



Pruna AI

Compression for Language Models



Bertrand Charpentier
Founder, President & Chief Scientist

How Do People Use Deep Learning?

Many Deep Learning Tasks



Image

- Classification
- Object detection
- Gen. from text
- Segmentation
- ...



Video

- Object tracking
- Gen. from text
- Gen. from image
- ...



Audio

- Transcription
- Translation
- Gen. from text
- ...



Text

- Question answering
- Summarization
- Classification
- ...



Proteins

- Folding
- De novo Gen
- Property prediction
- ...

Phases of Deep Learning Models

Development

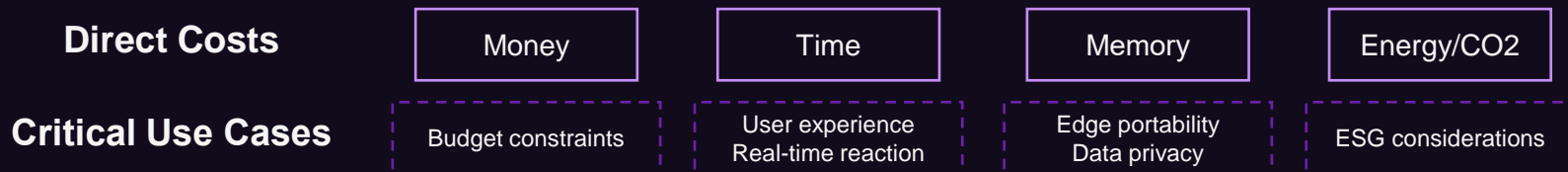
Training

Inference

80-90% of
DL workload



Why Do We Need Efficient Deep Learning?

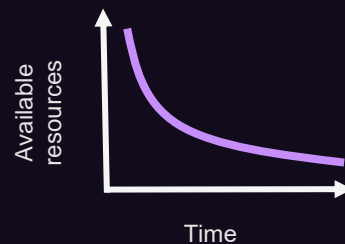
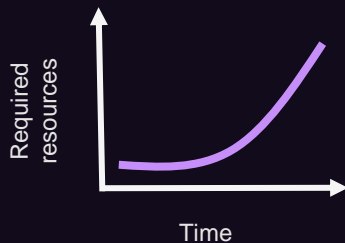


What Are Good Efficiency Metrics?

Metrics are often correlated

Metrics can be contradictory

Metrics should measure **direct real-world costs**



[1] The efficiency misnomer, ICLR 2022

[2] Power Hungry Processing: Watts Driving the Cost of AI Deployment?

How Does a Deep Learning Model Work?

