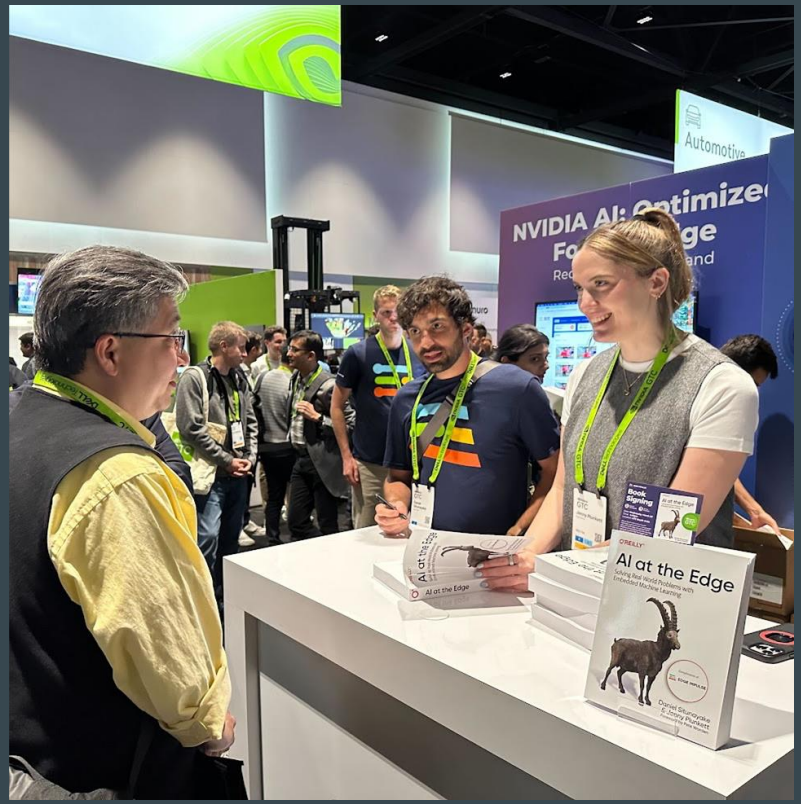
Solve edge AI problems with foundation models

...

Daniel Situnayake

Hello, I'm Daniel Situnayake! 👋



- Director of Machine Learning at Edge Impulse
- Wrote AI at the Edge and TinyML (O'Reilly)
- Previously worked on TensorFlow Lite and TFLM (Google)
- Superficial Intelligence newsletter (dansitu.substack.com)

Foundation models

- Pre-trained models
- Trained on broad datasets
- Applied to tasks outside their training
- Tend to be large! Hundreds of megabytes to terabytes.

