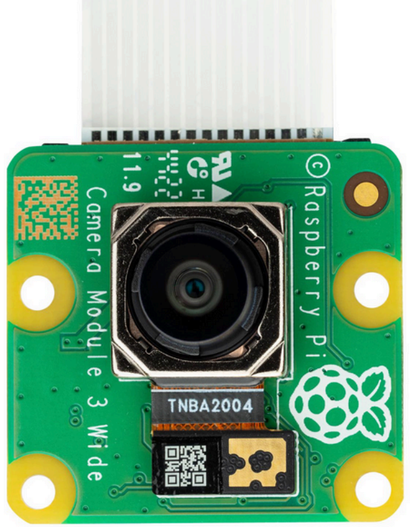
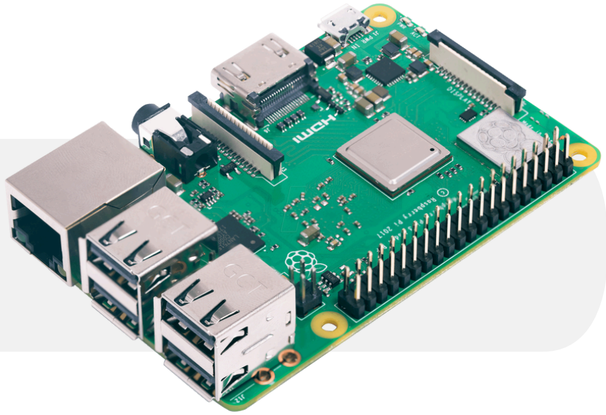
Mechanistic Interpretability for Vision Models Optimization

Andrea Petruzzi 1982874
Elisabetta Russo 1986112
Niccolò Buonfiglio 1983405

Computer Vision
A.Y. 2024-2025



Problem Statement



Goal

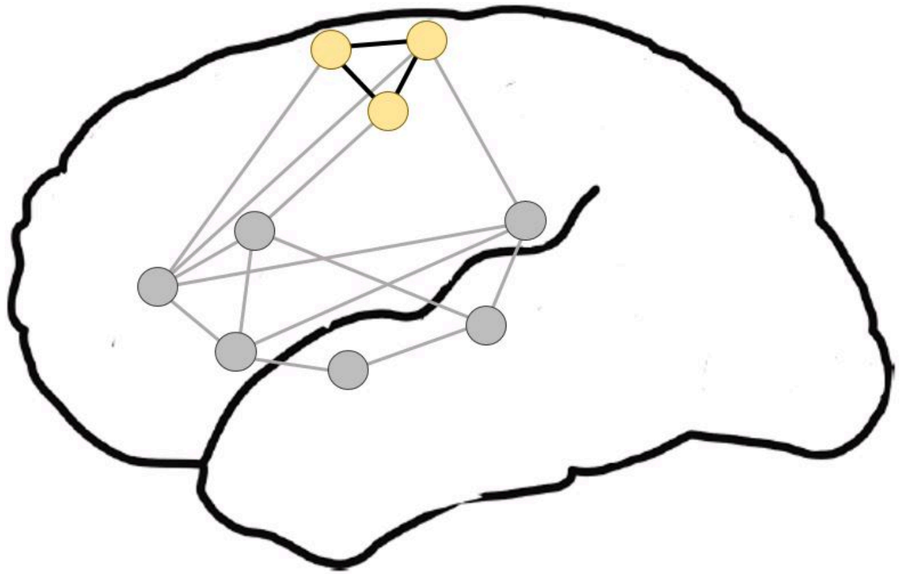
Improve the efficiency of Vision Transformers while preserving their accuracy

Approach

Apply mechanistic interpretability to identify and remove non-essential model components

State of the Art

Interpretability research aims to uncover internal neural network circuits that drive specific behaviors.
Mechanistic interpretability research aims to automate the process, allowing to find more and larger circuits.



State of the Art

First approach

Automated Circuit DisCovery (ACDC) finds these circuits via activation patching. It can be computationally costly, depending on the level of granularity it is applied on.

Algorithm 1: The ACDC algorithm.

Data: Computational graph G, dataset $(x_i)_{i=1}^n$, corrupted datapoints $(x_i')_{i=1}^n$ and threshold $\tau > 0$.

Result: Subgraph $H \subseteq G$.

