Sparse Logit Sampling: Accelerating Knowledge Distillation in LLMs

Paper & code!

Anshumann*, Mohd Abbas Zaidi*, Akhil Kedia*, Jinwoo Ahn, Taehwak Kwon, Kangwook Lee, Haejun Lee, Joohyung Lee

anshu.mann, abbas.zaidi, akhil.kedia, jinwoo.ahn, taehwak.kwon@samsung.com

Samsung Research

1. Contributions

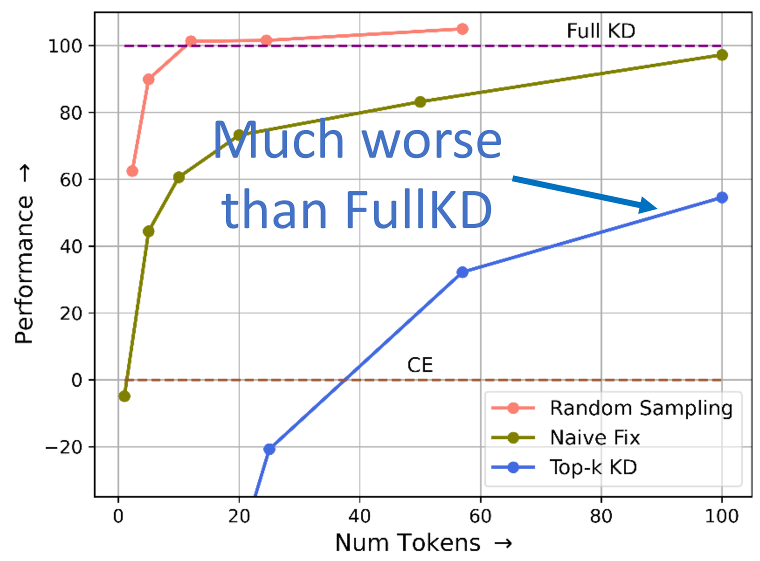
1. Run teacher once for KD for pre-training/SFT, pre-compute/**store teacher soft-labels**
2. Unlike top-K methods, provides **unbiased estimate** of teacher probs
3. Preserves KL Divergence **gradients** in expectation and empirically

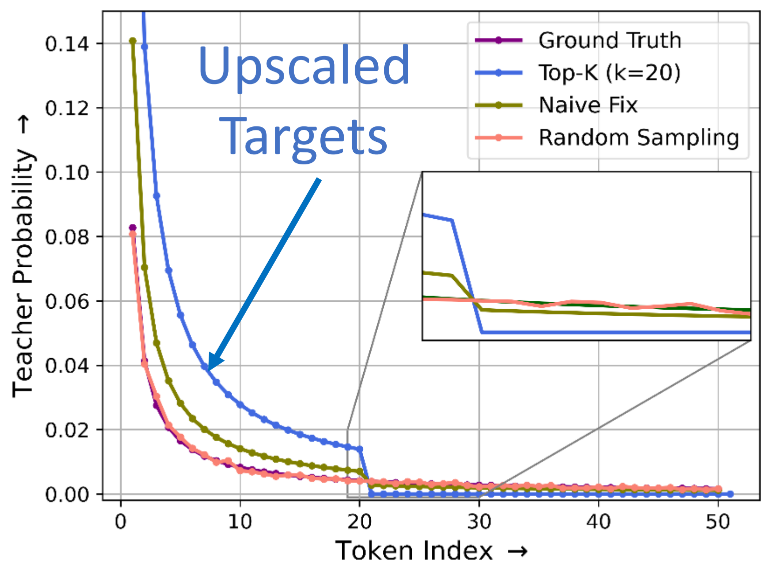
3. Vanilla Top-K Distillation

1. Store only Top-K largest probs, eg. Top-100
2. Using KLD loss, $L = \sum_{i=1}^{|V|} t_i \log \frac{t_i}{p_i}$,

   student learns **upscaled targets** - $p_i = \frac{t_i}{\sum t_i}$
3. Missing any supervision in the tail
4. **Worse than no KD for Top-25!**