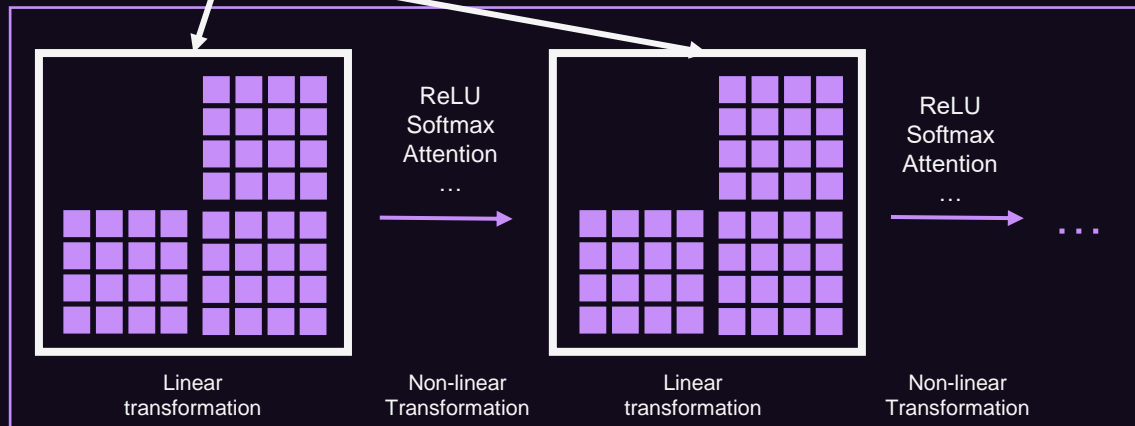
Input data

Output prediction

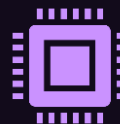


80-95% of
DL workload

Model



Language



Hardware



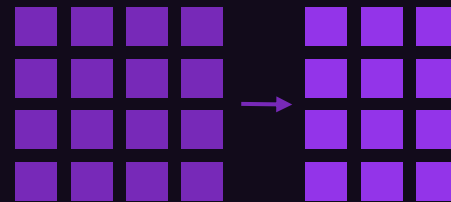
Pruning



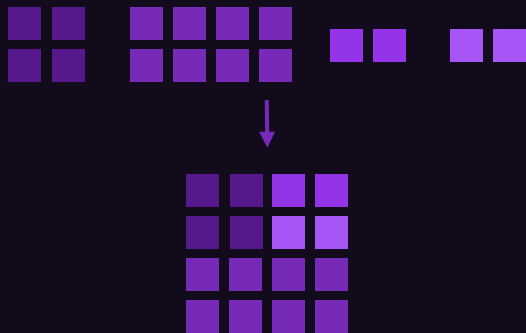
Quantization



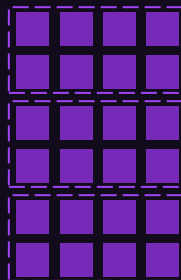
Distillation



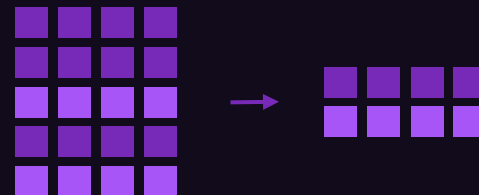
Compilation



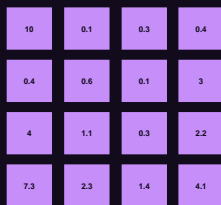
Batching



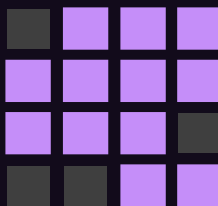
Caching



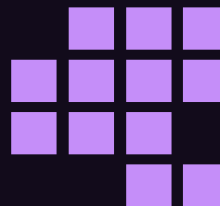
How to Prune Deep Learning Models?



Step 1
Score
structures



Step 2
Rank
structures
w.r.t. scores



Step 3
Prune structures
with lowest
scores

- **What structure to prune?** Unstructured pruning, structured pruning, ...
- **How to score structures?** Random, magnitude, gradient, hessian
- **What sparsity to prune?** Homogeneous, heterogeneous
- **When to prune?** Before, during, after training

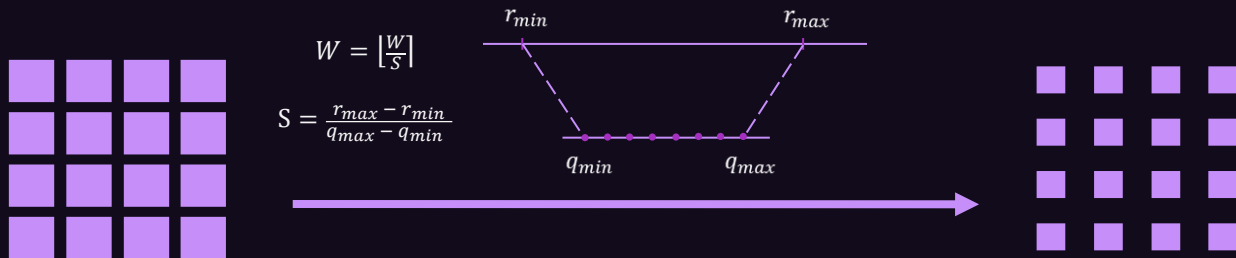


[1] Winning the Lottery Ahead of Time: Efficient Early Network Pruning. ICML 2022

[2] How Sparse Can We Prune A Deep Network: A Fundamental Limit Viewpoint

[3] Structurally Prune Anything: Any Architecture, Any Framework, Any Time.

How to Quantize Deep Learning Models?



