(\tau_i)}\right\|_2^2. \quad (12)$$

To minimize $\mathcal{L}_{\text{distill}}$, we need to maximize this gradient magnitude for accelerated convergence:

$$\min \mathcal{L}_{\text{distill}} \implies \max \sum_{\substack{i \\ I_{r\%}(y_i)=1}} \frac{1}{\tau_i^2}\left\|q_\theta^{(\tau_i)} - p^{(\tau_i)}\right\|_2^2. \quad (13)$$

The difficulty metric s_i is defined as the Hellinger distance:

$$s_i = \frac{1}{\sqrt{2}}\left\|\sqrt{p} - \sqrt{q_\theta}\right\|_2 \implies \left\|\sqrt{p} - \sqrt{q_\theta}\right\|_2^2 = 2s_i^2. \quad (14)$$

Temperature scaling modifies the distribution discrepancy:

$$\left\|q_\theta^{(\tau_i)} - p^{(\tau_i)}\right\|_2^2 \propto \frac{1}{\tau_i^2}\left\|q_\theta - p\right\|_2^2. \quad (15)$$

Combining these relationships yields:

$$\left\|q_\theta^{(\tau_i)} - p^{(\tau_i)}\right\|_2^2 \propto \frac{s_i^2}{\tau_i^2} \quad (16)$$

$$\left\|\nabla D_{\text{KL}}^{(\tau_i)}\right\|^2 \propto \frac{1}{\tau_i^2} \cdot \frac{s_i^2}{\tau_i^2} = \frac{s_i^2}{\tau_i^4} \quad (17)$$

Thus, the KL loss gradient is inversely related to the temperature τ. For difficult tokens, there is a larger discrepancy between the output distributions of the student and the teacher, student model require a larger gradient to approximate the teacher's distribution, which corresponds to a lower temperature. For easy tokens, the output distributions of the student and teacher are more similar, the student model need a smaller gradient to prevent itself from diverging, which corresponds to a higher temperature.

Experiments

Experimental Setup

Datasets and Models. Following the widely-adopted setup from Gu et al. (2024); Ko et al. (2024), we use the databricks-dolly-15k dataset for training and evaluate on five instruction-following benchmarks: Dolly-eval, Self-Instruct, Vicuna-eval, Super-Natural Instructions (S-NI), and Unnatural Instructions. We demonstrate the generalizability of our framework on two modern model families: Qwen2-7B distilled to Qwen2-1.5B and OpenLLaMA2-7B to OpenLLaMA2-3B.

Method	Dolly	S-NI	UnNI	Avg.
RKD (Baseline)	29.52	41.68	39.90	37.03
+ IDTS	**30.12**	43.70	41.83	38.55
+ LATF	29.50	41.88	39.82	37.07
AdaKD (Full)	30.03	**43.82**	**43.17**	**39.01**

Table 2: Ablation study of AdaKD's core components.

To save space, experiments on GPT-2 models and detailed information on the dataset are included in extended version.

Baselines and Implementation Details. We compare AdaKD with supervised fine-tuning (SFT) and state-of-the-art KD methods, including FKD, RKD (Gu et al. 2024), ABKD (Wang et al. 2025), GKD (Agarwal et al. 2024), and DistiLLM (Ko et al. 2024). For a fair comparison, all baselines are reproduced using their official implementations. Further information about baseline and the full implement details are deferred to the extended version.

Evaluation. We report the **ROUGE-L** (Lin 2004) score to measure the quality of generated text. Following standard practice, we generate responses using nucleus sampling with top-p=1.0 and a temperature of 1.0. To ensure statistical robustness, all reported scores are averaged over five runs with different random seeds. Results on other evaluation metrics are provided in the extended version.

Quantitative Results

Table 1 validates AdaKD as a universal plug-and-play enhancement. While advanced objectives (*e.g.*, RKD, ABKD) already outperform foundational methods and can even surpass complex SGO-based approaches(*e.g.*, GKD), AdaKD consistently elevates all of them to new state-of-the-art performance. This universal improvement demonstrates that dynamically adapting the distillation process to the student's real-time learning state is a robust and crucial element for effective knowledge transfer, offering a fundamental enhancement regardless of the underlying distillation objective.

Ablation Studies and Analyses

We conduct ablation studies on the Qwen2-7B $\rightarrow$ Qwen2-1.5B distillation task using RKD as the baseline to dissect the contribution of each component in AdaKD. For clarity, the following tables detail results on Dolly, S-NI, and UnNI, the three benchmarks with the most extensive test items.