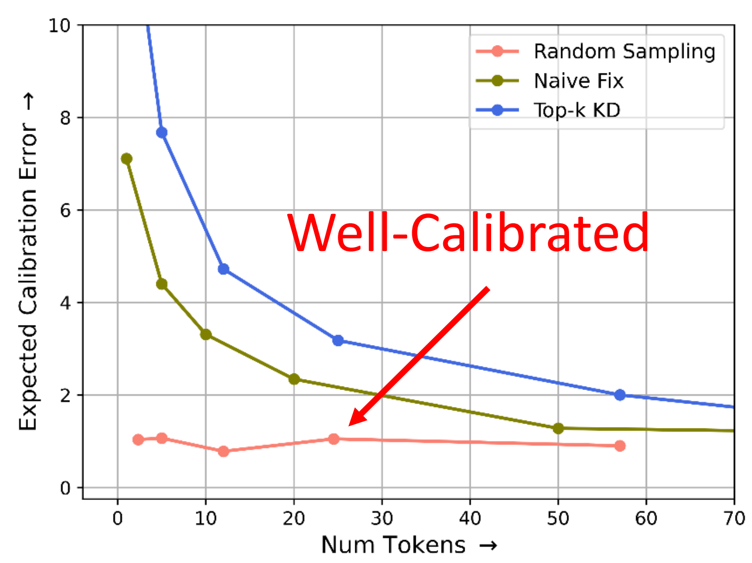
   60% of performance of Full-KD at Top-100





5. Random Sampling KD Analysis

1. Saved labels very sparse – 12 for $N = 50$
2. Only ≈ 36GB of storage for 1B training tokens.
3. $2 - 3$x compute savings of teacher
4. **Well-calibrated** student!
5. Mini-batch **gradient perfectly matches** FullKD



Method	Δ Angle ↓	Norm Ratio
Top-K 12	58°	2.4
Top-K 50	48°	1.8
Top-K 300	30°	1.3
Ours 12	4°	1.0

No Grad Error

7. Future Work

1. Larger Scale, Continual Pre-training, SFT/IF
2. Better sampling (without replacement?) Optimal sampling depends on student perf..
3. Cross-tokenizer Offline KD – Vocab mismatch?
4. Offline KD of Hidden States – Invert LM Head and Softmax?

