CATransformers:

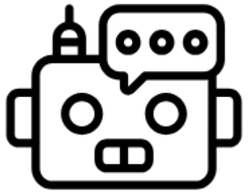
Carbon Aware Transformers Through Joint Model-Hardware Optimization

Irene Wang, Mostafa Elhoushi, H. Ekin Sumbul, Samuel Hsia, Daniel Jiang,
Newsha Ardalani, Divya Mahajan, Carole-Jean Wu, Bilge Acun

Meta AI Georgia Institute of Technology

Motivation

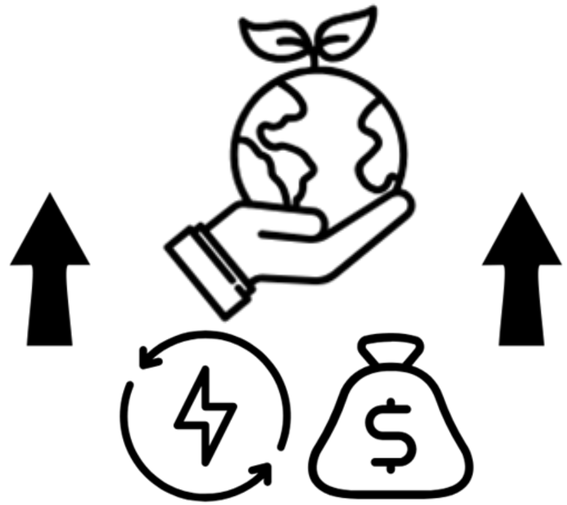
Rapid adoption of Machine learning solutions increases the associate lifecycle carbon footprint.