$H \leftarrow G$ // Initialize H to the full computational graph

$H \leftarrow H.reverse\_topological\_sort()$ // Sort H so output first

for $v \in H$ **do**

  for w *parent of* v **do**

    $H_{\text{new}} \leftarrow H \setminus \{w \rightarrow v\}$ // Temporarily remove candidate edge

    if $D_{KL}(G||H_{\text{new}}) - D_{KL}(G||H) < \tau$ **then**

      $H \leftarrow H_{\text{new}}$ // Edge is unimportant, remove permanently

    end

  end

end

return H

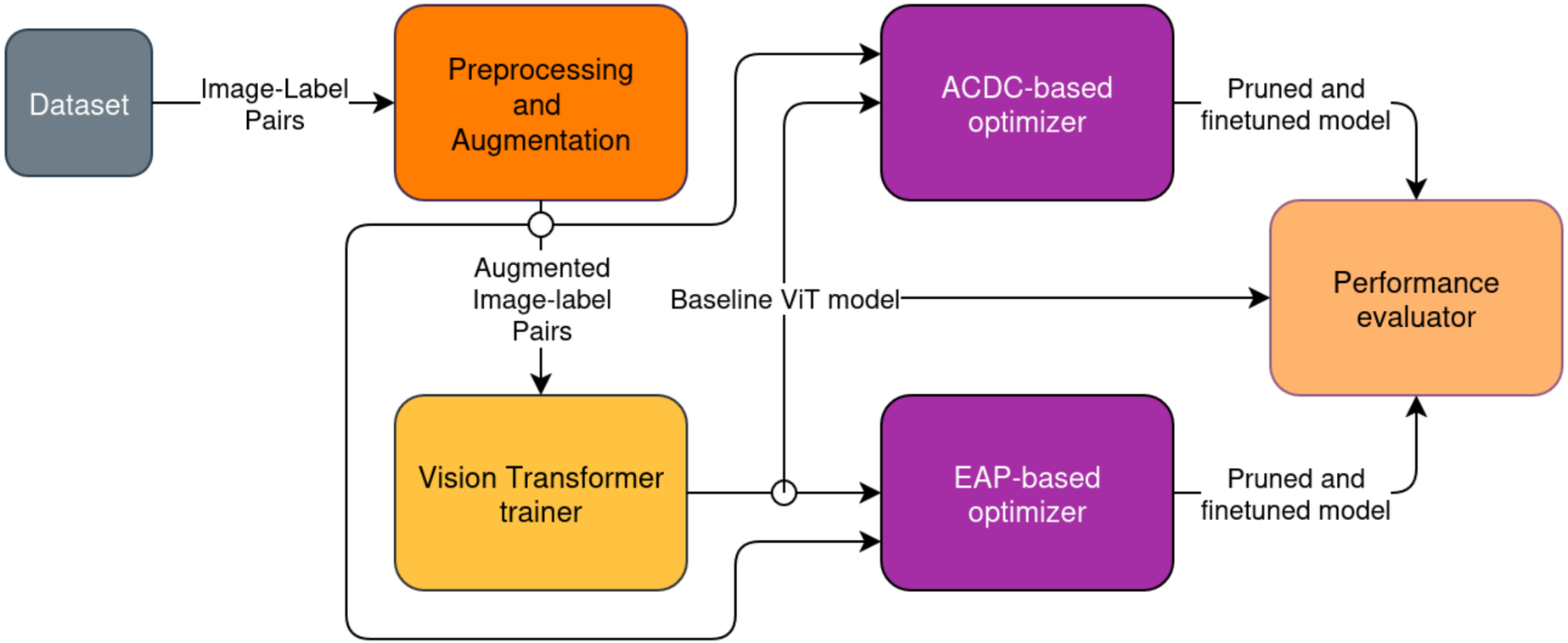
State of the Art

Newer approach

Edge Attribution Patching (EAP) is a faster, gradient-based approach requiring just two forward passes and one backward pass.

$$L(x_{\text{clean}} | \operatorname{do}(E = e_{\text{corr}})) \approx L(x_{\text{clean}}) + \underbrace{(e_{\text{corr}} - e_{\text{clean}})^\top \frac{\partial}{\partial e_{\text{clean}}} L(x_{\text{clean}} | \operatorname{do}(E = e_{\text{clean}}))}_{\text{Call this } \Delta_e L, \text{ the } \textbf{attribution score}.}.$$

