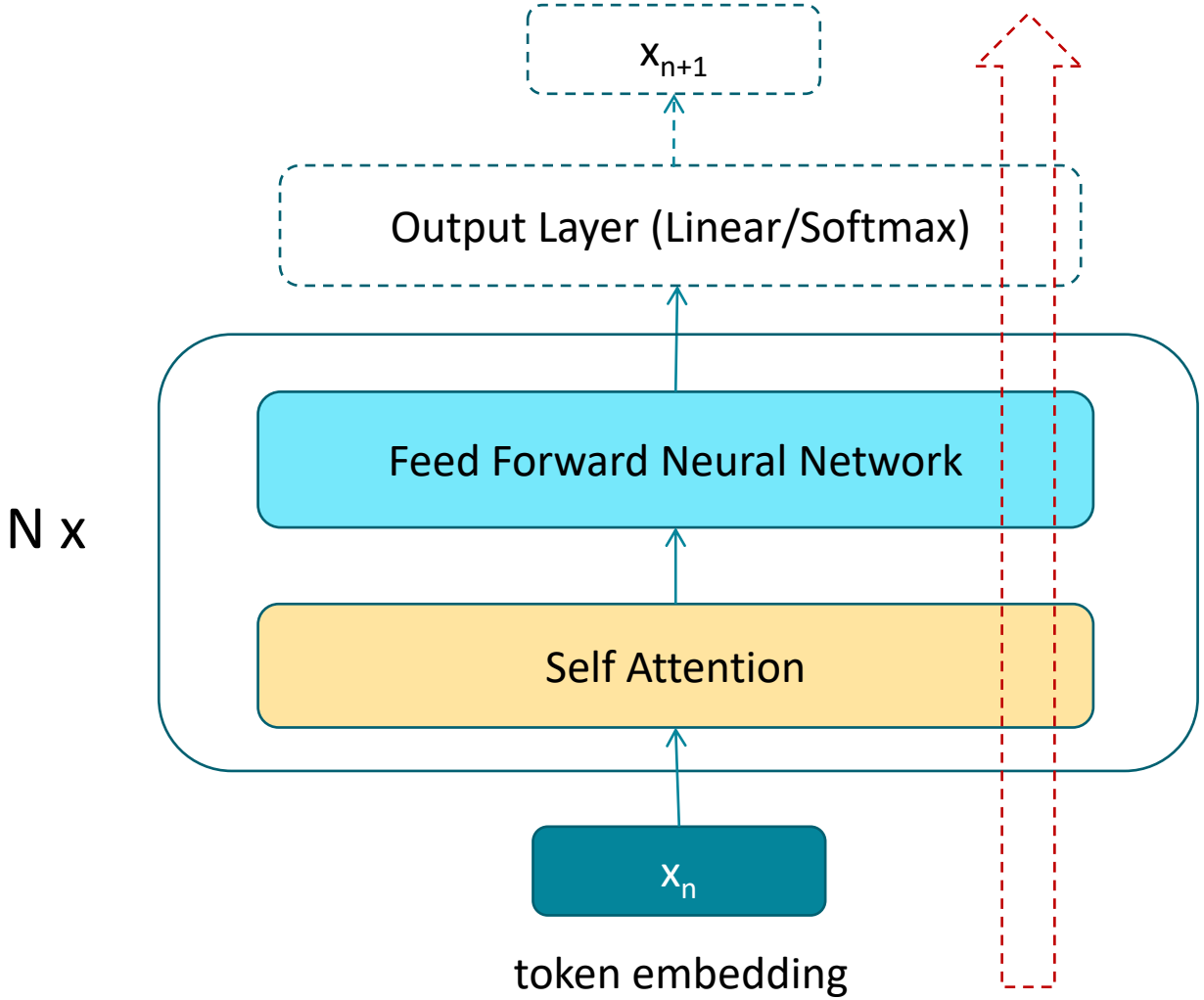
Adaptive Inference in Pre-trained LLMs Layer-skipping