AI Assistant

Autonomous Vehicles

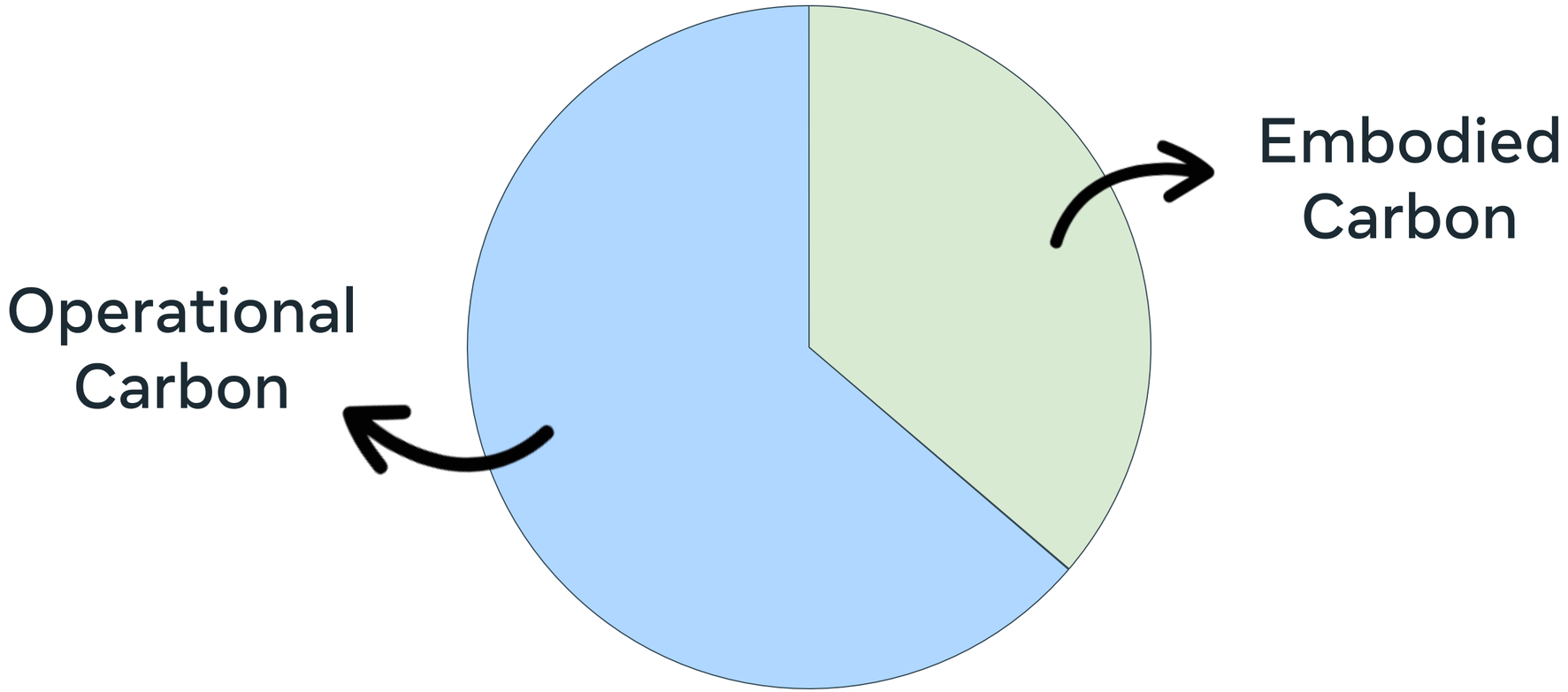
AR/VR Devices



High Operational Energy and Carbon Emissions

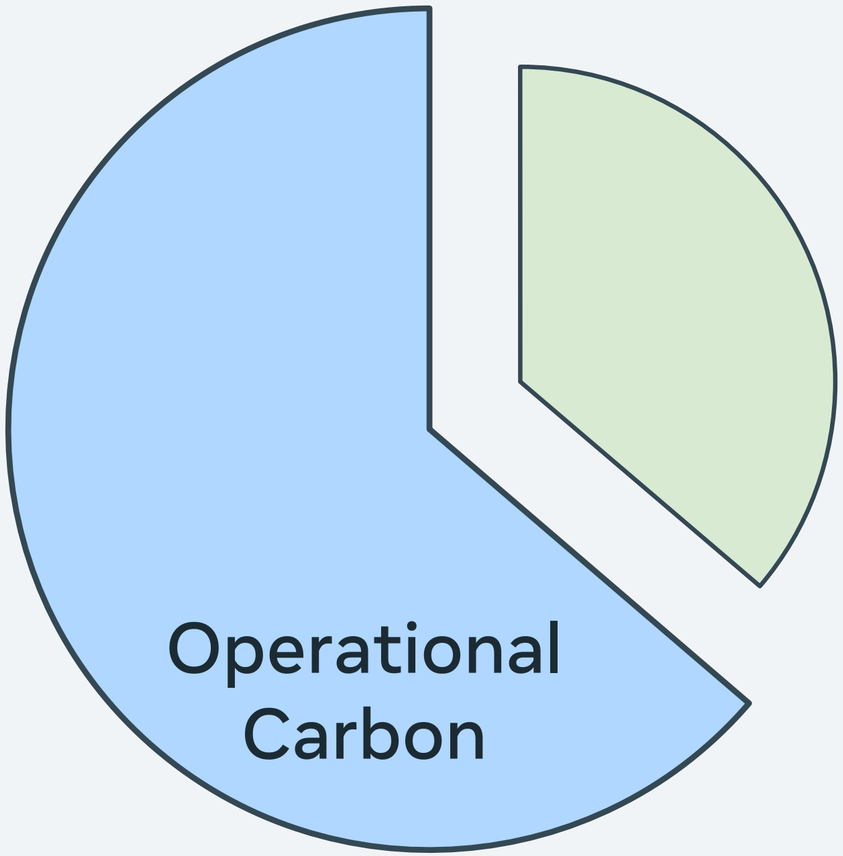
Motivation

The **Total Carbon Footprint** of these applications can be categorized into two forms:



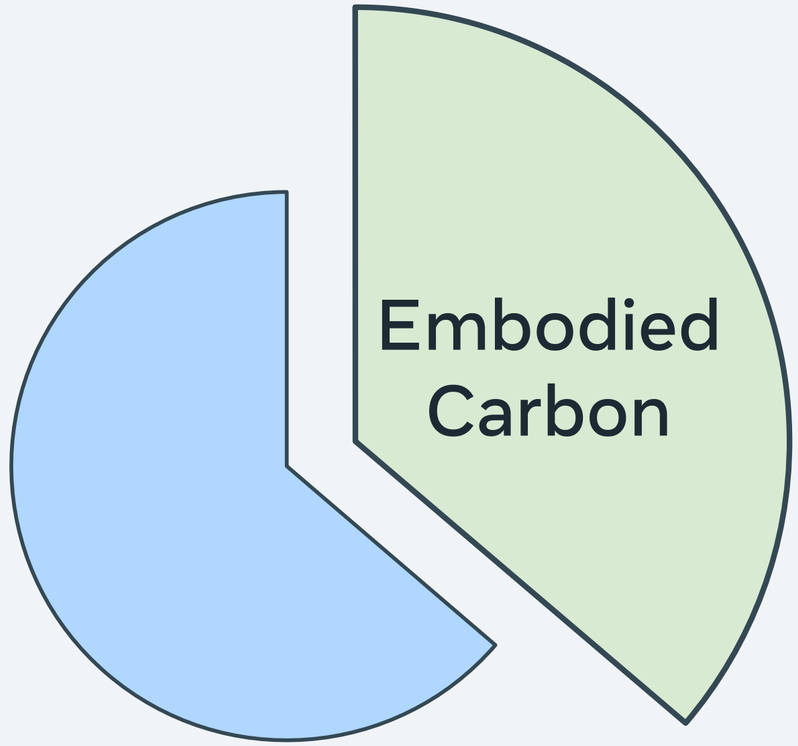
Operational Carbon

- Associated to the Electricity consume during Training and Inference
- Influenced by the location execution and energy source



Embodied Carbon

- Carbon Footprint from Manufacturing
- Influenced by the chip area and fabrication technology

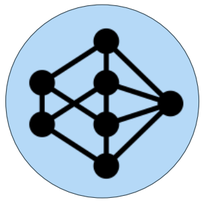


Carbon ≠ Single Dimension

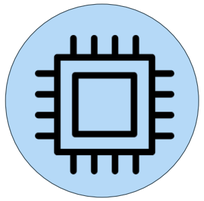
Total Carbon couples operational (latency, energy) and embodied (chip area) impacts. Achieving true sustainability requires joint optimization, not isolated trade-offs.

The Core Challenge

Model architecture and hardware design are tightly coupled.



Model architecture → drives *runtime behavior* → impacts operational carbon



Hardware design → determines *physical footprint* → impacts embodied carbon

These dimensions often conflict:
A larger, more complex chip can reduce runtime latency → higher embodied carbon, and increase power consumpiton → Impact on operational carbon is uncertain.

