One epoch is all you need

2021.08.11

주세준

- 현재 generative models 에선 scaling-up 방식으로 성능을 향상 시키고 있다
- 그러나 그 비용이 천문학적이기 때문에 현실적인 해결 방식이라고 하기는 그렇다

- Multi-epoch training은 데이터가 희귀한 supervised learning 에는 적합하지만 오히려 데이터가 풍부한 unsupervised-learning에는 부적합하다

- 위와 같이 모델별로 사용하는 epoch 의 횟수도 다르며 밝히지 않는 경우도 있다
- 따라서 모든 모델에게 공평하게 적용될 수 있는 one-epoch -standard dataset

Table 1: The number of epochs used for the training.

Model	Epochs
GPT (Radford et al., 2018)	100
SPN (Menick & Kalchbrenner, 2018)	Not reported
BERT (Devlin et al., 2018)	40
Mesh Transformer (Shazeer et al., 2018)	10
Transformer-XL (Dai et al., 2019)	Not reported
GPT-2 (Radford et al., 2019)	Not reported (20 or 100)
Sparse Transformer (Child et al., 2019)	70 - 120

TO SEE PERFORMANCE IMPROVEMENT OVER CONVENTIONAL SETTING

- 1. The dataset size is increased (e.g. by sampling from Internet a la WebText), so that, while training for the same number of iterations as before, the same sample is never reused.
- 2. Any regularization method is eliminated.
- 3. We set P and T according to some heuristics. For example, we can perform this by setting the ratio T/P as close to 5 as possible while keeping their product constant, or equivalently by solving the following:

$$\underset{P,T}{\arg\min} \, |\log(5) - \log(T/P)| \qquad \text{subject to} \qquad PT = P_0T_0.$$

- P is the number of parameters
- T=cI (c token number per mini batch,
  I number of iteration)

Justification

- Greater data size implies to greater diversity

both improves performance (Hestness et al., 2017; Radford et al., 2019).

Overfitting does not occur in an one-epoch setting
(sampling discrepancy is the the primary cause of overfitting)

Setup

- Base Tansformer decoder, LM1B, 65,000 iteration
- ”training with one epoch training” by S
- ”training for multiple epochs” by M
- ”using dropout” by D.
- “single epoch training and p = 0:1” by SD

One epoch training

- Speed up = M/MD 인 경우 best loss 도달 iteration 대비 single epoch 의 경우
- When dropout is used, the curve is shifted upward.

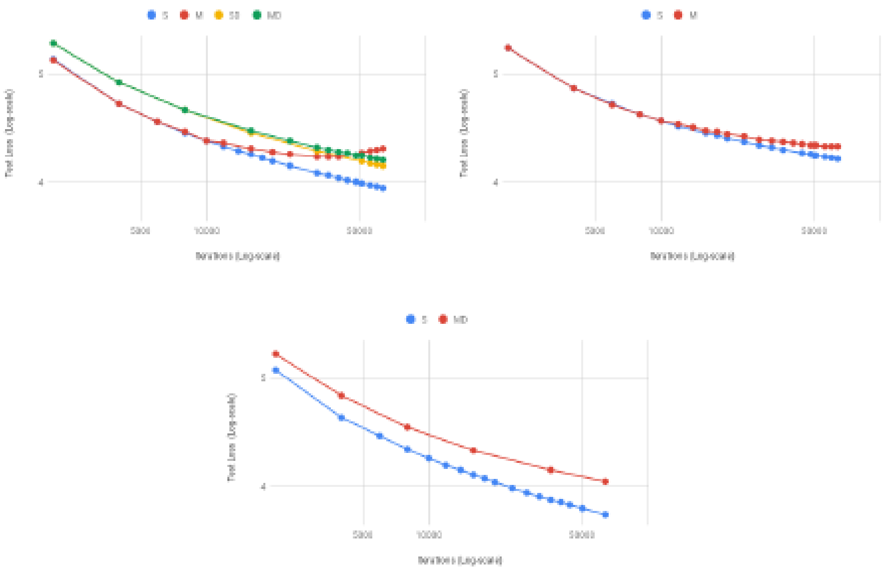


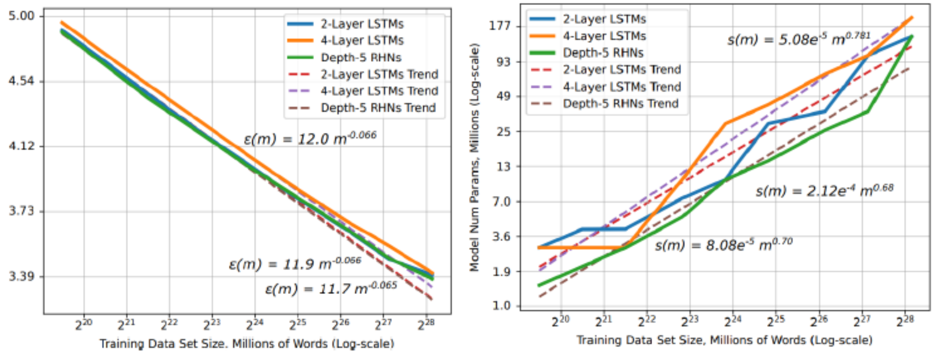
Figure 1: Learning curve of LM for 65,000 iterations on subsets of LM1B with different configurations.