Proposed method



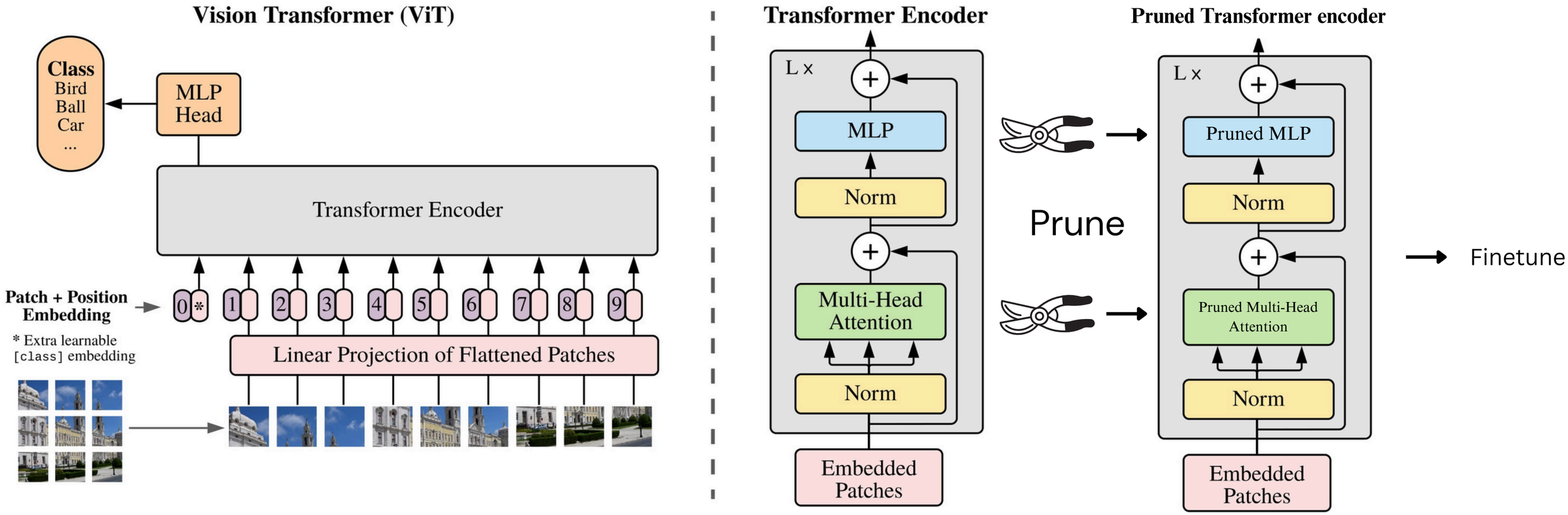
Dataset

Tiny ImageNet: a compact version of ImageNet

Size & Content:

- 100,000 images across 200 classes
- 64×64 resolution, colored (RGB)