Xuan Luo, Weizhi Wang, Xifeng Yan

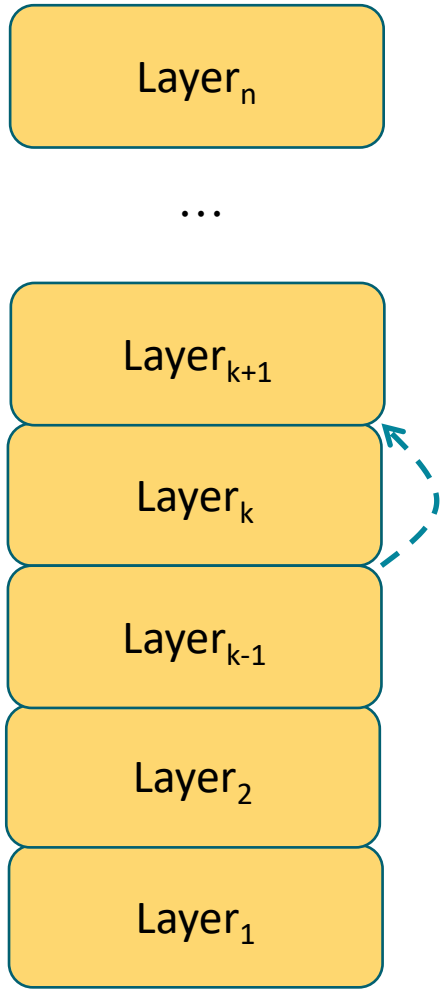
Computer Science
University of California, Santa Barbara

Transformer



No matter what the task is,
No matter what the tokens are,
Every token requires the same effort!

Can We Skip Some Layers Dynamically?



Do it at the token level: Different tokens may bypass different layers!

Layer-skipping: # of Layers needed

[Context] The grey seal was found at Cruden Bay on Tuesday. The Scottish SPCA removed the seal from beach and took it to the charity's rescue centre... Thankfully there were no injuries in this case and we were able to release him relatively quickly.

[Prompt] Where the grey seal was found?

The grey seal was found at Cruden Bay.

[Prompt] Please summarize this paragraph into a single sentence:

A grey seal was found tangled in netting at Cruden Bay , rescued by the Scottish S PCA , and released back into the water with no injuries.

[Prompt] Please calculate the sum of the 8 numbers in the list: [99, 45, 12, 78, 33, 66, 21, 54]

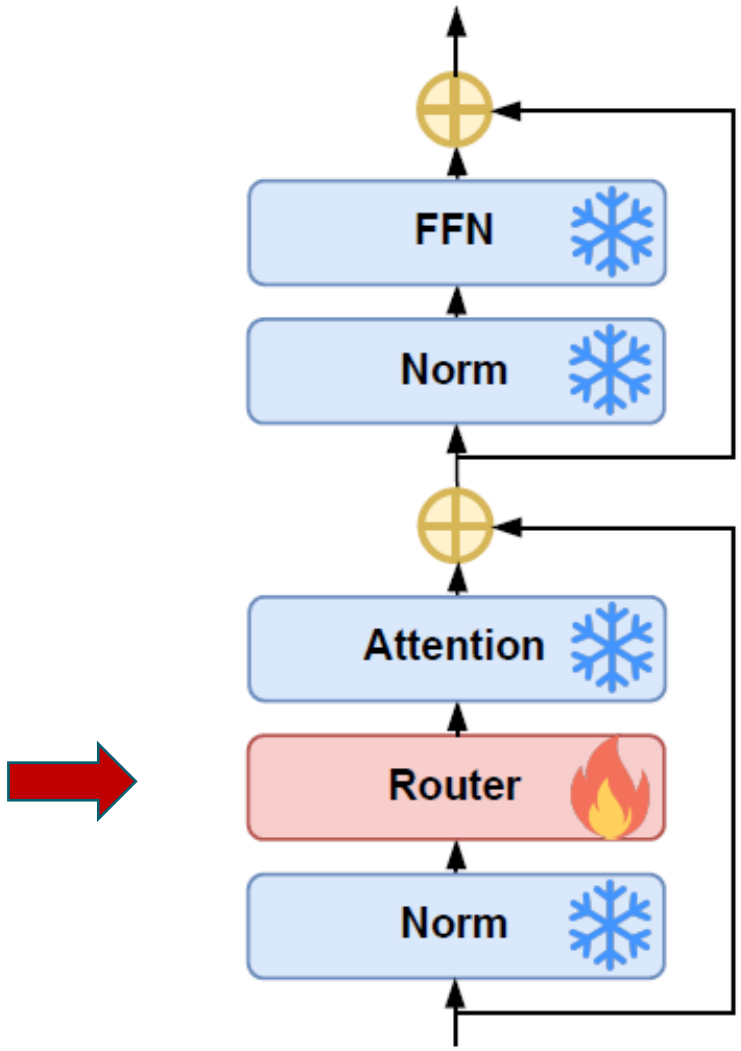
To calculate the sum of all 8 numbers in the list, I'll add each number one by one:

99 + 45 = 144
144 + 12 = 156
156 + 78 = 234
234 + 33 = 267
267 + 66 = 333
333 + 21 = 354
354 + 54 = 408

The sum of all 8 numbers in the list is 408.

Layer-skipping patterns (Llama-3-8B-Instruct) for a language task (left) and a math task (right). The light-to-dark blue gradient represents layer usage from 16 to 32.

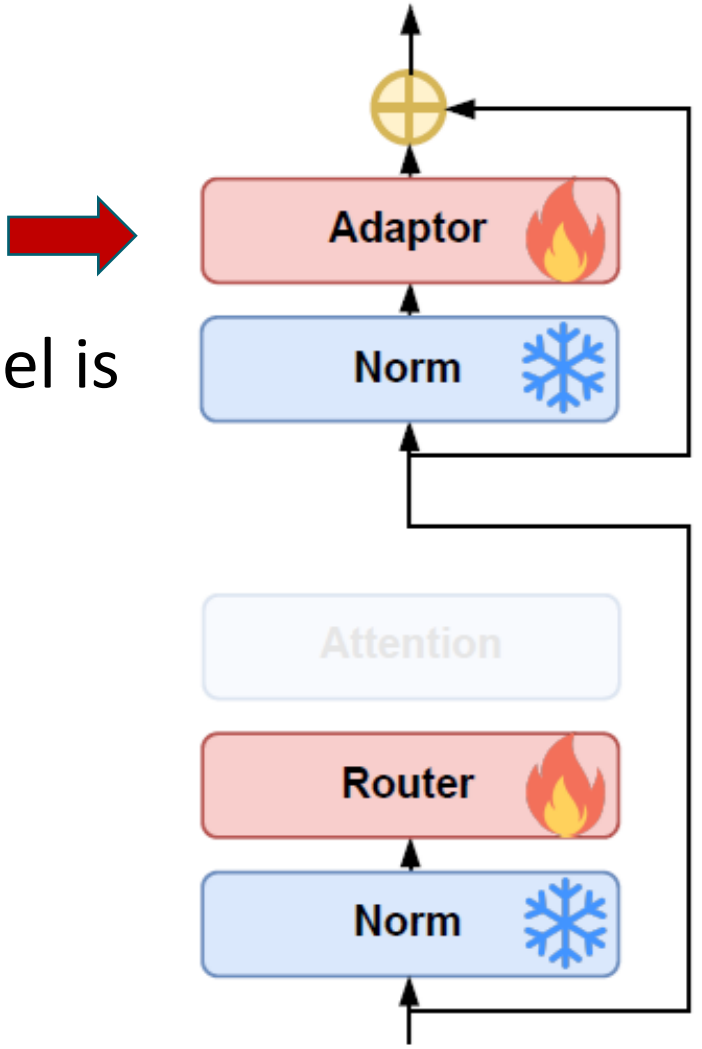
FlexiDepth: Router



$$\text{Router}(z) = W_{\text{r}} \cdot (W_{\uparrow} \cdot \text{Norm}(\tanh(W_{\downarrow} z)))$$

FlexiDepth: Adaptor

Adaptor is not needed if the model is fully retrained.



