MiniLLM: Knowledge Distillation of Large Language Models

Yuxian Gu[1], Li Dong[2], Furu Wei[2], Minlie Huang[1]

[1]CoAI Group, Tsinghua University

[2]Microsoft Research

Tsinghua University

Knowledge Distillation (KD)

- Background: Conventional KD
  - Common technique for model compression
  - Conventional KD based on **forward KLD** works well for image/text classification

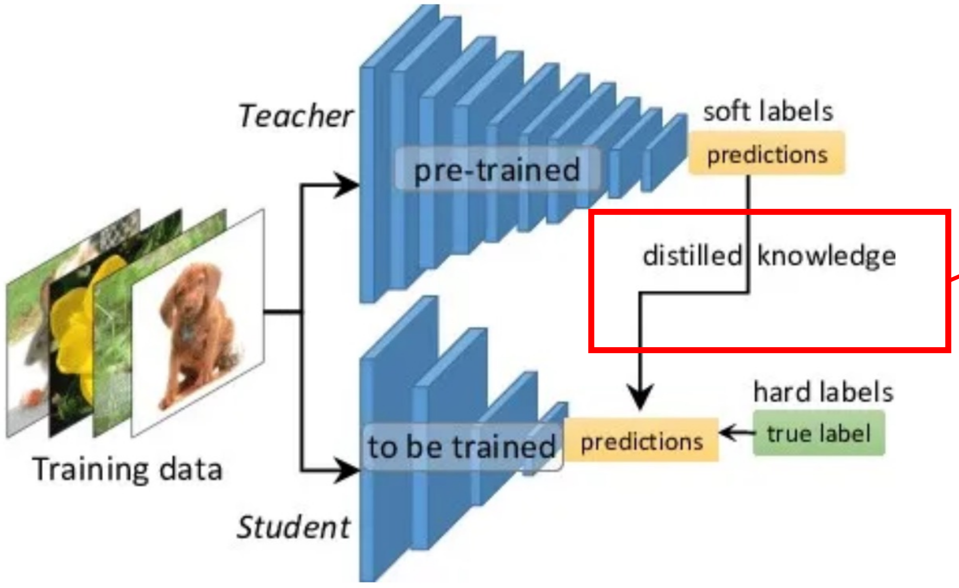
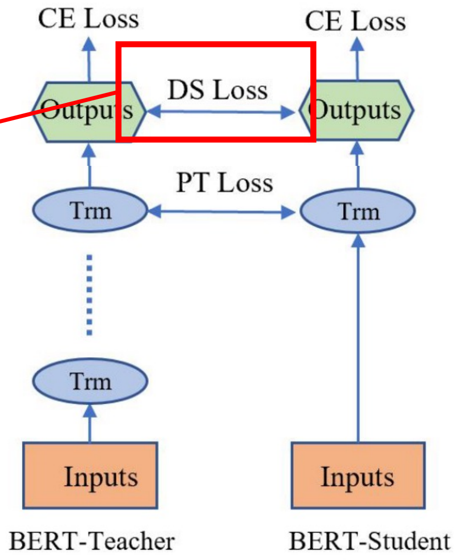


Image Classification[1]

