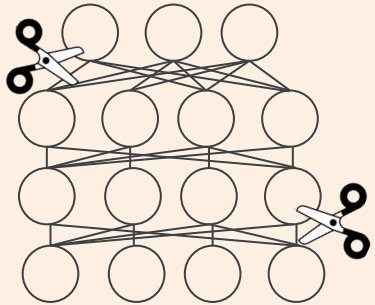
The costs of down-scaling LLMs

vaishnavh@
Oct 26, 2023

Tian Jin* (MIT), Nolan Clement* (MIT), Xin Dong* (Harvard),
Vaishnavh Nagarajan (Google Research), Michael Carbin (MIT),
Jonathan Ragan-Kelley (MIT), Gintare Karolina Dziugaite (DeepMind)

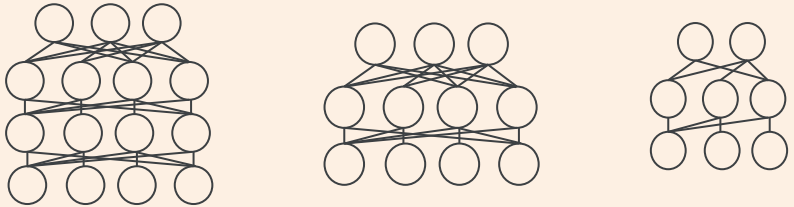
We study two types of down-scaling



Pruning

SparseGPT (Frantar & Alistarh, 2023)
Wanda (Sun et al., 2023)

(every layer pruned to some X%)



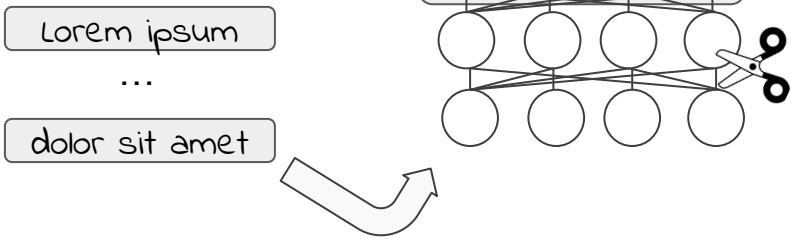
Dense downscaling

(independently training
smaller models from scratch)

Sidenote: How these pruning techniques work? *Roughly*:

Step 0: Pretrain as usual.

Step 1: Select tiny subset of pretraining data (e.g., 128 points with 2048 tokens).



Step 2: Prune each layer to X% in a way that minimally affects l2 distance of layer output on Step 1 data.

This is *one-shot* pruning that does not require retraining the network and does not require task-specific data. Outperforms magnitude pruning.

Pruning algorithms. We investigate pruning as one possible technique to (down-)scale LLMs. Few pruning algorithms currently scale to LLMs. We use SparseGPT (Frantar & Alistarh, 2023) in the main text and Wanda (Sun et al., 2023) in Appendix F. Both are one-shot pruning algorithms that scale to LLMs and outperform magnitude pruning (i.e., pruning the smallest magnitude weights), without computationally intensive re-training (Frantar & Alistarh, 2023). SparseGPT/Wanda prune each layer of the language model by minimizing the ℓ_2-distance between the outputs of the original dense layer and the pruned layer. SparseGPT/Wanda computes these outputs based on a small training dataset. See Frantar & Alistarh (2023, Sec. 3) for more details. While SparseGPT update the remaining weights after weights removal, Wanda does not.

