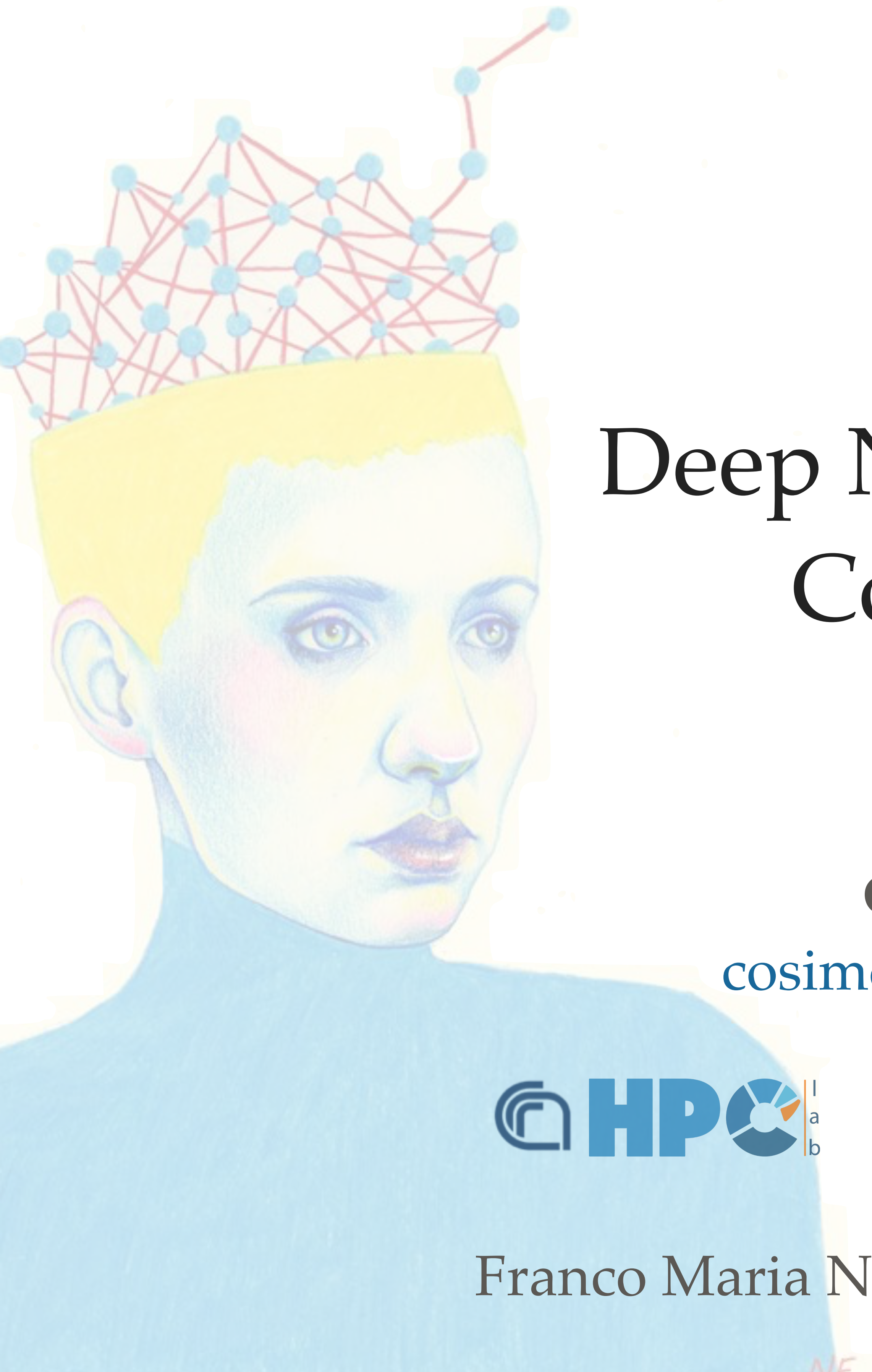


Deep Neural Network Compression

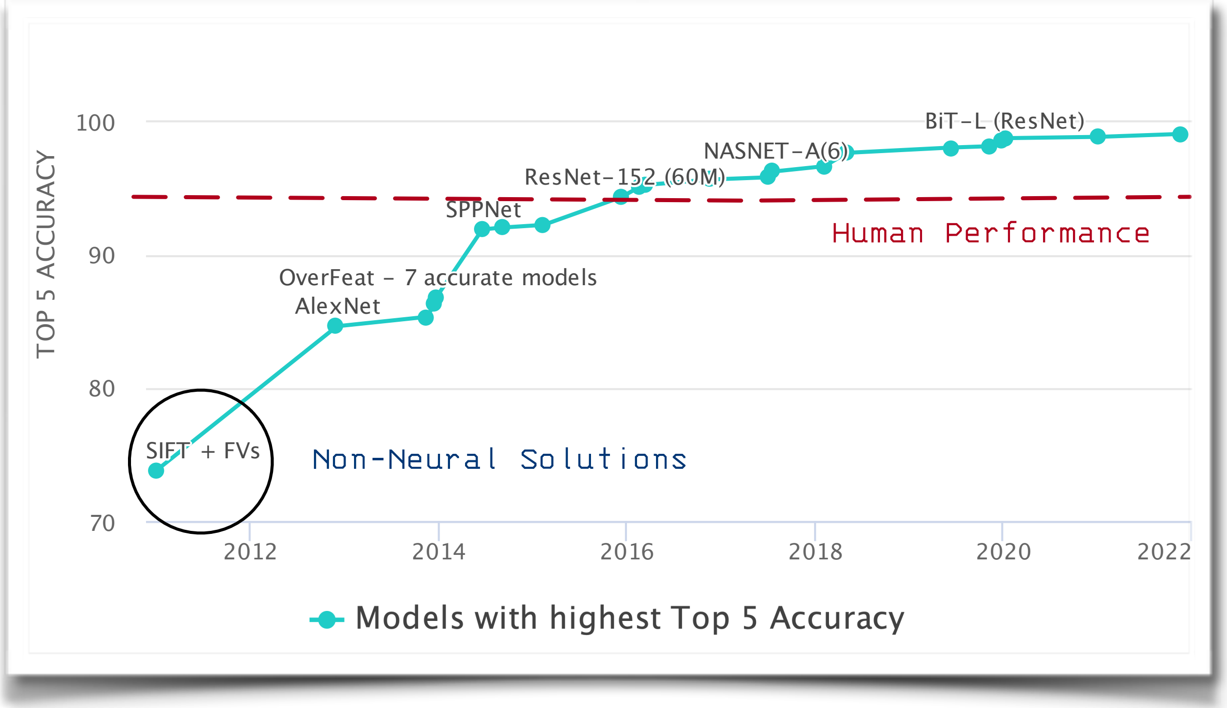
Cosimo Rulli
cosimo.rulli@phd.unipi.it

Università di Pisa

Supervisors
Franco Maria Nardini and Rossano Venturini

Deep Neural Networks..

