The Impact of Depth on Compositional Generalization in Transformer Language Models

Jackson Petty[1*], Sjoerd van Steenkiste[2], Ishita Dasgupta[3], Fei Sha[2], Dan Garrette[3], and Tal Linzen[2]

[1] NYU [2] Google Research [3] Google DeepMind

* *Work done as a Student Researcher at Google Research*

Question: Are deeper models more *compositional*, independent of total parameter count?

1. What is compositionality?

Generalize from **known pieces** to **(infinite) novel, well-formed combinations**

Necessary for semantic parsing (see COGS (vf) below), NLU, code generation, & more

Training input (hedgehog is subject)		*Output*
`the hedgehog ate the cake`		*eat*(***agent=hedgehog***, *theme=cake*)
`the hedgehog saw a child`		*see*(***agent=hedgehog***, *theme=child*)
`hedgehogs swim`	→	*swim*(***agent=hedgehog***)
Generalization (hedgehog is object)		
`the boy loves the hedgehog`		*love*(*agent=boy*, ***theme=hedgehog***)

2. Why might depth help?

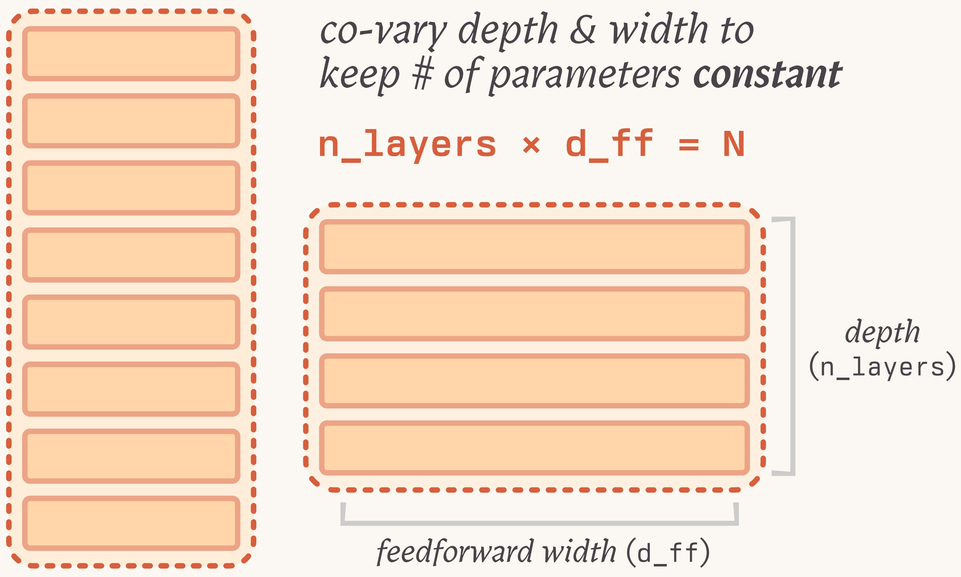
Theory:
- Expressive capacity is exponential in depth
- Each layer does successive function application

Empirically: Reducing depth harms linguistic generalization more than reducing width does

3. Controlling for # of parameters

Depth & total # of parameters are usually correlated

Many things improve w/ more parameters, so we must control for this confounder



*co-vary depth & width to keep # of parameters **constant***

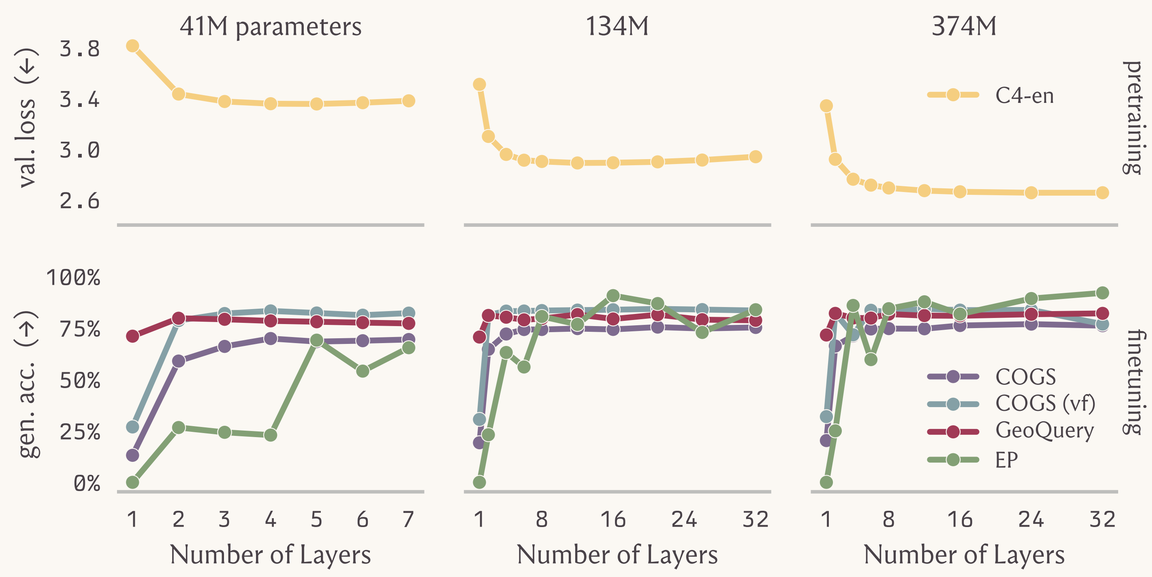
4. Experimental setup

Pretrain+finetune models of different depths within three size classes: 41M, 134M, and 374M parameters

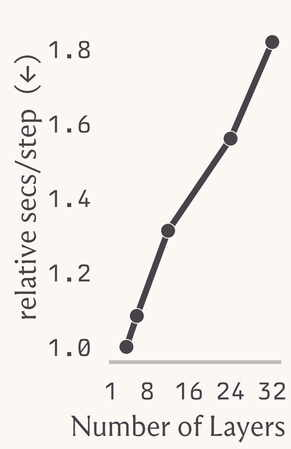
dataset	*type*	*metric*
C4-en	language modeling	validation loss
COGS	semantic parsing	full-sequence generalization accuracy
COGS (variable free)	semantic parsing	
GeoQuery	SQL generation	
English Passivization (EP)	natural language transformation	

5. Results: diminishing returns

Depth helps language modeling and compositional generalization, but ***marginal utility drops fast*** beyond ~ 6 layers



6. Depth is expensive



Latency/cost is ***linear*** in depth, but performance is ***sub-linear***

2× slower doesn't buy 2× better performance

Once a model is "deep enough," choosing depth over width is not efficient

Answer: Up to a point.

Depth aids compositionality and language modeling, but diminishing returns & linear latency cost mean choosing depth over width is an expensive choice beyond the first few layers.