

GreenAI

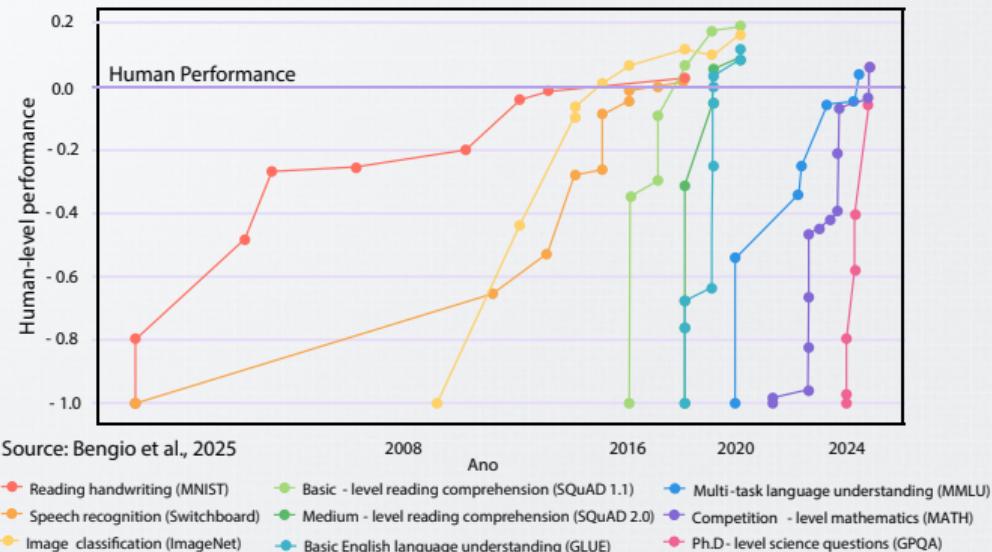
Artur Jordão
Escola Politécnica
Universidade de São Paulo

August 4, 2025

Introduction

GreenAI

- Deep learning drives unprecedented progress in various cognitive tasks and serves as the powerhouse for learning patterns from data
 - Modern models match or even surpass human performance¹



¹Bengio et al. *International Scientific Report on the Safety of Advanced AI*, 2025

Advances in Deep Learning

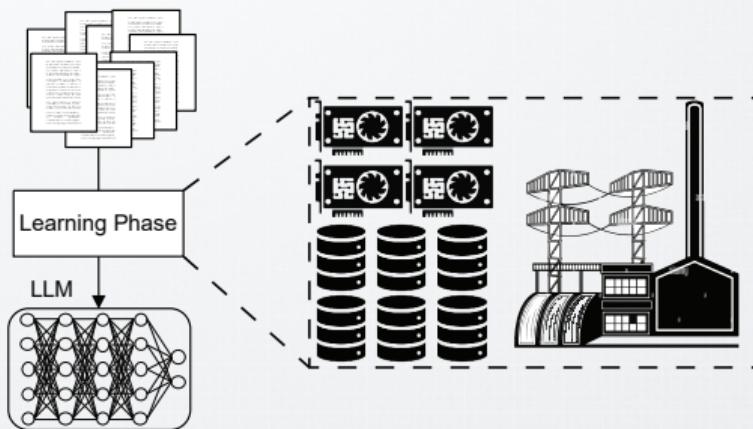
GreenAI

- Three factors drive the advance of Deep Learning
 - Algorithm innovation (predictive models)
 - Data
 - Hardware (amount of computing available)
- Algorithm innovation × data × hardware
 - The availability of large and well-curated datasets benefits the predictive ability of models
 - The need for massive computing exposes shortcomings of current algorithms
 - To avoid performance overhead, large datasets require high-capacity hardware

Energy Consumption

GreenAI

- There is a growing demand for energy to power AI workloads
 - Microsoft recently signed a deal to purchase the next 20 years of energy generated by reopening a nuclear power plant¹
 - Energy providers are extending the life of aging fossil fuel plants to keep up with demand¹



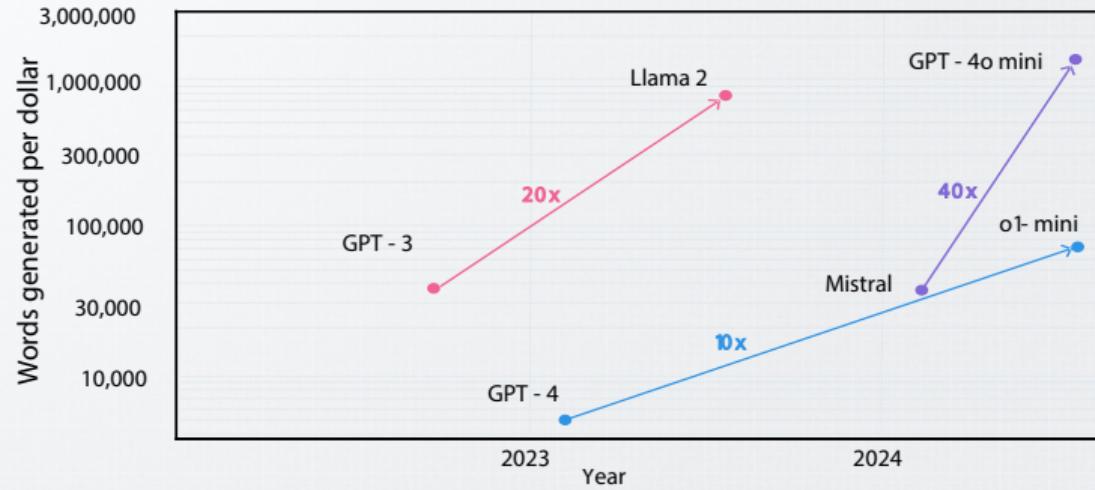
Modelo (#Params)	#Exec.	Equiv. CO2	Equi. Água (1 pessoa)
<1B	20	gallons of gasoline	3 months
1B	227	40× NY ↔ SF	1 year
7B	375	150 oil barrels	2 years, 7 month

¹Morrison et al. *Holistically Evaluating the Environmental Impact of Creating Language Models*. ICLR, 2025

Financial Cost

GreenAI

- Modern models are more efficient in terms of inference per cost¹
- The largest model providers are producing up to a hundred billion tokens per day²