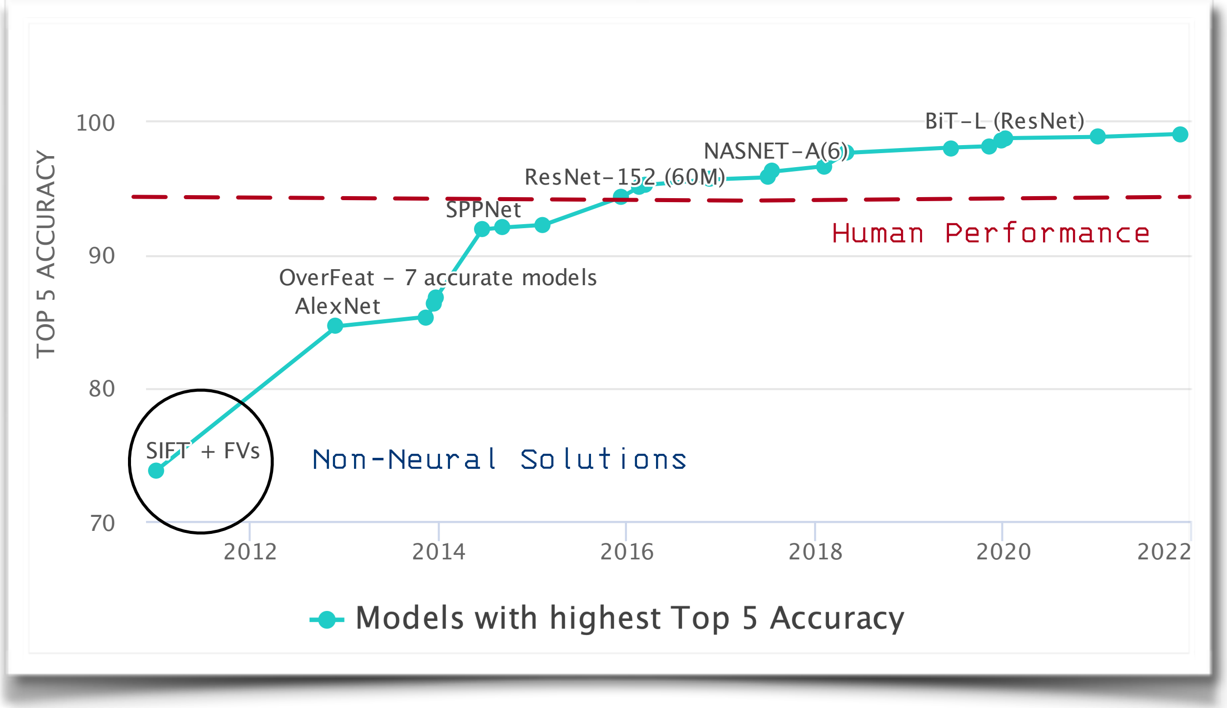
- **Leading** AI solution, **unprecedented** and **super-human** performance



Kaplan, Jared, et al. "Scaling Laws for Neural Language Models." *arXiv e-prints* (2020).

Deep Neural Networks..

- **Leading** AI solution, **unprecedented** and **super-human** performance

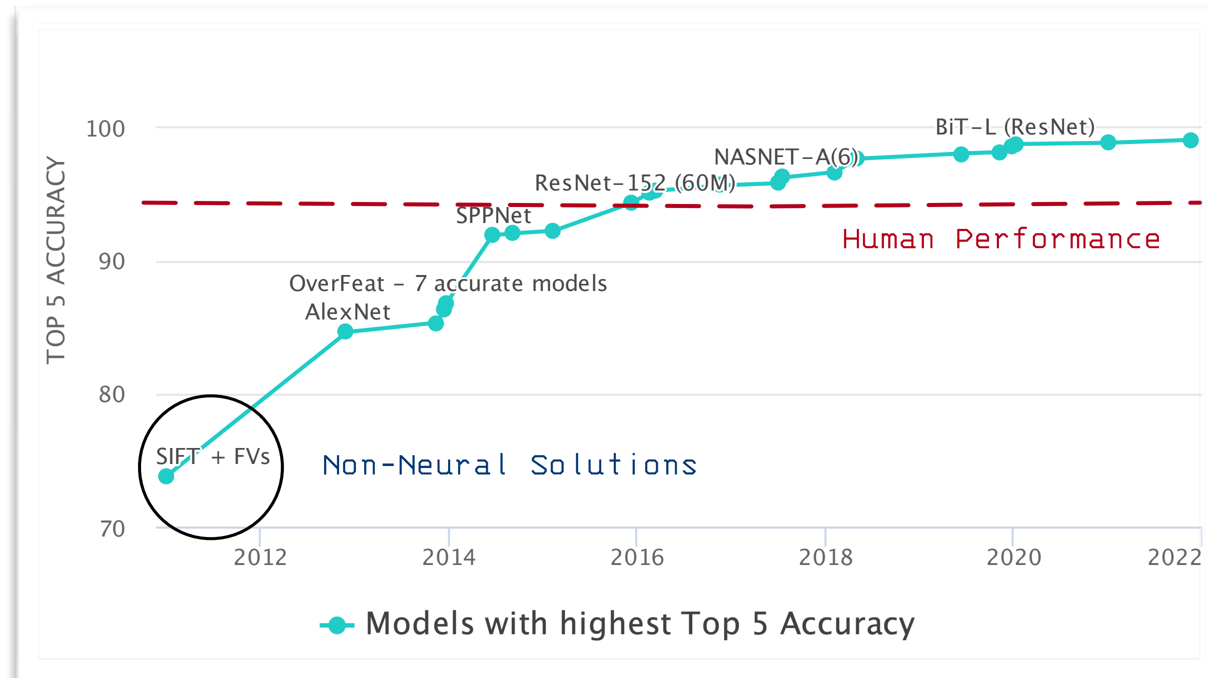


- Main **Features**

Kaplan, Jared, et al. "Scaling Laws for Neural Language Models." *arXiv e-prints* (2020).

Deep Neural Networks..