Impact of Core Components in AdaKD. Tab. 2 reveals the synergy between our components. While integrating IDTS alone brings a substantial performance boost, LATF alone yields no improvement. This is consistent with our gradient analysis (Fig. 1): LATF's primary role is to stabilize training by filtering out mastered tokens with unstable gradients, rather than directly advancing performance. The full AdaKD model achieves the best results, confirming a crucial synergy: LATF first removes noise to stabilize the learning process, which then allows IDTS to more effectively apply its adaptive teaching strategy to the remaining high-value tokens.

Analysis of the Difficulty Indicator. We evaluated several distribution metrics as the difficulty indicator, with results presented in Tab. 3. The results highlight that the optimal metric for an indicator differs from the distillation loss itself; for instance, FKL is a much more effective indicator than RKD; furthermore, symmetric metrics like Hellinger distance and JS-Divergence show a clear advantage over asymmetric ones on the large-scale S-NI and UnNI benchmarks. We also observe that Cross-Entropy, measured against the ground truth, performs best on the Dolly dataset, while NMTKD (Zhang et al. 2023), which focuses on aligning the top-k (k=5) predictions of each token, also demonstrates competitive performance. Ultimately, Hellinger distance achieves the highest average score, validating its use to provide the balanced and comprehensive disagreement signal crucial for our adaptive framework.

Method	Dolly	S-NI	UnNI	Avg.
RKD (Baseline)	29.52	41.68	39.90	37.03
AdaKD with different metrics:				
+ FKD	30.29	42.81	42.34	38.48
+ RKD	29.93	43.02	42.23	38.39
+ Cross-Entropy	**30.46**	43.46	42.63	38.85
+ JS-Divergence	30.15	43.39	42.58	38.71
+ NMTKD	30.27	43.61	42.64	38.84
+ Hellinger (Ours)	30.03	**43.82**	**43.17**	**39.01**

Table 3: Ablation study on the choice of difficulty indicator.

Method	Dolly	S-NI	UnNI	Avg.
fixed $r = 1.0$	30.12	43.70	41.83	38.55
LATF	30.03	**43.82**	**43.17**	**39.01**
fixed $r = 0.75$	30.27	43.49	42.02	38.59
linear $r : 1.0 \rightarrow 0.75$	29.63	43.37	41.83	38.28
cosine $r : 1.0 \rightarrow 0.75$	**30.29**	43.61	42.70	38.87
cosine $r : 1.0 \rightarrow 0.5$	29.74	42.71	41.74	38.06

Table 4: Ablation study on the design of LATF.

Analysis of LATF's Design. We compare LATF against static and scheduled strategies in Tab. 4, using a target ratio $r = 0.75$ for a fair comparison based on LATF's observed final value. While these schedules prove competitive, LATF is ultimately more robust. The reason is visualized in Fig. 3(a,b). For LATF (Fig. 3a), the sample ratio adapts to the training loss in real-time. The temporary increase in loss is an expected outcome of the model tackling a harder and more focused curriculum, a challenge it successfully overcomes. In contrast, scheduled methods (Fig. 3b) exhibit rising loss in later stages as they blindly enforce difficulty. This real-time adaptation makes LATF a more robust solution that eliminates the need for schedule-specific tuning. Further hyperparameter analysis for LATF and efficiency comparation is deferred to the extended version.

Analysis of Key Design Choices within IDTS. Tab. 5 validates IDTS's design against various temperature strategies. The failure of "Inverse Scaling" confirms our hypothesis that low temperatures are crucial for difficult tokens. However, simply using a low temperature globally is insuffi-

