Knowledge Distillation from Qwen2.5-7B to Qwen2.5-0.5B on GSM8K

Bui Hai Dang
Vietnam National University
University of Engineering and Technology
Institute for Artificial Intelligence
23020356@vnu.edu.vn

Abstract

Large Language Models (LLMs) have become a dominant paradigm in contemporary AI research; however, their strong reasoning capabilities typically come at the cost of substantial computational requirements, which limit their practical deployment in resource-constrained settings. In this work, we address this challenge through Knowledge Distillation (KD), a training paradigm that transfers knowledge from a large, high-capacity teacher model to a smaller and more efficient student model.

We present a case study that distills Qwen2.5-7B-Instruct (teacher, achieving 85.4% accuracy on GSM8K) into Qwen2.5-0.5B-Instruct as the student, evaluated on the GSM8K benchmark for mathematical reasoning. Experimental results show that the distilled student reaches an accuracy of 52.2%, demonstrating a clear improvement over both the pre-trained base model (41.6%) and the instruction-tuned student model (49.6%), as reported in the original Qwen2.5 report.[Qwen et al., 2025].

These results demonstrate that knowledge distillation can effectively enhance the reasoning performance of compact language models, enabling improved inference accuracy while significantly reducing computational overhead compared to large-scale models.

1 Introduction

Large Language Models (LLMs) have shown strong performance on a variety of natural language processing tasks, particularly in reasoning-intensive settings such as mathematical problem solving and multi-step inference. However, these capabilities are largely achieved through scaling model size, leading to high computational and memory costs that hinder deployment in resource-constrained environments.

Knowledge Distillation (KD) has been widely studied as a model compression technique that transfers knowledge from a large teacher to a smaller student model. While KD has proven effective for classification and representation learning, its application to reasoning-centric, instruction-tuned LLMs remains challenging. In particular, transferring structured reasoning behaviors from large models to compact students without significant performance degradation is still an open problem.

In this work, we investigate the effectiveness of knowledge distillation for improving mathematical reasoning in compact instruction-following language models. We distill a Qwen2.5-7B-Instruct teacher into a Qwen2.5-0.5B-Instruct student, achieving a substantial reduction in model capacity. Experiments on the GSM8K benchmark show that the distilled student attains 52.2% accuracy, outperforming both the pre-trained student baseline and a conventionally supervised fine-tuned model. These results demonstrate that knowledge distillation can effectively transfer reasoning-relevant knowledge to significantly smaller architectures while maintaining computational efficiency.

2 Related Work

Knowledge Distillation (KD) is a widely used technique for compressing large models into smaller ones by transferring knowledge from a teacher to a student model, typically via divergence minimization between their output distributions [Hinton et al., 2015]. Early extensions of KD to sequence generation tasks demonstrated its effectiveness in autoregressive settings such as machine translation [Kim and Rush, 2016], motivating subsequent work on distillation for language generation.

Recent studies adapt KD to large language models (LLMs) to reduce computational cost while preserving generation quality. Offline and sequence-level KD methods have been proposed to handle the large output space of autoregressive models, including reverse-KL–based formulations to mitigate exposure bias [Gu et al., 2025] and on-policy generalized KD objectives [Zhong et al., 2024]. These approaches show that carefully designed KD objectives can significantly improve student performance under strong capacity constraints.

Separately, incorporating chain-of-thought (CoT) reasoning has proven effective for enhancing reasoning capabilities in instruction-following models. Methods such as Self-Instruct [Wang et al., 2023] and subsequent instruction-tuned models [Taori et al., 2023] leverage teacher-generated rationales to transfer reasoning behavior, often relying on supervised fine-tuning over synthetic data.

Our approach combines CoT-based data augmentation with a two-stage offline top-K knowledge distillation framework tailored for mathematical reasoning. By selectively distilling high-confidence teacher predictions and enforcing strict numeric consistency, our method effectively transfers reasoning-relevant knowledge to compact models on GSM8K while maintaining computational efficiency.

3 Method

3.1 Problem Formulation

We distill reasoning and instruction-following behaviors from a large teacher autoregressive language model into a smaller student model. Let (x, y) be an input–output pair from dataset $\mathcal{D}$, where x is a natural-language problem and $y = (y_1, \ldots, y_L)$ is the target autoregressive output sequence (chain-of-thought reasoning followed by a final numeric answer [Wei et al., 2023].)