Loss Function

$$\mathcal{L}_{skip} = \frac{1}{T} \sum_{t=1}^{T} \left(\sum_{l=1}^{L} g_t^l \right)^2$$

# of layers kept

where g_t^l denotes the gating score for layer l at time step t.

total loss:

$$\mathcal{L} = \alpha \cdot \mathcal{L}_{\text{skip}} + \mathcal{L}_{\text{lm}}$$

Average # of Layers for Text Operations

Randomly sample 100 paragraphs from the XSum test set (Narayan et al., 2018), which is a collection of news articles. We evaluated FlexiDepth on three subtasks for each paragraph

- Please copy this paragraph
- Please summarize this paragraph into a single sentence
- Please continue writing this paragraph

Results: the average number of layers used per token

- Copy: 22.0 layers
- Summarization: 28.7 layers
- Continuation: 30.3 layers

Average # of Layers for Math Calculation

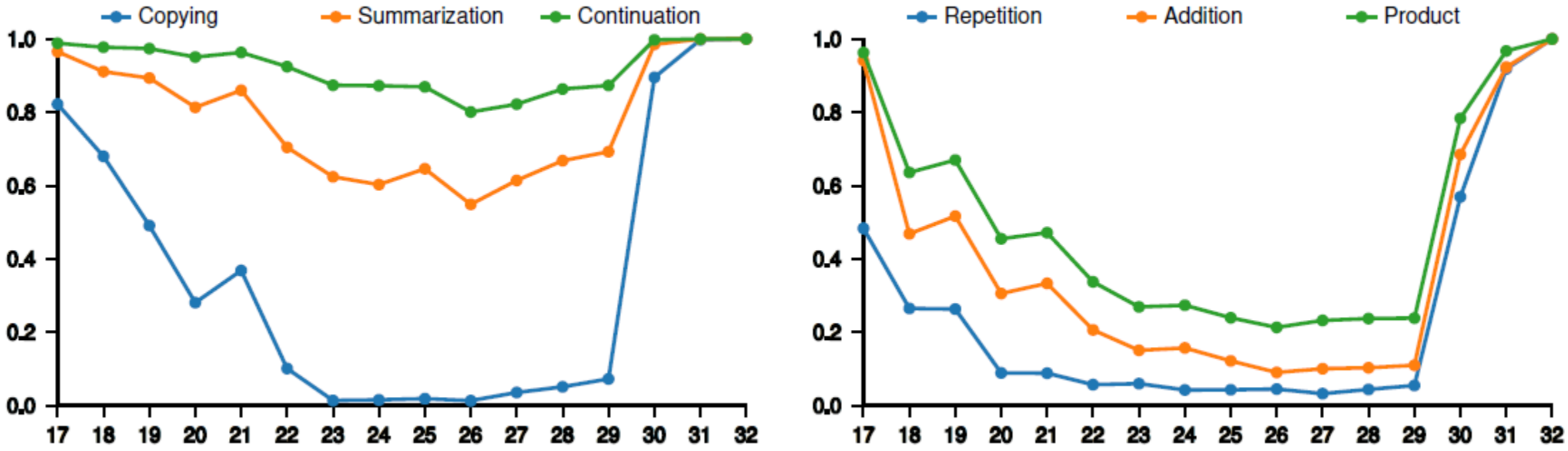
Generate a list of 5–10 integers between 10 and 99. Three subtasks:

> Please repeat the following list for 5 times: [...]
> Please calculate the sum of numbers in the following list: [...]
> Please calculate the product of numbers in the following list: [...]