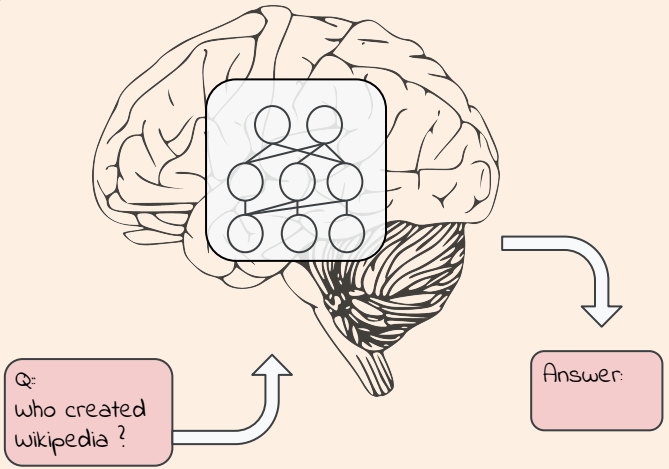
Following standard practice (Frantar & Alistarh, 2023; Frankle & Carbin, 2019), we only prune fully-connected layers. Since attention and feed forward modules consist of mostly fully-connected layers, parameters in fully-connected layers account for $> 97.5\%$ parameters for all models we examine. We do not prune embedding layers, language modeling heads and normalization layers.

Models. We evaluate 6 models from 3 families: OPT (Zhang et al., 2022), LLaMA (Touvron et al., 2023) and Pythia (Biderman et al., 2023b). We focus on OPT and LLaMA in our main text and present Pythia results in Appendix G. Pythia family models show consistent results as LLaMA and OPT family models. From the OPT family, we evaluate the two largest models that fit in our hardware setup – OPT-13B and OPT-30B, with 13 and 30 billion parameters, respectively. From the LLaMA family, we evaluate LLaMA-13B and LLaMA-33B, with 13 and 33 billion parameters, respectively.

A notable difference between OPT and LLaMA families is the ratio of training data to model parameters. Zhang et al. (2022) train the OPT family of models with 180 billion tokens, yielding approximately 14 and 5.5 tokens *per* parameter, respectively, for our two considered OPT models. Touvron et al. (2023) train the LLaMA-13B model with 1 trillion tokens (77 tokens/parameter) and the LLaMA-33B model with 1.4 trillion tokens (42 tokens/parameter).

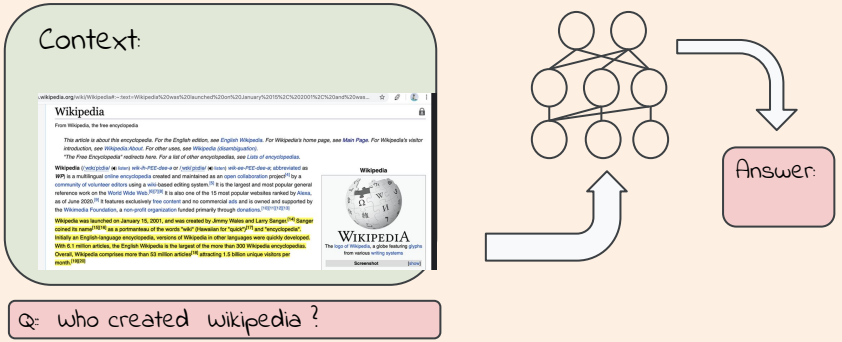
We study two core, *complementary* LLM capabilities

(Chan et al., 2022a, 2022b)



Fact Recall

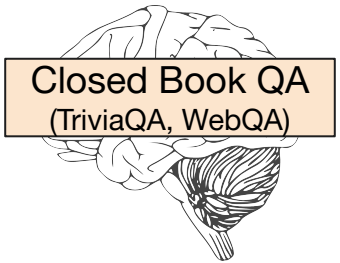
from pretraining data



Learning from context

Most generic tasks demand both capabilities.
How do we disentangle them?

We curate a suite of tasks to disentangle the two capabilities



Closed Book QA
(TriviaQA, WebQA)

Question: what is...

≠

Overriding QA
(DissentQA)
(Neeman et al. 2022,
Li et al., 2022;
Longpre et al., 2021)

Context:

Question: what is ...

Fact Recall

Open Book QA
(TriviaQA, NaturalQA)

Context:

Question: what is...