forward KLD

$$\min \mathrm{KL}(p||q_\theta)$$

p: teacher distribution
q_θ: student distribution

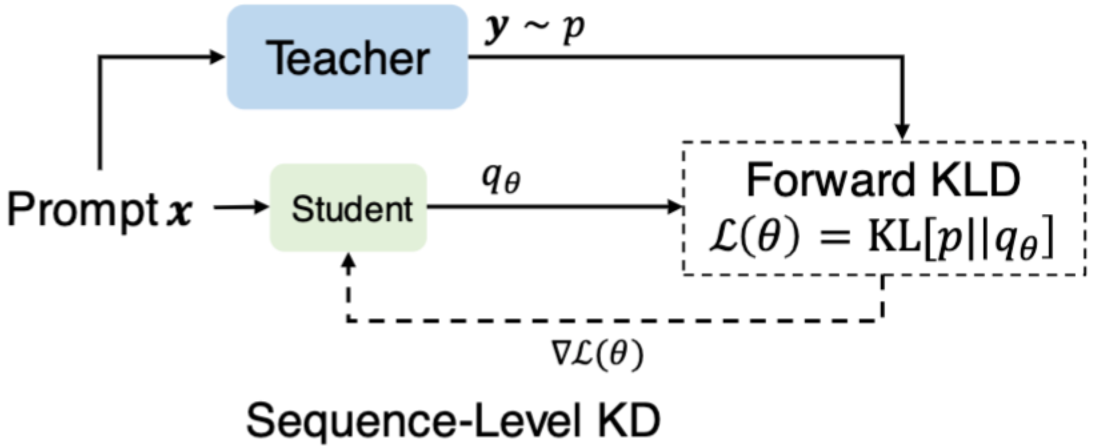


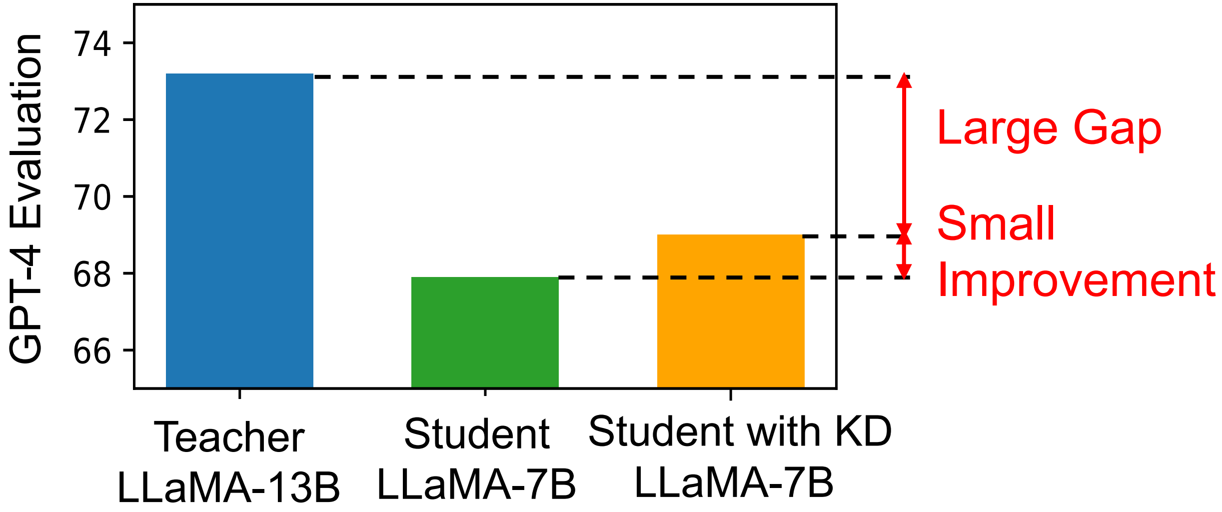
Text Classification[2]

[1] Hinton, et al. Distilling the Knowledge in a Neural Network. 2015. arxiv pre-print.
[2] Wang, et al. Patient knowledge distillation for BERT model compression. 2019. In Proceedings of EMNLP.

Motivation

- But ***forward KL***-based KD does not work well for language generation (the way LLMs perform tasks)





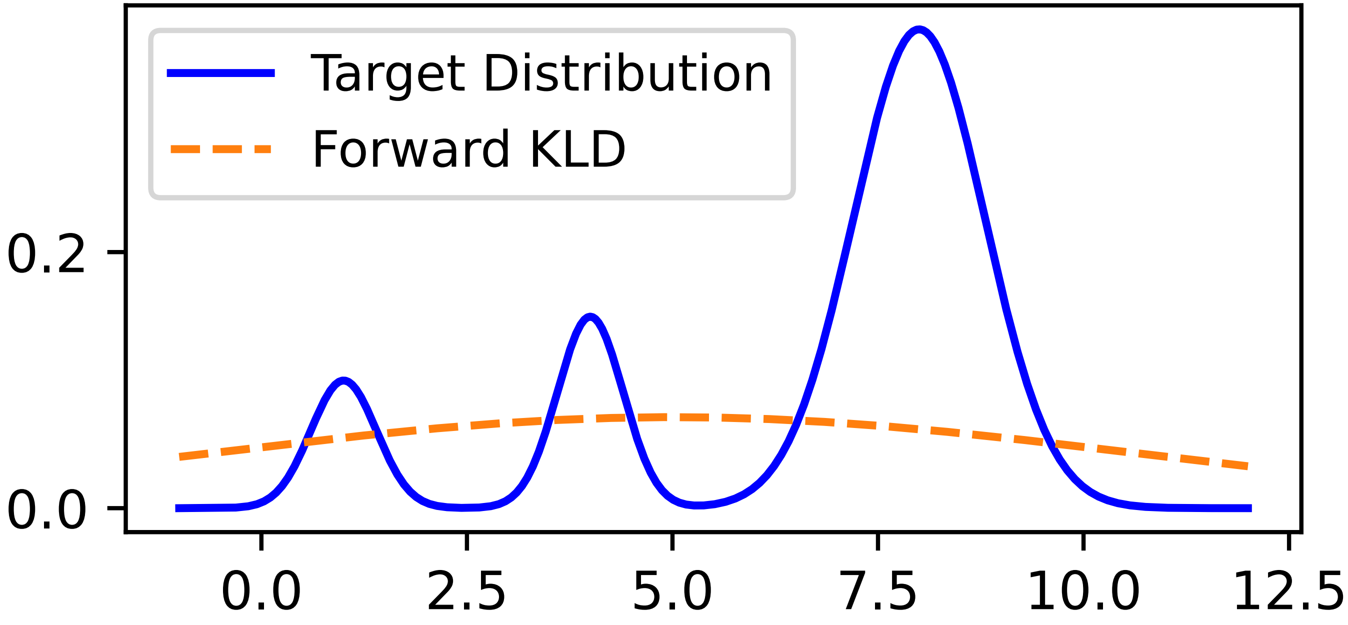
Forward KL-based KD for Language Generation[3]

Forward KL does not work well on LLMs

[3] Kim, et al. Sequence-Level Knowledge Distillation. 2016. In Proceedings of EMNLP.