- **Leading** AI solution, **unprecedented** and **super-human** performance

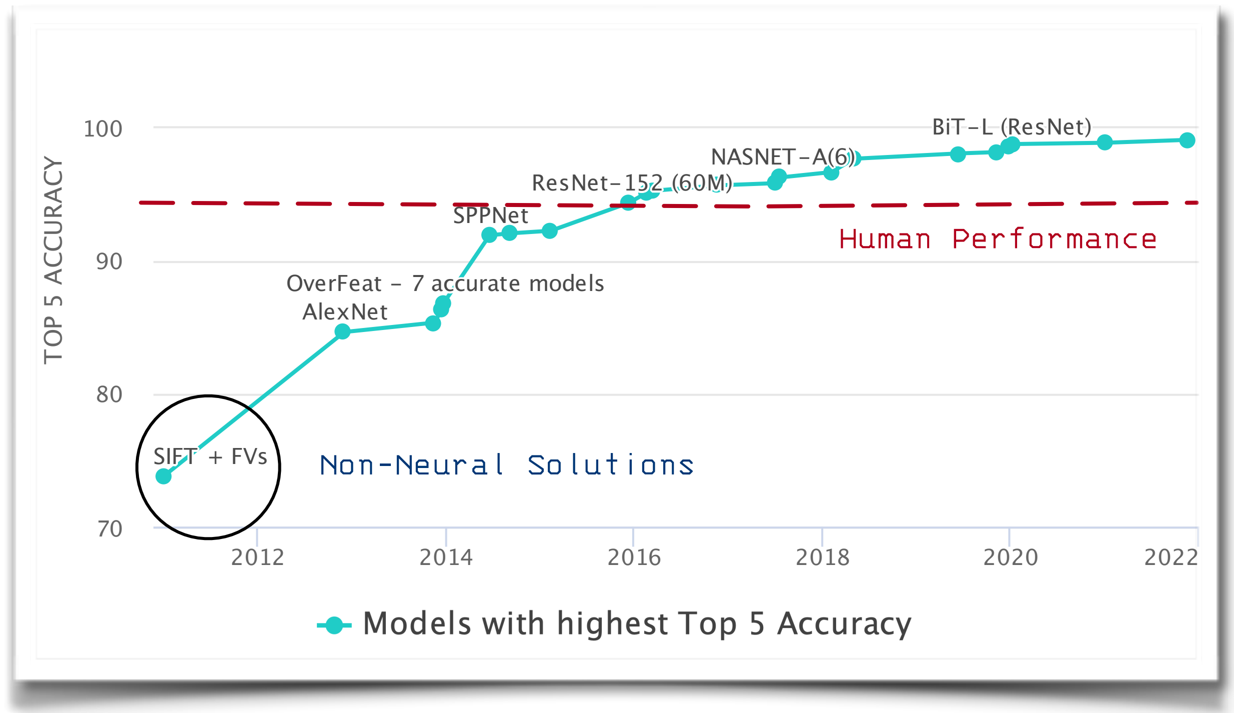


- Main **Features**
  - **Representation** Learning

Kaplan, Jared, et al. "Scaling Laws for Neural Language Models." *arXiv e-prints* (2020).

Deep Neural Networks..

- **Leading** AI solution, **unprecedented** and **super-human** performance

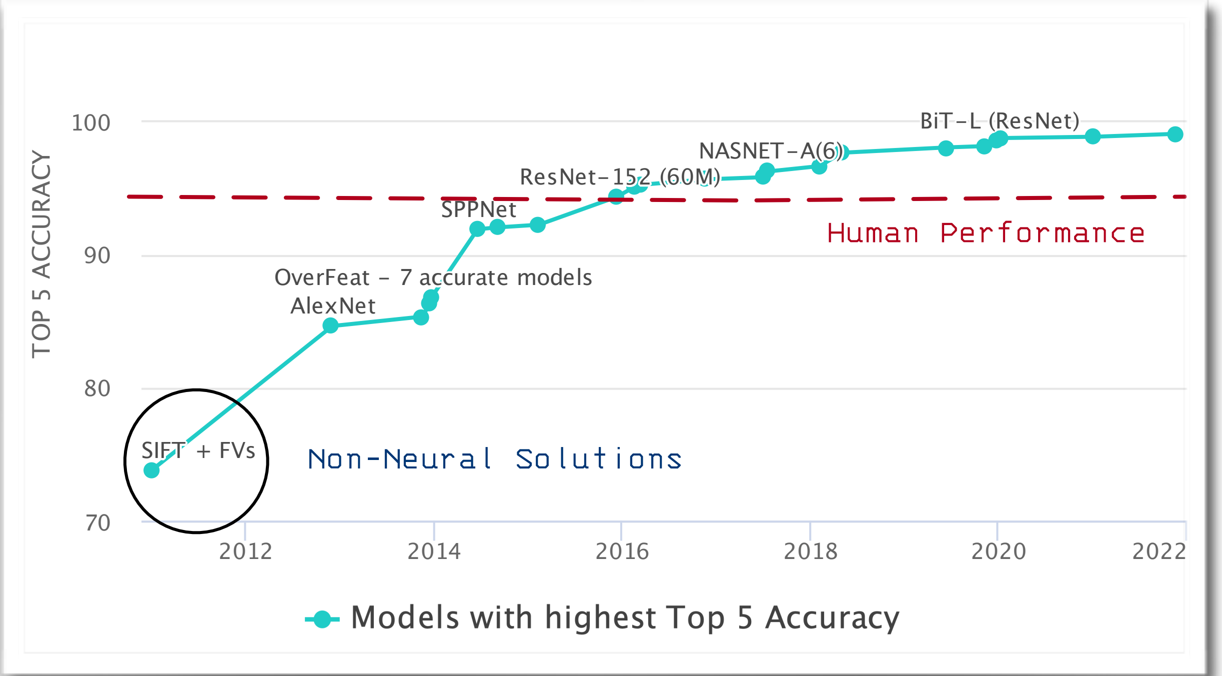


- Main **Features**
  - **Representation** Learning
  - Theoretical Universal Approximators

Kaplan, Jared, et al. "Scaling Laws for Neural Language Models." *arXiv e-prints* (2020).

Deep Neural Networks..

- **Leading** AI solution, **unprecedented** and **super-human** performance

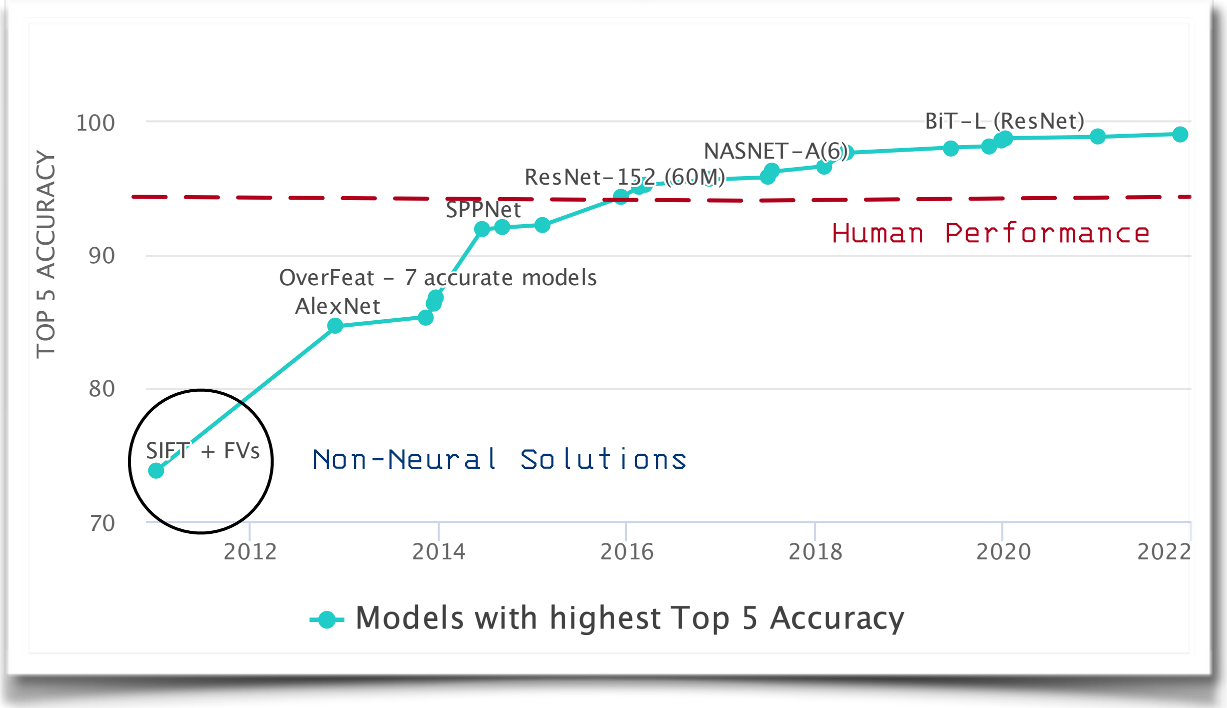


- Main **Features**
  - **Representation** Learning
  - Theoretical Universal Approximators
  - Accuracy **scales** with model size and training epochs