The teacher defines the next-token conditional distribution

$$p_{\mathcal{T}}(y_t \mid y_{<t}, x), \tag{1}$$

and the student parameterized by θ defines

$$p_\theta(y_t \mid y_{<t}, x). \tag{2}$$

Our goal is to train p_θ such that it produces valid and accurate reasoning traces and aligns its token-level predictive distribution with the teacher, under practical compute/memory constraints [Hinton et al., 2015].

3.2 Overall Distillation Pipeline

We adopt a two main stage distillation framework combining black-box and white-box KD:

- **Stage 0 (Generate CoT answers):** Use the teacher to generate multiple chain-of-thought (CoT) candidates per training dataset and filter by numeric correctness [Wei et al., 2023].
- **Stage 1 (Black-box KD / SFT):** Train the student by supervised fine-tuning on verified teacher CoT solutions [Ho et al., 2023, Magister et al., 2023].
- **Stage 2 (White-box KD):** Further align student token distributions to the teacher using an efficient Top-K distillation loss with offline teacher logits [Gu et al., 2025].

3.3 Stage 1: Black-Box Distillation with Chain-of-Thought Supervision

Teacher generation and filtering. For each input x, we sample K candidate solutions from the teacher:

$$y^{(k)} \sim p_{\mathcal{T}}(y \mid x), \quad k = 1, \ldots, K. \tag{3}$$

Outputs are post-processed to enforce a strict format (numbered steps and a final line ####<number>"). A candidate is accepted if its parsed final numeric answer matches the gold label under exact decimal arithmetic. Among accepted candidates, we keep a single target $y^\star$ (e.g., the shortest correct CoT) to reduce verbosity.

Masked supervised objective (prompt-masked NLL). Training uses teacher forcing on the concatenated prompt+completion sequence. Let $m_t \in \{0, 1\}$ be a mask indicating whether token position t belongs to the *completion* (student output) rather than the prompt. The Stage 1 objective is the masked negative log-likelihood:

$$\mathcal{L}_{\mathrm{SFT}}(\theta) = -\mathbb{E}_{(x,y^\star)\sim\mathcal{D}} \frac{1}{\sum_{t=1}^{L} m_t} \sum_{t=1}^{L} m_t \log p_\theta(y_t^\star \mid y_{<t}^\star, x). \tag{4}$$

In implementation, prompt labels are set to `-100` so loss is computed only where $m_t = 1$.

3.4 Stage 2: White-Box Knowledge Distillation with Top-K Logits

Stage 1 transfers behavioral supervision but does not explicitly match the teacher's uncertainty. We therefore add an efficient token-level KD objective based on offline teacher Top-K logits.

Vocabulary alignment. Because teacher and student vocabularies may differ, we build an alignment map

$$\phi : V_{\mathcal{T}} \to V_{\mathcal{S}} \cup \{-1\}, \tag{5}$$

where $\phi(v) = -1$ denotes an unmapped token (ignored in KD). After alignment, teacher Top-K indices are expressed in the student index space.

Offline teacher Top-K logit extraction (Top-M $\to$ filter $\to$ Top-K). For each training example, we run the teacher once offline on the full prompt+completion sequence and select KD positions that correspond to predicting completion tokens. At each valid position, we first take teacher Top-M logits, map indices through ϕ, drop unmapped tokens and duplicates, and retain the first K mapped entries. We store per example the KD position list and, for each position, the aligned Top-K indices and logits:

$$\Big\{\ t \in \mathcal{N},\ \mathbf{i}_t \in \{1, \ldots, |V_{\mathcal{S}}|\}^K,\ \mathbf{z}_t^{\mathcal{T}} \in \mathbb{R}^K\ \Big\}, \tag{6}$$

where $\mathcal{N}$ denotes the set of valid KD positions (completion-only, non-padding).

Top-K distributions and KD divergence. Let $T_{\mathrm{KD}} > 0$ be the distillation temperature. For each KD position $t \in \mathcal{N}$, let $\mathbf{i}_t = (i_{t,1}, \ldots, i_{t,K})$ be the aligned Top-K token indices and $\mathbf{z}_t^{\mathcal{T}} = (z_{t,1}^{\mathcal{T}}, \ldots, z_{t,K}^{\mathcal{T}})$ the stored teacher logits. Let $\mathbf{z}_t^{\mathcal{S}}$ be the student logits *gathered* at the same indices:

$$z_{t,k}^{\mathcal{S}} = \mathrm{logits}_\theta(t,\, i_{t,k}), \quad k = 1, \ldots, K. \tag{7}$$

