Rethinking Pruning Large Language Models:
Benefits and Pitfalls of Reconstruction Error Minimization

Sungbin Shin[1], Wonpyo Park[2], Jaeho Lee[1,2] and Namhoon Lee[1,2]

[1]POSTECH, [2]Google

- Pruning LLMs by minimizing reconstruction errors should be done carefully because otherwise it could easily overfit the calibration data.
- Leveraging the generative nature of LLMs to create its own calibration data can mitigate this issue and improve generalization of pruned LLMs.

Background

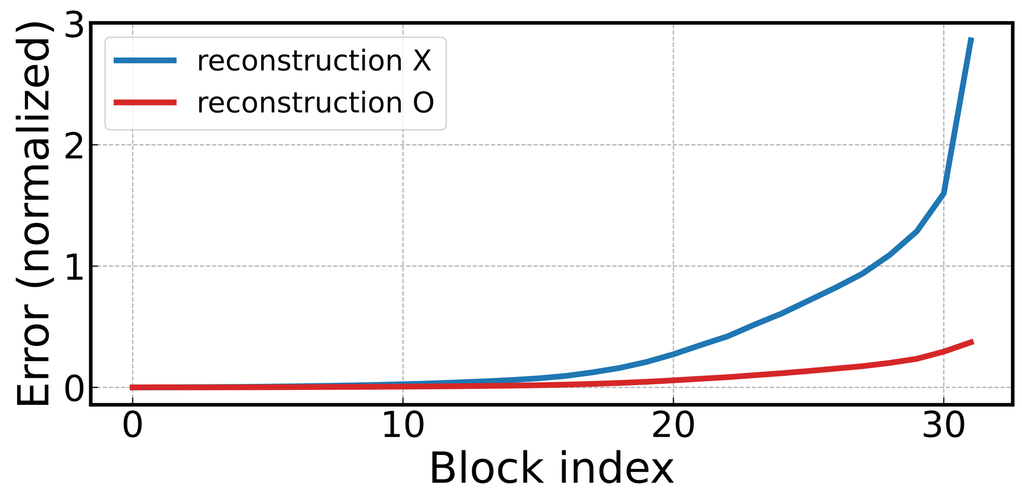
- Pruning has the potential to reduce the computational requirements of LLMs, yet the standard approaches are not feasible as they require an extensive training process as well as training data.
- Consequently, pruning LLMs is done post training, by finding a sparse mask (and updating remaining weights) such that it can *reconstruct* the original dense pre-trained model, as follows:

$$\min_{w,m} \|f(\bar{w};\mathcal{D}) - f(m \odot w;\mathcal{D})\|_2^2 \quad \text{s.t.} \quad \|m\|_0 \le k\,, \tag{1}$$

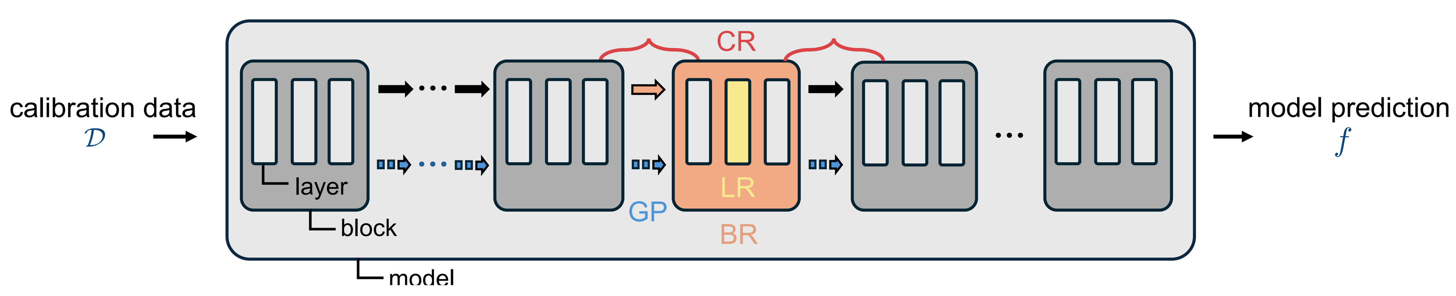
  i.e., given a pre-trained model $\bar{w}$, the goal is to find a pruning mask m such that the resulting sparse model $m \odot w$ reconstructs the predictions of the original dense model $f(\bar{w};\cdot)$ on some caliration data $\mathcal{D}$.
- However, this *reconstruction error minimization* process (1) still requires a lot of memory, and thus, existing works take a "divide-and-conquer" approach: *i.e.*, split model into smaller submodels, prune each submodel individually, and simply put all resulting sparse submodels together.