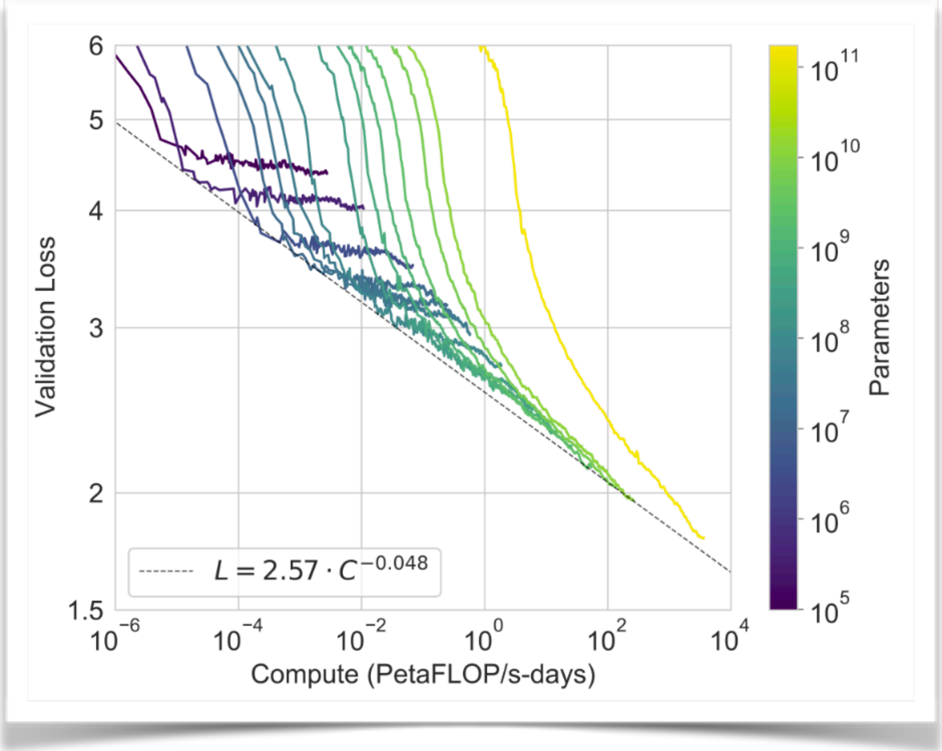
Kaplan, Jared, et al. "Scaling Laws for Neural Language Models." *arXiv e-prints* (2020).

Deep Neural Networks..

- **Leading** AI solution, **unprecedented** and **super-human** performance



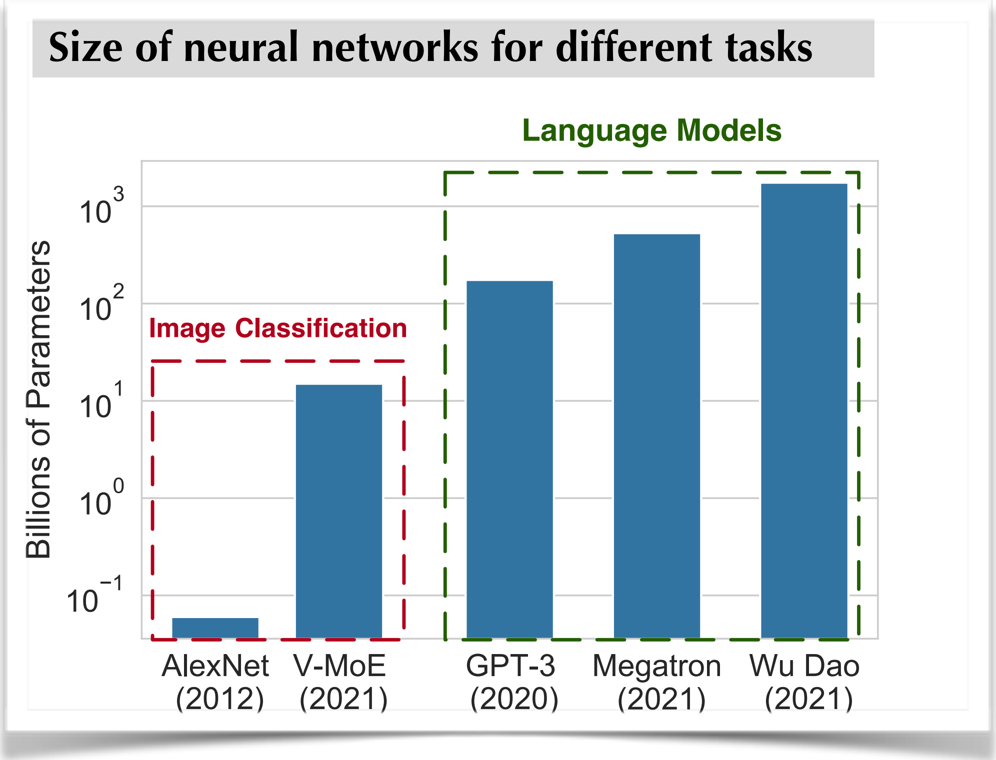
- Main **Features**
  - **Representation** Learning
  - Theoretical Universal Approximators
  - Accuracy **scales** with model size and training epochs



Kaplan, Jared, et al. "Scaling Laws for Neural Language Models." *arXiv e-prints* (2020).

..are Getting Huge

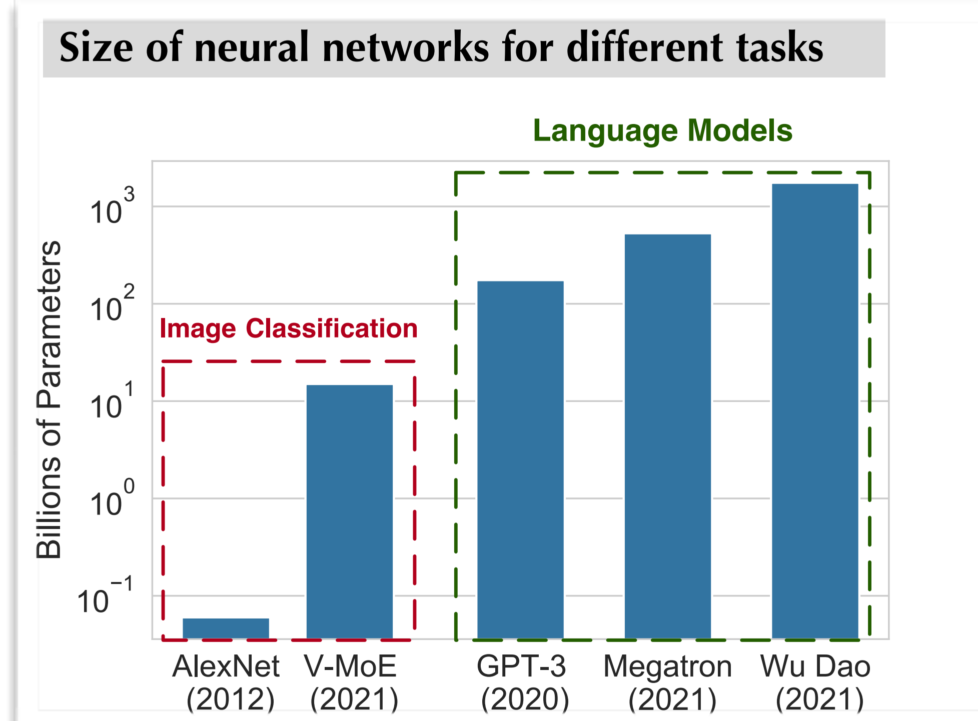
- **Image Classification.** current state-of-the-art **~100x** larger than AlexNet
- **Language Models.** Huge architectures up to 1.75 **trillions** of parameters