We then define *renormalized* Top-K distributions:

$$p_t^{\mathcal{T}}(k) = \frac{\exp(z_{t,k}^{\mathcal{T}}/T_{\mathrm{KD}})}{\sum_{j=1}^{K} \exp(z_{t,j}^{\mathcal{T}}/T_{\mathrm{KD}})}, \quad p_t^{\mathcal{S}}(k) = \frac{\exp(z_{t,k}^{\mathcal{S}}/T_{\mathrm{KD}})}{\sum_{j=1}^{K} \exp(z_{t,j}^{\mathcal{S}}/T_{\mathrm{KD}})}. \tag{8}$$

We support two divergences (matching implementation) [Gu et al., 2025, Kim and Rush, 2016]:

$$D_{\mathrm{FKL}}(t) = \sum_{k=1}^{K} p_t^{\mathcal{T}}(k) \Big(\log p_t^{\mathcal{T}}(k) - \log p_t^{\mathcal{S}}(k) \Big), \tag{9}$$

$$D_{\mathrm{RKL}}(t) = \sum_{k=1}^{K} p_t^{\mathcal{S}}(k) \Big(\log p_t^{\mathcal{S}}(k) - \log p_t^{\mathcal{T}}(k) \Big). \tag{10}$$

The KD loss averages over KD positions and scales by T_{KD}^2:

$$\mathcal{L}_{\mathrm{KD}}(\theta) = T_{\mathrm{KD}}^2 \cdot \mathbb{E}_{(x,y)\sim\mathcal{D}} \frac{1}{|\mathcal{N}|} \sum_{t\in\mathcal{N}} D(t), \tag{11}$$

where $D(t)$ is either $D_{\mathrm{FKL}}(t)$ or $D_{\mathrm{RKL}}(t)$.

Combined objective with linear warmup of KD ratio. Stage 2 optimizes a convex combination of the masked LM loss and the Top-K KD loss. Let $\alpha \in [0, 1]$ be the target KD ratio and let $\rho \in [0, 1]$ be the warmup ratio. Denoting the total training steps as $S_{\max}$ and the current step as s, we set warmup steps $S_w = \lfloor \rho S_{\max} \rfloor$ and:

$$\alpha_s = \alpha \cdot \min\left(1, \frac{s}{S_w}\right). \tag{12}$$

The Stage 2 objective is additive loss function:

$$\mathcal{L}(\theta) = \mathcal{L}_{\mathrm{SFT}}(\theta) + \alpha_s \, \mathcal{L}_{\mathrm{KD}}(\theta). \tag{13}$$

3.5 Numerical Correctness Evaluation

For math reasoning, string matching is brittle. We parse the final answer from the "####" marker, normalize numeric strings, and compare using exact decimal arithmetic:

$$\text{correct} \iff |\hat{a} - a| \leq \varepsilon, \quad \varepsilon = 0 \text{ (default)}. \tag{14}$$

This ensures measured gains reflect genuine numeric correctness rather than formatting artifacts.

3.6 Experimental Setup

Dataset and Preprocessing. We conduct experiments on the GSM8K dataset for grade-school mathematical reasoning. Chain-of-Thought (CoT) solutions are generated offline and stored in a JSONL format. For training, we retain only examples for which at least one generated solution yields a correct final answer (`DROP_WRONG=True`), ensuring high-quality supervision. The maximum sequence length is set to 1024 tokens for both Stage 1 and Stage 2 training.

Stage 1: Black-Box CoT Supervised Fine-Tuning. In Stage 1, the student model is trained using supervised fine-tuning (SFT) on verified CoT solutions. We train for 3 epochs with a per-device batch size of 8 and gradient accumulation of 4, resulting in an effective batch size of 32. The learning rate is set to 2×10^{-5} with a warmup ratio of 0.1, cosine learning rate scheduling, and weight decay of 0.01.

Stage 2: White-Box Top-K Logit Distillation. In Stage 2, we perform offline knowledge distillation using Top-K teacher logits. For each output position, we first extract the top-M logits from the teacher and then select the top-K aligned tokens after vocabulary mapping, with $K = 10$ and $M = 25$. Teacher logits are precomputed using a batch size of 32, and a temperature of $T = 2.0$ is applied when estimating the probability mass covered by the Top-K tokens.

The student is trained for 3 epochs with the same effective batch size as Stage 1. A smaller learning rate of 5×10^{-6} is used, with a warmup ratio of 0.1 and weight decay of 0.01. The final training objective is a weighted combination of the standard language modeling loss and the Top-K KL-divergence loss. The distillation weight is set to $\alpha = 0.35$ and is linearly warmed up over the first 15% of training steps. We use forward KL divergence (KL($p_T \| p_S$)) with a distillation temperature of $T = 2.0$.