Problem of Forward KLD

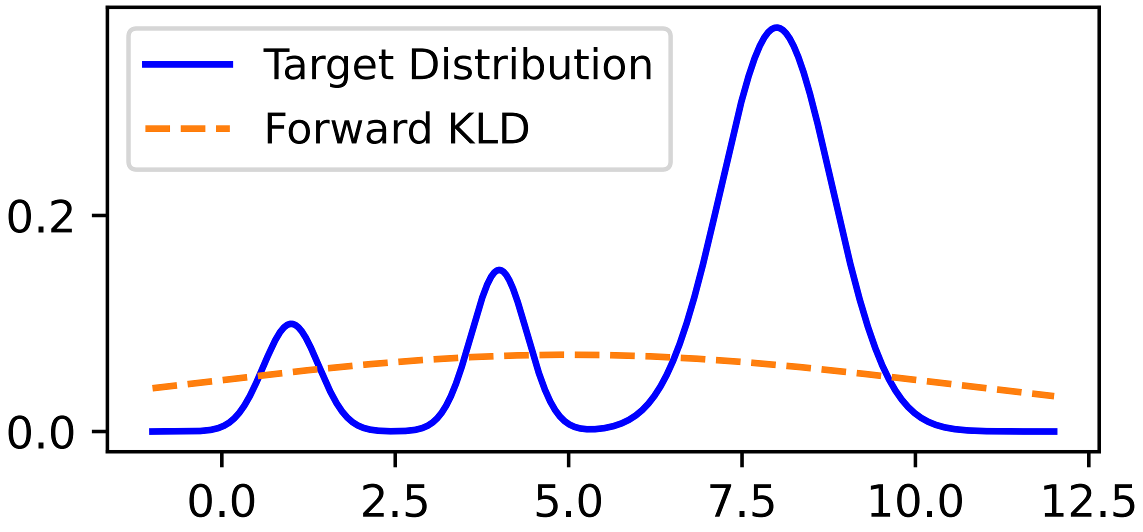
- **Zero-avoiding**: Try to cover all non-zero parts of the target distribution
- **Mean-seeking**: Try to match the mean of the target distribution



Over-estimating void regions!

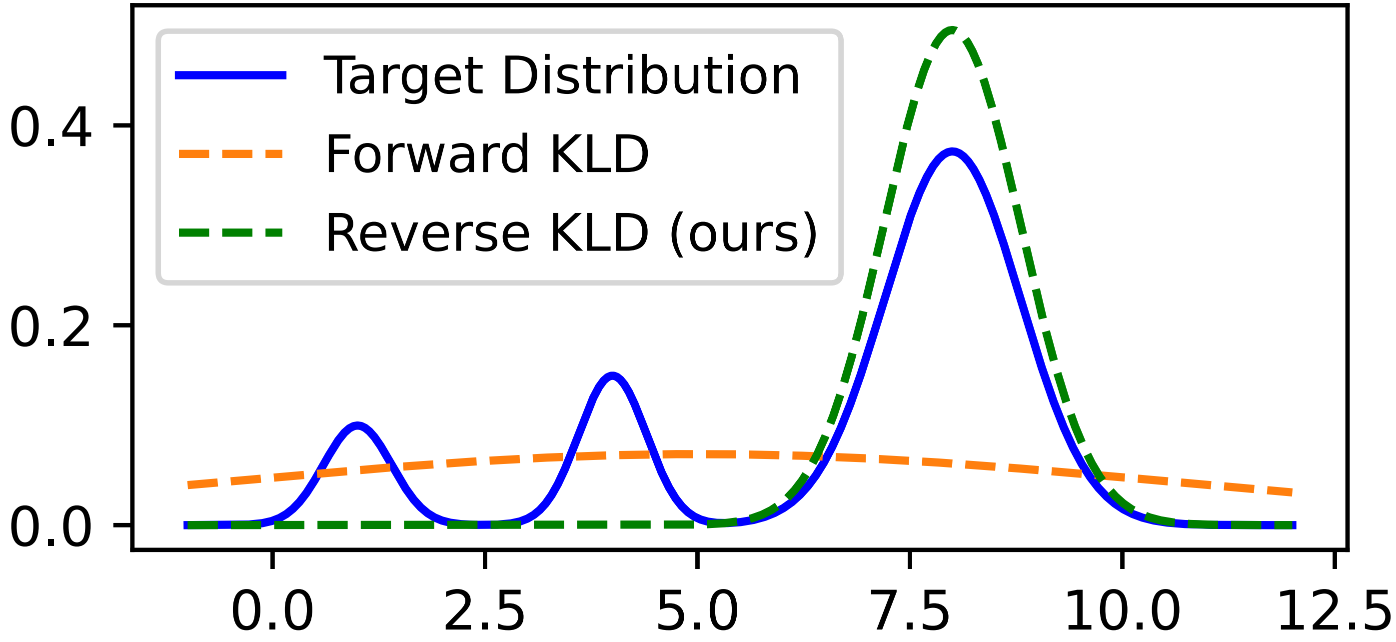
Problem of Forward KLD

- Classification: target distribution has few modes
  - Output space: 1K/10K classes 🤔
- Generation: target distribution has much more modes
  - Output space: **32000^{2048} sequences** 😭