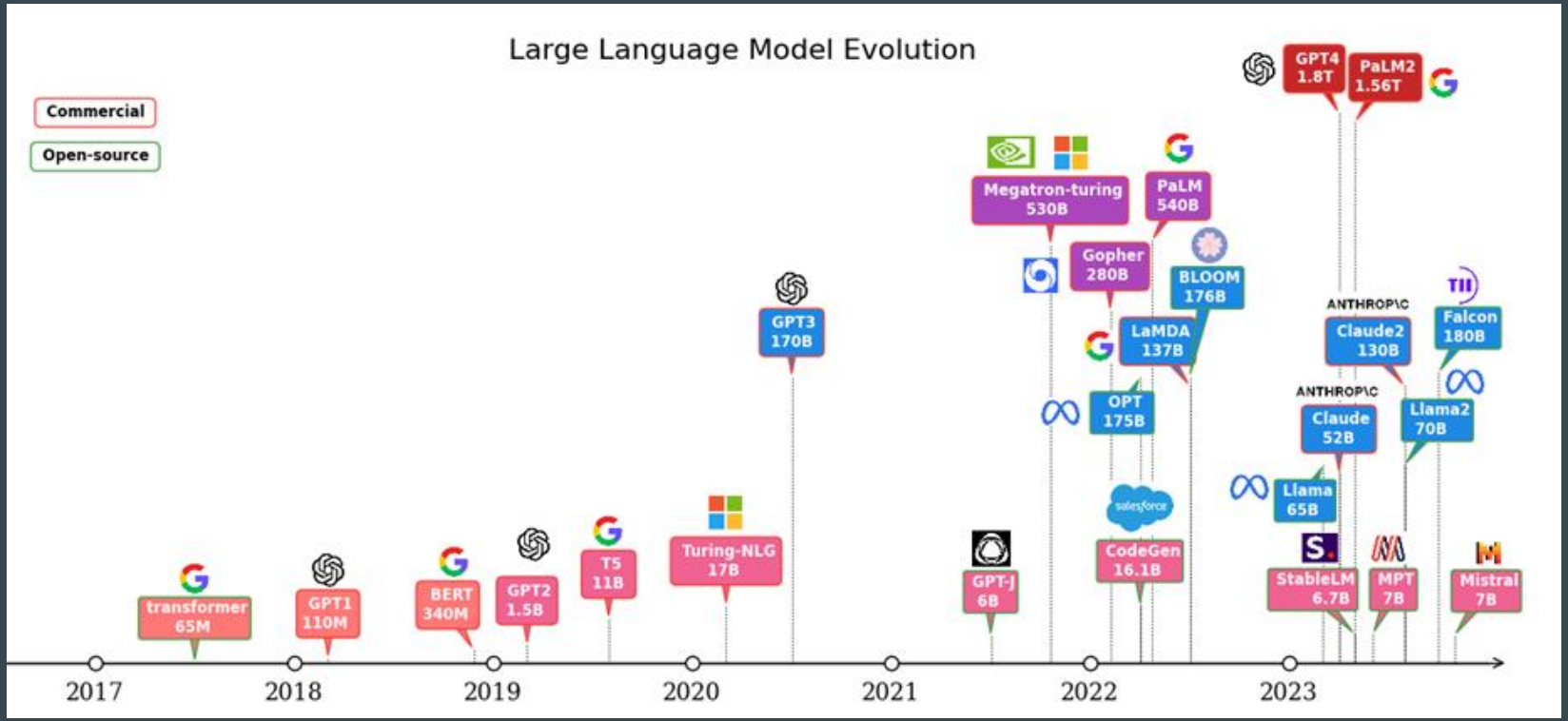
Text
Audio
Image
Code
Genomics
Time series

Generative AI

- Create data in addition to consuming
- Can be implemented using foundation models
- Size can vary greatly depending on task