(*f* is linear/decision tree/perceptron)

Generic Classification

Context:

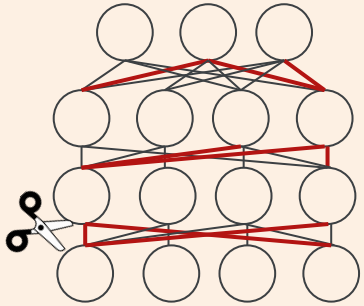
x_1, $f(x_1)$
x_2, $f(x_2)$
x_3, $f(x_3)$

Question: x_{test}

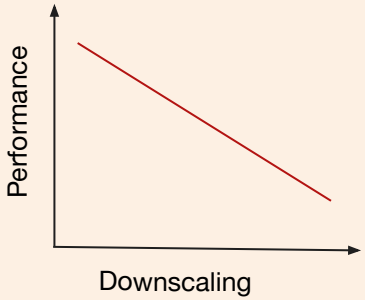
In-context Learning

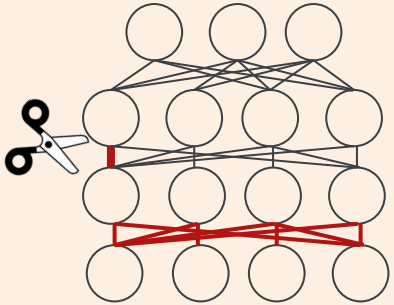
Datasets. We use these datasets: (a) *TriviaQA*. Joshi et al. (2017) developed the TriviaQA dataset with questions and supporting evidence. We use its Wikipedia validation partition consisting of 7993 questions. (b) *WebQuestions*. Berant et al. (2013) collected question-answer pairs from the Freebase knowledge database. We use its test set consisting of 2032 questions. (c) *NaturalQuestions*. Kwiatkowski et al. (2019) compiled the NaturalQuestions dataset from Google search queries. We sampled a 7700-question subset of its validation partition (the same size as the following dataset derived from it), to make our evaluation computationally feasible. (d) *DissentQA*. Neeman et al. (2022) constructed the DissentQA dataset from the NaturalQuestions dataset. It contains pairs of questions and evidence for a made-up answer that is different from the factual one. It assesses whether the model can override its memory formed during pre-training with new context. We use its validation partition consisting of 7700 questions.

How could down-scaling affect a capability?