The number of layers used per token:

- Copy: 20.1 layers
- Addition: 22.5 layers
- Multiplication: 23.9 layers

Layer Traffic



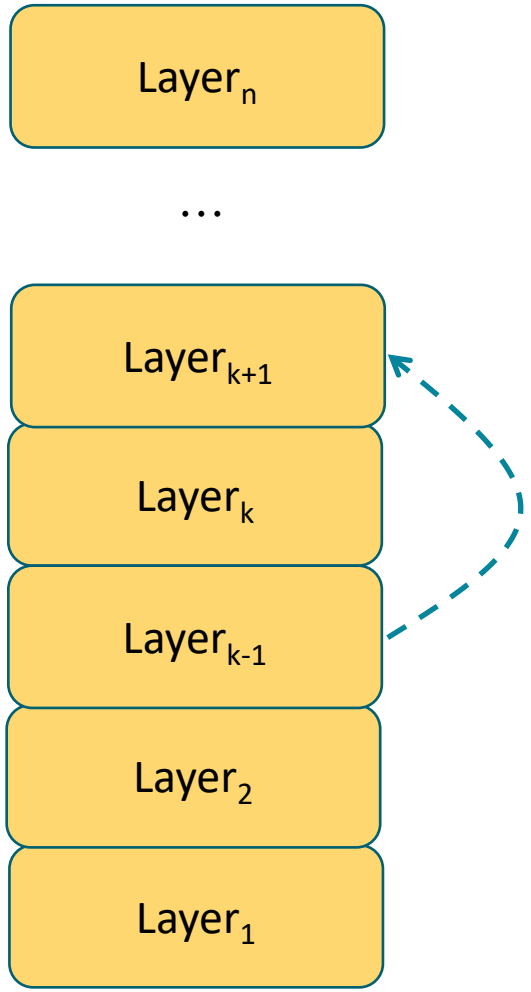
Percentage of tokens processed by each layer (Layer 17th to 32nd, Llama-3-8B-Instruct)

Method Comparison

Methods	Single-Token Generation			Multi-Token Generation			Retain %
	MMLU	Hellaswag	Winogrande	GSM8K	HumanEval	CoQA	
Vanilla	0.673	0.706	0.744	0.679	0.299	0.784	100.0%
Skip 4 Layers							
LayerSkip	0.659	0.636	0.676	0.004	0.0	0.350	54.0%
ShortGPT	0.664	0.662	0.700	0.536	0.092	0.145	69.1%
LaCo	0.671	0.693	0.724	0.581	0.031	0.778	81.7%
MindSkip	0.664	0.698	0.722	0.378	0.189	0.720	84.2%
Ours	0.663	0.724	0.756	0.695	0.390	0.810	106.5%
Skip 8 Layers							
LayerSkip	0.650	0.525	0.640	0.0	0.0	0.049	43.9%
ShortGPT	0.307	0.462	0.597	0.001	0.0	0.005	32.0%
LaCo	0.656	0.628	0.695	0.065	0.006	0.707	65.3%
MindSkip	0.602	0.650	0.646	0.039	0.024	0.620	60.2%
Ours	0.616	0.705	0.735	0.662	0.341	0.801	100.7%

Llama-3-8B-Instruct, which consists of 32 layers.