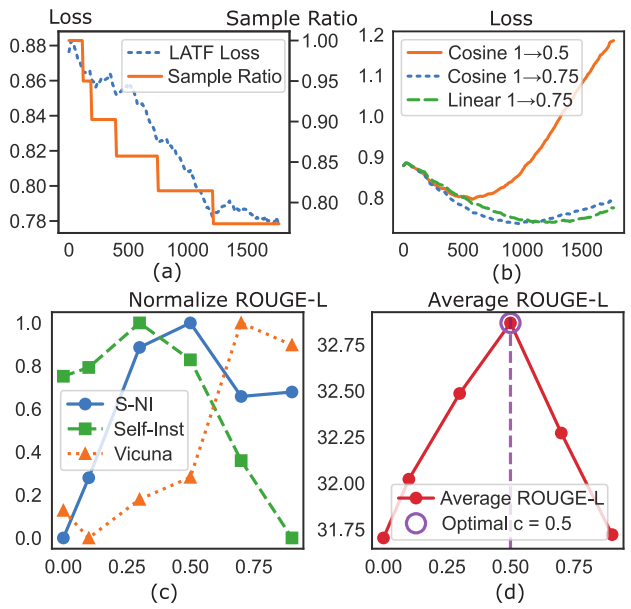

Figure 3: **Top**: Learning curves plotted over training steps for (a) our adaptive LATF and (b) fixed scheduling strategies. **Bottom**: ROUGE-L scores of the IDTS modulation intensity c, showing (c) per-dataset results and (d) the overall average.

Method	**Dolly**	**S-NI**	**UnNI**	**Avg.**
$T = 1.0$	29.50	41.88	39.82	37.07
AdaKD ($c = 0.5$)	30.03	**43.82**	**43.17**	**39.01**
Inverse Scaling ($-c$)	28.93	40.02	38.06	35.67
$T = 0.8$	30.00	41.41	40.10	37.17
$T = 1.2$	29.55	42.07	40.05	37.22
$T \approx e^{-0.5}$	30.19	42.87	41.38	38.15
CTKD	**30.22**	41.99	40.30	37.50
Logit Std.	26.74	40.08	37.46	34.76

Table 5: Ablation on temperature scaling strategies.

cient; AdaKD surpasses not only fixed-temperature baselines but also one using our method's optimal lower bound ($T \approx e^{-0.5}$). This proves the dynamic, token-level application is the key to success. Furthermore, AdaKD's superior performance over other adaptive methods like CTKD (Li et al. 2023) and Logit Std (Chi et al. 2023; Sun et al. 2024) highlights the effectiveness of our specific token-level design.

We analyze the impact of the IDTS modulation intensity c in Fig. 3(c,d). While the optimal c varies for individual datasets (Fig. 3c), the average performance across all benchmarks (Fig. 3d) robustly peaks at $c = 0.5$. We therefore adopt this value for our main experiments.

Analysis of the Dynamic Mechanisms in AdaKD. Fig. 4 illustrates the dynamic synergy of AdaKD's mechanisms by comparing distributions at the start and end of training. Crucially, our IDTS module aligns the information entropy of all tokens, adjusting the learning objective to guide output

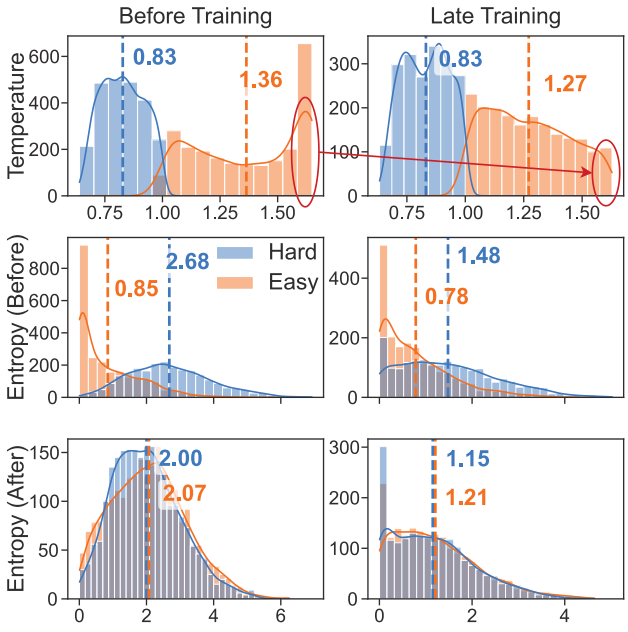

Figure 4: These histograms display the token count (y-axis) distribution over different metrics, comparing the state "Before Training" (left column) with "Late in Training" (right column). The rows, from top to bottom, show the distribution of assigned temperatures, the student's output entropy before IDTS, and the entropy after IDTS. Tokens are split by median into "hard" (blue) and "easy" (orange) groups, with dashed lines indicating their respective means.

distributions toward a uniform uncertainty, irrespective of their initial difficulty.

This strategy is reflected in the temperature distribution: IDTS assigns lower temperatures to hard tokens and higher ones to easy tokens. However, the distribution's evolution (see red circles) reveals the synergy with LATF: early in training, the temperature for easy tokens peaks sharply due to many trivial examples in the "easy" set. Conversely, late in training, LATF removes these mastered tokens, which refines the learning process by causing IDTS to assign a smoother high-temperature range to the remaining valuable "easy" tokens.