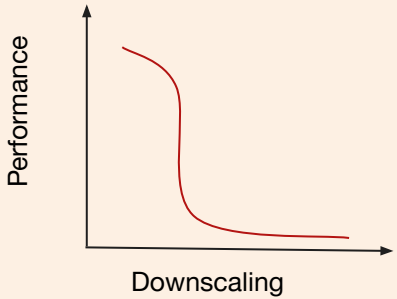


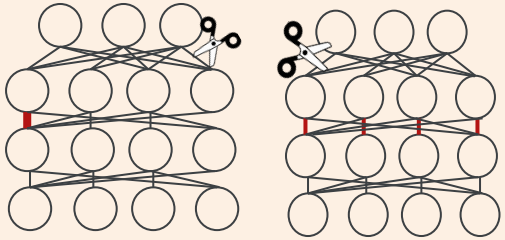
World 1: Highly distributed capability



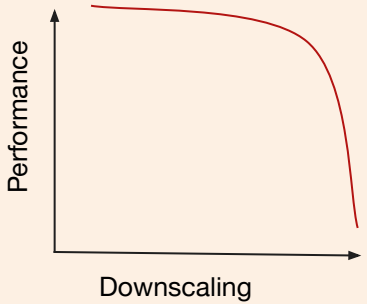


World 2: Vulnerable bottleneck exists

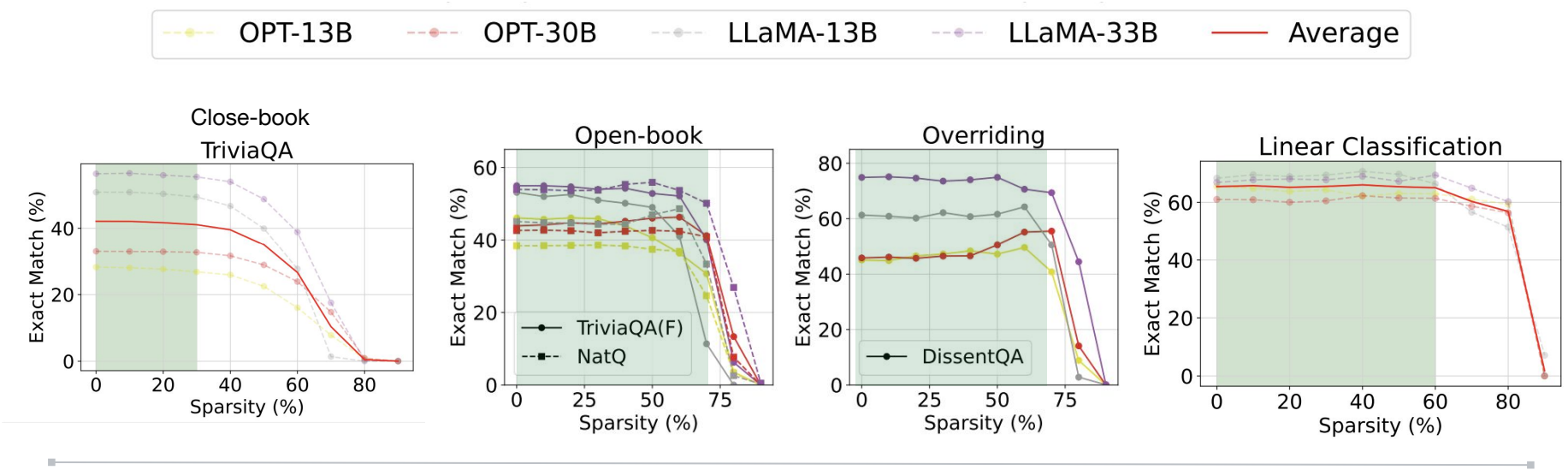




World 3:
Concentrated in safe spot (or) implemented redundantly



Pruning has disparate effects on the two capabilities



Fact Recall

In-context Learning

Fact recall deteriorates quicker (5% drop around 30% of pruning) while in-context learning withstands as much as 60-70% pruning

Wanda (the other pruning approach) shows the same results

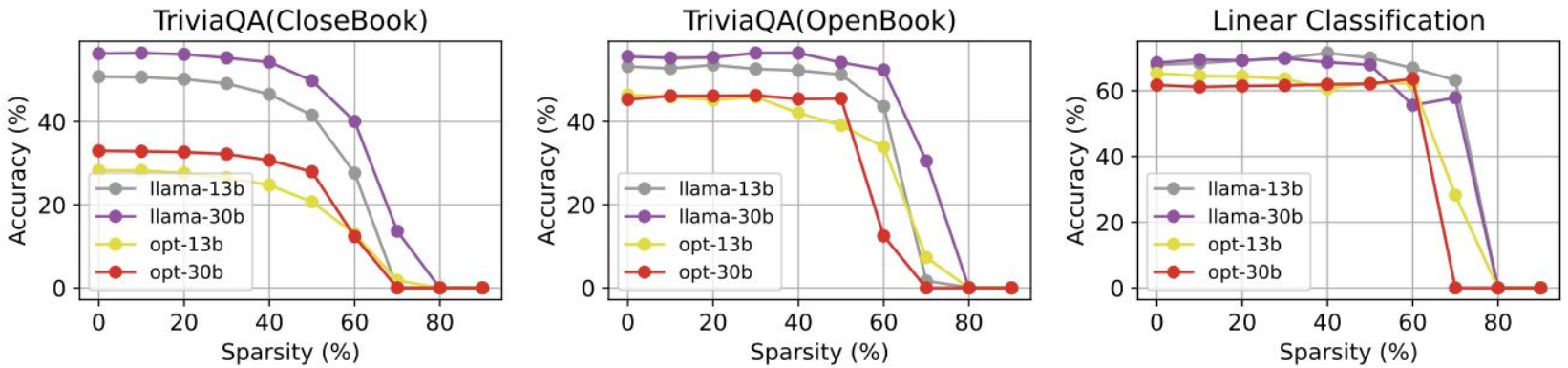
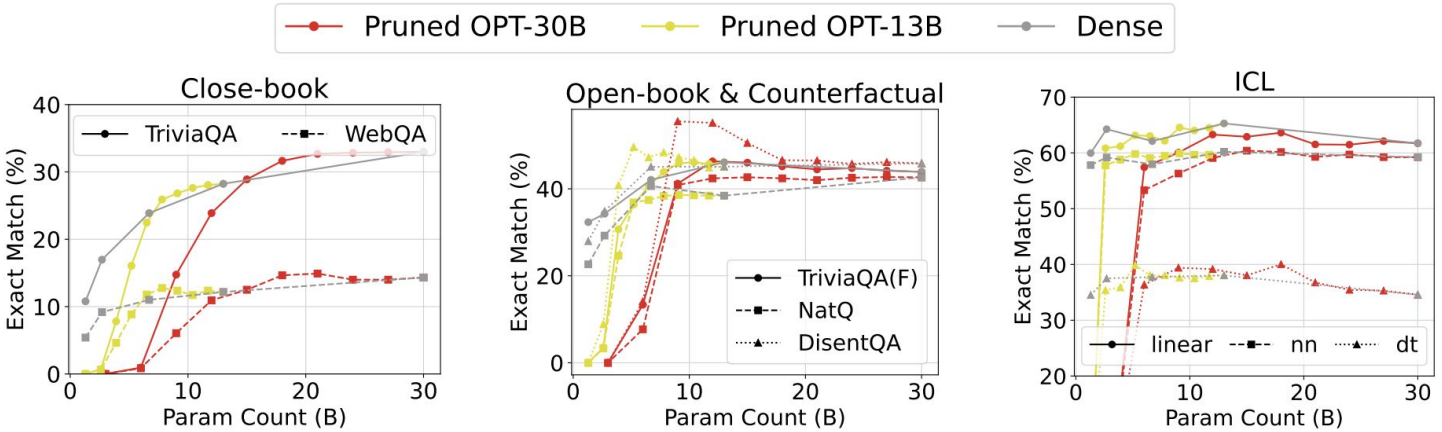