Writing
Speech
Denoising
Code
Images
Music

Model sizes

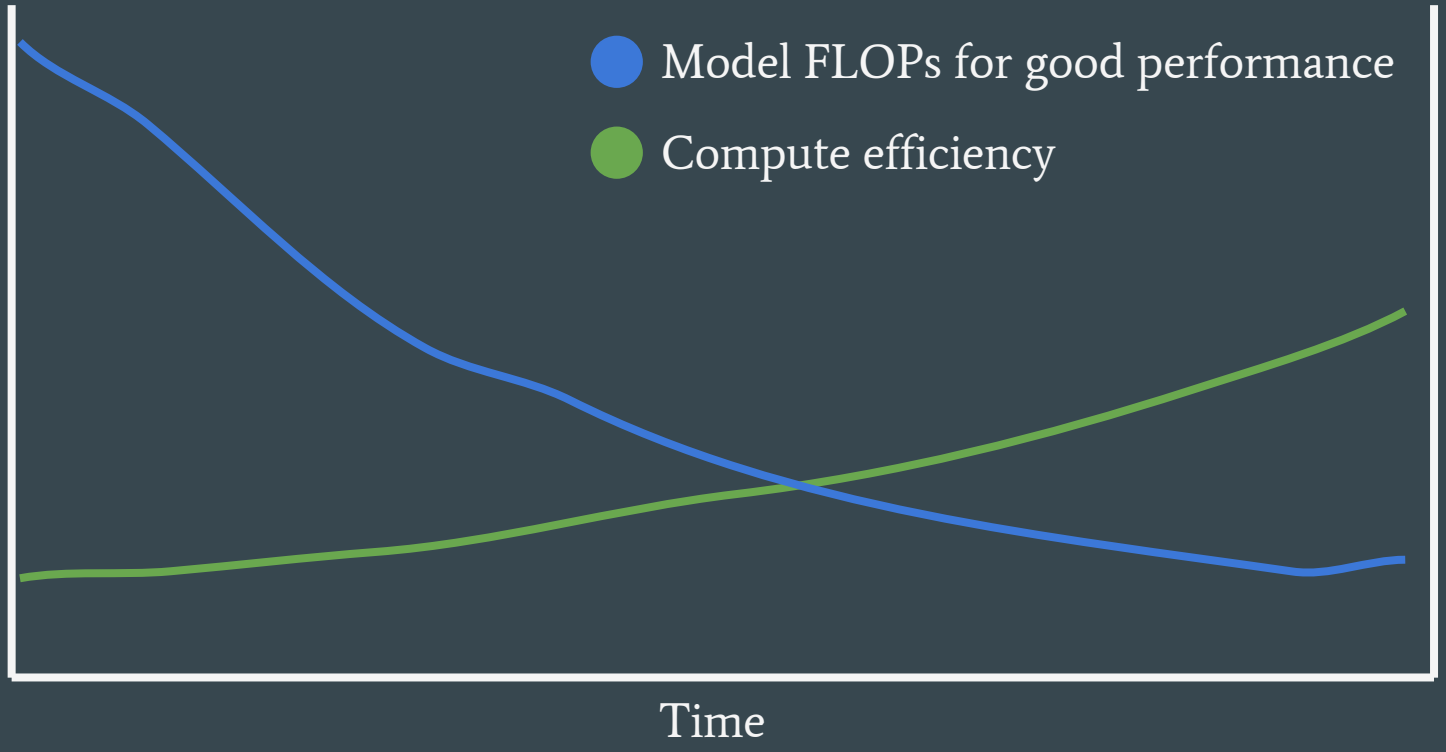


https://infohub.delltechnologies.com/en-US/p/investigating-the-memory-access-bottlenecks-of-running-llms/

Model	N Params	Max Tokens	HF Average
Phi-2	2.7B	2048	61
TinyLlama	1.1B	2048	53
Rocket	3B	1024	51
Mamba GPT	3B	2048	44
Guanaco Uncensored	3B	2048	39
Incite	3B	2048	39
OpenLLama	3B	196K	36
Orca	3B	1024	35
Pythia	1.4B	2048	35
OPT	1.3B	2048	35
Lamini Neo	1.3B	2048	35
Lamini GPT	1.5B	1024	35
Lamini GPT	774M	1024	32
Pythia	410M	2048	31
Lamini Cerebras	1.3B	2048	30
Pythia	160M	2048	29
Lamini Neo	125M	2048	29
Pythia	70B	2048	28
Lamini GPT	124M	1024	28
Lamini Cerebras	590M	2048	28
Lamini Cerebras	256M	2048	28
Lamini Cerebras	111M	2048	28

https://medium.com/@daniellefranca96/battle-of-the-smallest-llms-e923e2cac1ff

Where we're headed (warning, unscientific chart)



“Large” models will eventually arrive on cheap, low power devices

“Large” models will eventually arrive on cheap, low power devices

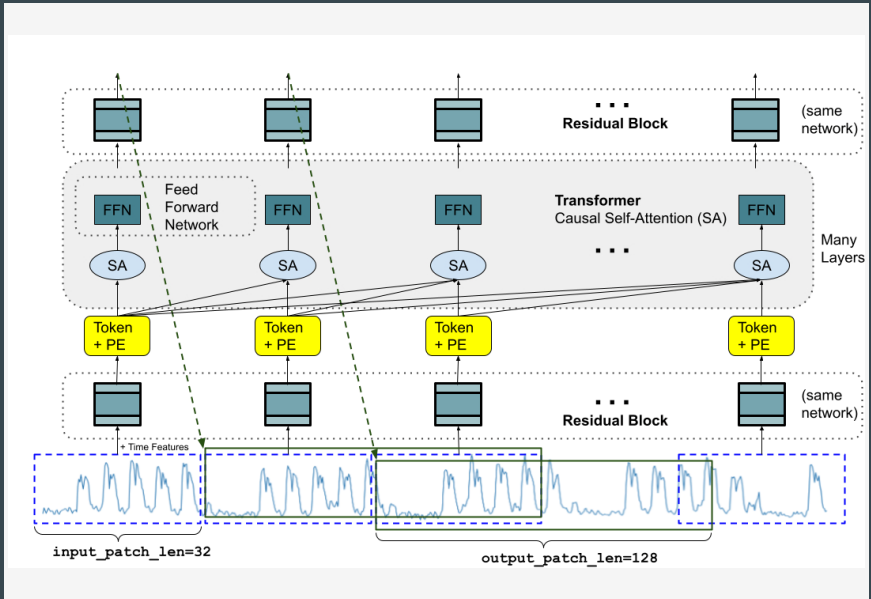
But we don’t need to wait.