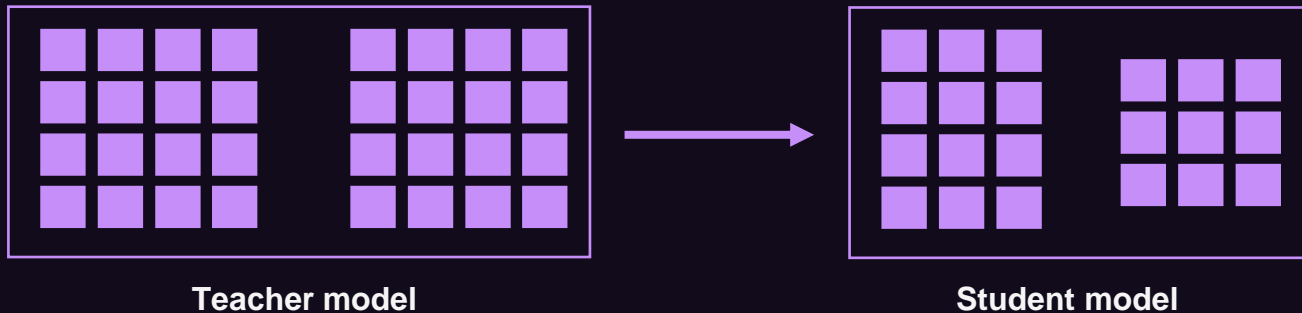
- **What structure to quantize?** Per tensor/channel/group/outliers, weight/activation
- **How to quantize structures?** Linear quantization, code books
- **What precision to quantize?** 16, 8, 4, 2, 1 bits
- **When to quantize?** Quantization-aware, post-training



[1] AWQ: Activation-aware Weight Quantization for LLM Compression and Acceleration. NeurIPS 2023

[2] GPTQ: Accurate Post-Training Quantization for Generative Pre-trained Transformers. ICLR 2023

How to Distill Deep Learning Models?



- **What information to distill?** Response, feature, weights...
- **What model to distill into?** Architecture, size, precision
- **When to distill?** Offline, online

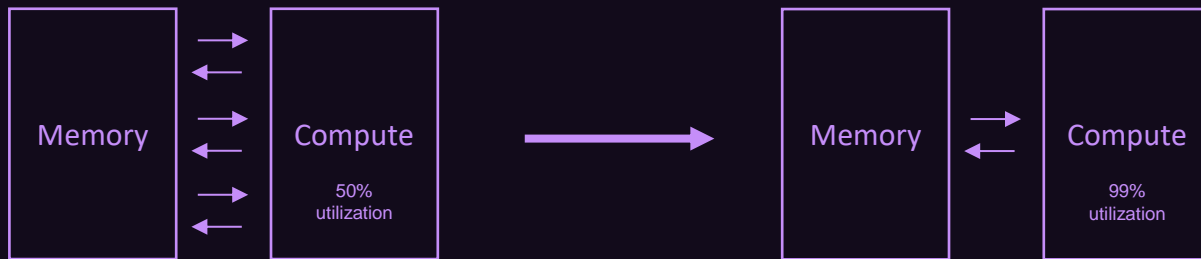


[1] Does Knowledge Distillation Really Work?. NeurIPS 2021

[2] Improved Knowledge Distillation via Teacher Assistant. AAAI 2019

[3] Fitnets: Hints for Thin Deep Nets. ICLR 2015

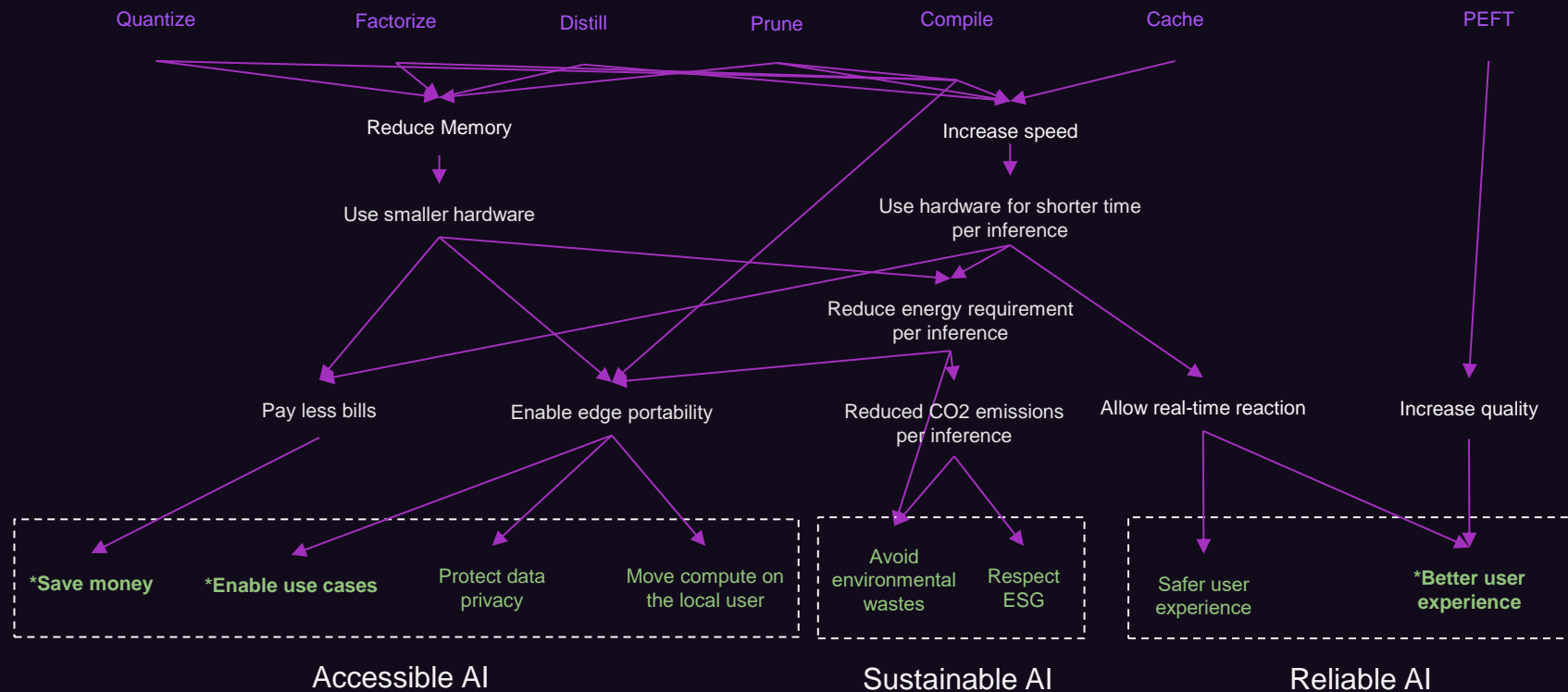
How to Compile Deep Learning Models?