Layer Skipping Shows Underutilized Computation

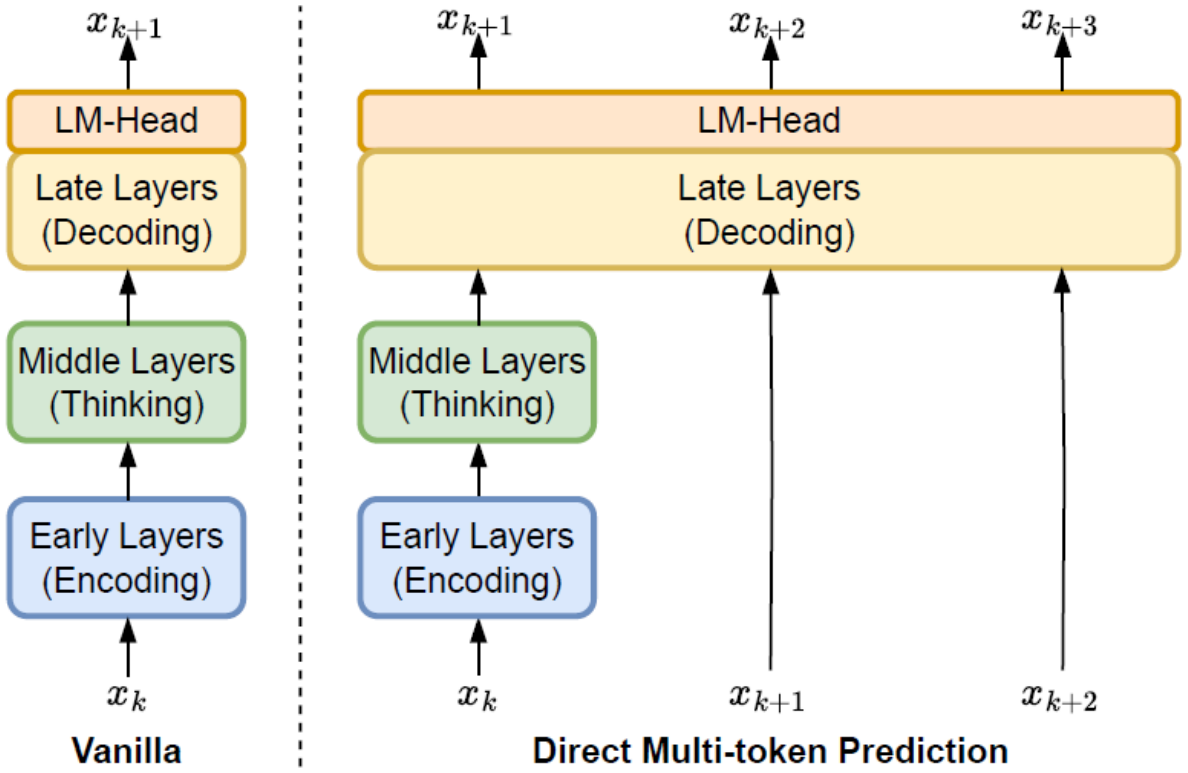


Layer skipping is not synchronized during token generation; different tokens bypass different layers.

Another Perspective: Direct Multiple Token Decoding

Instead of skipping layers, can these layers be used to compute future tokens?

Goal: Does not require verification, unlike speculative decoding.



Preliminary Results on Qwen3-4B

	ARC-E	ARC-C	WinoGrande	GSM8K	CoQA	Overall
Vanilla	0.934	0.922	0.657	0.907	0.805	100%
MTD2	0.930	0.897	0.701	0.901	0.798	100.0%
MTD3	0.921	0.886	0.673	0.889	0.780	98.4%
MTD4	0.916	0.881	0.652	0.866	0.749	96.3%
MTD6	0.872	0.801	0.601	0.500	0.672	82.1%

- Achieved by fine-tuning Qwen3-4B with 1.5B tokens
- Utilized the last 8 of the 36 layers for predicting the second and subsequent tokens
- Up to 2x speedup (likely more speedup if fully retrained)

Luo, Wang, Yan, “Direct Multiple Token Decoding,” will be on arxiv soon.

Links

- Project Website/Model/Code/Data
- We seek collaboration with additional compute and training data to extend these ideas, including MoE.
- Thanks!

Room 710, Poster **#45**