Four key capabilities of foundation models

- Zero-shot learning
- Reasoning
- Information retrieval
- Data generation

Zero-shot learning

Zero-shot time series forecasting



https://blog.research.google/2024/02/a-decoder-only-foundation-model-for.html

Zero-shot image classification with multimodal LLM

Prompt: “Classify this image as hotdog or not hotdog”

Response: “hotdog”

Zero-shot question answering with BERT

Prompt: “How do I change the batteries?”

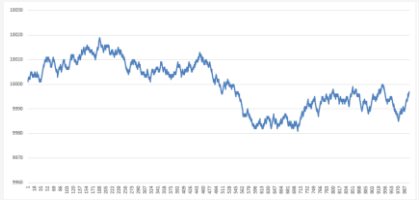
Document:

Response: “In order to change the batteries...”

Reasoning

Determining the right action

Prompt: “Plan a maintenance window based on the production line status”



Response: “A reasonable maintenance window is...”

Intent matching

Intents: dispense_drink, dispense_food

User: “I want a soda please”

Match: dispense_drink

Reasoning based on documents

Prompt: “Is the proposed solution legal?”

Document:

Response: “Yes, the solution proposed is...”

Information retrieval

Looking up facts with LLM + RAG

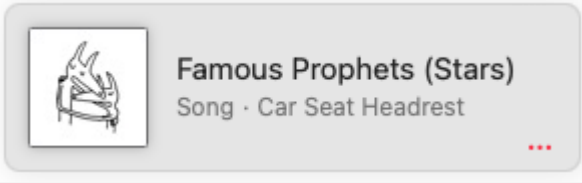
Prompt: “How can I treat this plant disease?”

Response: “This looks like <disease>, which can be treated with <treatment>.”

Multimodal lookup

Prompt: “Play a song with heavy guitar I have not heard before”

Response:



Question answering with BERT

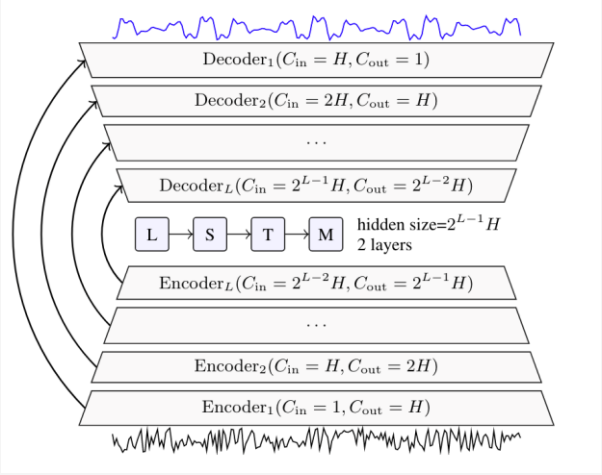
Prompt: “How do I change the batteries?”

Document:

Response: “In order to change the batteries...”

Data generation

Denoising and upscaling



https://github.com/facebookresearch/denoiser

Generating text and audio

Prompt: “Tell me a story about unicorns, with pictures”

Response: “Once upon a time...”

Video and audio generation

https://openai.com/sora

Are foundation models capable of these?

Yes.

- Zero-shot learning
- Reasoning
- Information retrieval
- Data generation

Are foundation models required?

No!

- Zero-shot learning
- Reasoning
- Information retrieval
- Data generation

Zero-shot learning on the edge

Benefits of large foundation models

- Reduces training data requirements
- Allows task to be adjusted on-the-fly

Alternatives

- Can implement in other ways (embeddings + nearest neighbor lookup, etc)
- Use smaller, domain-specific models (custom BERT)
- Can use zero-shot model for data labelling then train a conventional model

Reasoning on the edge

Benefits of large foundation models

- Understand complex user communication
- Match inputs to states
- Make sophisticated decisions

Alternatives

- Language - intent matching and slot filling
- State machines (game design)
- Smaller, domain-specific models (perhaps created via distillation)

Information retrieval on the edge

Benefits of large foundation models

- Convenient retrieval of information
- Language-based interface
- Answer any possible question

Alternatives

- Smaller, domain-specific models (custom BERT)

Data generation on the edge

Benefits of large foundation models

- Create and manipulate signals
- Generate multimodal content

Alternatives

- Smaller, domain-specific models
  - Visual question answering
  - Signal-to-signal for specific use cases
- Small, distilled generative models

Designing with foundation models
at the edge

1. Frame your problem

- Which special capabilities do you require?
  - Zero-shot learning
  - Reasoning
  - Information retrieval
  - Data generation

- Can it be framed more simply? (classification, regression, clustering, etc.)