- **What structure to compile?** Linear/Attention/..., Fuse operators
- **What compilation backend/kernels to use?** CUDA, Triton, ARM, Custom backend
- **What hardware is supported by compilation?** CPU, GPU, others
- **How to compile?** Memory vs compute bound



How Does Compression Benefit Deep Learning?



How Well Does Compression Methods Work?

	Acc.	Speed	Mem.
Base	~75%	x1	x1
Prun.	~76%	x2	X0.5

ResNet50 - ImageNet

	Perpl.	Speed	Mem.
Base	~5.6	x1	x1
Quant.	~6.0	x2	X0.5

Llama 7B - WikiText

	Speed	Mem.
Base	x1	x1
Distill.	x2	X0.5

Stable Diffusion

- **Remark 1:** There are many, many, many other compression methods.
- **Remark 2:** Compression methods can be combined
- **Remark 3:** The best (combination of) compression methods depends on the final application setup (incl. architecture, hardware, data,...).

Compression of DL model is complex!



[1] Structurally Prune Anything: Any Architecture, Any Framework, Any Time.

[2] GPTQ: Accurate Post-Training Quantization for Generative Pre-trained Transformers. ICLR 2023

[3] SSD-1B. Segmind

How Well Does Compression Methods Work?



Base



Pruna



How Well Does Compression Methods Work?