	d	Parameters	Epochs	Iters./Epoch	Speedup ($E = 10$)	Speedup ($E = 5$)
Left	512	45M	10	6500	3.3	1.8
Right	256	18M	10	6500	1.9	1.5
Bottom	1024	128M	10	6500	3.3	2.6

Table 2: Configuration of each figure of Fig. 1.

Power law

- the curve enters a linear region, which is, in fact, a power-law region, since the plot is log-log
- analyzing the training becomes simpler



igure 3: Learning curve of LM on subsets of LM1B with varying size (cited from Hestness et al. 2017)).

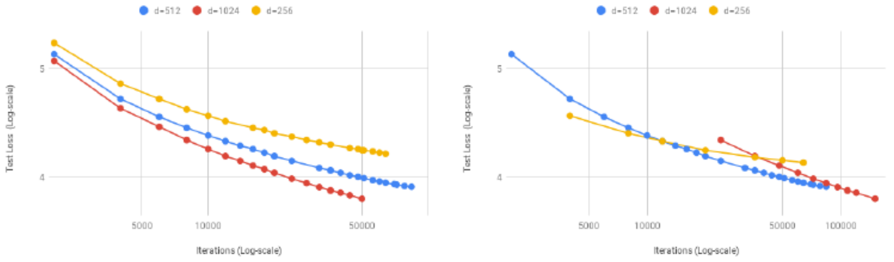
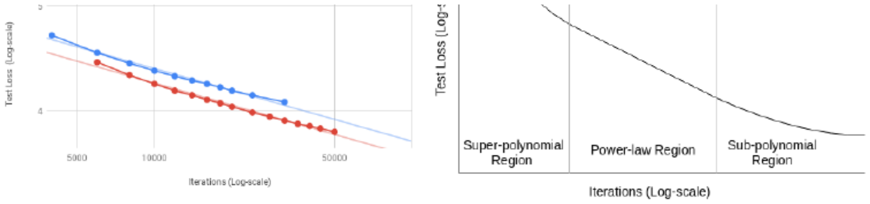


Figure 2: (Left) Log-log plot of learning curve over iterations. (Right) Log-log plot of the learning curve scaled according to the per-iteration FLOPS with respect to the $d = 512$ curve, which is fixed at its original position as with the scaling of the x-axis.

SIZE/ITERATION ADJUSTMENT

- FLOPS = Floating point operations per second (계산 효율)
- FLOPs = Floating point operations (실제 계산량)
- total FLOPS of the training is proportional to PI

- This result suggests that five words per parameter can be the most efficiently compressed, at least in our setting.
- [1.8,11.5] geometric mean = 5



igure 4: (Left) Log-log plot of partial learning curve of LM over iterations with a line fit. (Right) etch of learning curve over iterations.

d	Parameters	Optimal Iters.	(Optimal Tokens)/Params.
256	18M	[0, 30000]	[0, 11.5]
512	45M	[12000, 84000]	[1.8, 12.9]
1024	128M	[28000, ∞)	[1.5, ∞)

Table 3: Optimal number of iterations and ratio.

시간이 부족했나봐요

- STATE-OF-THE-ART MODELS ARE LIKELY TO UNDERGO BETTER SPEEDUP
- RANGE OF APPLICABILITY
- EFFICIENTNET SCALING WITH THE NUMBER OF ITERATIONS
- CAVEATS ON FINE-TUNING
- SAMPLE EFFICIENCY OF LEFT-TO-RIGHT LANGUAGE MODEL AND BERT
- SHIFT OF ATTENTION FROM REGULARIZATION TO MODEL CAPACITY
- CREATION OF NEW DATASETS AND COMPARISON OF MODELS
- DATA AUGMENTION WITH INTERNET
- ON SAMPLING DATA FROM INTERNET

그나마...

- 인터넷 크롤링의 기준 (인용수, meta data)
- BERT 와 LtoR LM의 성능 차이 bert 매번 다른 마스킹, one epoch면 그 효과 사라짐 left to right model 이점?? Text generatio에 대해선 LtoR이 낫다
- Regularization을 하지 않는 것이 fine tuning 시 overfitting을 야기할 수 있다 -> GPT-2 도 그렇게 많이 안한다

- Speed up 해도 성능이 안 나오면 그만 아닌가…