2. Determine your constraints

- Do you need to run on-device?
  - Bandwidth
  - Latency
  - Economics
  - Reliability
  - Privacy

- What are your hardware capabilities?
  - GPU
  - NPU
  - CPU
  - MCU

3. Is there a non-ML solution, or an existing solution, that works?

- Algorithm choice
  - Rule-based AI
  - Digital signal processing
  - State machines

- Pre-trained deep learning models
  - TinyBERT
  - Small LLMs
  - Quantization?

4. If you have to use an on-device model, make it simple

- Use a simple, non-foundation model where possible
  - For zero-shot can you just use embeddings and k-nearest neighbors?

- Transfer knowledge from foundation models to domain-specific simple ones
  - Label data with zero-shot learning models
  - Generate synthetic data with generative models

5. Increase complexity only when required

- Watch your costs and constraints
- Fine-tune instead of training from scratch
- Try to predict performance before spending money on training

How to design with foundation models at the edge

1. Frame your problem. Which capabilities do you require? (zero-shot, data generation, etc.)

2. Determine your constraints. Do you need to run on-device?

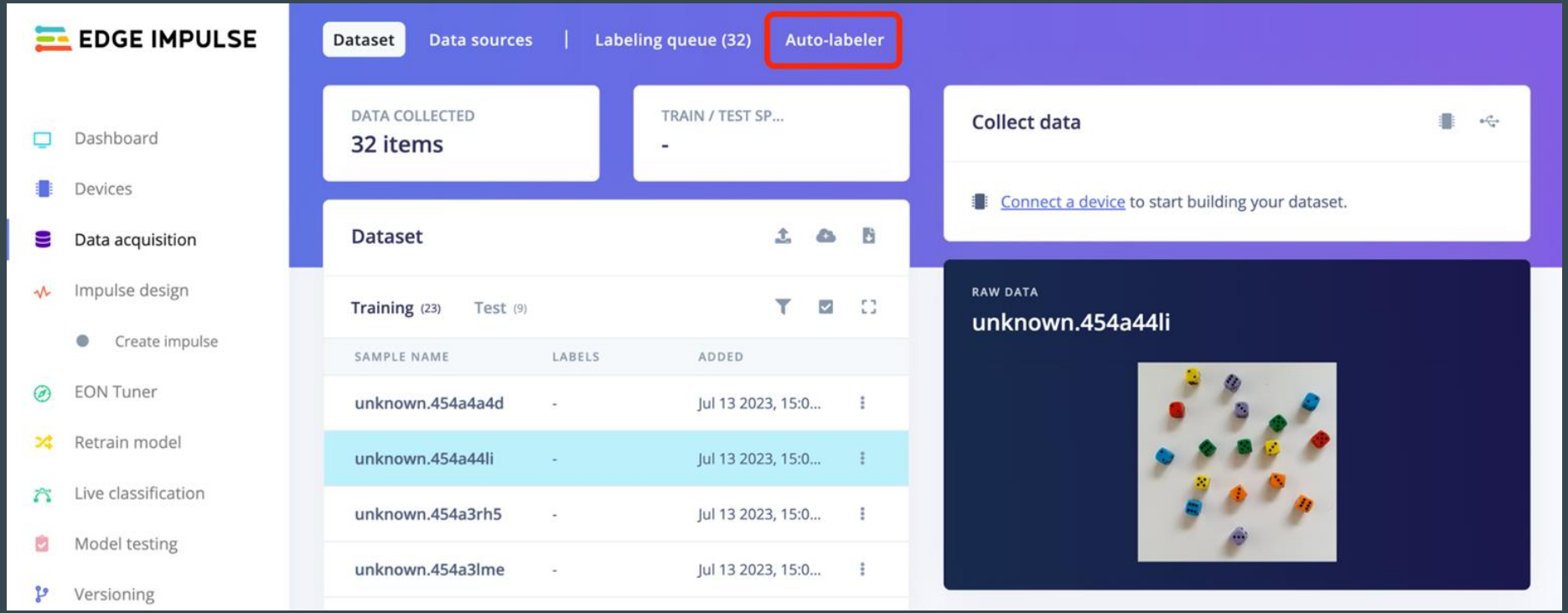
3. Look for a non-ML solution, or an existing solution that already works.