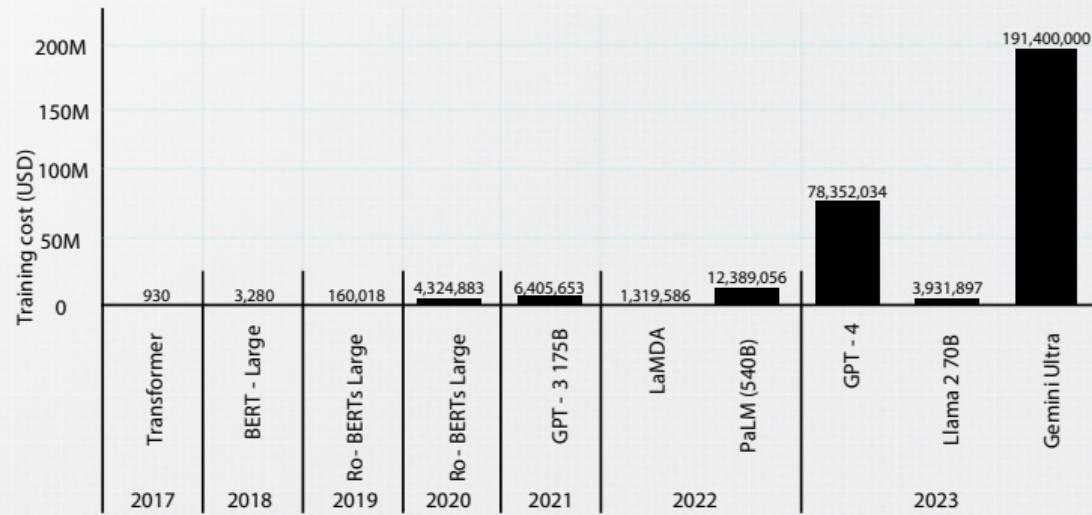
¹Bengio et al. *International Scientific Report on the Safety of Advanced AI*, 2025

²Morrison et al. *Holistically Evaluating the Environmental Impact of Creating Language Models*. ICLR, 2025

Financial Cost

GreenAI

- The development of state-of-the-art models requires enormous financial investment¹
- Few companies can afford to train models at such a high cost¹

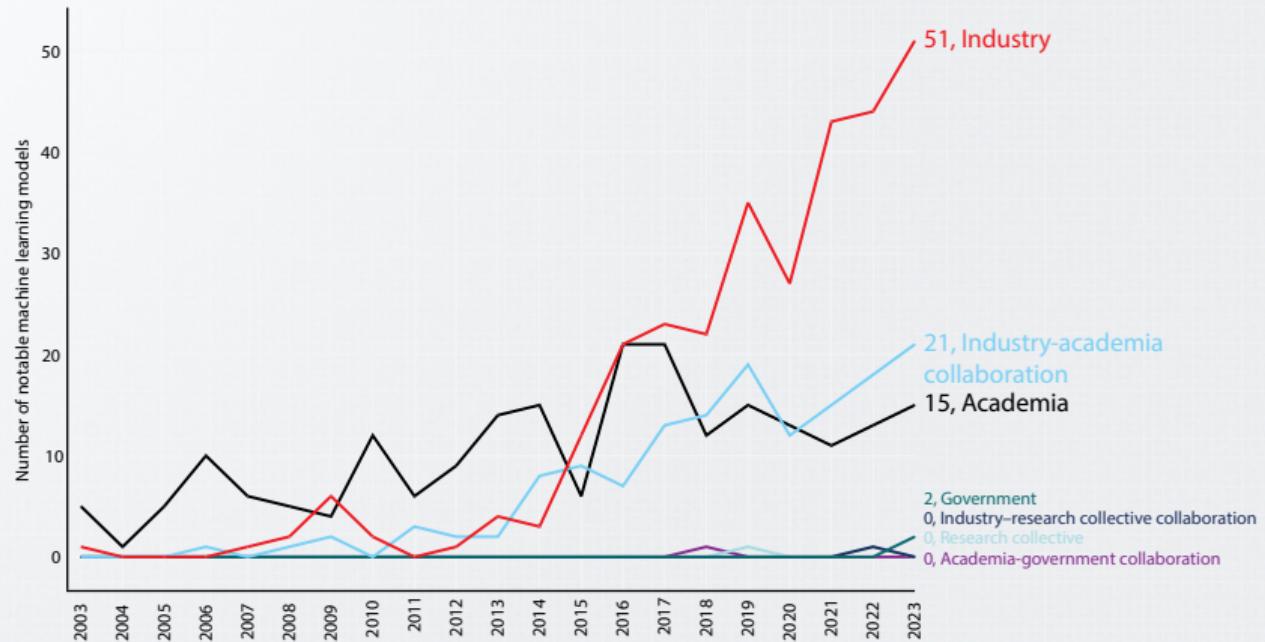


¹Bengio et al. *International Scientific Report on the Safety of Advanced AI*, 2025

Notable Models by Sector

GreenAI

- Companies typically have more access to computational resources than academic institutions¹



¹Maslej et al. *Artificial Intelligence Index Report*, 2024

GreenAI Metrics

GreenAI

- CO₂-eq. footprint¹ = Energy Consumption × Carbon Energy Efficiency
 - Energy Consumption (kWh): W*hours/1000
 - Carbon Energy Efficiency: CO₂ emissions per kWh (Default value 0.432)
- Since other gases have a warming effect, the standard measure describes how much warming a given amount of gas causes, expressed in CO₂-equivalents
- User-friendly toolkits
 - (Deep learning-specific)¹ <https://mlco2.github.io/impact/>
 - (LLM-specific)² <https://github.com/SotaroKaneda/MLCarbon>
 - (General Purpose)³ <http://calculator.green-algorithms.org/>

Lacoste et al. *Quantifying the carbon emissions of machine learning*. NeurIPS, 2019

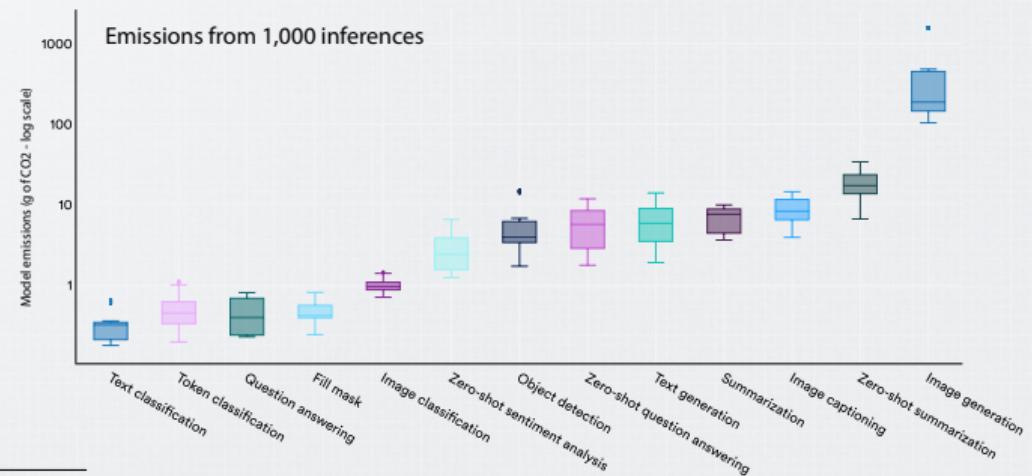
Faiz et al. *LLMCarbon: Modeling the End-to-End Carbon Footprint of Large Language Models*. ICLR, 2024

Lannelongue et al. *Green Algorithms: Quantifying the Carbon Footprint of Computation*. Advanced Science, 2021

Environmental Impact of Inference

GreenAI

- Models requiring more computation often have larger environmental footprints¹
- While the emissions from inference may be relatively low, the total impact can surpass that of training when models are queried thousands, if not millions, of times daily



¹Morrison et al. *Holistically Evaluating the Environmental Impact of Creating Language Models*. ICLR, 2025