https://github.com/akhilkedia/RandomSamplingKD

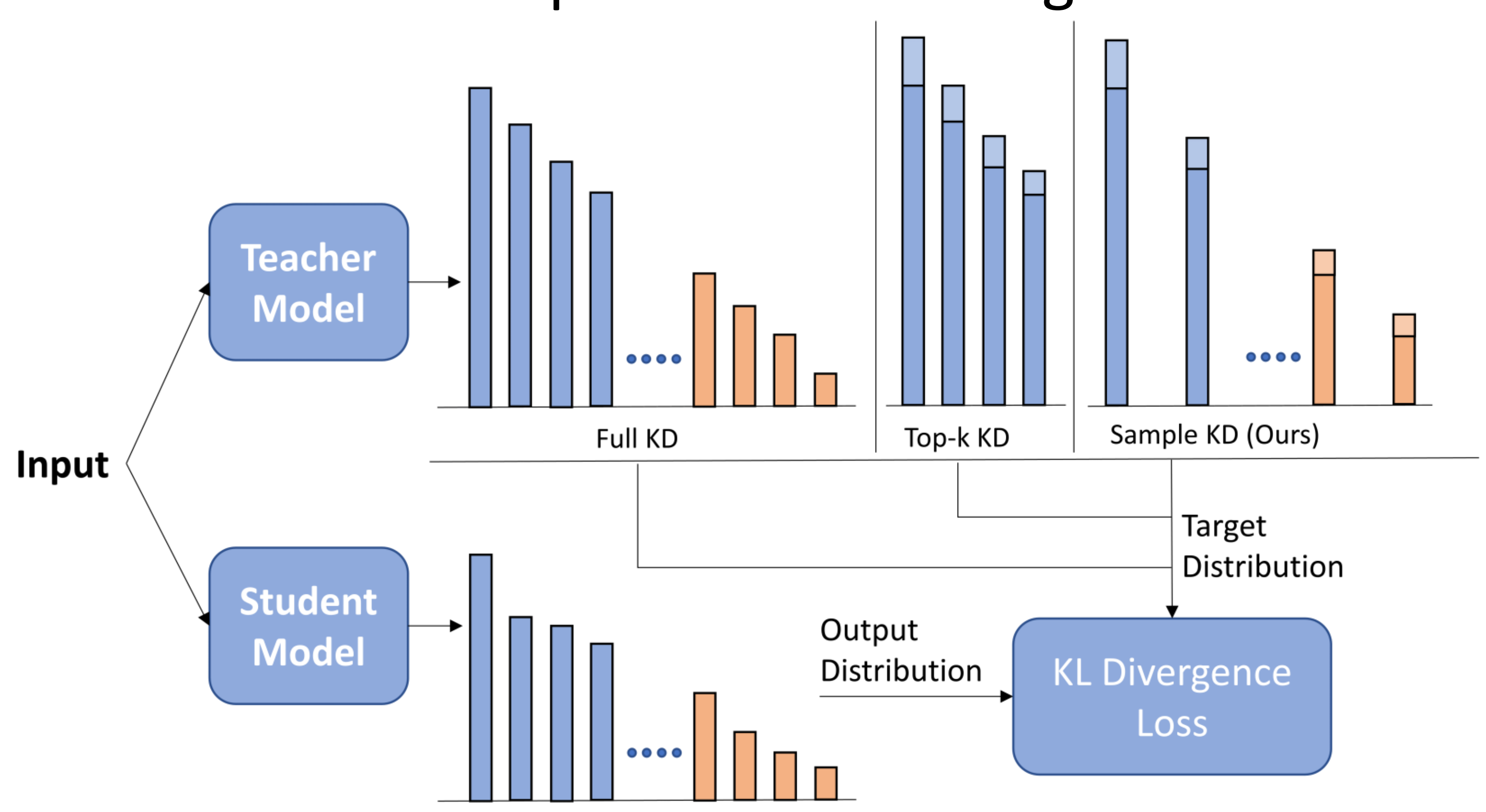
2. Advantages

1. Store only **12 soft-labels/token,** $< 10\%$ compute overhead
2. Comparable performance to using Full KD
3. The student in **well-calibrated**, improves 0-shot, IF scores, speculative decoding
4. 300M to 3B student, for 10B to 1T train tokens

4. Proposed: Random Sampling KD

Don't truncate to top-K, sample instead!

1. **Sample tokens** (with replacement) from teacher probability dist. for N rounds
2. For each token i, save **sampled freqs** $\frac{\text{count}_i}{N}$
3. Use saved freqs as soft label targets for KLD.



6. Results

Unique Tokens	LM Loss ↓	ECE % ↓	Speculative Accept % ↑	0-shot Score ↑
CE	2.81	0.4	59.95	40.4
2.4	2.77	1.0	61.47	42.1
5.0	2.75	1.1	61.83	42.6
12.1	2.75	0.8	61.85	43.0
24.5	2.75	1.1	61.93	**43.1**
57.0	**2.74**	0.9	61.97	42.9
FullKD	2.75	**0.7**	**62.02**	42.1

≈ **12 labels sufficient**

3B → 300M

Method	LM Loss ↓	ECE % ↓	Speculative Accept % ↑	0-shot Score ↑	IF SFT Score ↑
CE	2.37	0.3	71.1	55.6	54.5
Top-K 12	2.50	4.7	73.0	56.6	57.7
Top-K 50	2.40	1.8	73.1	57.1	58.3
Ours (12)	2.35	**0.2**	73.2	57.5	**59.4**
FullKD	2.34	**0.2**	73.4	57.5	58.4

8B → 3B

Dataset	CE	Top-K 12	Top-K 50	Ours 12	FullKD
Dolly	64.2	59.0	65.4	**71.3**	66.1
SelfInst	64.6	60.9	63.4	**73.1**	66.1
Vicuna	49.1	48.9	53.1	**58.2**	56.9
S-NI	62.4	63.4	62.6	**63.8**	60.7
UnNI	60.4	58.0	58.3	**61.4**	61.0
Avg	60.2	58.0	60.6	**65.6**	62.2

Better than FullKD for NLG

Sampling noise =Regularization?