4. If you have to use an on-device model, make it simple.

5. Increase complexity only when required.

Foundation models in the edge AI toolchain

Labelling assistance



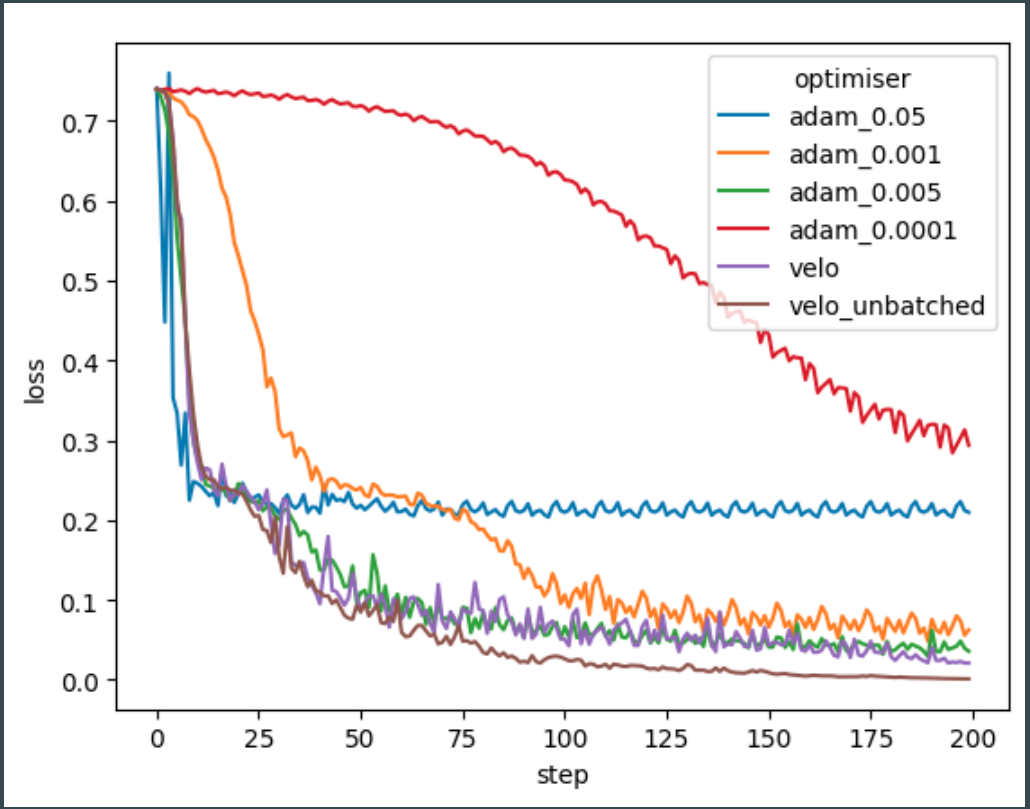
Labelling assistance

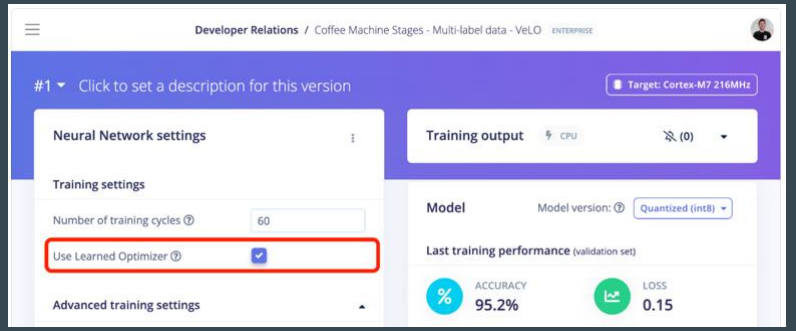
Synthetic data

- Text to image
  - Dall-E, stable diffusion, etc.
- Audio
  - Generate data for keyword spotting
- Many other things!
  - NeRF (2D to 3D)
    - https://blogs.nvidia.com/blog/instant-nerf-research-3d-ai/
  - 3D scene synthesis
    - https://machinelearning.apple.com/research/roomdreamer

https://docs.edgeimpulse.com/docs/tutorials/ml-and-data-engineering/generate-synthetic-datasets

Training!