MiniLLM: Knowledge Distillation for LLMs

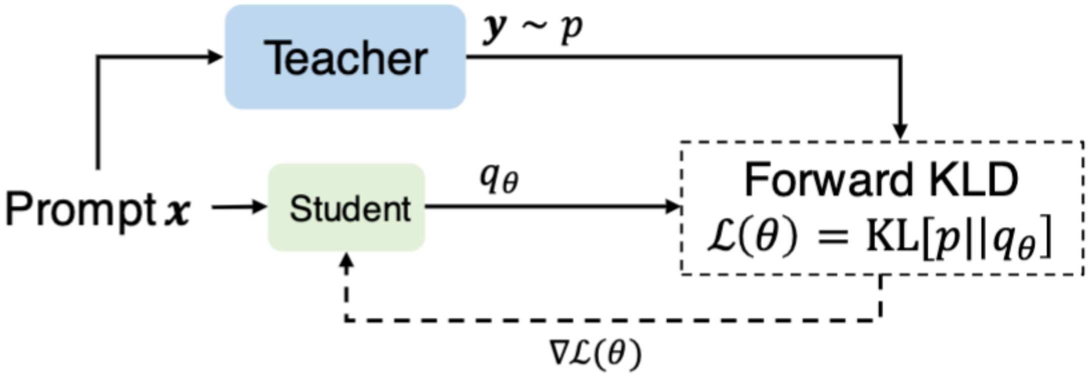
- Forward KLD → Reverse KLD
- Reverse KLD exhibits **mode-seeking** behavior: find the important modes



Gu, et al. MiniLLM: Knowledge Distillation for Large Language Models. 2024. In Proceedings of ICLR.

Method of MiniLLM

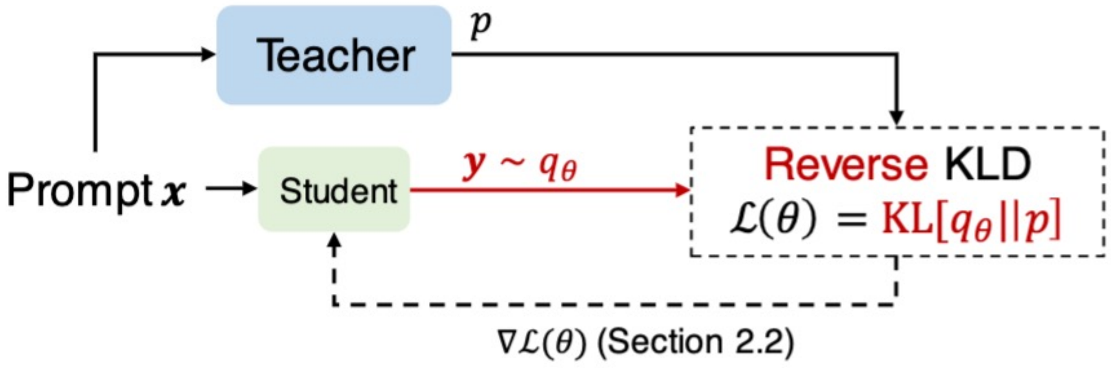
Minimizing Forward KLD



Sequence-Level KD

$$\arg\min_{\theta} \mathrm{KL}[p||q_\theta] = \arg\min_{\theta} \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_{\boldsymbol{x}}, \boldsymbol{y}\sim p'} \log \frac{p(\boldsymbol{y}|\boldsymbol{x})}{q_\theta(\boldsymbol{y}|\boldsymbol{x})}$$

Minimizing Reverse KLD (Ours)



MiniLLM (Ours)

$$\arg\min_{\theta} \mathrm{KL}[q_\theta||p] = \arg\min_{\theta} \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_{\boldsymbol{x}}, \boldsymbol{y}\sim q_\theta} \log \frac{q_\theta(\boldsymbol{y}|\boldsymbol{x})}{p(\boldsymbol{y}|\boldsymbol{x})}$$

MiniLLM and Inverse RL

- Minimizing Reverse KL:
- Inverse RL from Model's Feedback:

$$\arg\min_{\theta} \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_{\boldsymbol{x}},\boldsymbol{y}\sim q_\theta} \log \frac{q_\theta(\boldsymbol{y}|\boldsymbol{x})}{p(\boldsymbol{y}|\boldsymbol{x})} \iff \arg\max_{\theta} \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_{\boldsymbol{x}},\boldsymbol{y}\sim q_\theta} \sum_t r(y_t, y_{<t}) + \mathcal{H}(q_\theta)$$

$$r(y_t, y_{<t}) = \log p(y_t|\boldsymbol{y}_{<t}, \boldsymbol{x})$$

$$\mathcal{H}(q_\theta) = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_{\boldsymbol{x}},\boldsymbol{y}\sim q_\theta} \log q_\theta(\boldsymbol{y}|\boldsymbol{x})$$

*Proof in our paper: The equivalence between MiniLLM (reverse KLD) and **Inverse RL from the teacher model***

Optimization: Gradient Derivation

- Compute the gradient of the objective
- Optimize the sampling model: **Policy Gradient Theorem**

$$\nabla \mathcal{J}(\theta) = -\nabla \mathop{\mathbb{E}}_{\substack{\boldsymbol{x} \sim p_{\boldsymbol{x}} \\ \boldsymbol{y} \sim q_\theta(\cdot|\boldsymbol{x})}} \log \frac{p(\boldsymbol{y}|\boldsymbol{x})}{q_\theta(\boldsymbol{y}|\boldsymbol{x})}$$

$$\Rightarrow \quad \nabla \mathcal{J}(\theta) = - \mathop{\mathbb{E}}_{\boldsymbol{x} \sim p_{\boldsymbol{x}}, \boxed{\boldsymbol{y} \sim q_\theta(\cdot|\boldsymbol{x})}} \sum_{t=1}^{T} (R_t - 1) \nabla \log q_\theta(y_t|\boldsymbol{y}_{<t}, \boldsymbol{x}),$$

where $T = |\boldsymbol{y}|$ and $R_t = \sum_{t'=t}^{T} \log \frac{p(y_{t'}|\boldsymbol{y}_{<t'}, \boldsymbol{x})}{q_\theta(y_{t'}|\boldsymbol{y}_{<t'}, \boldsymbol{x})}$