Conclusion

In this paper, we introduced Token-Adaptive Knowledge Distillation (AdaKD), a novel framework that dynamically adapts the distillation process to each token's learning state, overcoming the limitations of static distillation strategies. AdaKD synergistically combines Loss-driven Adaptive Token Focusing (LATF) to concentrate on valuable tokens and Inverse Difficulty Temperature Scaling (IDTS) to apply a highly effective temperature strategy for both error correction and generalization. Extensive experiments demonstrate that AdaKD, as a plug-and-play enhancement, consistently improves the performance of various distillation methods across multiple architectures.

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Agarwal, R.; Vieillard, N.; Zhou, Y.; Stanczyk, P.; Garea, S. R.; Geist, M.; and Bachem, O. 2024. On-policy distillation of language models: Learning from self-generated mistakes. In *ICLR*.

Anil, R.; Dai, A. M.; Firat, O.; Johnson, M.; Lepikhin, D.; Passos, A.; Shakeri, S.; Taropa, E.; Bailey, P.; Chen, Z.; et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.

Anthropic. 2023. Introducing Claude. https://www.anthropic.com/news/introducing-claude.

Bai, G.; Chai, Z.; Ling, C.; Wang, S.; Lu, J.; Zhang, N.; Shi, T.; Yu, Z.; Zhu, M.; Zhang, Y.; et al. 2024. Beyond efficiency: A systematic survey of resource-efficient large language models. *arXiv preprint arXiv:2401.00625*.

Chi, Z.; Zheng, T.; Li, H.; Yang, Z.; Wu, B.; Lin, B.; and Cai, D. 2023. Normkd: Normalized logits for knowledge distillation. *arXiv preprint arXiv:2308.00520*.

Geng, X.; and Liu, H. 2023. Openllama: An open reproduction of llama.

Grattafiori, A.; Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Vaughan, A.; et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

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

Guo, J. 2020. Reducing the teacher-student gap via adaptive temperatures. *arXiv preprint arXiv:2010.07485*.

Hellinger, E. 1909. Neue begründung der theorie quadratischer formen von unendlichvielen veränderlichen. *Journal für die reine und angewandte Mathematik*, 1909(136): 210–271.

Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Ho, N.; Schmid, L.; and Yun, S.-Y. 2022. Large language models are reasoning teachers. *arXiv preprint arXiv:2212.10071*.

Hsieh, C.-Y.; Li, C.-L.; Yeh, C.-K.; Nakhost, H.; Fujii, Y.; Ratner, A.; Krishna, R.; Lee, C.-Y.; and Pfister, T. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. *arXiv preprint arXiv:2305.02301*.

Jafari, A.; Rezagholizadeh, M.; Sharma, P.; and Ghodsi, A. 2021. Annealing knowledge distillation. *arXiv preprint arXiv:2104.07163*.

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 preprint arXiv:2001.08361*.

Ko, J.; Kim, S.; Chen, T.; and Yun, S.-Y. 2024. DistiLLM: Towards Streamlined Distillation for Large Language Models. In *Forty-first International Conference on Machine Learning*.

Li, Z.; Li, X.; Yang, L.; Zhao, B.; Song, R.; Luo, L.; Li, J.; and Yang, J. 2023. Curriculum temperature for knowledge distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 1504–1512.

Liang, C.; Zuo, S.; Zhang, Q.; He, P.; Chen, W.; and Zhao, T. 2023. Less is more: Task-aware layer-wise distillation for language model compression. In *International Conference on Machine Learning*.

Lin, C.-Y. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, 74–81.

Liu, X.; Zhang, Y.; Li, W.; Li, S.; Huang, X.; Chen, H.; Tang, Y.; Hu, J.; Xiong, Z.; and Wang, Y. 2024. Multi-Granularity Semantic Revision for Large Language Model Distillation. *arXiv preprint arXiv:2407.10068*.

Long, J.; Yin, Z.; Han, Y.; and Huang, W. 2024. Mkdat: Multi-level knowledge distillation with adaptive temperature for distantly supervised relation extraction. *Information*, 15(7): 382.

Peng, T.; and Zhang, J. 2025. Enhancing Knowledge Distillation of Large Language Models through Efficient Multi-Modal Distribution Alignment. In *Proceedings of the 31st International Conference on Computational Linguistics*, 2478–2496.

Piantadosi, S. T. 2014. Zipf's word frequency law in natural language: A critical review and future directions. *PBR*.