Challenges

GreenAI

- A major challenge in evaluating the environmental impacts of AI models is a lack of transparency about emissions
- The environmental impact of vital pieces of all model development pipelines is still unknown¹ (embodied emissions)
 - GPUs manufacturing
 - **Data center construction**

¹Morrison et al. *Holistically Evaluating the Environmental Impact of Creating Language Models*. ICLR, 2025

Positive Side of Large Models

GreenAI

- The community widely acknowledges the environmental costs of developing AI systems
- AI can help promote environmental sustainability

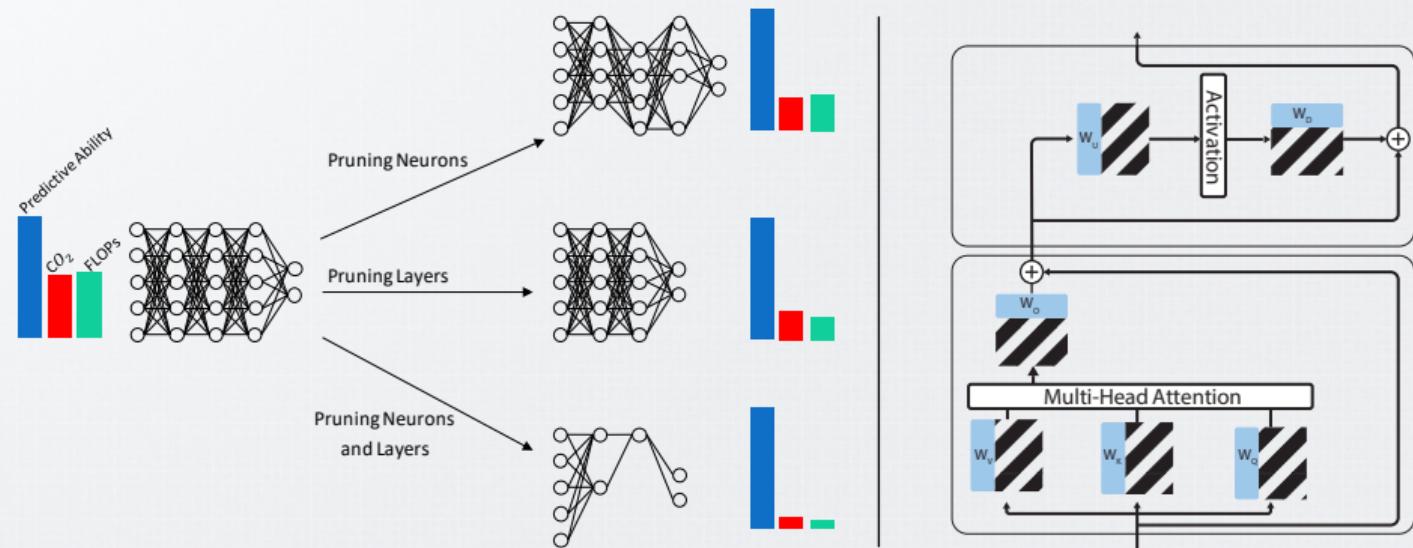
Use Case	AI Contribution
Management of thermal energy storage systems	Anticipating thermal energy needs and managing thermal energy storage systems
Agriculture	Identifying and eliminating pests in commercial harvests
Enhancing urban air quality	Forecasting and predicting air quality in urban cities
Predicting extreme events	Early identification of extreme storm tide events

Research Projects

Compression of Deep Models

Research Projects

- Pruning strategies remove structures from neural networks
- *How to identify unimportant structures?*



Compression of Deep Models

Research Projects

Effective Layer Pruning Through Similarity Metric Perspective

Luis Pons¹, Bruno Yamamoto¹, Anna H. Reali Costa¹, Arthur Jardim¹

Abstract

Deep neural networks have been the predominant paradigm in machine learning for solving cognitive tasks. However, they are often characterized by a high computational overhead, limiting their applicability and hindering advancements in the field. Therefore, pruning layers and removing structures from these models is a straightforward approach to reducing network complexity. In this work, we propose a similarity metric for weights or neurons. Studies have also been devoted to layer pruning as it promotes superior computational gains. However, ignoring other layers in the model can reduce its predictive ability (i.e., high compression rates). This work introduces an effective layer-pruning strategy that makes an underlying assumption about the relative importance of a layer using the Centered Kernel Alignment (CKA) metric. We show the relationship between the representations of the unpruned model and a candidate layer pruning. We compare our method with other state-of-the-art direct architectures and benchmarks, in which it outperforms existing layer-pruning strategies and achieves up to 95% of compression. Particularly, we notice that the 95% of compression while improving predictive ability. At higher compression ratios, our method exhibits negligible memory requirements and maintains high predictive model accuracy. Apart from these benefits, our pruned models exhibit robustness to adversarial and out-of-distribution samples.

1. Introduction

It is well known that deep neural networks are capable of obtaining remarkable results in various cognitive fields, obtaining better performance than simple baseline methods [1].

Yamamoto, Universidade de São Paulo. Correspondence to: Luis Pons. © 2023, the authors. Licensee MDPI, Basel, Switzerland.

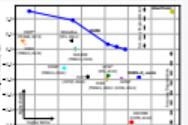


Figure 1. Comparison with state-of-the-art on the popular ResNet18 + CIFAR-10 setting. (Blue: for illustration purposes, no points (pp); however, it may be higher). Overall, our method obtains the best compromises between accuracy and computational cost. The proposed method is compared with three other approaches: (i) Direct pruning, which is a standard approach; (ii) Filter, our method; (iii) Filter+ (our improved layer pruning method, indicated by symbol “+”) by a remarkable margin. Compared to the state-of-the-art, our method obtains a reduction of 73.7% of FLOPs while keeping accuracy (sometimes improving it). Other methods, however, decrease accuracy when operating at these low FLOPs. (Left) Comparison of the number of layers (number of FLOPs) versus accuracy (percentage). (Right) Comparison of the number of layers (number of FLOPs) versus accuracy (percentage). The legend for the layers in the figure is consistent with other layers and architectures.

Pruning Everything, Everywhere, All at Once

Gustavo Henrique da Nascimento, Ian Pons, Anna Helena Reali Costa and Amar Jardim
Escola Politécnica, Universidade de São Paulo, Brazil

Abstract—Deep learning stands as the modern paradigm for solving complex tasks. However, as the problem size increases, models grow deeper and deeper, leading to significant overheads hindering advancements in real-world and resource-constrained applications. Therefore, pruning layers and removing structures from these models efficiently reduces model complexity and improves computational efficiency. However, pruning layers must include constraints (i.e., filters, neurons, or layers), but both together. Therefore, considering pruning different structures simultaneously is key. To this end, this work explores the benefits of eliminating neurons and layers at once, we propose a novel pruning strategy that combines both approaches within a model as filters. Given two candidate subnetworks (groups of neurons) for layer pruning and one other layer neuron pruning, this work proposes a new metric to select the layer with the highest representation similarity to its peers. After pruning the selected layer, this work performs a Centered Kernel Alignment (CKA) metric. Iteratively repeating this process until all layers are removed, we obtain the original predictive ability. Through extensive experiments we analyze the proposed strategy and highlight how our solution effectively optimizes the network. Additionally, we confirm the effectiveness of our approach and highlight how our solution outperforms the state-of-the-art in terms of accuracy and computational cost. (Left) Comparison of the number of layers (number of FLOPs) versus accuracy (percentage). (Right) Comparison of the number of layers (number of FLOPs) versus accuracy (percentage).