- Training with PPO (or other RL algorithms)

Optimization: Strategies

- Decompose Single-Step & Long-Range Gradients
  - Pay more attention to the single-step generation quality

$$\nabla \mathcal{J}(\theta) = \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p_{\boldsymbol{x}} \\ \boldsymbol{y}\sim q_\theta(\cdot|\boldsymbol{x})}} \left[-\sum_{t=1}^{T} \nabla \mathop{\mathbb{E}}_{y_t\sim q_\theta(t)} [r_t] \right] + \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p_{\boldsymbol{x}} \\ \boldsymbol{y}\sim q_\theta(\cdot|\boldsymbol{x})}} \left[-\sum_{t=1}^{T} R_{t+1} \nabla \log q_\theta(y_t|\boldsymbol{y}_{<t}, \boldsymbol{x}) \right]$$
$$= (\nabla \mathcal{J})_{\text{Single}} + (\nabla \mathcal{J})_{\text{Long}},$$

- Teacher-mixed Sampling
  - Mix teacher and student distribution when doing sampling

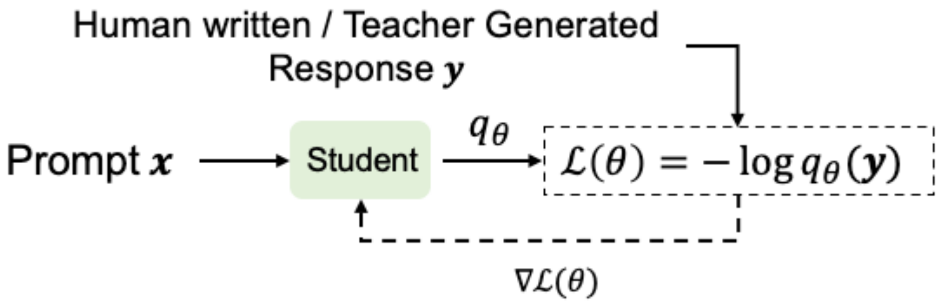
$$\widetilde{p}(y_t|\boldsymbol{y}_{<t}, \boldsymbol{x}) = \alpha \cdot p(y_t|\boldsymbol{y}_{<t}, \boldsymbol{x}) + (1-\alpha) \cdot q_\theta(y_t|\boldsymbol{y}_{<t}, \boldsymbol{x}),$$

- Length Normalization
  - Overcoming the length bias of reverse KL

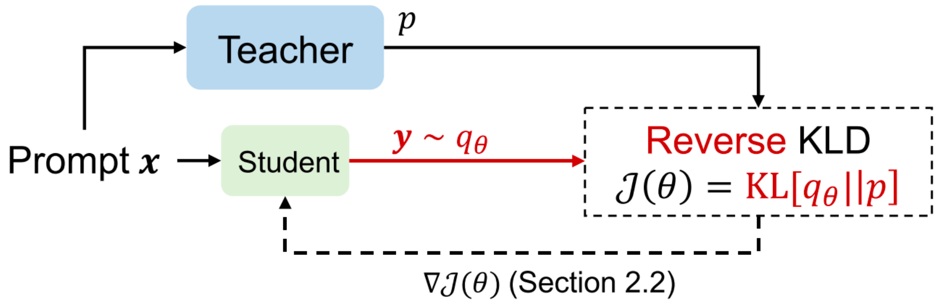
$$R_{t+1}^{\text{Norm}} = \frac{1}{T-t-1} \sum_{t'=t+1}^{T} \log \frac{p(y_{t'}|\boldsymbol{y}_{<t'}, \boldsymbol{x})}{q_\theta(y_{t'}|\boldsymbol{y}_{<t'}, \boldsymbol{x})}$$

How to Train: Much like RLHF

- **Distillation at the Instruction-tuning Stage**
- Step 1: Supervised Find-Tuning or Sequence KD



- Step 2: PPO (no value network, no KL penalty)
  - + 3 Strategies



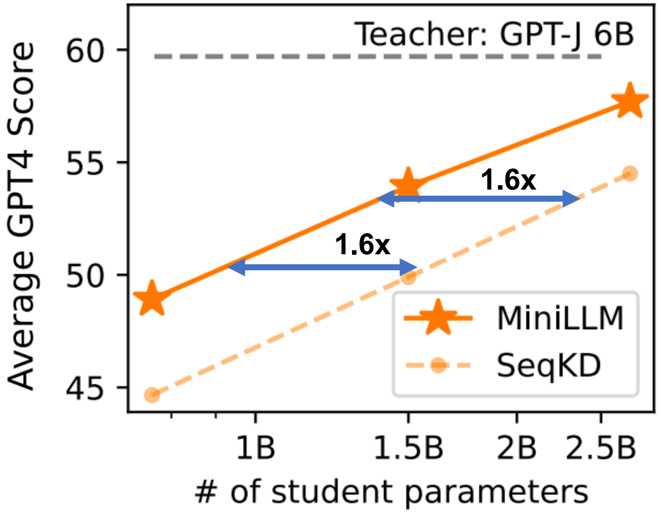
Overall Performance

- Training: Dolly dataset
- Evaluation Data:
  - Dolly dataset
  - Self-Instruct
  - Vicuna-Eval
  - Supernatural Instructions
  - Unnatural Instructions
- Evaluation Metrics
  - Rouge-L
  - GPT-4 Scoring