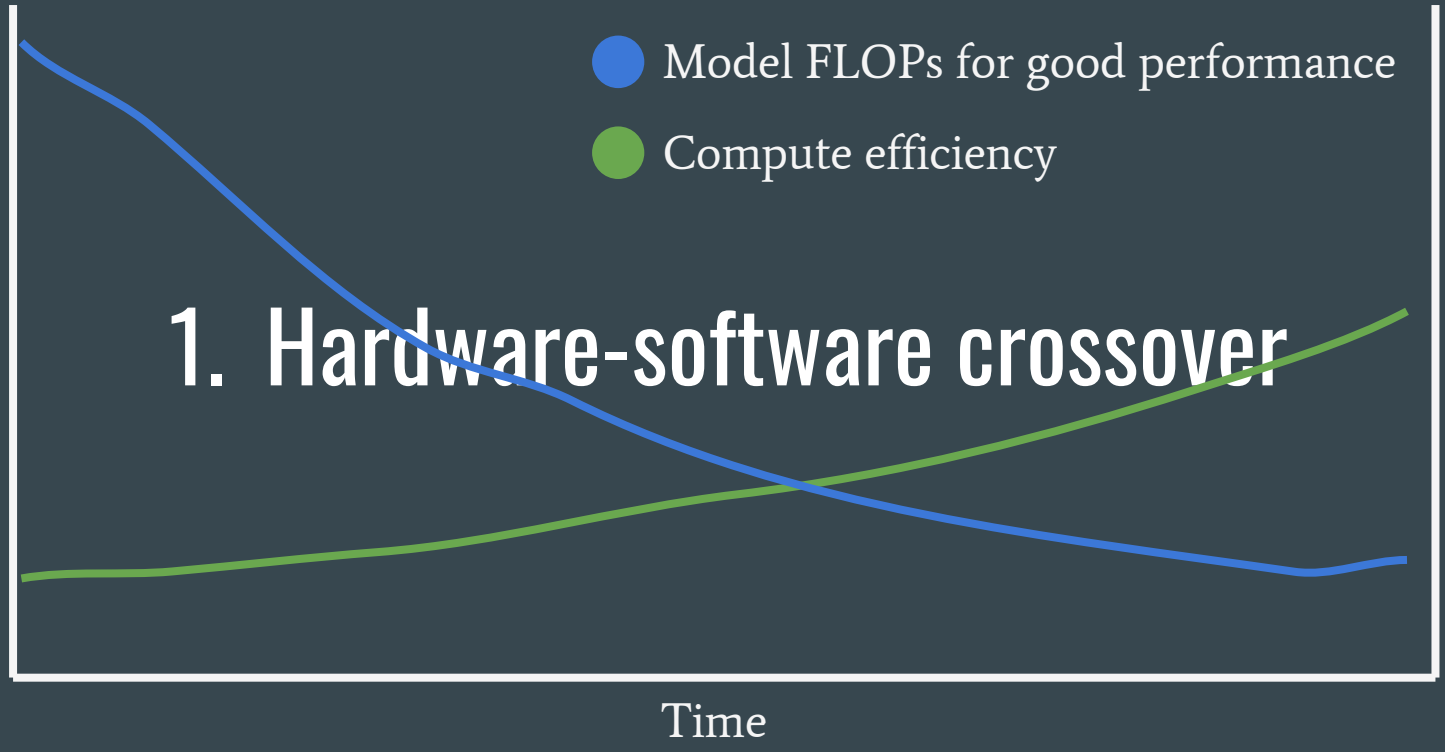




VeLO: Training Versatile Learned Optimizers by Scaling Up

https://arxiv.org/abs/2211.09760

Edge AI and foundation models in the future


Model FLOPs for good performance
Compute efficiency
1. Hardware-software crossover
Time

2. Disconnectivity

No more subscriptions,
models as IP

3. The curse of generality

Goodbye, GPT

3. Embodiment


AI

Thank you!

•••

edgeimpulse.com

dansitu.substack.com