I. INTRODUCTION

Deep learning drives unprecedented progress in various complex domains [1]. However, this performance comes with high computational cost and storage demand. Advances in the field have led to the development of more efficient models trained on a broad range of data with the capacity to transfer its knowledge to unseen (demonstrative) tasks [2]. In this context, large models play a crucial role in transferring knowledge to downstream tasks (Bonmassari et al., 2021). The latter, namely structured neurons, often hardware-specific

non-performing neurons. From image recognition to complex games such as go (Silver et al., 2016; Hasselmo et al., 2023). However, this performance comes with high computational cost and storage demand. Advances in the field have led to the development of more efficient models trained on a broad range of data with the capacity to transfer its knowledge to unseen (demonstrative) tasks [2]. In this context, large models play a crucial role in transferring knowledge to downstream tasks (Bonmassari et al., 2021). The latter, namely structured neurons, often hardware-specific non-performing neurons. From image recognition to complex games such as go (Silver et al., 2016; Hasselmo et al., 2023). However, this performance comes with high computational cost and storage demand. Advances in the field have led to the development of more efficient models trained on a broad range of data with the capacity to transfer its knowledge to unseen (demonstrative) tasks [2]. In this context, large models play a crucial role in transferring knowledge to downstream tasks (Bonmassari et al., 2021). The latter, namely structured neurons, often hardware-specific

non-performing neurons. From image recognition to complex games such as go (Silver et al., 2016; Hasselmo et al., 2023). However, this performance comes with high computational cost and storage demand. Advances in the field have led to the development of more efficient models trained on a broad range of data with the capacity to transfer its knowledge to unseen (demonstrative) tasks [2]. In this context, large models play a crucial role in transferring knowledge to downstream tasks (Bonmassari et al., 2021). The latter, namely structured neurons, often hardware-specific

non-performing neurons. From image recognition to complex games such as go (Silver et al., 2016; Hasselmo et al., 2023). However, this performance comes with high computational cost and storage demand. Advances in the field have led to the development of more efficient models trained on a broad range of data with the capacity to transfer its knowledge to unseen (demonstrative) tasks [2]. In this context, large models play a crucial role in transferring knowledge to downstream tasks (Bonmassari et al., 2021). The latter, namely structured neurons, often hardware-specific

Efficient LLMs with AMP: Attention Heads and MLP Pruning

Leandro Giusti Mugnaini^{1*}, Bruno Bottino¹, Lucas Yamamoto², Lucas Leites de Alencar³, Victor Zucarini⁴, Edson Bozzo¹, Lucas Peñico¹, Anna Reali¹ and Amar Jardim¹
¹ Escola Politécnica de Engenharia, Universidade de São Paulo, São Paulo, Brazil
² Instituto de Ciência e Tecnologia Itaú (ICTI), São Paulo, Brazil

Abstract—Deep learning drives a new wave in computing research and triggers the need for more efficient models to solve problems. In particular, Large Language Models (LLMs) have revolutionized the way we interact with computers, enabling us to comprehend human-level performances. However, their computational cost is high, and they require significant resources to run, especially in edge devices. Therefore, the challenge is to make these models smaller without compromising quality (i.e., without compromising the user experience).

Among the strategies to overcome the aforementioned challenges, pruning is a promising technique that can reduce the model size while maintaining predictive ability. In this paper we introduce AMP: Attention Heads and MLP Pruning. LLMs are trained with attention heads and MLPs. AMP proposes a new approach for pruning layers and neurons. AMP also proposes a new approach for pruning attention heads. AMP filters layers but not neurons, but a naive strategy for obtaining high-layer compression is to eliminate deeper structures

involving layers achieves both superior prediction preservation and better efficiency despite reduced performance metrics such as perplexity and loss. In this work, we propose a new approach for pruning layers and neurons at once, we propose a new metric to select the layer with the highest representation similarity to its peers. This means that we can skip the first step and reaching a leaf directly. (Left) Comparison of the number of layers (number of FLOPs) versus accuracy (percentage). (Right) Comparison of the number of layers (number of FLOPs) versus accuracy (percentage).

Following the path to the left from the leaf (indicated by arrow) in Figure 1 (Left). Technically speaking, existing filtering approaches are not able to skip the first step. AMP proposes the following focus on eliminating individual neurons [25], [26]. AMP filters layers but not neurons, but a naive strategy for obtaining high-layer compression is to eliminate deeper structures

in short in flexibility or efficiency. In particular, AMP proposes the use of the set union operation for obtaining a 90% pruning rate with minimal impact on averaged task performance. Moreover, AMP proposes a new approach for pruning attention heads and MLPs deployed in resource-constrained environments. We confirm the utility of AMP on different families of LLMs, including LLaMA and LLaMA-2.

** Author to whom correspondence should be addressed. Code: <https://github.com/LGiustiMugnaini/PruningEverythingAllAtOnce>.*

L. INTRODUCTION

Within the evolving landscape of Artificial Intelligence, Large Language Models (LLMs) stand as a pivotal force, propelling breakthroughs in various applications. However, the demand for these models has led to significant challenges, particularly regarding their computational cost and storage demands. Among these challenges, one of the most pressing is the high computational cost associated with training and deploying these models, often necessitating specialized hardware and significant energy consumption. This high cost is particularly evident in edge devices, where power efficiency and cost are critical factors.

Regrettably, the current landscape of AI models is characterized by a lack of balance between performance and efficiency. While some models achieve remarkable performance, they often come at the expense of high computational costs and slow inference times. Conversely, other models sacrifice performance for lower costs, but they may lack the necessary features and capabilities required for certain applications.

Given these challenges, there is a clear need for more efficient and cost-effective AI models. One promising approach is to focus on pruning, which involves removing less important structures while maintaining the model's overall performance. This can be achieved through various techniques, such as layer pruning, neuron pruning, and head pruning. These methods aim to identify and remove components that have a minimal impact on the model's performance, thereby lowering computational overhead.