Model	#Params	Method	DollyEval GPT4	DollyEval R-L	SelfInst GPT4	SelfInst R-L	VicunaEval GPT4	VicunaEval R-L	S-NI R-L	UnNI R-L
OPT	13B	Teacher	70.3	29.2	56.1	18.4	58.0	17.8	30.4	36.1
	1.3B	SFT w/o KD	52.6	26.0	37.7	11.4	40.5	15.6	23.1	28.4
		KD	52.7	25.4	36.0	12.2	40.8	14.9	21.9	27.0
		SeqKD	51.0	26.1	36.6	12.7	42.6	16.6	21.4	28.2
		MiniLLM	**60.7**	**26.7**	**47.0**	**14.8**	**50.6**	**17.9***	**28.6**	**33.4**
	2.7B	SFT w/o KD	55.4	27.1	38.9	13.9	44.8	16.6	24.9	32.3
		KD	60.5	25.9	48.6	13.8	51.3	16.7	26.3	30.2
		SeqKD	57.6	27.5	40.5	13.3	44.5	16.5	25.3	32.3
		MiniLLM	**63.2**	**27.4**	**52.7**	**17.2**	**55.9**	**19.1***	**30.7***	**35.1**
	6.7B	SFT w/o KD	67.9	27.6	56.4	16.4	57.3	17.8	30.3	28.6
		KD	68.6	28.3	58.0	17.0	57.0	17.5	30.7*	26.7
		SeqKD	69.6	28.5	54.0	17.0	57.6	17.9*	30.4	28.2
		MiniLLM	**70.8***	**29.0**	**58.5***	**17.5**	**60.1***	**18.7***	**32.5***	**36.7***
LLaMA	13B	Teacher	79.0	29.7	75.5	23.4	65.1	19.4	35.8	38.5
	7B	SFT w/o KD	73.0	26.3	69.2	20.8	61.6	17.5	32.4	35.8
		KD	73.7	27.4	70.5	20.2	62.7	18.4	33.7	37.9
		SeqKD	73.6	27.5	71.5	20.8	62.6	18.1	33.7	37.6
		MiniLLM	**76.4**	**29.0**	**73.1**	**23.2**	**64.1**	**20.7***	**35.5**	**40.2***

Scaling Results of MiniLLM

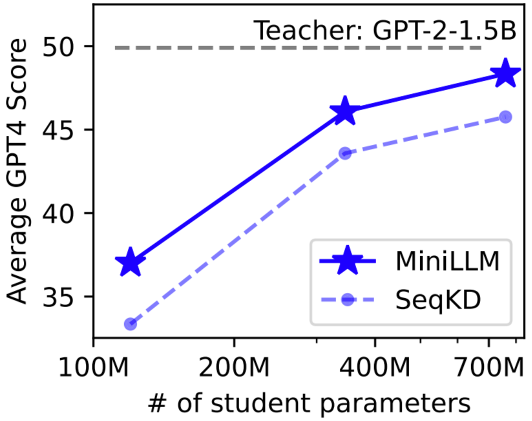
- Improves the coefficients in the scaling law (*Qualitatively*)
  - The model compression ratio preserves with different model sizes



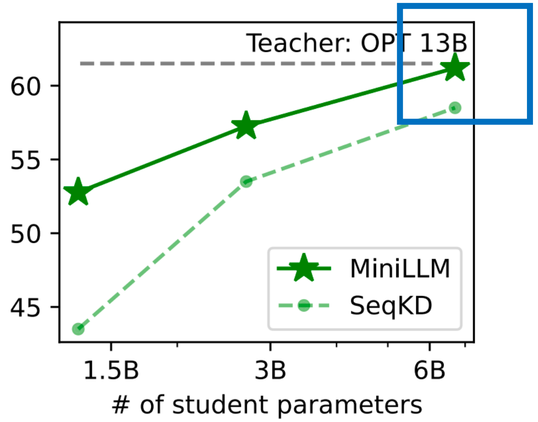
Improved (*Qualitatively*)

$$L(N,S) = \left(\frac{\boxed{N_c}}{N}\right)^{\alpha_N} + \left(\frac{S_c}{S_{\min}(S)}\right)^{\alpha_S}$$