Figure 8: An alternative pruning algorithm (Wanda) shows the same patterns of accuracy drop as findings in our original paper: moderate pruning hurts fact recall (e.g., left, TriviaQA in Closebook setting) while ICL (e.g., right, in-context linear classification) survives to higher sparsity. Specifically, accepting a 5% relative drop in accuracy, one may remove 30%, 40% and 50% weights on TriviaQA(Closebook), TriviaQA(Openbook) and Linear Classification tasks, respectively.

Dense downscaling too has the *same* disparate effects

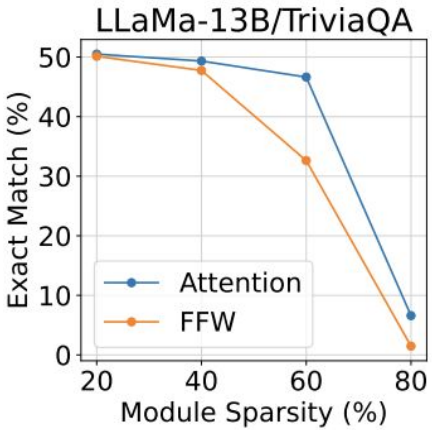


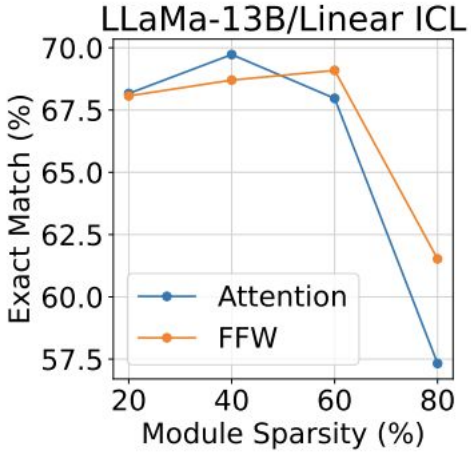
Fact Recall

In-context Learning

Even under dense downscaling, fact recall deteriorates much quicker than in-context learning.

What happens if we pruned only Attention or only MLP?