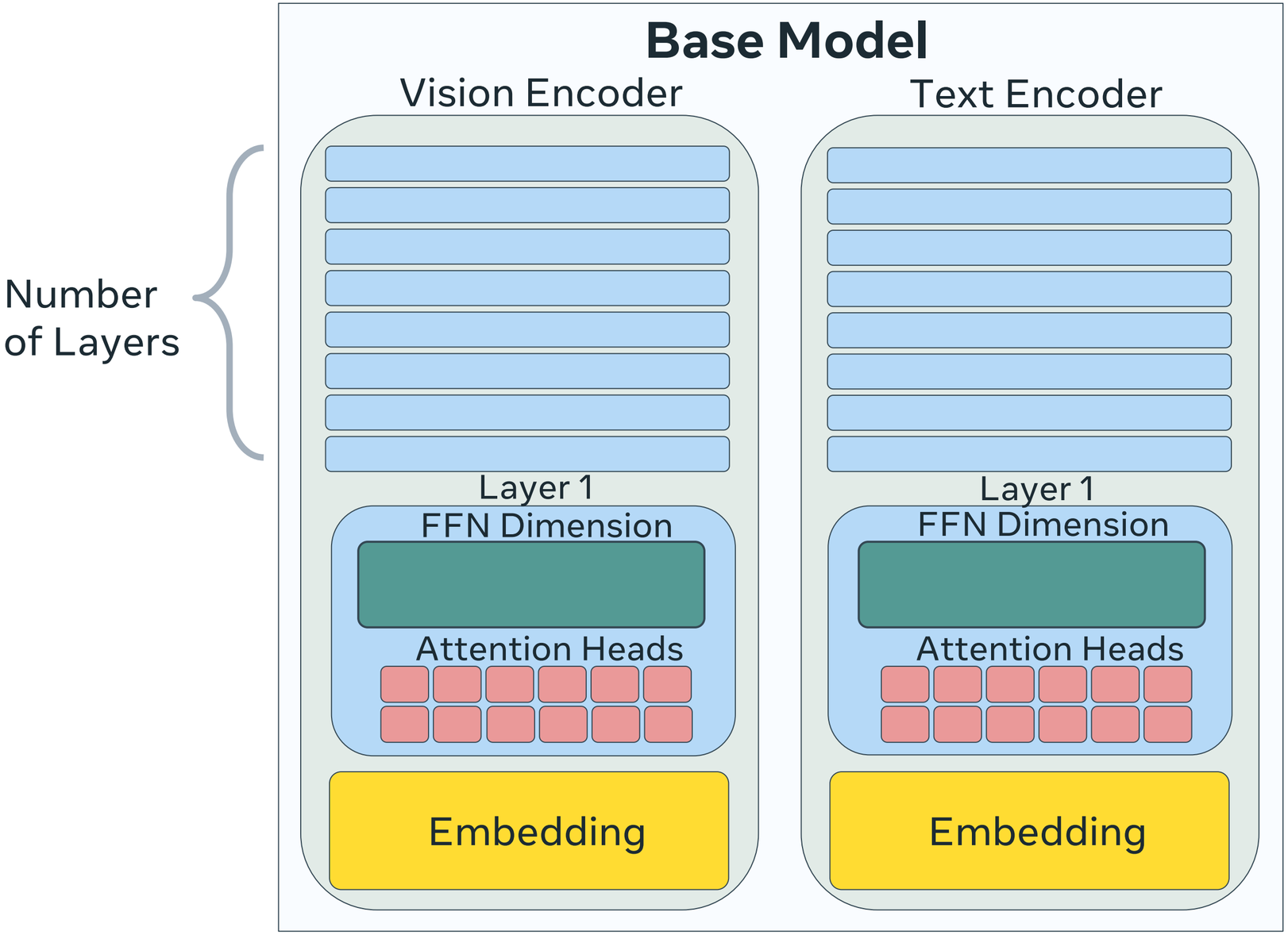
Our Approach: Joint HW-Model Design

Carbon optimization requires joint model-hardware exploration, particularly during early accelerator design, to uncover opportunities that align model demands with hardware capabilities in a carbon-efficient way.