- Other sizes / models
  - Not perfectly follows the law, but still scales well

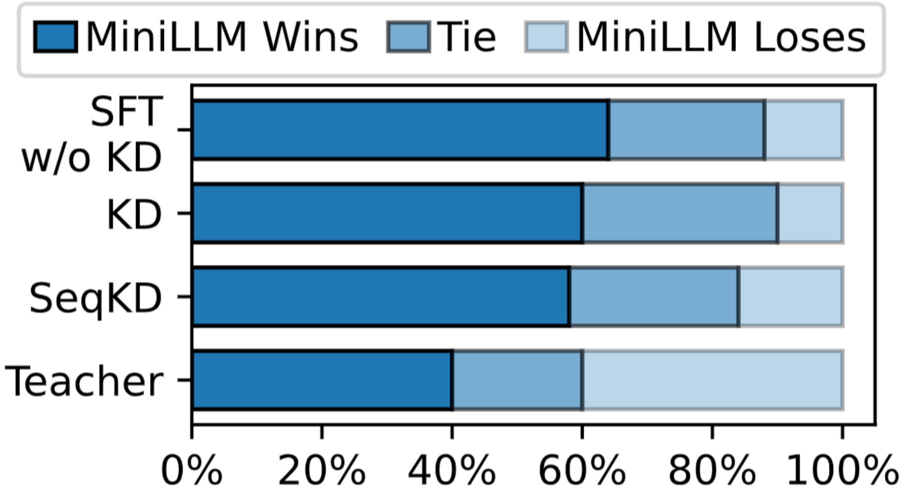


2.0x



Other Results of MiniLLM

- Benefits brought by learning from teacher model



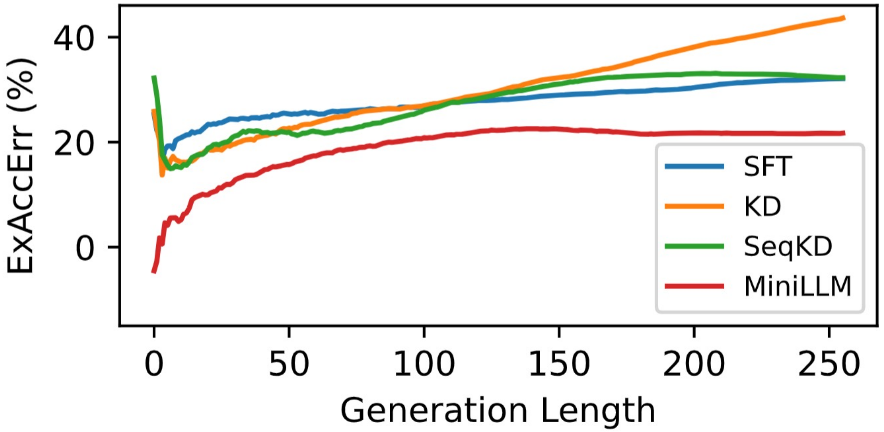
Better Human Preference

	SST2		BoolQ	
	ECE	Acc.	ECE	Acc.
Teacher	0.025	93.0	0.356	74.5
KD	0.191	84.7	0.682	63.5
SeqKD	0.243	66.5	0.681	62.8
MiniLLM	**0.099**	**89.7**	**0.502**	**67.8**

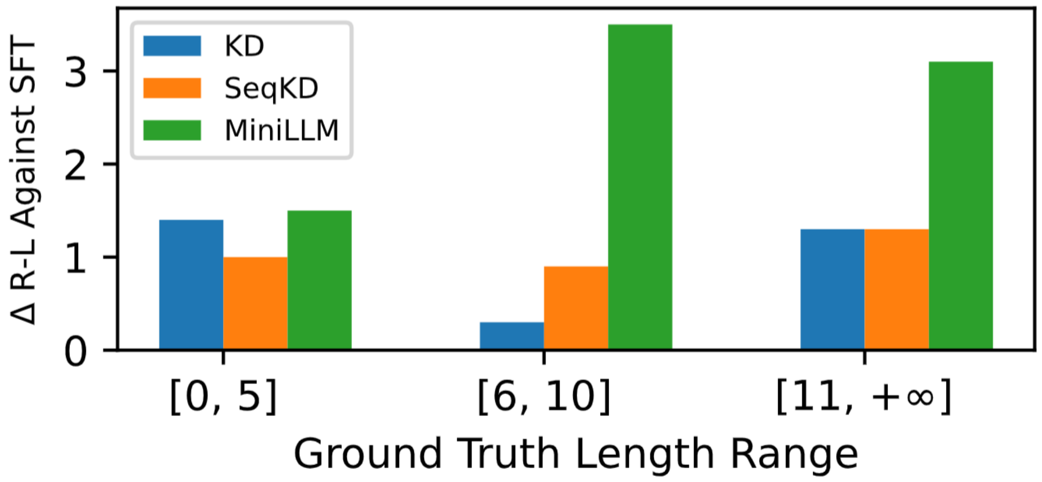
Better Model Calibration

Other Results of MiniLLM

- Benefits brought by policy optimization training



Lower Exposure Bias



Better Long-Text Generation Performance

Without PPO?

- There may be some implementation/stability issues of PPO

- Can we optimize the objective without PPO?

$$\arg\min_{\theta} \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_{\boldsymbol{x}},\boldsymbol{y}\sim q_\theta} \log \frac{q_\theta(\boldsymbol{y}|\boldsymbol{x})}{p(\boldsymbol{y}|\boldsymbol{x})}$$

  - Alternative #1: Single-step gradient
  - Alternative #2: Ranking loss

Alternative #1: Single-Step Gradient

- Introducing approximation of the gradient
- Only consider the **Single-Step Gradient**

$$
\begin{aligned}
\nabla \mathcal{J}(\theta) &= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p_{\boldsymbol{x}} \\ \boldsymbol{y}\sim q_\theta(\cdot|\boldsymbol{x})}} \left[-\sum_{t=1}^{T} \nabla \mathop{\mathbb{E}}_{y_t \sim q_\theta(t)} [r_t] \right] \\
&= (\nabla \mathcal{J})_{\text{Single}} \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p_{\boldsymbol{x}} \\ \boldsymbol{y}\sim q_\theta}} \sum_{t=1}^{T} \nabla_\theta \operatorname{KL} \left[q_\theta(\cdot|\boldsymbol{y}_{<t}, \boldsymbol{x}) || p(\cdot|\boldsymbol{y}_{<t}, \boldsymbol{x}) \right]
\end{aligned}
$$