Base

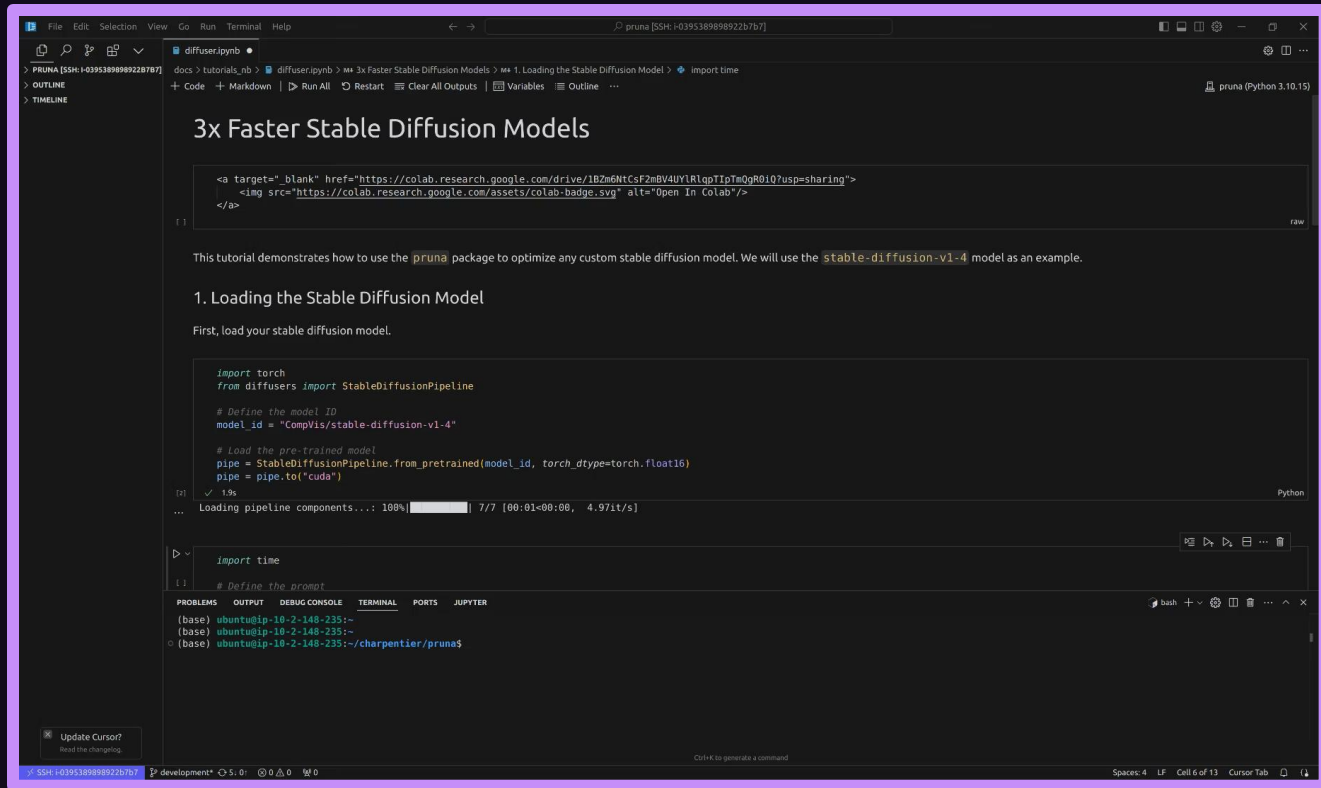


Pruna

Compression of DL model does not
necessarily means quality loss!



How to Compress Your Deep Learning Model?



```
diffuser.ipynb
> PRUNA [SSH: i439538989892257b7] docs > tutorials.nb > 3x Faster Stable Diffusion Models > 1. Loading the Stable Diffusion Model > import time
+ Code + Markdown | ▶ Run All | ⏮ Restart | 🧹 Clear All Outputs | 📄 Variables | 📖 Outline | ...
```

3x Faster Stable Diffusion Models

<https://colab.research.google.com/drive/1B7mFntCsF2mBV4UY1RlqTIpTm0pR0IQ?usp=sharing>

This tutorial demonstrates how to use the `pruna` package to optimize any custom stable diffusion model. We will use the `stable-diffusion-v1-4` model as an example.

1. Loading the Stable Diffusion Model

First, load your stable diffusion model.

```
import torch
from diffusers import StableDiffusionPipeline

# Define the model ID
model_id = "CompVis/stable-diffusion-v1-4"

# Load the pre-trained model
pipe = StableDiffusionPipeline.from_pretrained(model_id, torch_dtype=torch.float16)
pipe = pipe.to("cuda")
```

[1] ✓ 1.9s Python

... Loading pipeline components...: 100% [████████████████████] 7/7 [00:01:00:00, 4.97it/s]

```
import time

# Define the prompt
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS JUPYTER

(base) ubuntu@ip-10-2-148-235:~
(base) ubuntu@ip-10-2-148-235:~
(base) ubuntu@ip-10-2-148-235:~/charpentier/pruna\$

Update Cursor?
Read the changelog

SSH: i439538989892257b7 development* 0:0:0 @ 0 0 0 0



How to Compress Your Deep Learning Model?

Manual solution

1. Read & understand research papers.
2. Implement & test compression methods
3. Integrate compression methods for your specific model/hardware.
4. Test & evaluate all hyperparameters.
5. *Hopefully* get efficiency gains

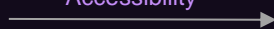
Pruna AI solution

1. Install Pruna package
2. Smash your AI model
3. Get *significant* efficiency gains



Long research exploration

Accessibility

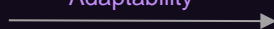


Easy-to-use compressions



Painful model/hardware debugging

Adaptability

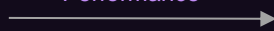


Model/hardware adaptability



Resources wasted/Impossible project

Performance



Reliable efficiency gains



Try our 10,000+ smashed models on
Hugging Face!