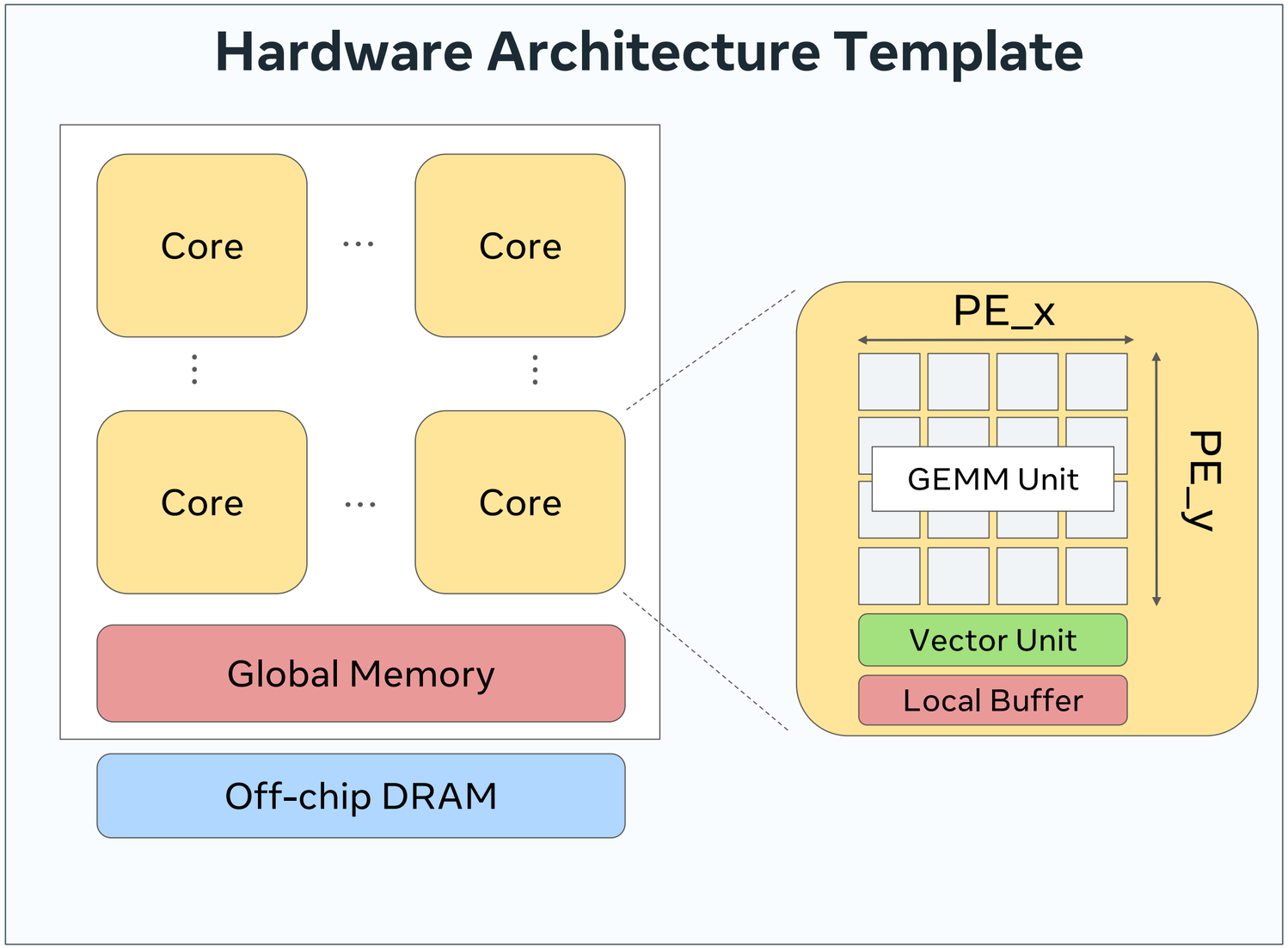
CATransformers

The first framework to co-optimize models and hardware to minimize total carbon emissions — discovering greener, more efficient Transformer architectures.