Evaluation Protocol. We evaluate all models on the GSM8K test set using deterministic decoding (greedy decoding).

4 Results

We evaluate the performance of our distilled model on the GSM8K benchmark, a widely used dataset for arithmetic reasoning. Our distilled model achieves an accuracy of 52.2%, demonstrating a clear improvement both the pre-trained base model (41.6%) and the instruction-tuned student model (49.6%) [Qwen et al., 2025]. This result demonstrates that the proposed knowledge distillation framework effectively enhances the reasoning capability of compact language models beyond standard instruction tuning.

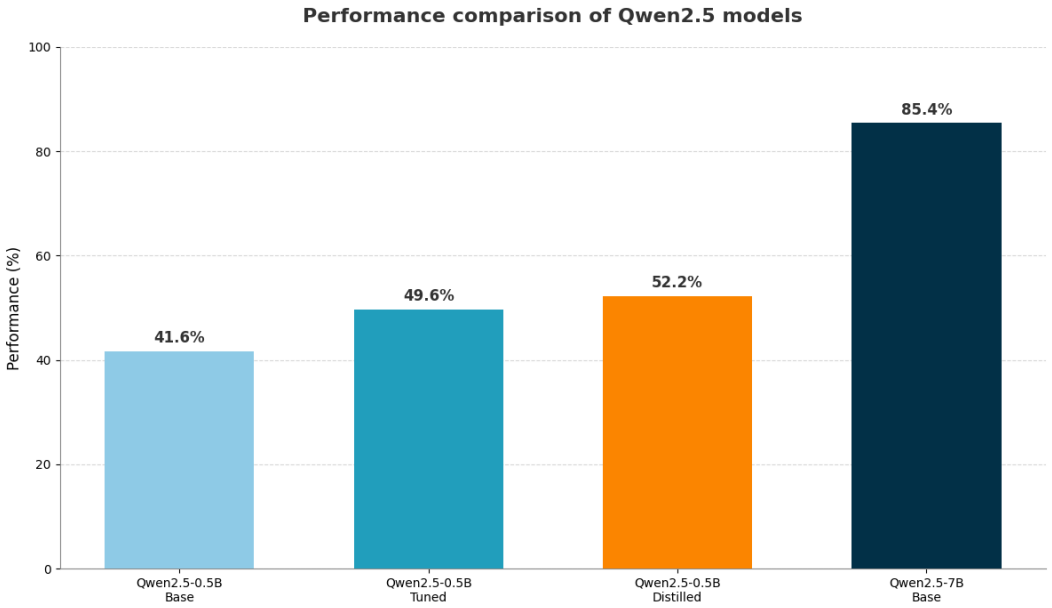

Figure 1: Accuracy comparison on the GSM8K benchmark between the pre-trained base model,instruction-tuned student model, and our distilled student model.

The performance gain can be primarily attributed to the two-stage distillation strategy that combines chain-of-thought (CoT) supervision with token-level white-box distillation. In the first stage, CoT-based supervision encourages the student model to learn structured and explicit reasoning patterns, enabling it to decompose arithmetic problems into intermediate steps rather than directly predicting the final answer. This behavioral alignment with the teacher model leads to more stable reasoning trajectories and reduces error propagation across decoding steps.

In the second stage, token-level distillation further refines the student's predictive behavior by aligning its output distribution with the teacher's Top-K logits. This stage is crucial for capturing the teacher model's uncertainty structure, which is not preserved by instruction tuning alone. By matching the relative probabilities among the most likely candidate tokens, the student model learns to better discriminate between competing arithmetic operations and intermediate values, particularly in ambiguous or multi-step reasoning scenarios.

An important empirical observation underlying the effectiveness of this approach is the strong concentration of the teacher's predictive probability mass on a small number of tokens. We observe that, at each decoding step, the cumulative probability of the top-10 tokens is close to 1, indicating that nearly all informative teacher knowledge is contained within this restricted subset of the vocabulary. This property justifies the use of Top-K distillation and enables an efficient white-box KD framework that avoids full-vocabulary alignment.

Specifically, by distilling only the aligned Top-K teacher logits (with $K = 10$ by default) after token-level vocabulary alignment, the student model is exposed to almost all meaningful teacher signals while significantly reducing memory and computational overhead. This selective distillation strategy focuses learning on semantically relevant alternatives and suppresses noise from the long tail of the output distribution. Such behavior is particularly beneficial for arithmetic reasoning tasks like GSM8K, where small token-level deviations can propagate into incorrect final answers.

