ReLU Strikes Back: Exploiting Activation Sparsity in Large Language Models

Iman Mirzadeh, Keivan Alizadeh, Sachin Mehta, Carlo C Del Mundo, Oncel Tuzel, Golnoosh Samei, Mohammad Rastegari, Mehrdad Farajtabar
ICLR 2024 · Apple

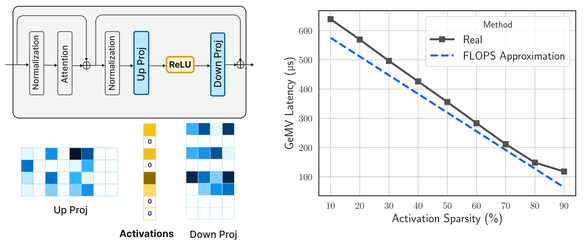
TL;DR

1- ReLU is significantly more inference-friendly that other activation functions
ReLU activations are very sparse, which can be used for faster inference.

2- Most of the modern LLMs are trained without ReLU.
Thus non-sparse activation and more costly inference.

3- Non-ReLU activations do not improve the performance significantly. However, they lead to non-sparse activations leading to much more costly inference.

ReLU & Activation Sparsity

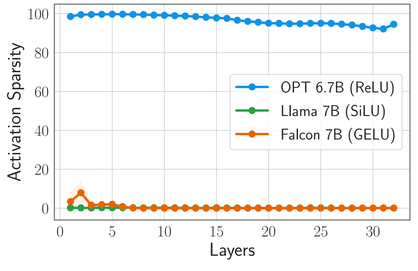


The zeroed elements do not need to be multiplied with the associated rows from Down Proj

Sparse MatVec on M2-Macbook: Near perfect speedup due to skipped multiplications

LLMs & Activation Functions

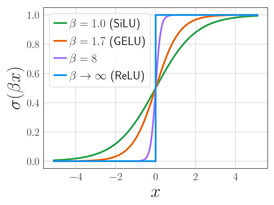
ReLU is rarely being used in SoTA LLMs



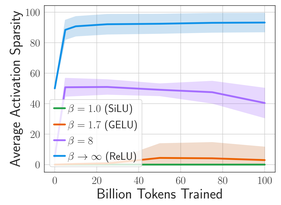
ReLU leads to significantly higher activation sparsity.

The Impact of the Activation Function

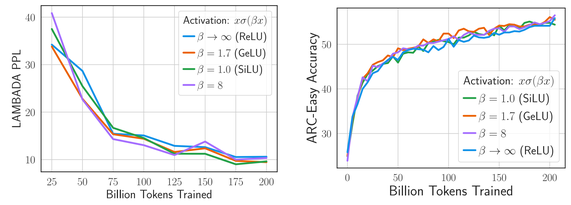
When trained from scratch, the choice of activation function has a negligible impact on performance



Behavior of activation functions

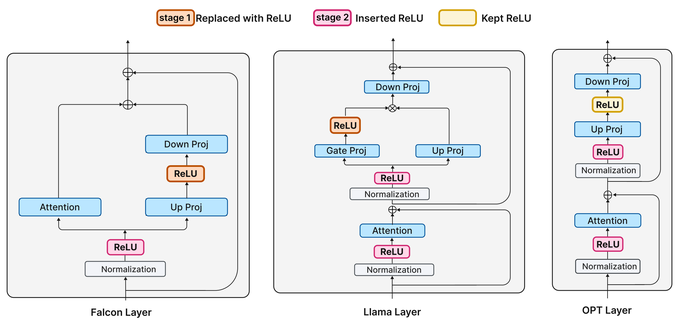


Induced activation sparsity

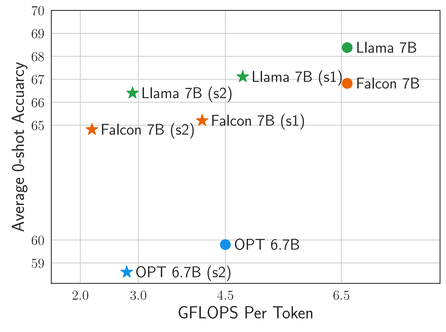


Training an OPT model from scratch with various activation functions results in comparable performance across different functions.

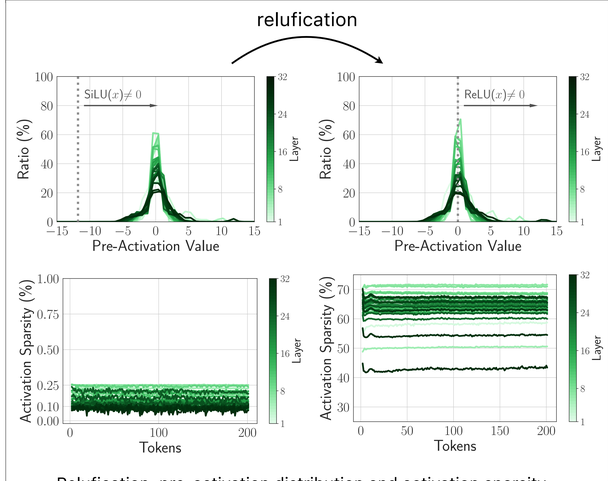
Relufication: Replace/insert ReLU layers to LLMs that were pre-trained w/o ReLU



Model	Input Sparsity (%)			FLOPS (G)	Avg 0-shot
	QKV	UpProj	DownProj		
OPT 6.7B	0	0	97	4.5	59.8
OPT 6.7B (relufied-s2)	50	40	97	2.8	58.6
Llama 7B	0	0	0	6.6	68.4
Llama 7B (relufied-s1)	0	0	62	4.8	67.1
Llama 7B (relufied-s2)	51	67	65	2.9	66.4
Falcon 7B	0	1	0	6.6	66.8
Falcon 7B (relufied-s1)	0	0	94	4.1	65.2
Falcon 7B (relufied-s2)	56	56	95	2.2	64.8



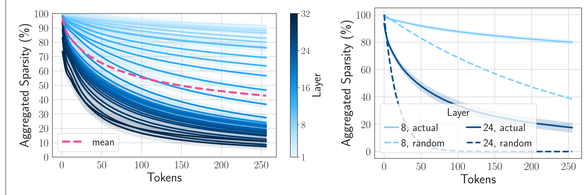
Relufication: Additional Results



Relufication, pre-activation distribution and activation sparsity

Aggregated Sparsity

Definition: How much the neurons haven't been used for processing the first t tokens?



For the OPT-6.7B model, on average, about 50% of all the neurons will be unused across the first 150 tokens of the prompt.

Takeaway

Favorable accuracy-efficiency tradeoff of ReLU:
Generally, using ReLU has a minimal impact on performance, yet it can significantly speed up token generation.

Call for more work on inference-aware architecture design
inference costs generally outweigh training costs over the long term. This factor should be considered more carefully in architecture design.