Recent studies confirm pruning as a promising solution to compress models as it maintains predictive ability and is often hardware-agnostic [9]–[11]. Within the field of LLMs, pruning

Pons et al. *Effective Layer Pruning Through Similarity Metric Perspective*. ICPR (Oral) and ICML Workshops, 2024

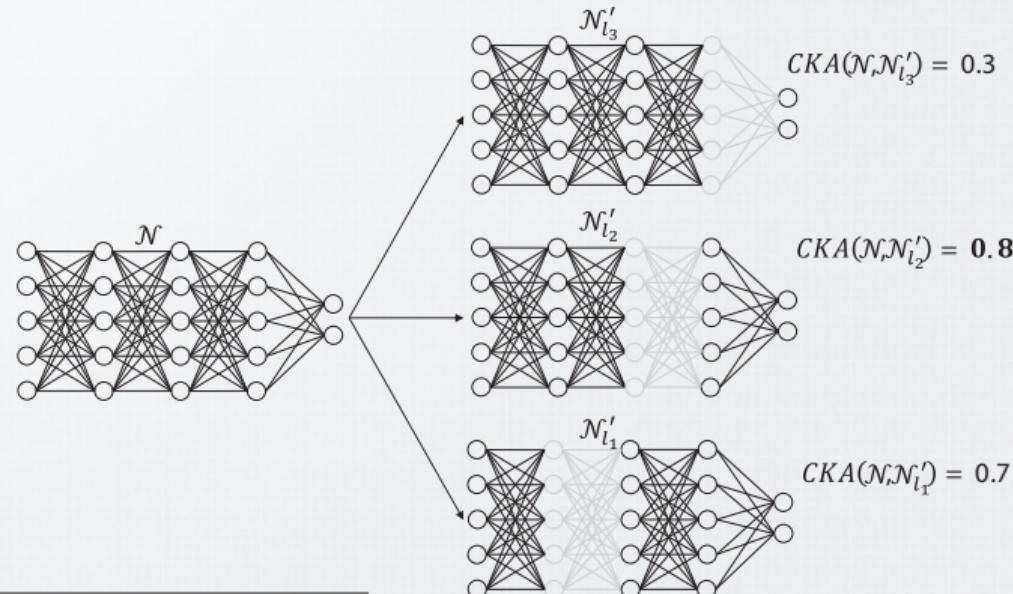
Nascimento et al. *Pruning Everything, Everywhere, All at Once*. IJCNN, 2025

Mugnaini et al. *Efficient LLMs with AMP: Attention Heads and MLP Pruning*. IJCNN, 2025

Compression of Deep Models

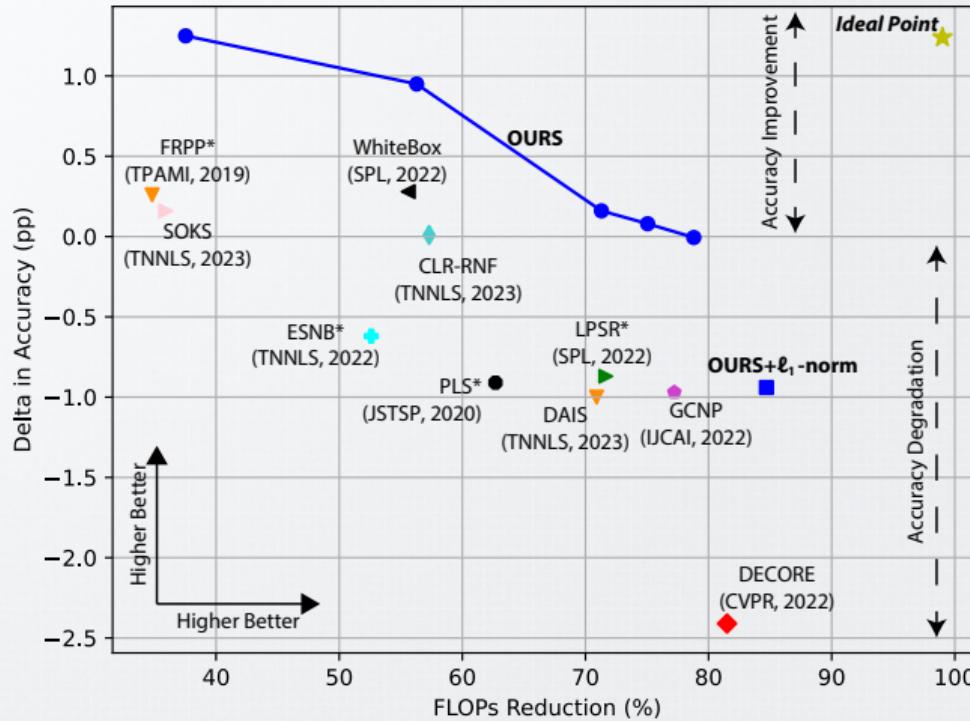
Research Projects

- We showed that similar representations between a dense (unpruned) network and its optimal pruning candidate indicate lower relative importance, thus capturing underlying properties exhibited by layers and preventing model collapse