Raman, M.; Mani, P.; Liang, D.; and Lipton, Z. 2023. For distillation, tokens are not all you need. In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*.

Shannon, C. E. 1948. A mathematical theory of communication. *The Bell system technical journal*, 27(3): 379–423.

Sun, S.; Cheng, Y.; Gan, Z.; and Liu, J. 2019. Patient knowledge distillation for bert model compression. *arXiv preprint arXiv:1908.09355*.

Sun, S.; Ren, W.; Li, J.; Wang, R.; and Cao, X. 2024. Logit standardization in knowledge distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 15731–15740.

Team, G.; Anil, R.; Borgeaud, S.; Alayrac, J.-B.; Yu, J.; Soricut, R.; Schalkwyk, J.; Dai, A. M.; Hauth, A.; Millican, K.; et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Wan, Z.; Wang, X.; Liu, C.; Alam, S.; Zheng, Y.; Liu, J.; Qu, Z.; Yan, S.; Zhu, Y.; Zhang, Q.; et al. 2023. Efficient large language models: A survey. *arXiv preprint arXiv:2312.03863*.

Wang, F.; Yan, J.; Meng, F.; and Zhou, J. 2021. Selective Knowledge Distillation for Neural Machine Translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 6456–6466.

Wang, G.; Yang, Z.; Wang, Z.; Wang, S.; Xu, Q.; and Huang, Q. 2025. ABKD: Pursuing a Proper Allocation of the Probability Mass in Knowledge Distillation via α-β-Divergence. *arXiv preprint arXiv:2505.04560*.

Wang, W.; Wei, F.; Dong, L.; Bao, H.; Yang, N.; and Zhou, M. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in neural information processing systems*, 33: 5776–5788.

Wu, S.; Chen, H.; Quan, X.; Wang, Q.; and Wang, R. 2023. Ad-kd: Attribution-driven knowledge distillation for language model compression. *arXiv preprint arXiv:2305.10010*.

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 preprint arXiv:2404.02657*.

Yang, A.; Yang, B.; Hui, B.; Zheng, B.; Yu, B.; Zhou, C.; Li, C.; Li, C.; Liu, D.; Huang, F.; Dong, G.; Wei, H.; Lin, H.; Tang, J.; Wang, J.; Yang, J.; Tu, J.; Zhang, J.; Ma, J.; Yang, J.; Xu, J.; Zhou, J.; Bai, J.; He, J.; Lin, J.; Dang, K.; Lu, K.; Chen, K.; Yang, K.; Li, M.; Xue, M.; Ni, N.; Zhang, P.; Wang, P.; Peng, R.; Men, R.; Gao, R.; Lin, R.; Wang, S.; Bai, S.; Tan, S.; Zhu, T.; Li, T.; Liu, T.; Ge, W.; Deng, X.; Zhou, X.; Ren, X.; Zhang, X.; Wei, X.; Ren, X.; Liu, X.; Fan, Y.; Yao, Y.; Zhang, Y.; Wan, Y.; Chu, Y.; Liu, Y.; Cui, Z.; Zhang, Z.; Guo, Z.; and Fan, Z. 2024. Qwen2 Technical Report. *arXiv preprint arXiv:2407.10671*.

Yang, S.; Yang, X.; Ren, J.; Xu, L.; Yang, J.; Huang, Z.; Gong, Z.; and Wang, W. 2025. Adaptive Temperature Distillation method for mining hard samples' knowledge. *Neurocomputing*, 636: 129745.

Yu, P.; Xu, J.; Weston, J.; and Kulikov, I. 2024. Distilling system 2 into system 1. *arXiv preprint arXiv:2407.06023*.

Zhang, S.; Liang, Y.; Wang, S.; Han, W.; Liu, J.; Xu, J.; and Chen, Y. 2023. Towards Understanding and Improving Knowledge Distillation for Neural Machine Translation. *arXiv preprint arXiv:2305.08096*.

Zheng, Y.; Chen, Y.; Qian, B.; Shi, X.; Shu, Y.; and Chen, J. 2025. A review on edge large language models: Design, execution, and applications. *ACM Computing Surveys*, 57(8): 1–35.

Zhou, Q.; Li, P.; Liu, Y.; Guan, Y.; Xing, Q.; Chen, M.; and Sun, M. 2023. AdaDS: Adaptive data selection for accelerating pre-trained language model knowledge distillation. *AI Open*, 4: 56–63.