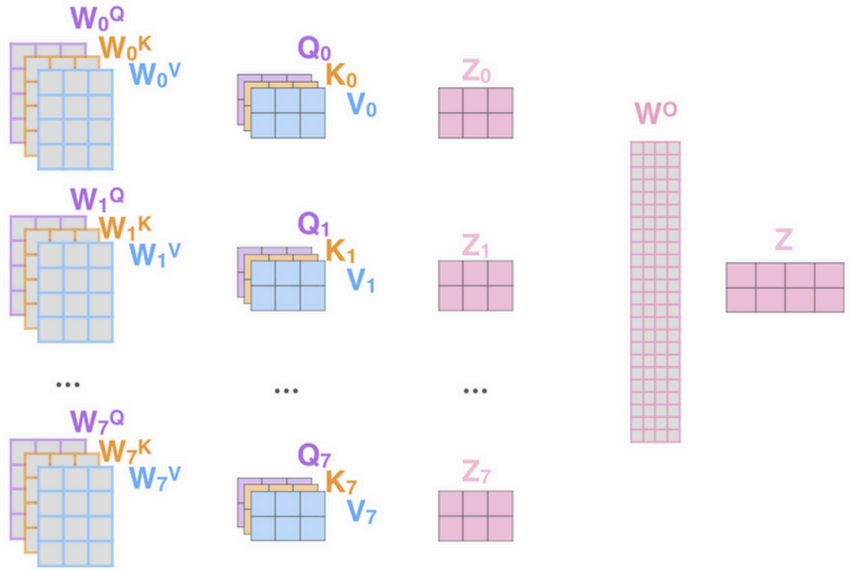
Experimental Setup



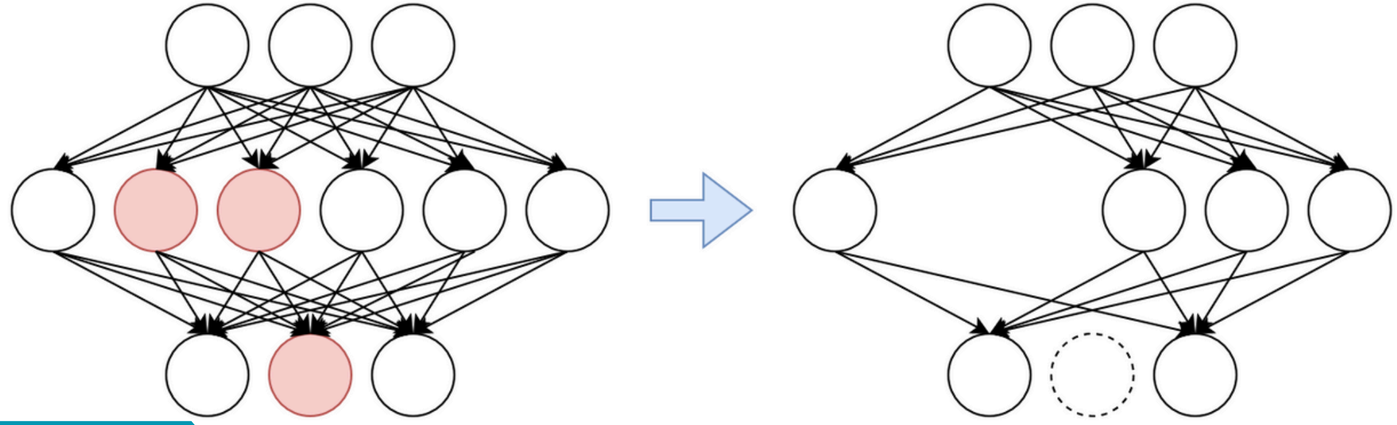
- **ACDC pruning:** corrupt attention heads and neuron chunks, keep only those that negatively affect the KL divergence above a certain threshold
- **EAP pruning:** corrupt attention heads and neuron chunks, evaluate their score, then keep only the top performers

Experimental Setup

The pruned multi-head attention is formed by pruning entire attention heads, and computing smaller QKV matrices.