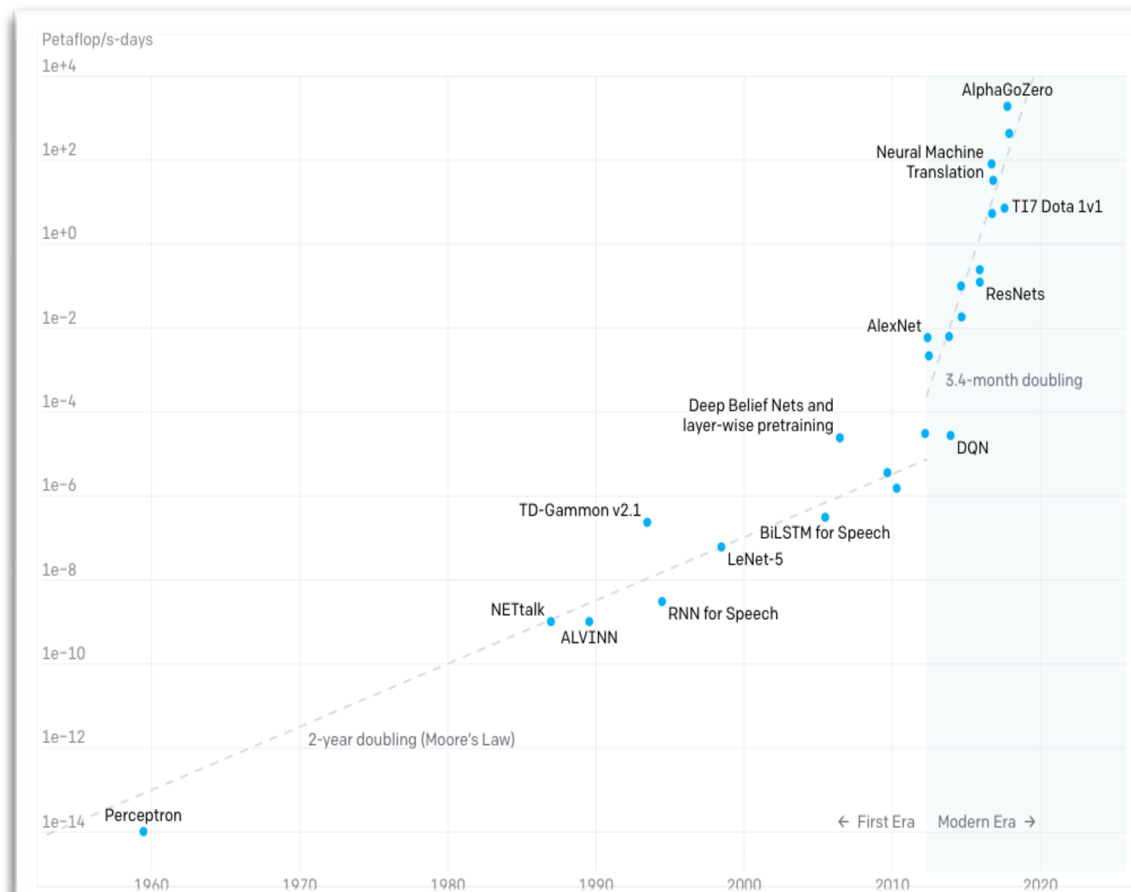
https://openai.com/blog/ai-and-compute/

..are Getting Huge

- **Image Classification.** current state-of-the-art **~100x** larger than AlexNet
- **Language Models.** Huge architectures up to 1.75 **trillions** of parameters



- Consequent growth of **computational burden**
- **Petaflop/s-day** increase faster than Moore's law

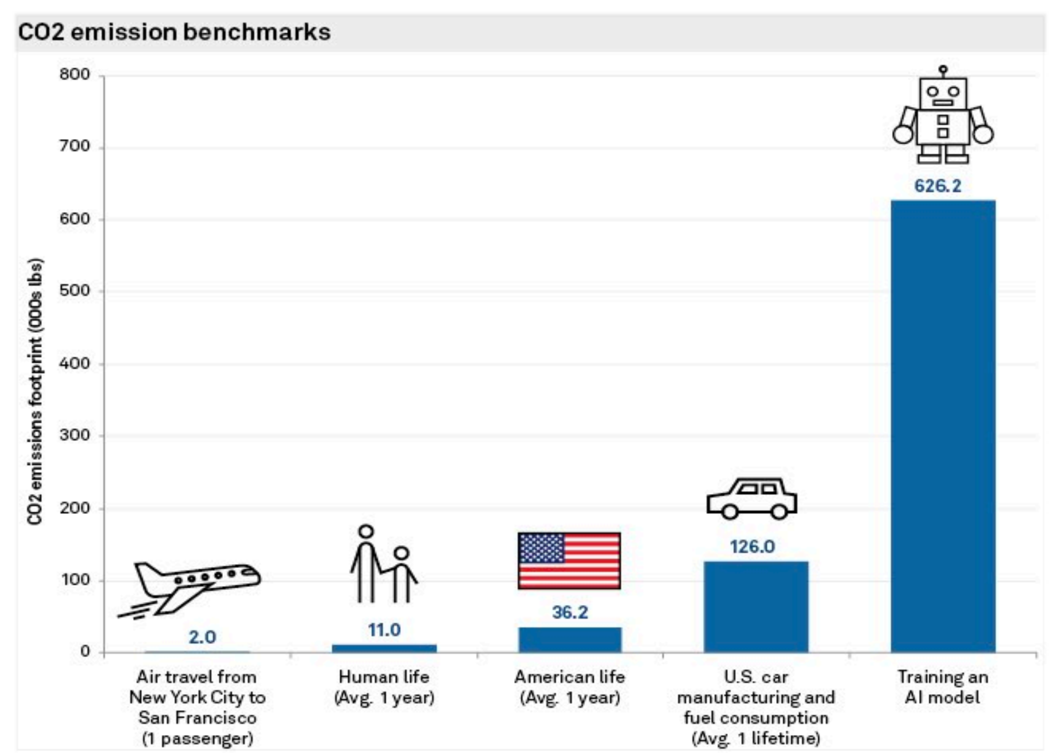


https://openai.com/blog/ai-and-compute/

Training is costly

Model	Hardware	Power (W)	Hours	kWh·PUE	CO_2e	Cloud compute cost
$T2T_{base}$	P100x8	1415.78	12	27	26	\$41–\$140
$T2T_{big}$	P100x8	1515.43	84	201	192	\$289–\$981
ELMo	P100x3	517.66	336	275	262	\$433–\$1472
$BERT_{base}$	V100x64	12,041.51	79	1507	1438	\$3751–\$12,571
$BERT_{base}$	TPUv2x16	—	96	—	—	\$2074–\$6912
NAS	P100x8	1515.43	274,120	656,347	626,155	\$942,973–\$3,201,722
NAS	TPUv2x1	—	32,623	—	—	\$44,055–\$146,848
GPT-2	TPUv3x32	—	168	—	—	\$12,902–\$43,008

Table 3: Estimated cost of training a model in terms of CO_2 emissions (lbs) and cloud compute cost (USD).[7] Power and carbon footprint are omitted for TPUs due to lack of public information on power draw for this hardware.