Compression of Deep Models

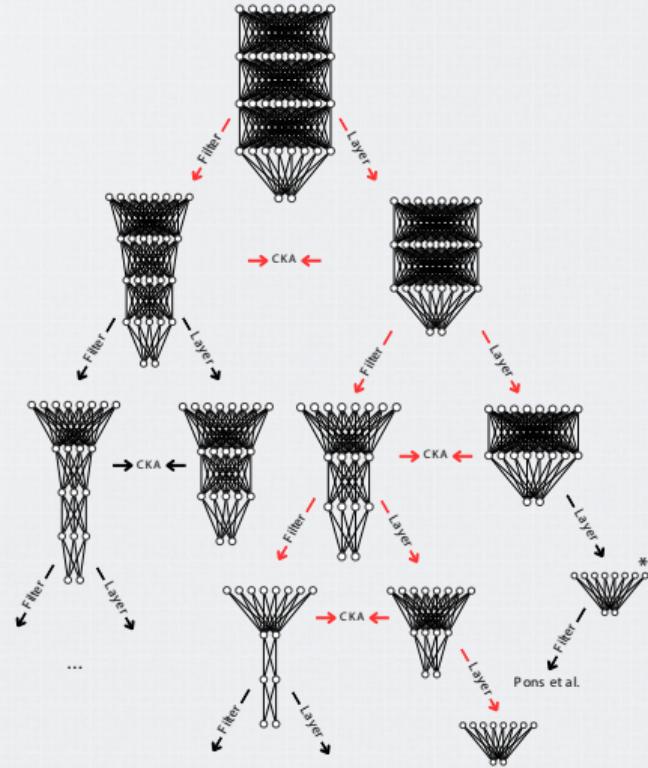
Research Projects



Compression of Deep Models

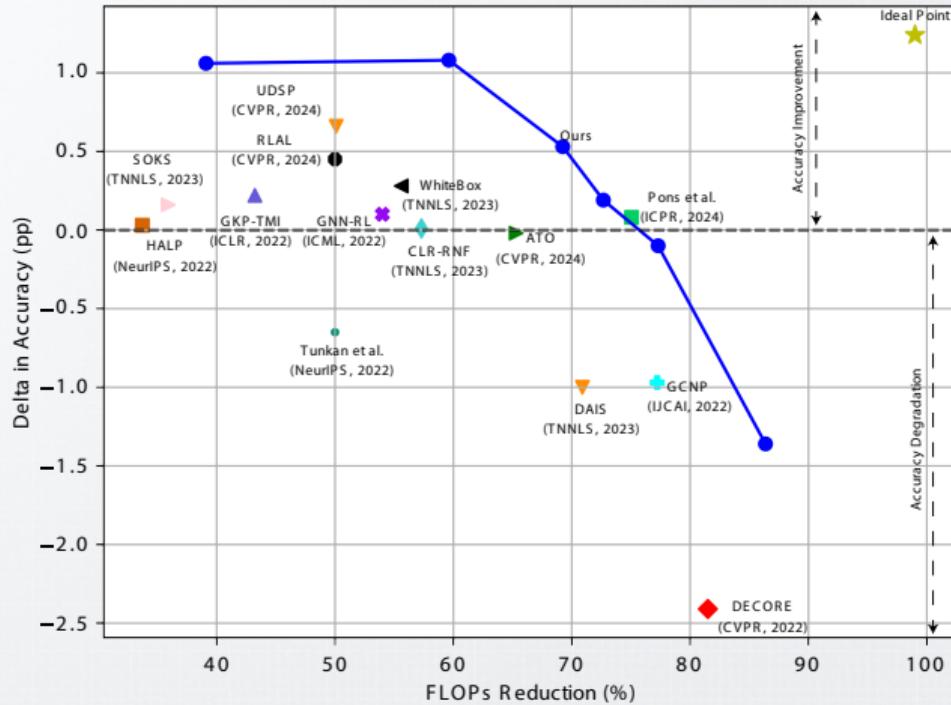
Research Projects

- Given two candidate subnetworks, one from layer pruning and the other from filter pruning, we can effectively decide which to choose by selecting the one with the highest representation similarity to its parent



Compression of Deep Models

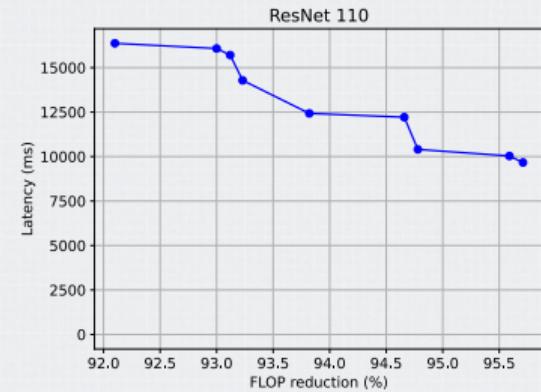
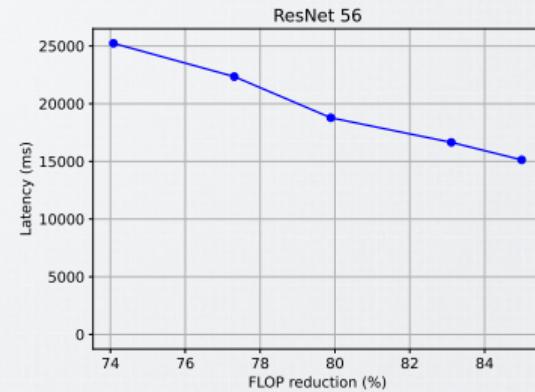
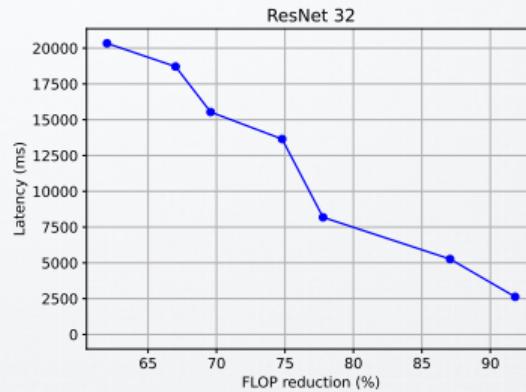
Research Projects



Compression of Deep Models

Research Projects

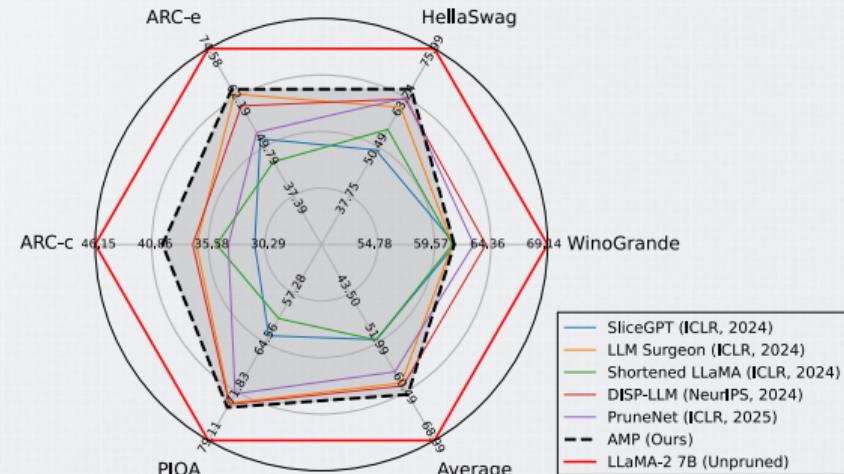
- Effectiveness on a resource-constrained device
 - ESP32



Compression of Deep Models

Research Projects

- We effectively estimate the importance of internal components in LLMs by projecting input data onto the weights, thereby quantifying their importance to the final representation. Consequently, removing components with the lowest contributions results in models that are lighter, faster, and maintain their predictive capacity



Compression of Deep Models

Research Projects

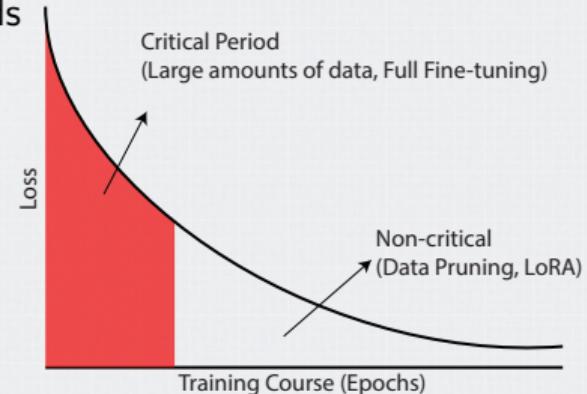
- Short demo: <https://github.com/c2d-usp/Efficient-LLMs-with-AMP>

Compression Ratio	Method	WinoGrande	HellaSwag	ARC-e	ARC-c	PIQA	Avg. ↑
0%	LLaMA 7B	69.85	76.21	72.81	44.71	79.16	68.55
	LLM Pruner (NeurIPS, 2023)	61.33	65.34	59.18	37.12	75.57	59.71
	LLM Pruner (+ fine-tuning) (NeurIPS, 2023)	65.11	68.11	63.43	37.88	76.44	62.19
	Shortened LLaMA (ICLR, 2024)	68.82	69.82	64.06	39.93	74.65	63.46
	DISP-LLM (NeurIPS, 2024)	64.72	68.39	64.81	37.12	76.66	62.34
	PruneNet (ICLR, 2025)	62.12	65.40	64.65	36.52	75.51	60.82
20%	AMP (Ours)	63.93	70.34	69.82	41.38	77.15	64.52
	LLaMA-2 7B	69.14	75.99	74.58	46.15	79.11	68.99
	SliceGPT (ICLR, 2024)	61.33	49.62	51.77	31.23	63.55	51.50
	LLM Surgeon (ICLR, 2024)	61.09	60.72	63.09	36.69	73.56	59.03
	Shortened LLaMA (ICLR, 2024) (‡)	61.09	54.97	45.96	34.81	60.99	51.56
	DISP-LLM (NeurIPS, 2024)	63.93	62.87	60.1	37.03	73.72	59.53
30%	PruneNet (ICLR, 2025)	61.09	58.30	53.20	32.94	71.11	55.33
	PruneNet (+ fine-tuning) (ICLR, 2025)	62.90	63.21	53.37	33.70	72.20	57.08
	AMP (Ours)	61.25	65.47	64.31	39.85	74.21	61.02