Overall, the consistent improvement over the instruction-tuned baseline highlights that reasoning performance gains are not solely due to improved instruction following, but also depend on effectively transferring the teacher's token-level predictive structure. The combination of CoT supervision and Top-K logit distillation enables the student model to achieve stronger generalization and more reliable reasoning performance under constrained model capacity.

5 Discussion

Why does the proposed distillation work? The performance gain on GSM8K (52.2%) indicates that transferring both reasoning behavior and token-level uncertainty is effective under limited model capacity. In Stage 1, chain-of-thought (CoT) supervised fine-tuning improves procedural competence by encouraging the student to externalize intermediate computations and follow a consistent reasoning structure, which stabilizes multi-step decoding. Stage 2 further refines local decision-making by aligning the student's predictions with the teacher's Top-K token distribution. Since the teacher's predictive probability mass is highly concentrated on a small set of tokens, distilling only these candidates preserves most informative signals while remaining computationally efficient. Together, these stages improve both global reasoning trajectories and critical token-level choices in arithmetic problems.

Comparison with instruction tuning. Although instruction tuning enhances adherence to prompt formats and output constraints, it does not explicitly transfer the teacher's relative preferences among plausible next tokens, such as competing operations or intermediate values. Top-K token-level distillation complements instruction tuning by preserving the teacher's uncertainty structure and calibrating the student's local predictions. This reduces error accumulation in long reasoning chains and explains the consistent improvement over the instruction-tuned baseline, beyond gains attributable to formatting or instruction following alone.

Limitations. Our evaluation focuses on final-answer accuracy on GSM8K and does not directly measure reasoning faithfulness; improvements may arise from better step structuring rather than exact alignment with the teacher's internal rationale. In addition, the Top-K approximation, while efficient, may omit low-probability tokens that are relevant in rare or ambiguous cases, and the optimal value of K may vary across domains. Finally, filtering teacher-generated CoT by numeric correctness biases training toward easier examples and token-level alignment based on string matching may be imperfect for heterogeneous tokenizers.

Future directions. Future work includes targeted ablation studies to isolate the respective contributions of CoT supervision and Top-K distillation, as well as evaluation on additional reasoning benchmarks to assess robustness. We also plan to analyze solution stability by generating and evaluating Top-K candidate answers, which can provide insights into the consistency and reliability of the distilled model's reasoning. On the distillation side, adaptive selection of K for high-entropy steps, improved vocabulary alignment, and correctness-aware weighting of KD positions remain promising directions for further improving transfer efficiency and reasoning reliability.

6 Conclusion

We presented a practical two-stage knowledge distillation framework for transferring mathematical reasoning capability from a large teacher LLM to a compact student model. The approach combines black-box CoT distillation via supervised fine-tuning on verified teacher solutions with efficient white-box Top-K logit distillation based on offline teacher inference and vocabulary alignment. On GSM8K, our distilled student achieves 52.2% accuracy, outperforming both the pre-trained student baseline (41.6%) and the instruction-tuned student baseline (49.6%).

These results demonstrate that coupling structured reasoning supervision with token-level distribution alignment is an effective strategy for improving small-model reasoning performance under limited computational budgets. Importantly, the proposed framework is task-agnostic and can be readily applied to other autoregressive reasoning tasks where teacher inference is expensive or unavailable at training time. By decoupling teacher execution from student optimization, our method enables scalable distillation into compact models without incurring additional runtime cost during training.

Future work includes extending the framework to broader reasoning domains, studying sensitivity to Top-K and temperature hyperparameters, and exploring adaptive KD strategies that dynamically select informative distillation positions.

References

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models, 2025. URL https://arxiv.org/abs/2306.08543.

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

Namgyu Ho, Laura Schmid, and Se-Young Yun. Large language models are reasoning teachers, 2023. URL https://arxiv.org/abs/2212.10071.

Yoon Kim and Alexander M. Rush. Sequence-level knowledge distillation, 2016. URL https://arxiv.org/abs/1606.07947.

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. Teaching small language models to reason, 2023. URL https://arxiv.org/abs/2212.08410.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, , et al. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Alpaca: A Strong, Replicable Instruction-Following Model, March 2023. URL https://crfm.stanford.edu/2023/03/13/alpaca.html.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions, 2023. URL https://arxiv.org/abs/2212.10560.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023. URL https://arxiv.org/abs/2201.11903.

Qihuang Zhong, Liang Ding, Li Shen, Juhua Liu, Bo Du, and Dacheng Tao. Revisiting knowledge distillation for autoregressive language models, 2024. URL https://arxiv.org/abs/2402.11890.