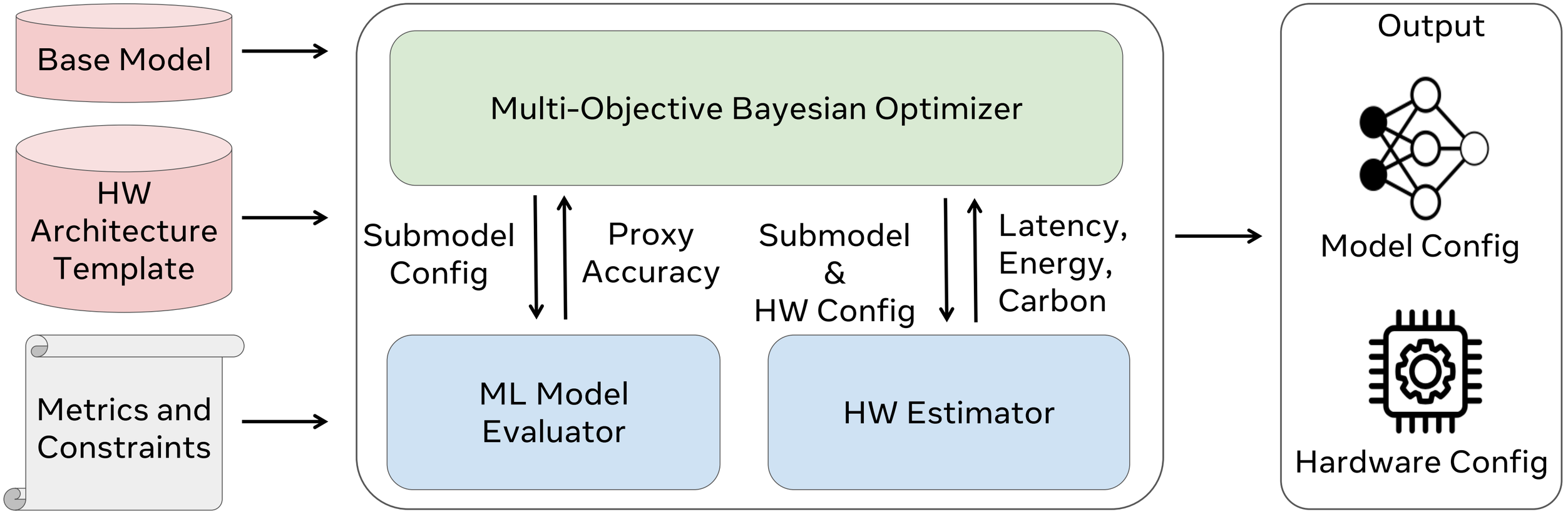
CATransformers: Inputs



CATransformers: Inputs

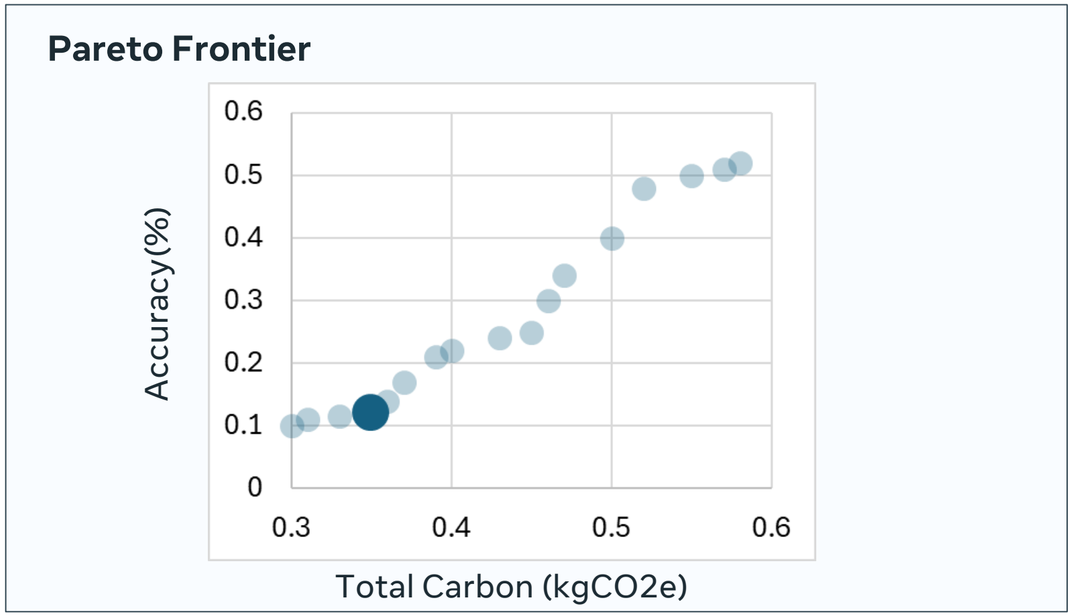


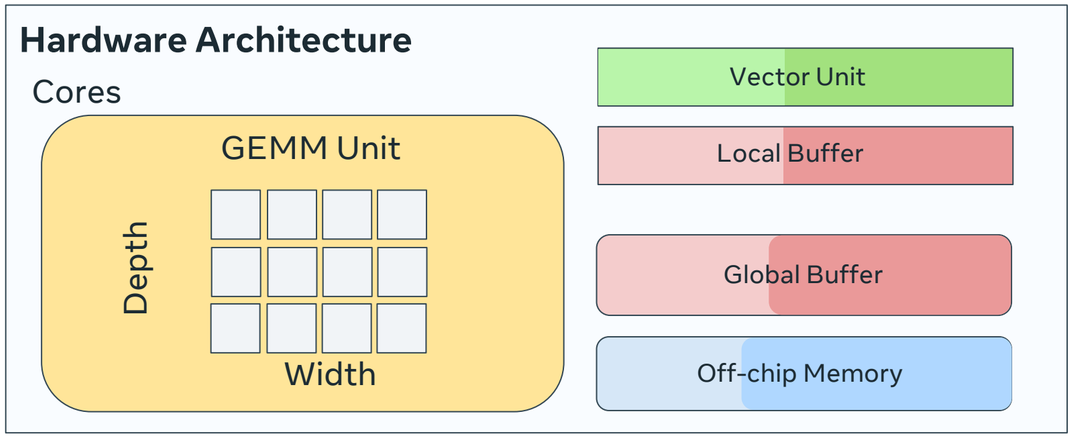
CATransformers: Overview

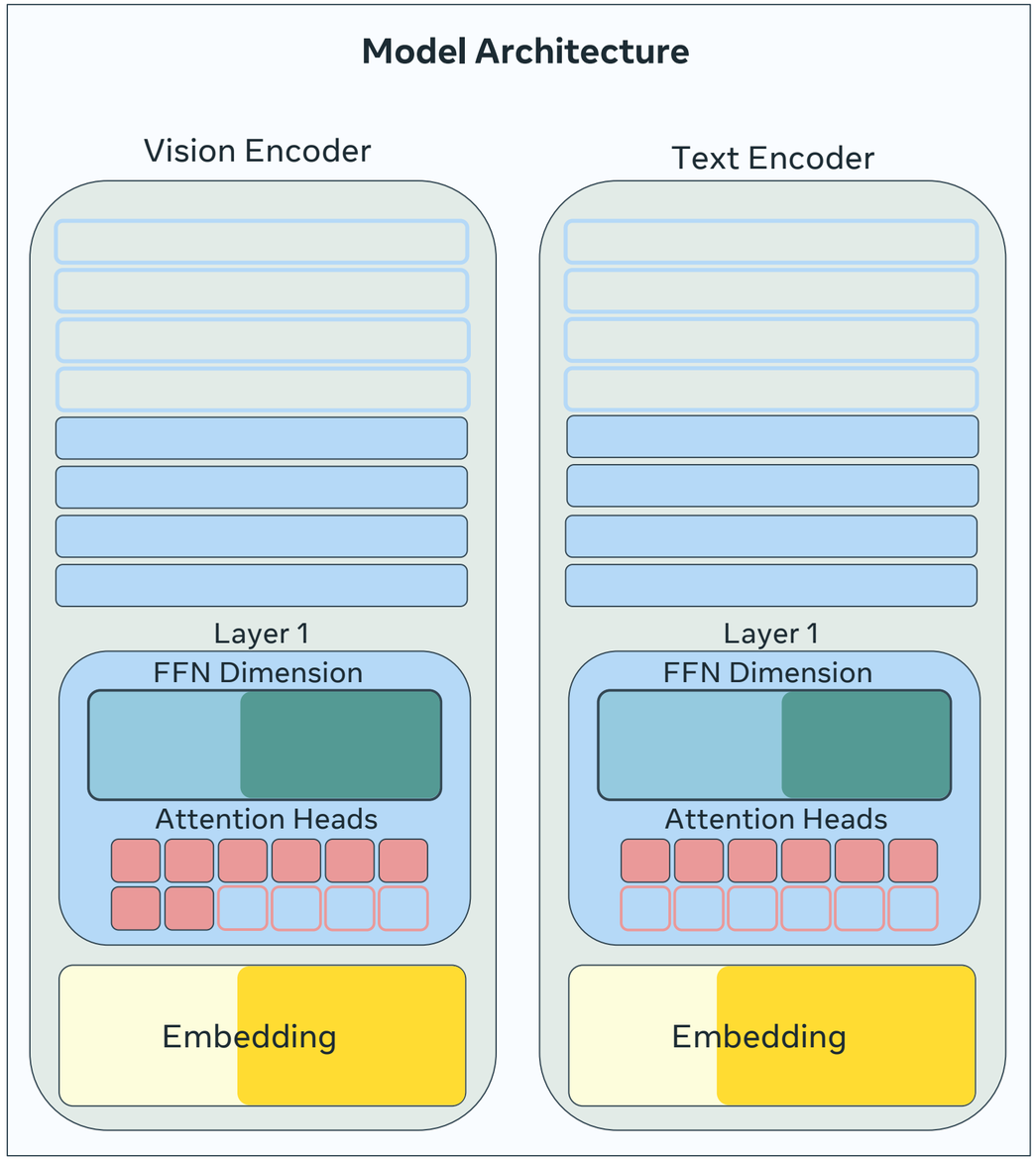


(Optional)
Fine-tuning to recover Accuracy Loss

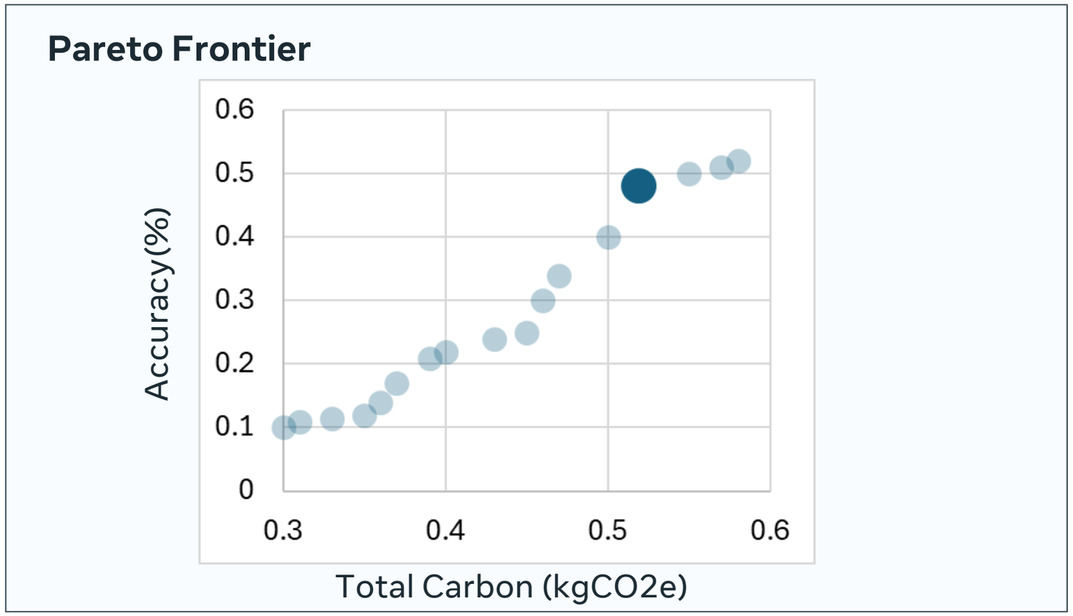
CATransformers: Search Process