Efficient Training – Critical Periods

Research Projects

- Critical Learning Periods comprise an important phenomenon involving deep learning¹²
 - Early epochs play a decisive role in the success of many training recipes
- *How to identify critical periods during the course of training?*
 - Given calls for more efficient training in the era of foundation models, answering this question yields notable gains



¹Golatkar et al. *Time Matters in Regularizing Deep Networks: Weight Decay and Data Augmentation Affect Early Learning Dynamics, Matter Little Near Convergence*. NeurIPS, 2019

²Achille et al. *Critical Learning Periods in Deep Networks*. ICLR, 2019

Efficient Training – Critical Periods

Research Projects

- *Throughout the course of training, a simple generalization estimation enables successful identifying when the critical period emerges*

ONE PERIOD TO RULE THEM ALL: IDENTIFYING CRITICAL LEARNING PERIODS IN DEEP NETWORKS

Vinicius Yutti Fukase^{*1}, Heitor Gama^{*1}, Barbara Bueno¹,
Lucas Libanio¹, Anna Helena Reali Costa¹, and Artur Jordao¹

¹Escola Politécnica, Universidade de São Paulo, Brazil

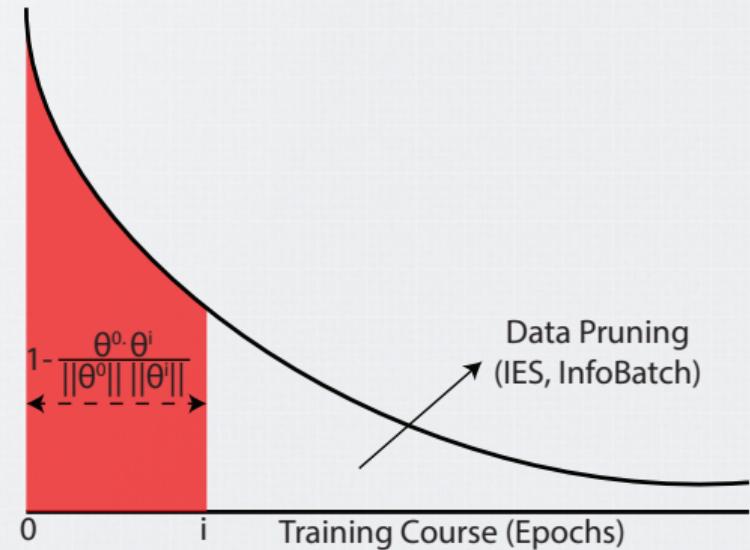
ABSTRACT

Critical Learning Periods comprehend an important phenomenon involving deep learning, where early epochs play a decisive role in the success of many training recipes, such as data augmentation. Existing works confirm the existence of this phenomenon and provide useful insights. However, the literature lacks efforts to precisely identify when critical periods occur. In this work, we fill this gap by introducing a systematic approach for identifying critical periods during the training of deep neural networks, focusing on eliminating computationally intensive regularization techniques and effectively applying mechanisms for reducing computational costs, such as data pruning. Our method leverages generalization prediction mechanisms to pinpoint critical phases where training recipes yield maximum benefits to the predictive ability of models. By halting resource-intensive recipes beyond these periods, we significantly accelerate the learning phase and achieve reductions in training time, energy consumption, and CO₂ emissions. Experiments on standard architectures and benchmarks confirm the effectiveness of our method. Specifically, we achieve significant milestones by reducing the training time of popular architectures by up to 59.67%, leading to a 59.47% decrease in CO₂ emissions and a 60% reduction in financial costs, without compromising performance. Our work enhances understanding of training dynamics and paves the way for more sustainable and efficient deep learning practices, particularly in resource-constrained environments. In the era of the race for foundation models, we believe our method emerges as a valuable framework. The repository is available at <https://github.com/bauinihamarga/critical-periods>.

Efficient Training – Critical Periods

Research Projects

- Layer rotation¹: $1 - \frac{\theta^0 \cdot \theta^i}{\|\theta^0\| \|\theta^i\|}$
 - θ^0 indicates the initially randomized weights
 - θ^i indicates weights at epoch i



¹Carbonnelle et al. *Layer rotation: a surprisingly simple indicator of generalization in deep networks?* ICML, 2019

Efficient Training – Critical Periods

Research Projects

- Setup
 - Original: Data pruning method by Yuan et al.¹ (without critical period)
 - Ours: Data pruning after critical period
 - Baseline: Standard training using full data

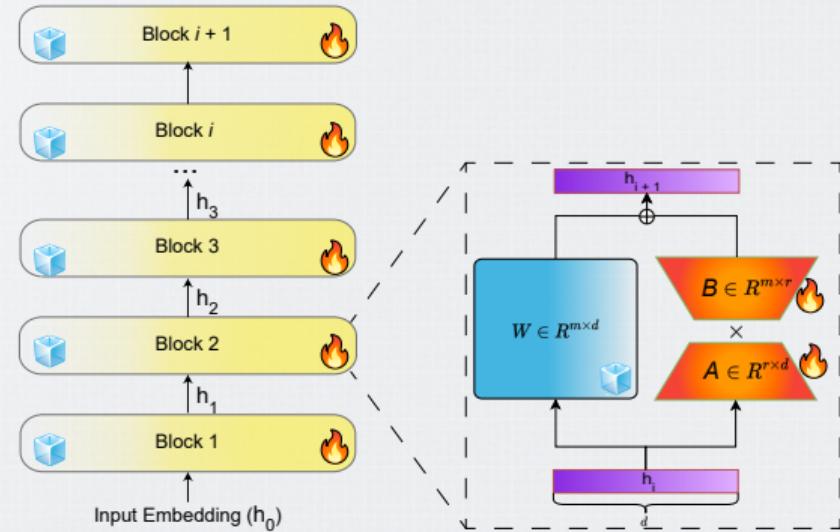
Dataset	CIFAR-10			CIFAR-100		
Architecture	ResNet18	ResNet50	VGG16	ResNet34	ResNet101	DenseNet121
IES ¹	0.9469	0.9464	0.9312	0.7747	<u>0.7799</u>	0.7912
	<u>0.9489</u>	<u>0.9474</u>	0.9331	0.7754	0.7788	0.7930
	0.9504	0.9488	0.9329	0.7752	0.7815	0.7907

¹Yuan et al. *Instance-dependent Early Stopping* ICLR, 2025

Parameter Efficient Fine-Tuning

Research Projects

- LoRA¹ constitutes an important technique in the age of large models
 - It allows the exploration of the capacity of these models for specialized tasks with efficient fine-tuning
 - $W + \Delta W \approx W + BA$
- Are all LoRA modules necessary?