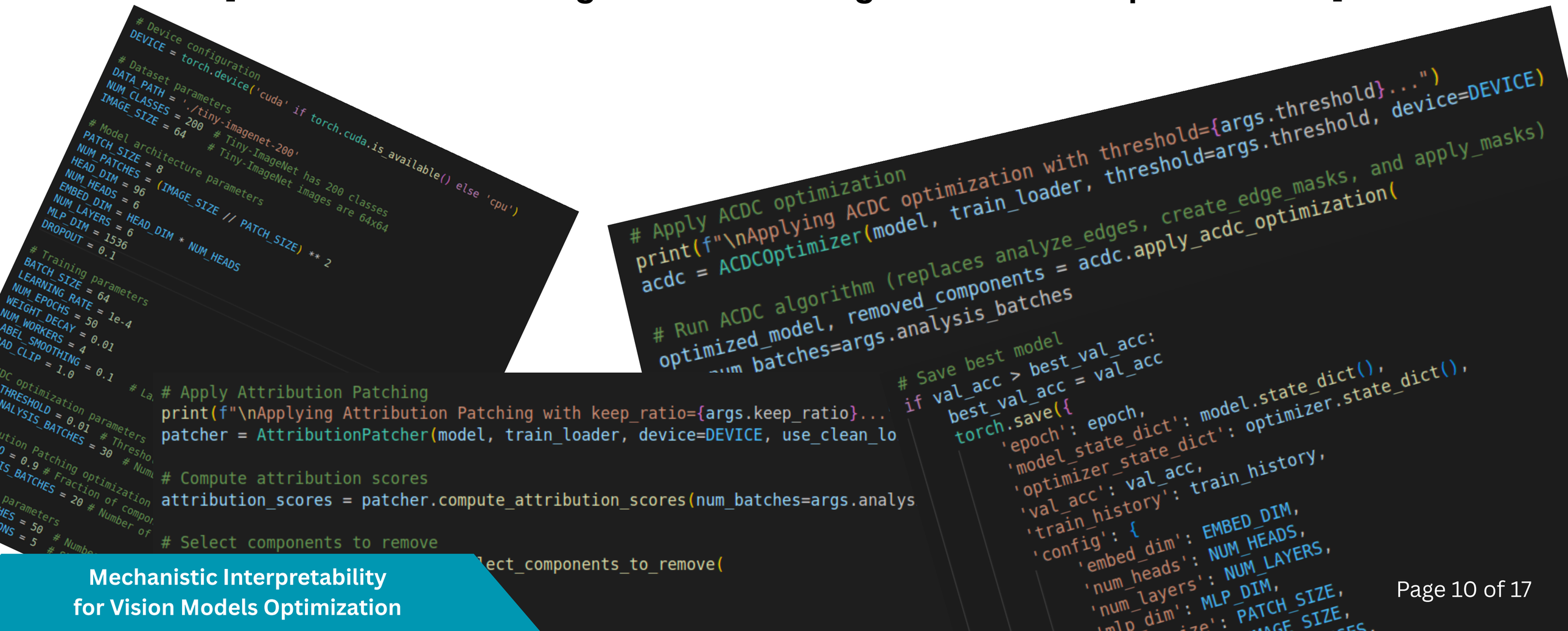


Pruned MLP have entire nodes removed. If output nodes are removed, their missing response gets “padded” with zeroes to keep the original embedding dimensions for subsequent layers.



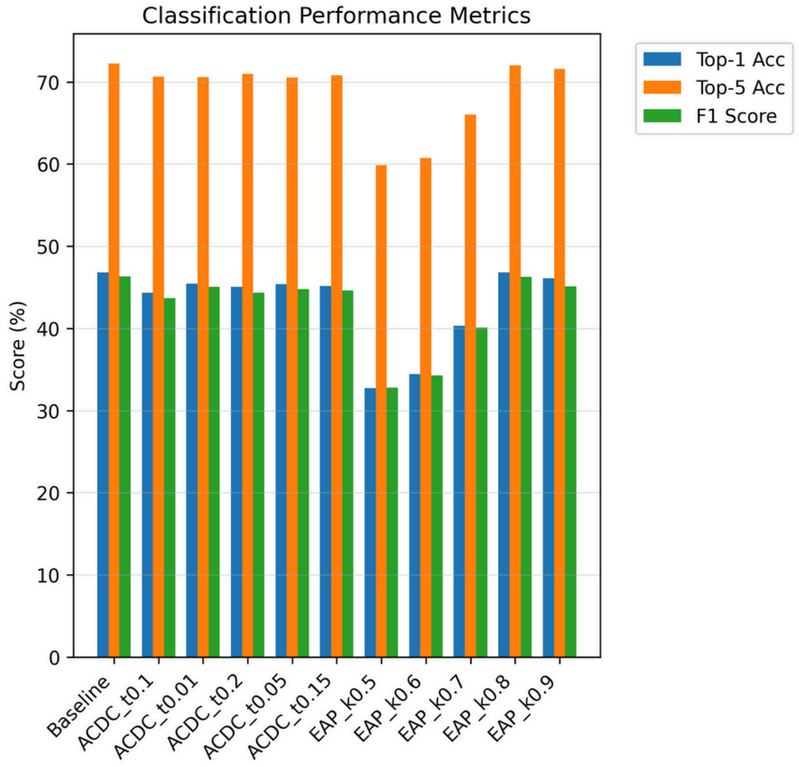
Experimental Setup

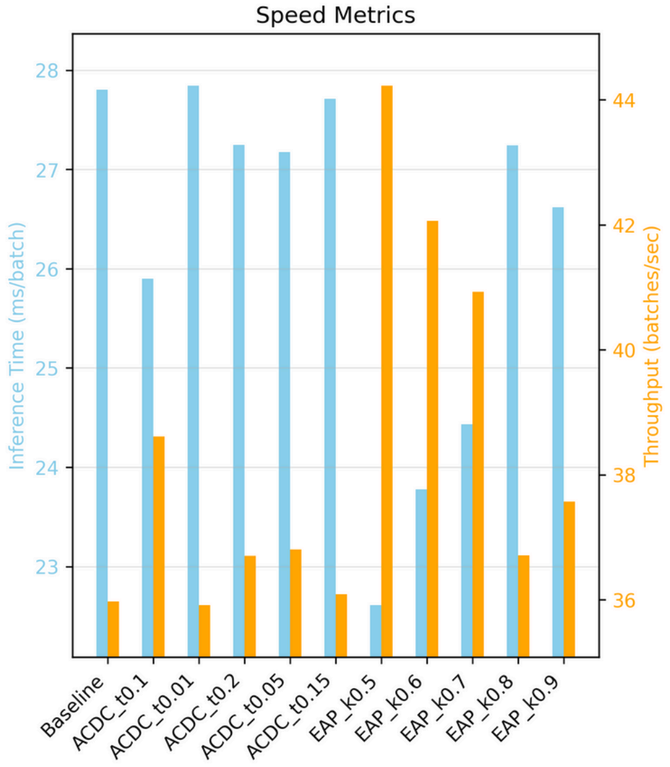
[Insert lots of training and and testing with different parameters]