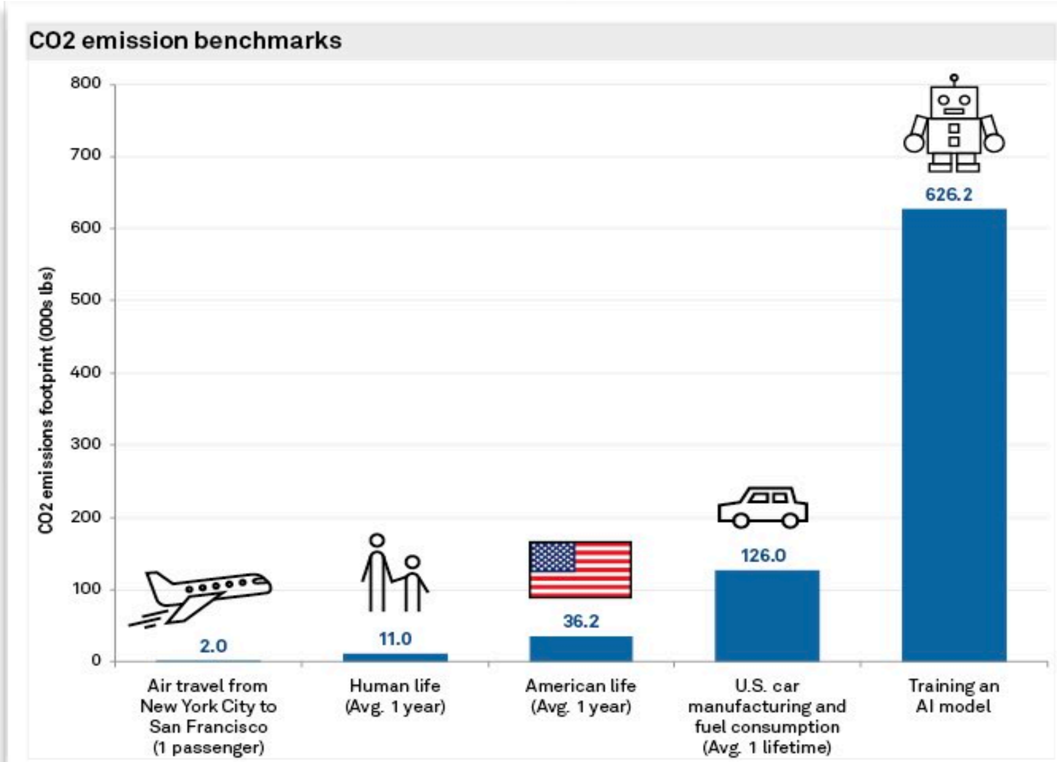


Strubell, Emma, Ananya Ganesh, and Andrew McCallum. "Energy and Policy Considerations for Deep Learning in NLP." *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2019.

Training is costly

Model	Hardware	Power (W)	Hours	kWh·PUE	CO_2e	Cloud compute cost
$T2T_{base}$	P100x8	1415.78	12	27	26	\$41–\$140
$T2T_{big}$	P100x8	1515.43	84	201	192	\$289–\$981
ELMo	P100x3	517.66	336	275	262	\$433–\$1472
$BERT_{base}$	V100x64	12,041.51	79	1507	1438	\$3751–\$12,571
$BERT_{base}$	TPUv2x16	—	96	—	—	\$2074–\$6912
NAS	P100x8	1515.43	274,120	656,347	626,155	\$942,973–\$3,201,722
NAS	TPUv2x1	—	32,623	—	—	\$44,055–\$146,848
GPT-2	TPUv3x32	—	168	—	—	\$12,902–\$43,008

Table 3: Estimated cost of training a model in terms of CO_2 emissions (lbs) and cloud compute cost (USD).[7] Power and carbon footprint are omitted for TPUs due to lack of public information on power draw for this hardware.



Strubell, Emma, Ananya Ganesh, and Andrew McCallum. "Energy and Policy Considerations for Deep Learning in NLP." *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2019.

Inference

- **A lot** of inferences
  - 200 trillions of inference per day at Facebook[1]
  - 90% of **workload** spent on inference at Amazon, NVIDIA[2]
- Inference is resource **constrained** on the **edge** (IoT, Industry 4.0)

Phase	**Freq.**	**FLOPs**	**Devices**	**Constraints**
Training	1	10^{15} (day)	Cloud, Servers	None
Inference	∞	$10^{9\div12}$	Embedded smartphones PC	Memory Time Energy

[1]https://engineering.fb.com/data-center-engineering/accelerating-infrastructure/

[2]https://arxiv.org/pdf/2104.10350.pdf

Over-parametrization

- More **equations** (parameters) than **unknowns** (data samples)

- In general

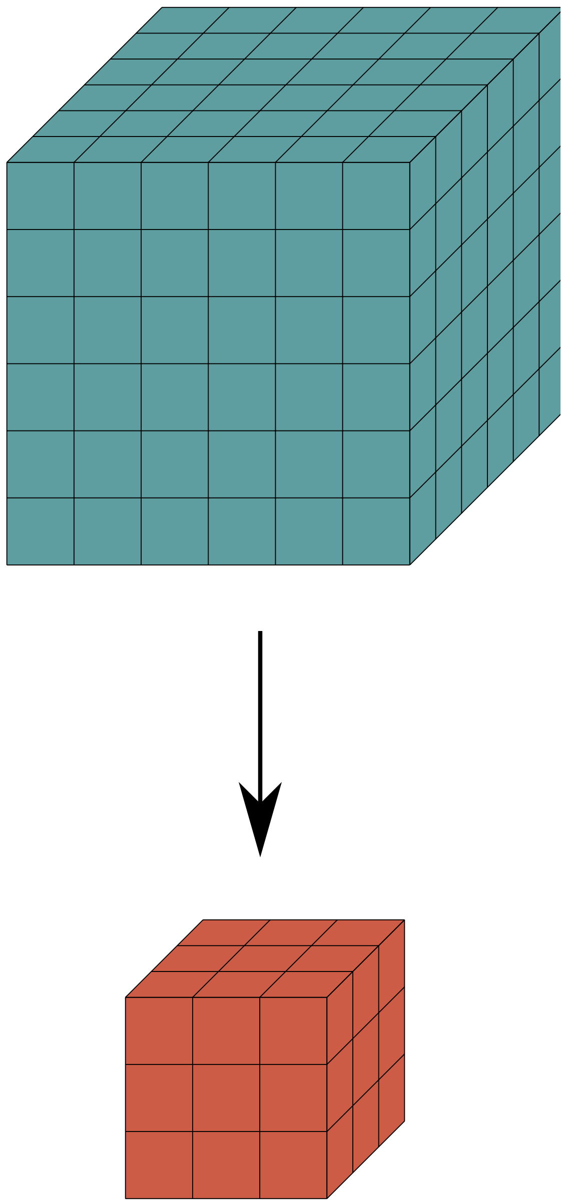
  ↓ Over-fitting

  ↓ Poor performances

- Neural Networks

  ↑ **Eases** optimization

  ↑ **Increases** generalization



Bubeck, Sébastien, and Mark Sellke. "A universal law of robustness via isoperimetry." *Advances in Neural Information Processing Systems* 34 (2021).