Fact Recall

In-context Learning

For fact recall, MLP layers are more important than Attention.
For in-context learning, not as much difference.

Pythia Models show same behavior

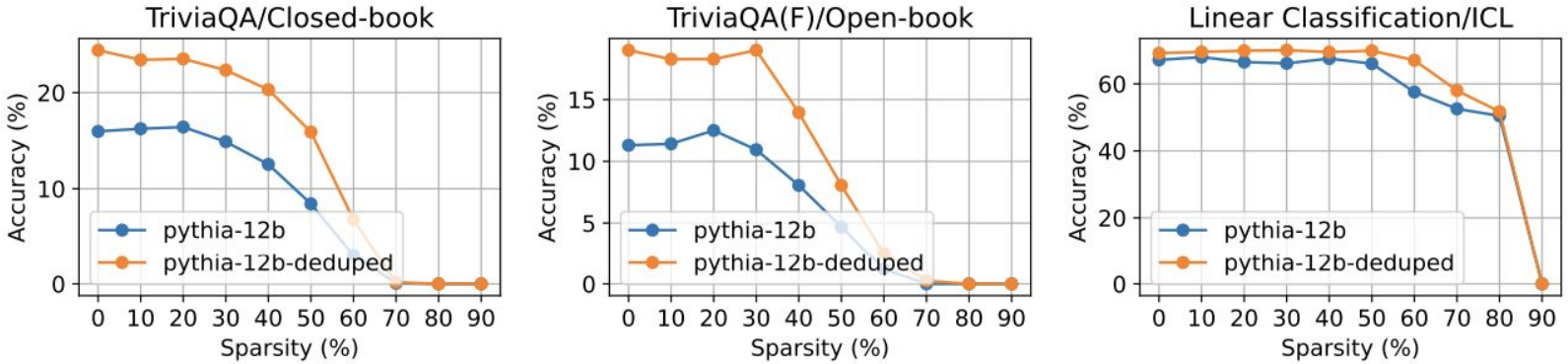
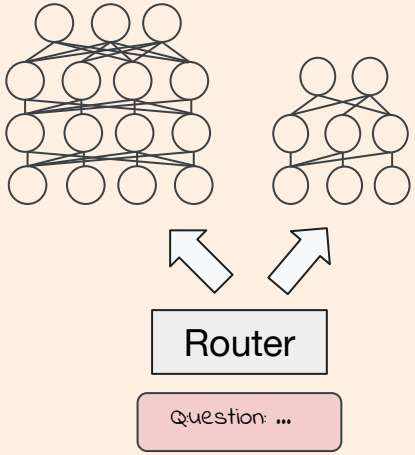


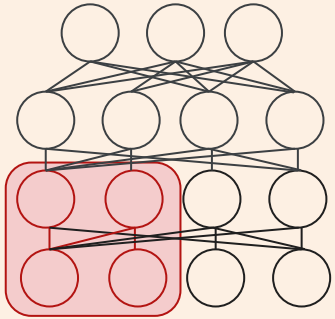
Figure 9: Two variants of Pythia-12B model shows the same patterns of accuracy drop as findings in our original paper: moderate pruning hurts fact recall (e.g., left, TriviaQA in Closebook setting) while ICL (e.g., right, in-context linear classification) survives to higher sparsity. Specifically, accepting a 5% relative drop in average accuracy, one may remove 20%, 30% and 50% weights on TriviaQA(Closebook), TriviaQA(Openbook) and Linear Classification tasks, respectively.

So what? Practical implications:



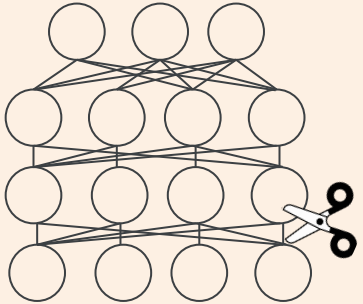
Improve inference-time compute efficiency

Retrieve evidence into context + route to smaller model



Improve interpretability

Is there a small module that is responsible for in-context learning?



Improve pruning

Prune MLP layers more than attention?