(word-level reverse KL)

- Similar approximation can also be found in concurrent works[4][5]

[4] GKD: Generalized Knowledge Distillation for Auto-regressive Sequence Models. 2024. In Proceedings of ICLR.
[5] f-Divergence Minimization for Sequence-Level Knowledge Distillation. 2023. In proceedings of *ACL.*

Alternative #2: Ranking Loss

- Inspired by RLHF
- Replace RL with **Ranking**:
  - Step 1: Sample various responses from the student
  - Step 2: Rank the responses based on the teacher probability $p(\boldsymbol{y}|x)$
  - Step 3: Optimize with the ranking (margin) loss:

$$\mathcal{J}(\theta) = \max(0, \delta - \log q_\theta(\boldsymbol{y}^+|\boldsymbol{x}) + \log q_\theta(\boldsymbol{y}^-|\boldsymbol{x})) - \lambda \log q_\theta(\boldsymbol{y}^*|\boldsymbol{x})$$

- Useful to stabilize training in RLHF works[1]

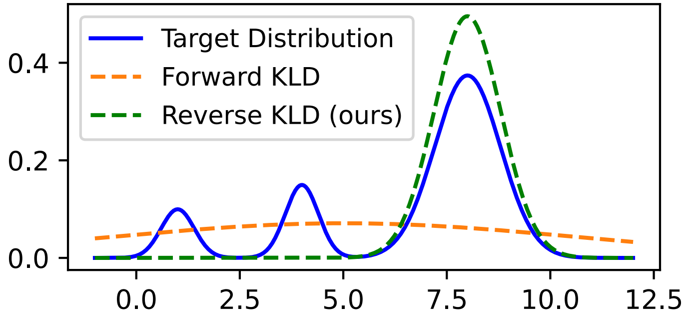
[6] SLiC-HF: Sequence Likelihood Calibration with Human Feedback. 2023. *arxiv pre-print.*

Effect of Two Alternatives

- No teacher mixed-in
- No length normalization
- There are some performance drop. But still outperforms baselines.

	DollyEval		SelfInst		VicunaEval	
	GPT4	R-L	GPT4	R-L	GPT4	R-L
SFT w/o KD	38.6	23.3	26.3	10.3	30.4	14.7
SeqKD	41.2	22.7	26.2	10.1	31.0	14.3
MiniLLM	**44.7**	**24.6**	**29.2**	**13.2**	**34.1**	**16.9**
MiniLLM (Single-Step)	<u>43.9</u>	<u>24.0</u>	<u>28.3</u>	<u>12.5</u>	<u>33.1</u>	<u>16.3</u>
MiniLLM (Ranking)	41.3	23.0	28.6	10.9	32.5	14.9

Will MiniLLM Lose Diversity?



- Definition of "Diversity"
  - Knowledge Coverage
    - The long-tail knowledge in LLMs, measured by PPL
  - Linguistic Complexity
    - The diversity of word use in a single sentence, measured by Distinct-4-Grams
  - One to more Generation
    - Given one prompt, how many different responses a model can generate

Will MiniLLM Lose Diversity?

- Experiments on "Diversity"
  - Does not lose much knowledge coverage and linguistic complexity 😄

	DollyEval		SelfInst	
	Dist-4	Loss	Dist-4	Loss
Teacher	99.3	3.55	99.1	4.44
SFT	99.5	3.89	99.0	5.28
MINILLM	99.0	3.95	98.6	5.33

  - Tend to generate similar responses given one prompt 🤔
    - (may not be bad in practice)
- Reverse KL ignores modes in $p(y|x)$, not $p(x,y)$

The ability to generate multiple responses given one prompt

The ability to model knowledge/complex sentences

More Than Reverse KLD?

- J-S Divergence [4]

$$J_{\text{JS}} = \frac{1}{2} \mathop{\mathbb{E}}_{\mathbf{Y} \sim p} \left[\log \frac{p(\mathbf{Y})}{m(\mathbf{Y})} \right] + \frac{1}{2} \mathop{\mathbb{E}}_{\boxed{\mathbf{Y}' \sim q_\theta}} \left[\log \frac{q_\theta(\mathbf{Y}')}{m(\mathbf{Y}')} \right]$$

- Total Variational Distance (TVD)[5]

$$J_{\text{TVD}} = \frac{1}{2} \sum_{\boxed{\mathbf{Y} \sim q_\theta}} |q_\theta(\mathbf{Y}) - p(\mathbf{Y})|$$

The Lesson we Learn:

Students should learn from their mistakes, not just imitate the teacher!

Behavior Cloning ⟶ (Inverse) Reinforcement Learning

[4] Wen et al. f-Divergence Minimization for Sequence-Level Knowledge Distillation. 2023. In Proceedings of ACL.
[5] Agarwal et al. GKD: Generalized Knowledge Distillation for Auto-regressive Sequence Models. 2024. In Proceedings of ICLR.

Summary

- MiniLLM: Knowledge Distillation of Large Language Models
- Method:
  - Minimizing Reverse KL Divergence
  - Optimized by PPO (like RLHF)
- Results
  - 1.6x - 2.0x model compression
  - Consistent improvement across model families/sizes
- Takeaway/Insights from MiniLLM:
  - **Students should learn from their mistakes, not just imitate the teacher**

Thanks for Your Attention！

Paper Link: https://arxiv.org/abs/2306.08543

Code Link: https://github.com/microsoft/LMOps/tree/main/minillm