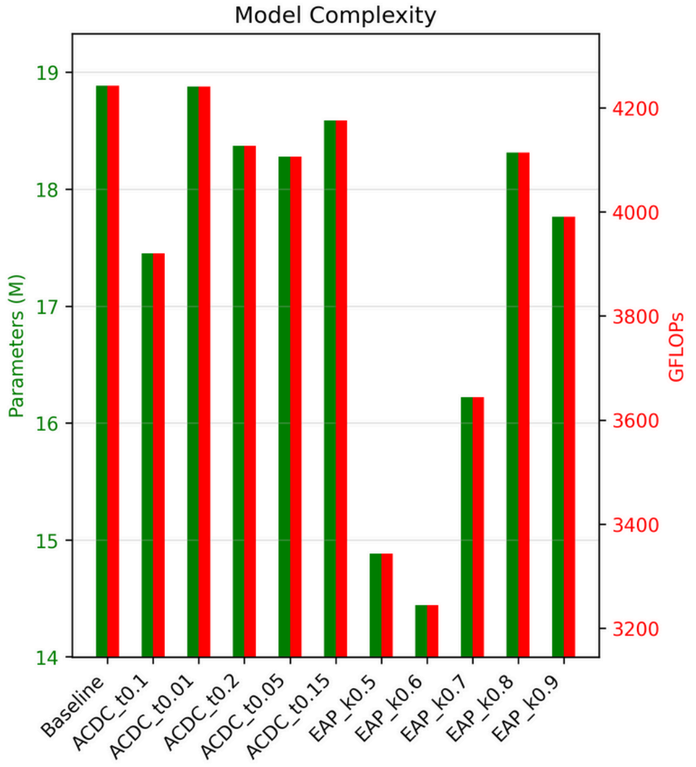


Model Evaluation

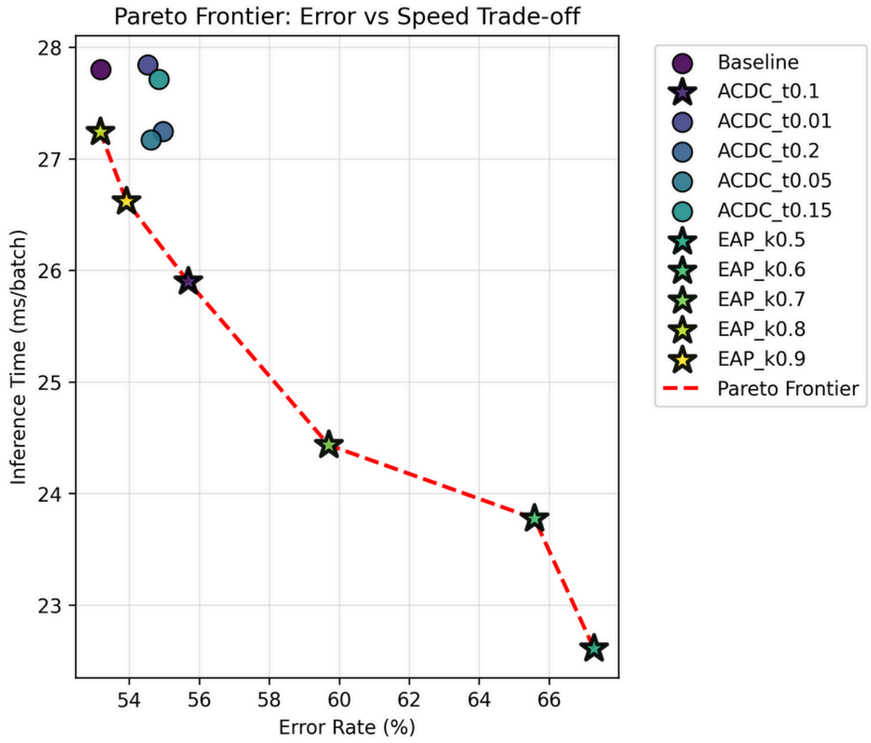
Typical results for a sample evaluation:

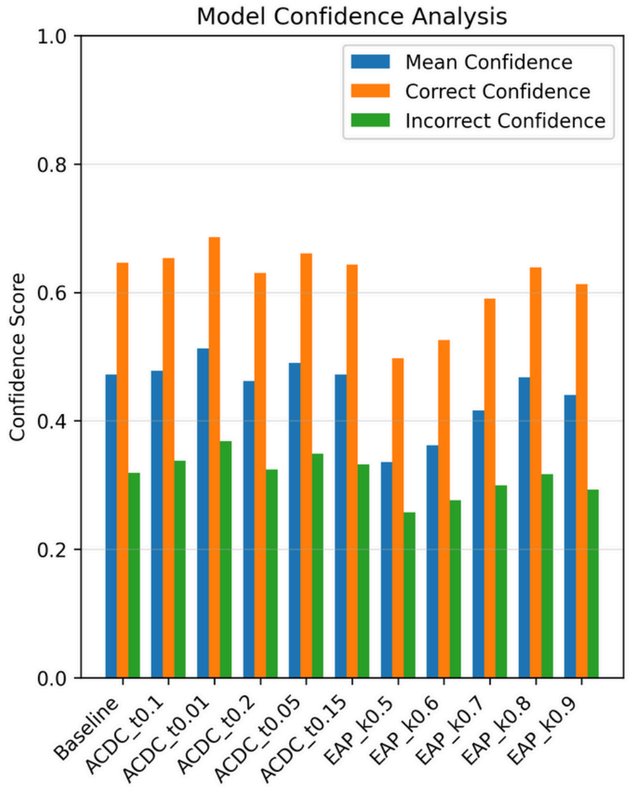


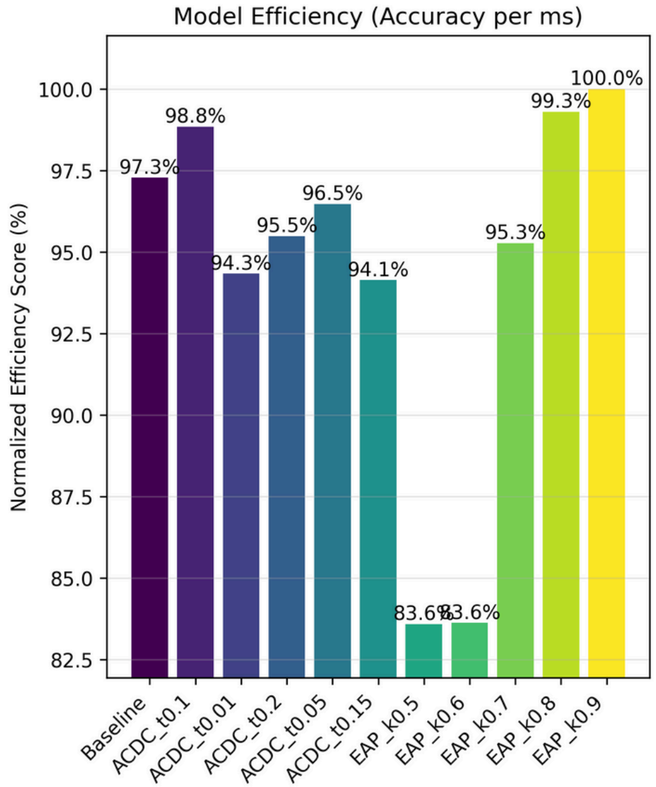




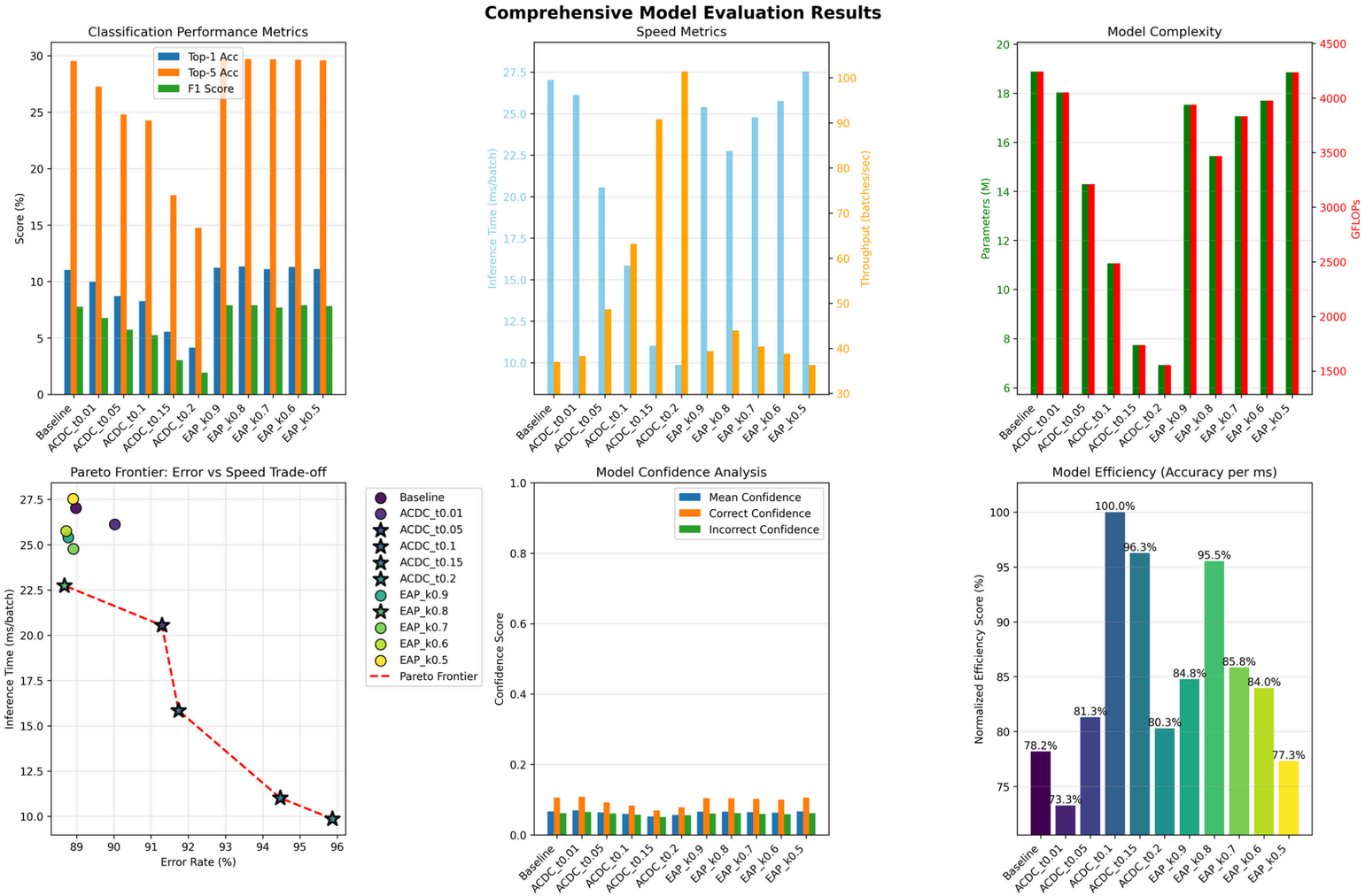
Model Evaluation







Model Evaluation



Among the different ViT sizes, hyperparameters and optimization parameters tested, the **only** scenarios where ACDC managed to beat EAP, was with undertrained (badly performing) starting baseline models

Conclusions

Proposed approach benefits:

- Enables smaller, faster networks while maintaining accuracy
- Consistently pushes the Pareto frontier (inference time vs. error rate)
- Reduces parameters, making models better for edge devices

EAP consistently outperforms ACDC in optimization time and optimized model results.

ACDC greatly benefits from the fine-tuning steps, regaining a lot of final model performance, allowing it to be competitive with EAP in some occasions.

Future work

Future work could explore:

- Testing on larger base models for broader applicability
- Exploring more granular patching
- Dynamic pruning: automatically find the minimal set of circuits needed to preserve model performance, optimize pruning by targeting specific output thresholds

References:

1 A. Conmy et al. (2023). *Towards Automated Circuit Discovery for Mechanistic Interpretability*. In: Advances in Neural Information Processing Systems 36 (NeurIPS 2023)

2 A. Syed, C. Rager and A.Conmy, (2024). *Attribution Patching Outperforms Automated Circuit Discovery*, BlackboxNLP 2024.

3 A. Amangeldi, A. Taigonyrov, M. H. Jawad, C. E. Mbonu (2025) CNN and ViT Efficiency Study on Tiny ImageNet and DermaMNIST Datasets Available at: https://arxiv.org/pdf/2505.08259

References:

4 C. Schiavella, L. Cirillo, L. Papa, P. Russo, and I. Amerini, (2023). Optimize vision transformer architecture via efficient attention modules: a study on the monocular depth estimation task.

5 PyTorch Documentation. (2024). *PyTorch: Tensors and Dynamic neural networks in Python with strong GPU acceleration*. Available at: https://pytorch.org/docs/.