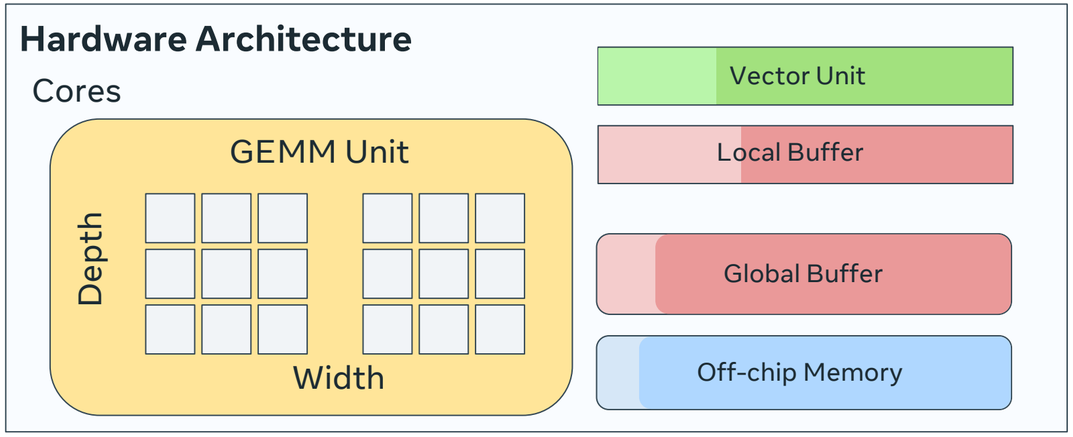


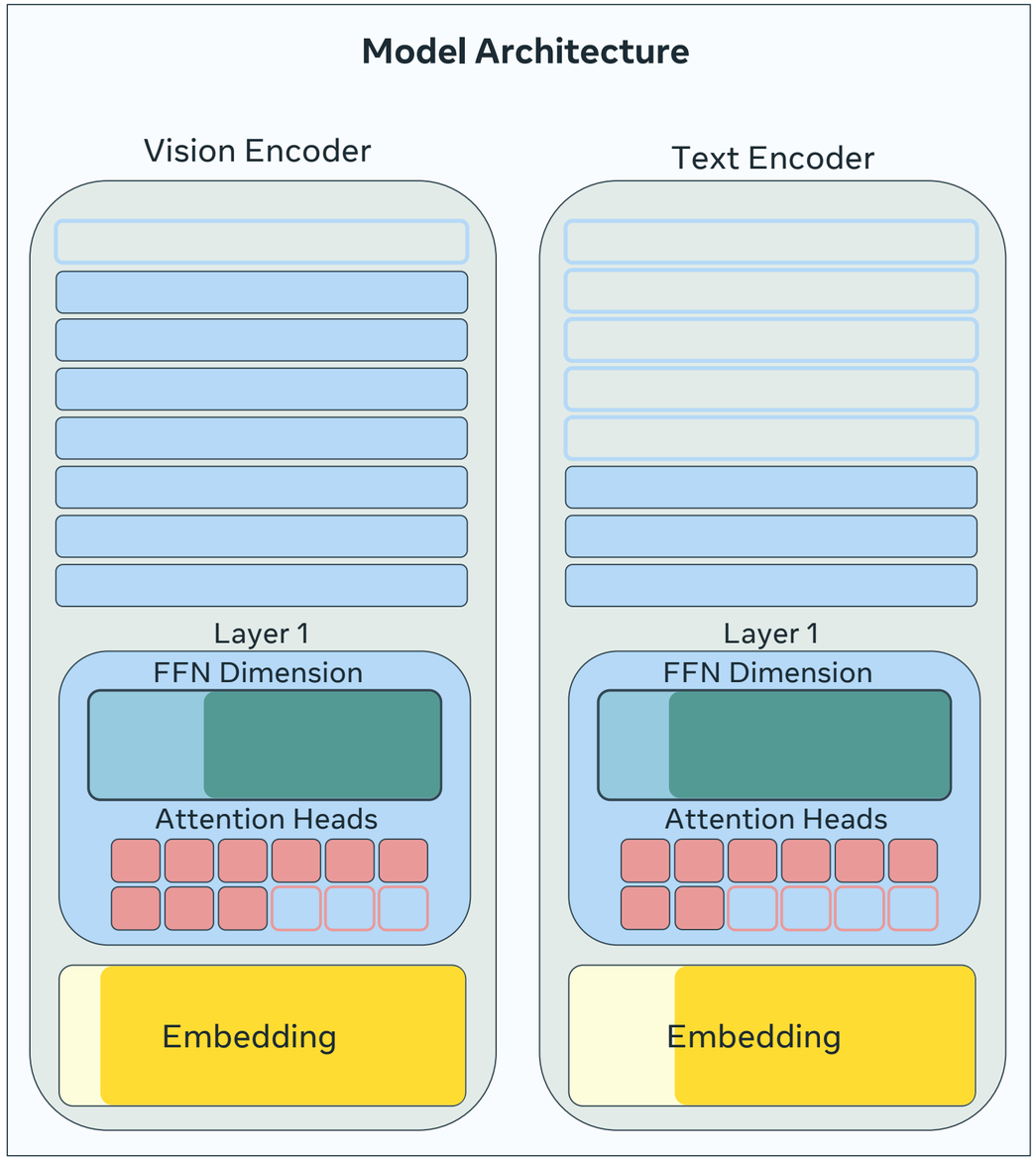




CATransformers: Search Process







Evaluation

Experimental Setup

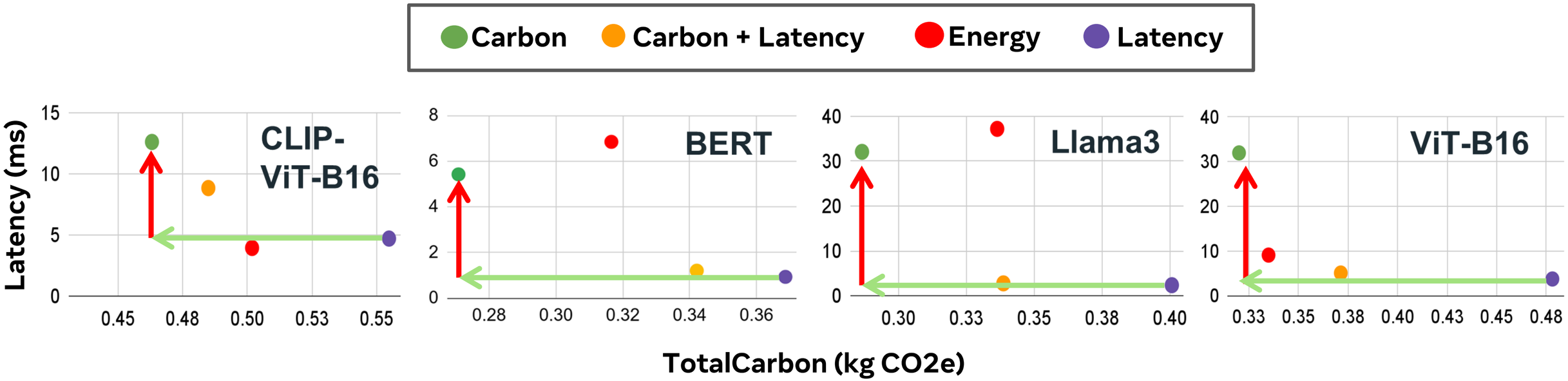
Optimization:

- 100 iterations-> 5 to 20 hours
- Repeated 3 times each
- 8 x V100 GPUs

Design choices:

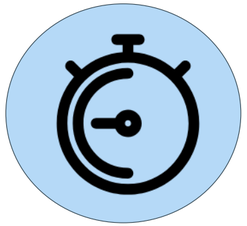
- Maximum 20 TOPS Compute Budget
- Latency Constraint: 50ms
- California grid intensity, Manufacturing in Taiwan
- 3 year device life span

Key Results: Carbon-Awareness Creates New Trade-offs

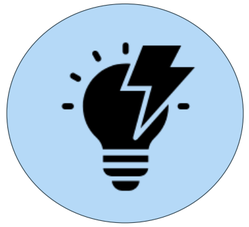


Prioritizing carbon as a primary objective reveals fundamentally different and more sustainable design choices compared to traditional latency or energy-focused methods.