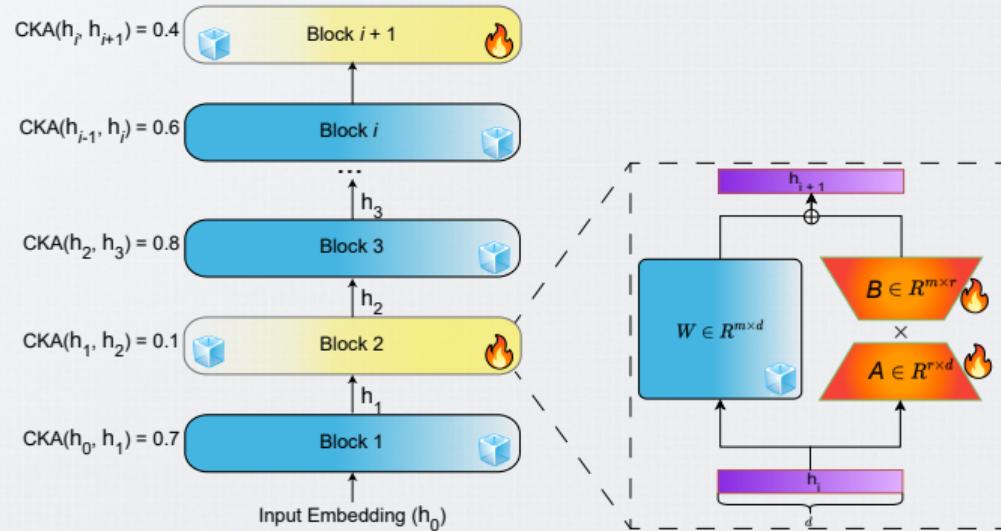


¹Hu et al. LoRA: Low-Rank Adaptation of Large Language Models. ICLR, 2022

Parameter Efficient Fine-Tuning

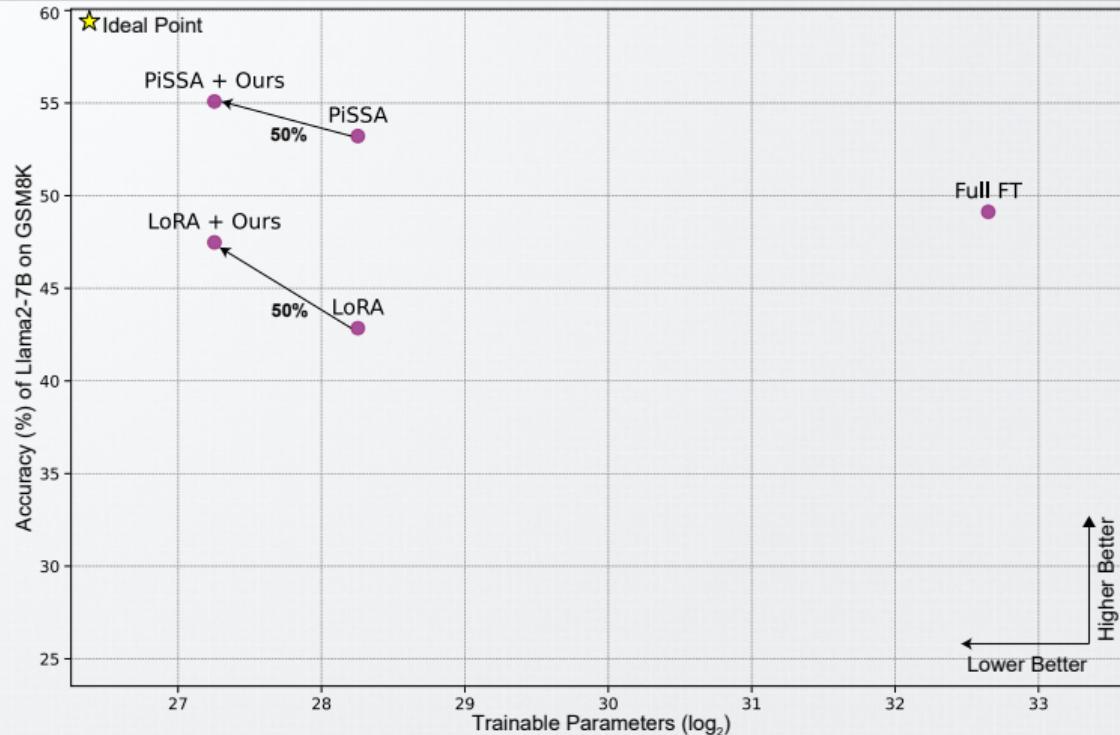
Research Projects

- Given a LLM and a task, we can effectively choose a subset of n transformer blocks to fine-tune by measuring their participation on the construction of internal representations. We quantify this relevance to the task by assessing the similarity between the input and output representations of a block



Parameter Efficient Fine-Tuning

Research Projects



¹Hu et al. LoRA: Low-Rank Adaptation of Large Language Models. ICLR, 2022

²Meng et al. PiSSA: Principal Singular Values and Singular Vectors Adaptation of Large Language Models. NeurIPS, 2024

Parameter Efficient Fine-Tuning

Research Projects

Model	Params	Strategy	GSM8K	MATH	HumanEval	MBPP
LLaMA 2-7B	6738M	Full FT	49.13±0.21	7.29±0.22	21.20±0.30	35.59±0.25
	320M	LoRA	42.85±0.12	5.50±0.33	18.35±0.31	35.50±0.14
	320M	PiSSA	53.22±0.55	7.47±0.34	21.92±0.38	37.24±0.63
	159M	LoRA + Ours	47.48±1.29	5.90±0.25	22.77±1.56	34.26±2.23
	159M	PiSSA + Ours	55.09±1.14	8.28±0.33	24.60±0.69	40.86±1.50
Mistral-7B	7240M	Full FT	69.91±0.25	18.64±0.35	45.31±0.14	51.46±0.13
	335M	LoRA	69.50±0.42	20.08±0.20	43.78±0.34	58.46±0.37
	335M	PiSSA	73.31±0.23	23.12±0.52	46.88±0.25	62.55±0.58
	167M	LoRA + Ours	72.63±0.08	20.69±0.13	50.00±1.58	62.43±0.55
	167M	PiSSA + Ours	73.29±0.84	21.56±0.23	50.20±4.09	60.57±1.59
Gemma-7B	8540M	Full FT	72.09±0.32	22.71±0.34	47.02±0.27	55.67±0.50
	400M	LoRA	75.11±0.64	30.41±0.48	53.70±0.23	65.58±0.29
	400M	PiSSA	77.78±0.32	31.33±0.33	54.31±0.28	66.17±0.43
	200M	LoRA + Ours	73.54±0.66	28.39±0.43	49.80±2.50	65.00±0.66
	200M	PiSSA + Ours	73.76±0.50	27.38±0.53	51.03±3.67	61.80±0.53