Reconstruction techniques

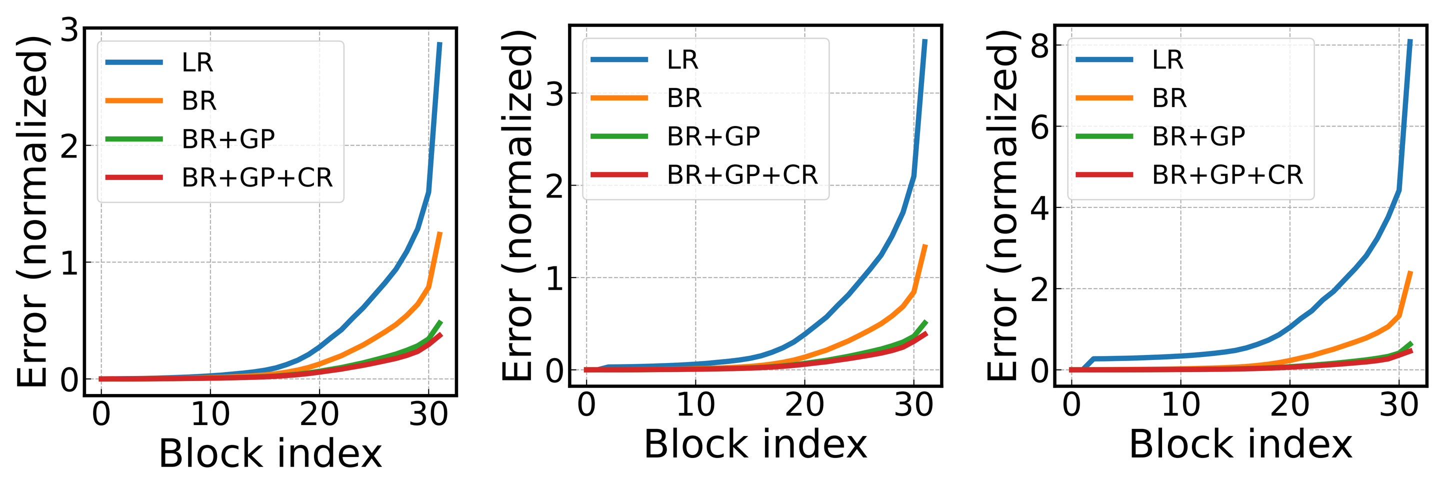
- We first show that the "divide-and-conquer" approaches create critically high compounding errors, and subsequently, that various engineering techniques can reduce this error quite significantly as seen in the following plots:



- Specifically, we apply the following three techniques:



  (i) block-wise reconstruction (BR) to extend the unit of optimization target from a layer to a block of layers; (ii) global propagation (GP) to use "global propagation" from the original dense model as input for the target reconstruction; (iii) cross-block reconstruction (CR) to stitch between blocks.
- More results of reconstruction techniques on LLaMA-7B:



(a) SparseGPT (b) Wanda (c) Magnitude

Overfitting calibration data

- While one can reduce the reconstruction error, it turns out that this does not necessarily mean a "better" pruning result.
- More specifically, we find that reducing the reconstruction error often leads to overfitting the calibration data:

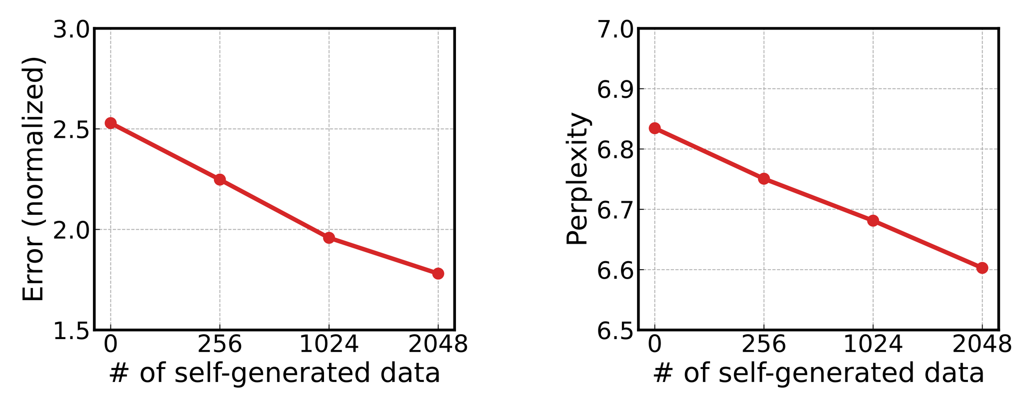
Reconstruction	Error (normalized)	Perplexity	Zero-shot accuracy
LR	3.56	9.77	54.24
BR	1.33	9.02	55.14
BR+GP	0.51	**8.83**	**56.22**
BR+GP+CR	**0.38**	9.18	54.65

CR	Error (normalized) Calibration	Test
X	0.51	**2.23**
O	**0.38**	2.48

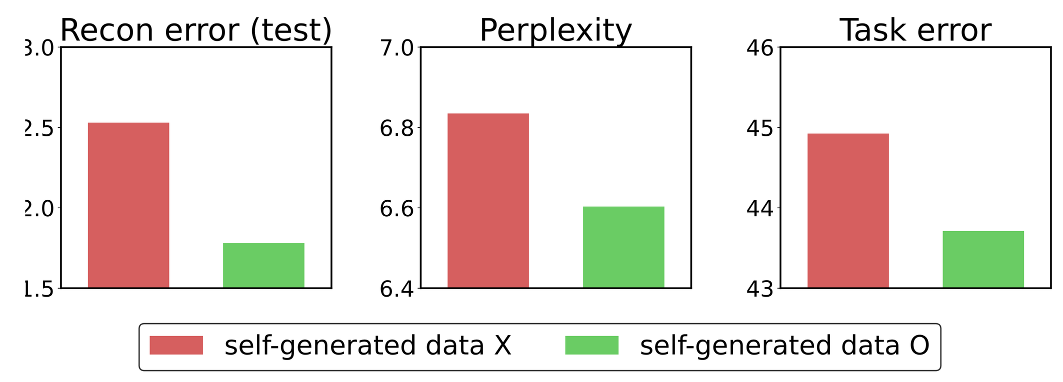
  where it is seen that a method with a lower reconstruction error does not necessarily yield a lower perplexity or higher zero-shot accuracy.
- We note that this phenomenon seems to be more pronounced in larger models.

Leveraging self-generated calibration data to improve generalization

- We have seen that reconstruction techniques are useful but they can lead to undesirable overfitting.
- This can be explained by our intuition that the calibration data is highly limited in two aspects: it is too little (compared to optimization variables), and may not represent the training data (as it is arbitrarily given).
- Crucially, noticing that what we are dealing with is a generative model, we suggest creating calibration data on our own, that is potentially much bigger in size and closer to the data that the original model is trained on.
- Here, we create the calibration data similarly to Liu et al. (2023), and the results are as follows:



(a) Test error (b) Perplexity



  where we can see that leveraging the self-generated calibration data reduces both test error and perplexity, mitigating overfitting quite effectively.

Conclusion

- Minimizing reconstruction errors can have both benefits and pitfalls, suggesting fundamentally rethinking the current practice of pruning LLMs.
- Leveraging self-generated calibration data can potentially mitigate this issue.

Contact: {ssbin4, jaeho.lee, namhoonlee}@postech.ac.kr, wppark@google.com