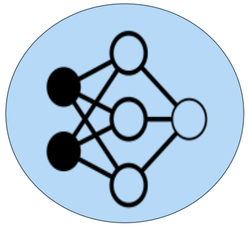
Key Takeaways – Design choices



Latency-optimized designs favor large, high-throughput hardware; carbon-optimized designs use compact, low-power accelerators

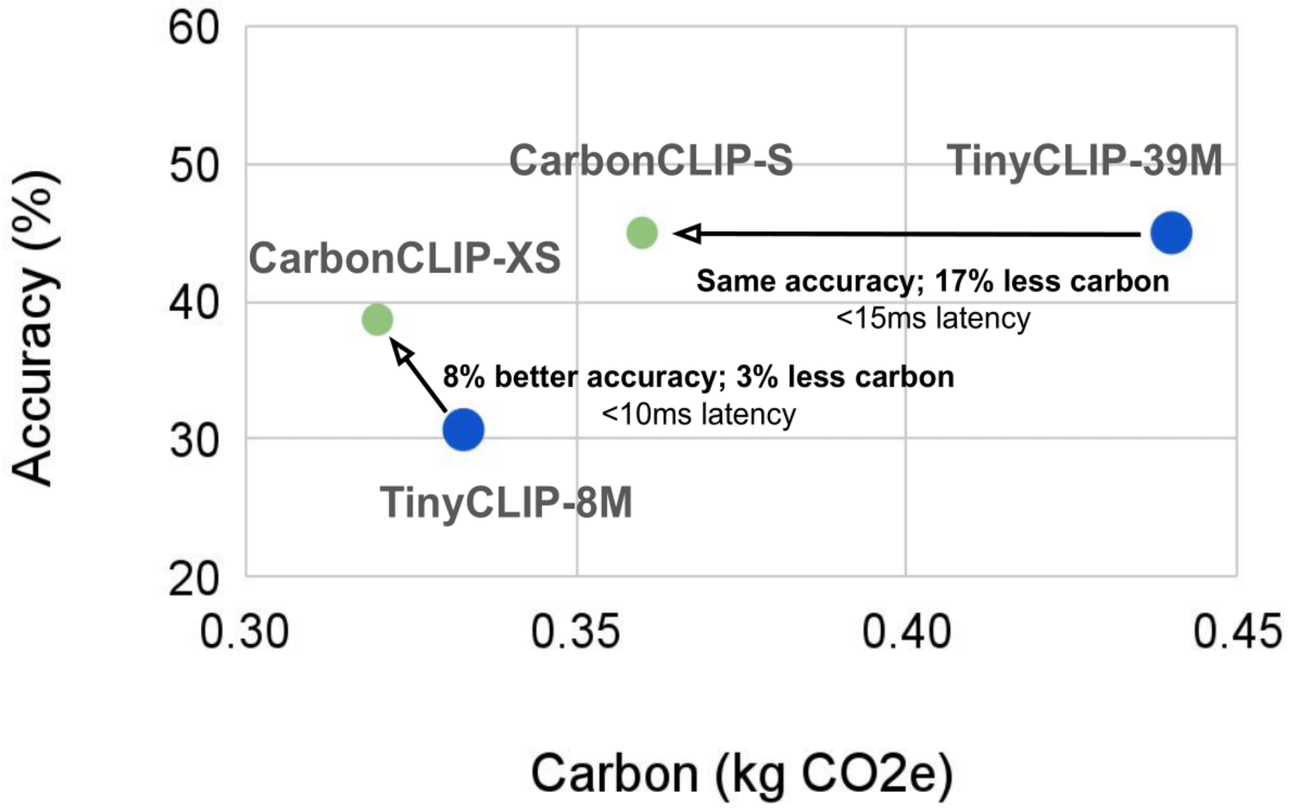


Energy optimization pair smaller models with larger accelerators, indirectly reducing latency, but not as effective as direct latency optimization.



Model sensitivity to pruning varies by model architecture: CLIP and ViT are especially sensitive to hidden dimension pruning.

Key Results: CarbonCLIP

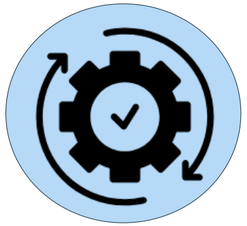


Using CATransformers, we find CarbonCLIP family of models for edge devices.

Carbon-aware co-design can yield models that are more sustainable while matching or even surpassing traditional baselines in performance.

Conclusion

We have introduced CATransformers, a sustainability-driven framework :



Co-optimizes model and hardware design by accounting for both operational and embodied carbon.

Demonstrated substantial potential in carbon reductions without sacrificing performance across Transformer models

Code is Open-Sourced:
https://github.com/facebookresearch/CATransformers

This work provides a path toward sustainable AI, making carbon efficiency a first-class objective in next-generation ML system design.

Code & Paper


Meta AI

Additional Experiments

- Fix model + hardware optimization
- Optimization under different compute constraints
- Under different latency constraints
- Varying operational regions
- Model architecture search with GPUs + Energy and Latency Validations

Demonstrate the effectiveness and generalizability of the method