"Pluralitas non est ponenda sine necessitate"

- novacula Occami

Model Compression

Model Compression

- Leverages over-parametrization to **compress** DNNs **without** accuracy **degradation**

- **Reducing**
  - Memory impact
  - Inference time
  - Energy consumption

- Main methods
  - **Pruning**
  - **Quantization**
  - **Knowledge Distillation**
  - and more..

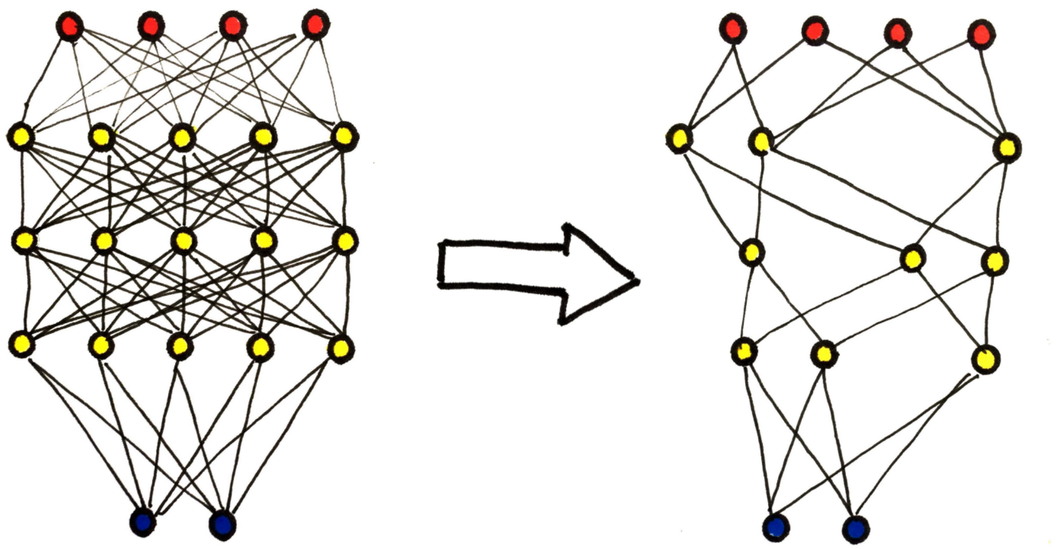
Pruning

Pruning

- Pruning techniques remove **unnecessary** parameters from neural networks

- **Removing = set to 0**

- Reduces **memory** impact, **energy** consumption and speedup **inference**

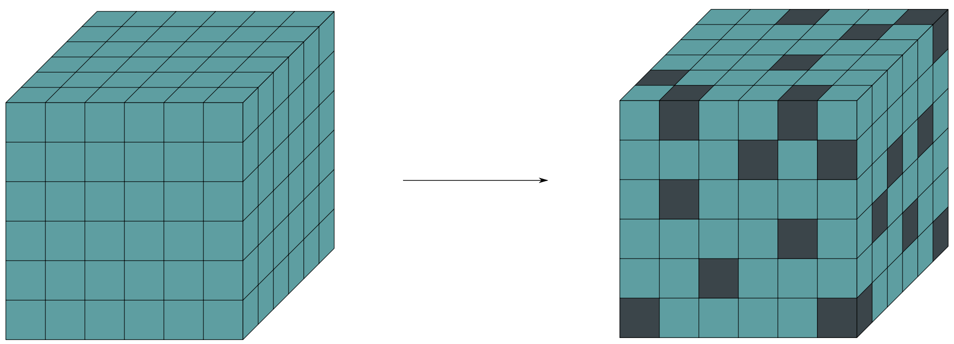


Element-wise vs Structured

- **Element-wise.** Removes single weights producing sparse tensors

  ↑ High memory compression

  ↓ Requires sparse multiplication

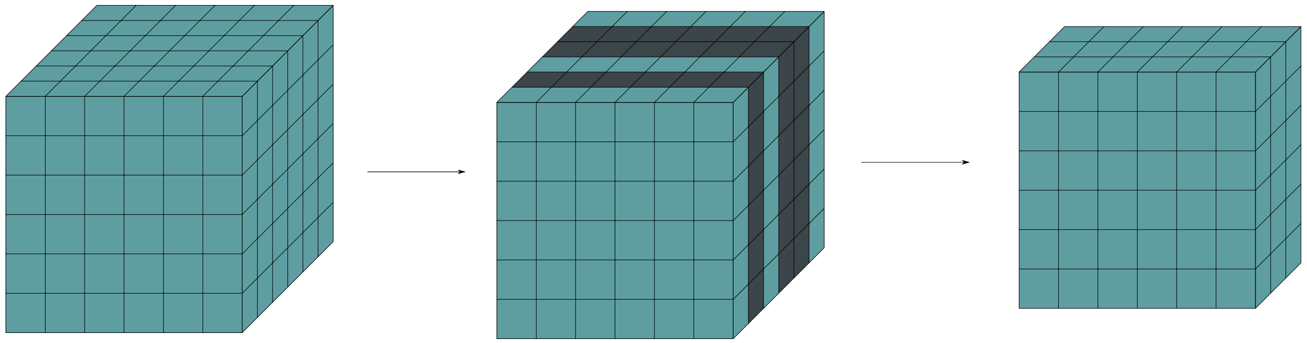


Element-wise

- **Structured.** Removes entire structures (columns, filters)

  ↑ Direct speedup

  ↓ Reduced memory compression



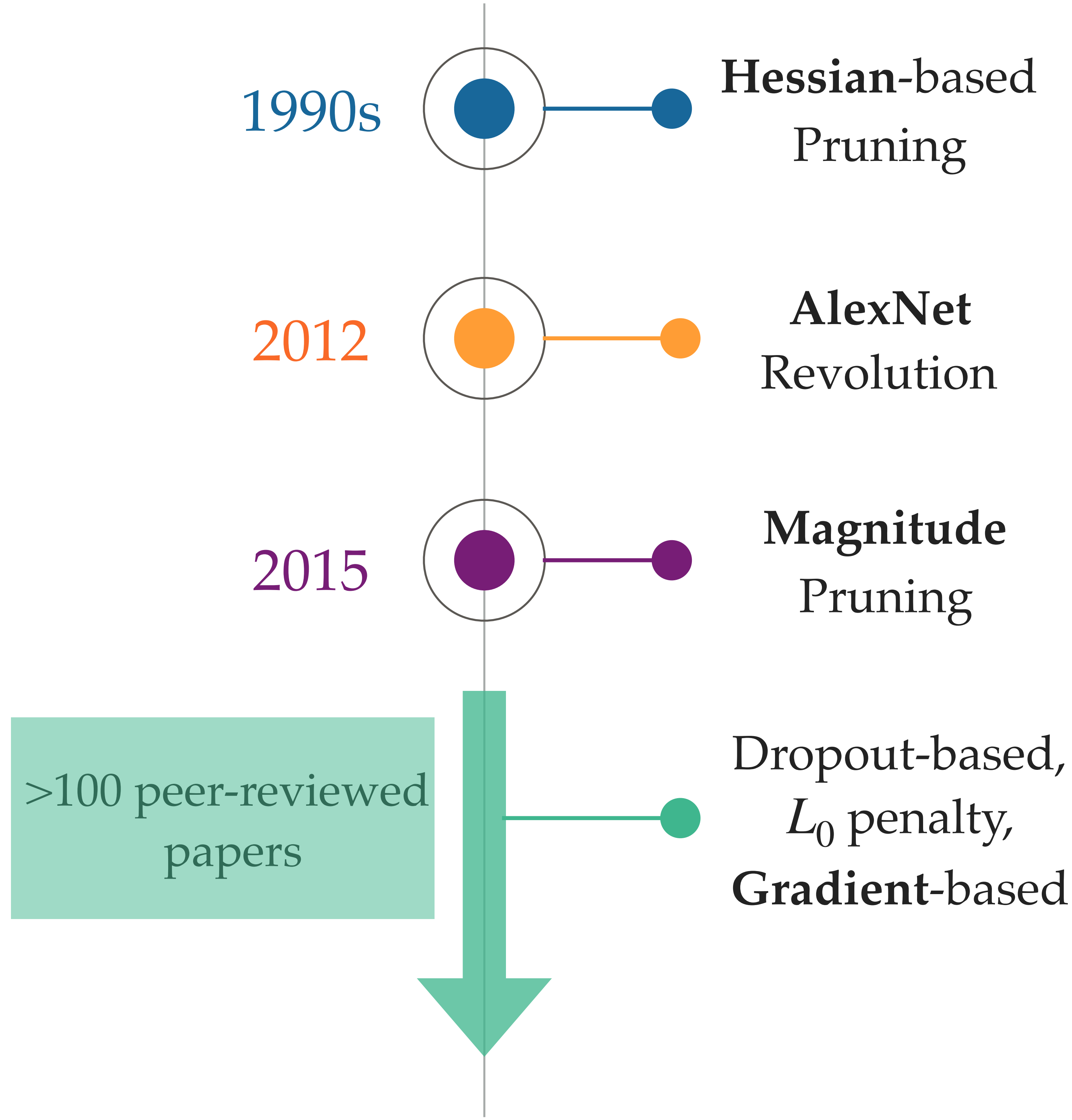
Structured

What to Prune?

- How to **select** which the parameters to prune?
- With n parameters, 2^n possible pruning patterns
- **Heuristic** to estimate weight importance, or **penalty** to induce sparsity

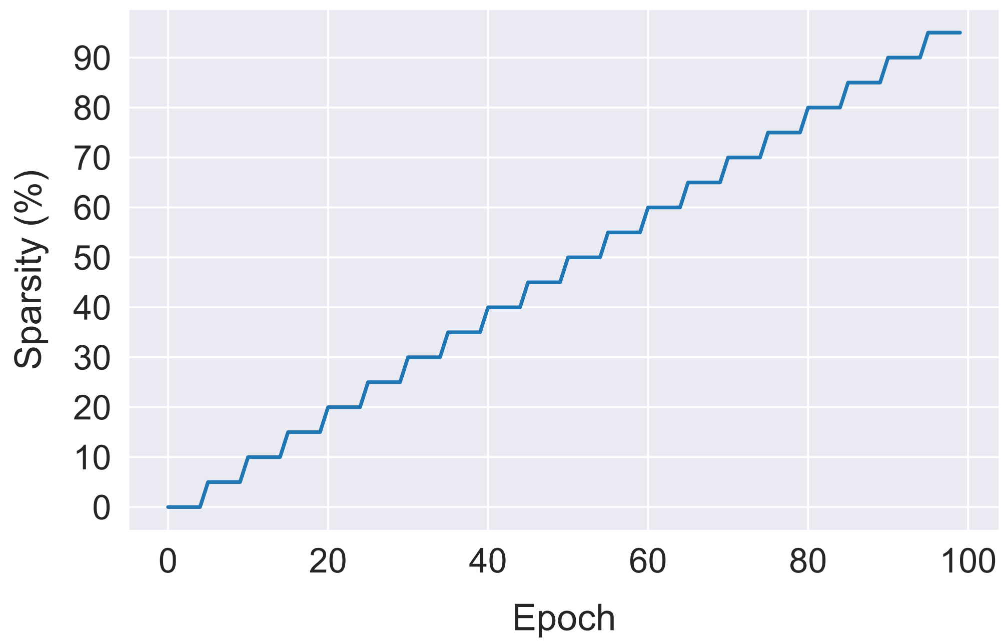
What to Prune?

- How to **select** which the parameters to prune?
- With n parameters, 2^n possible pruning patterns
- **Heuristic** to estimate weight importance, or **penalty** to induce sparsity

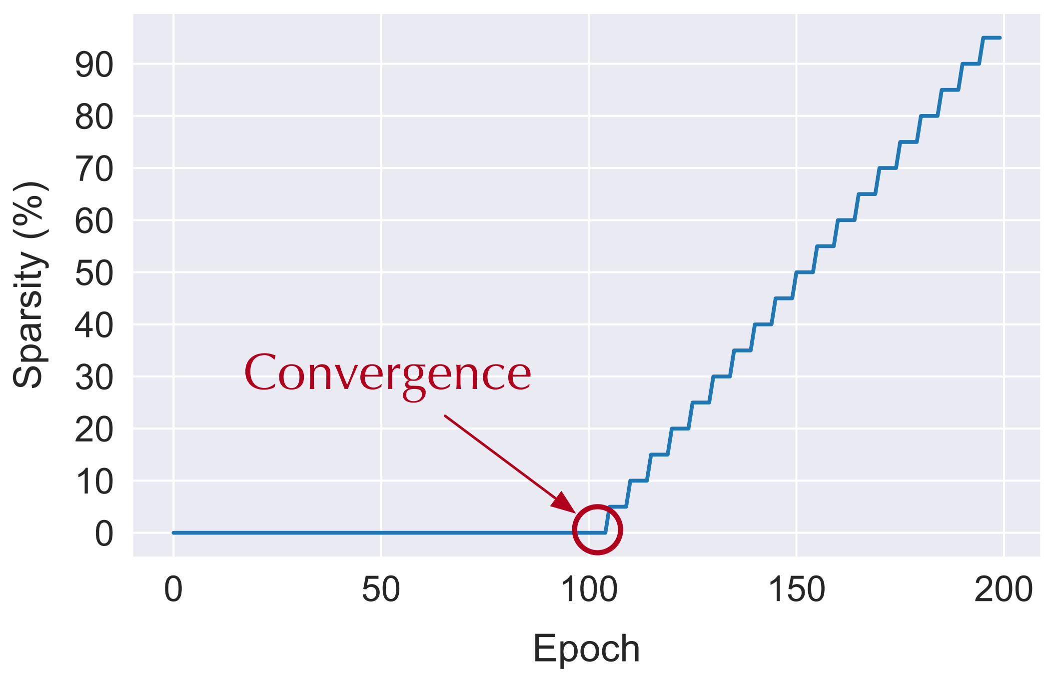


When to Prune?

- **During Training.** The model is **trained** to be sparse
  - Same budget as standard training



- **Fine-tuning.** Pruning is applied on a trained, dense model.
  - Better accuracy



Pruning Performance

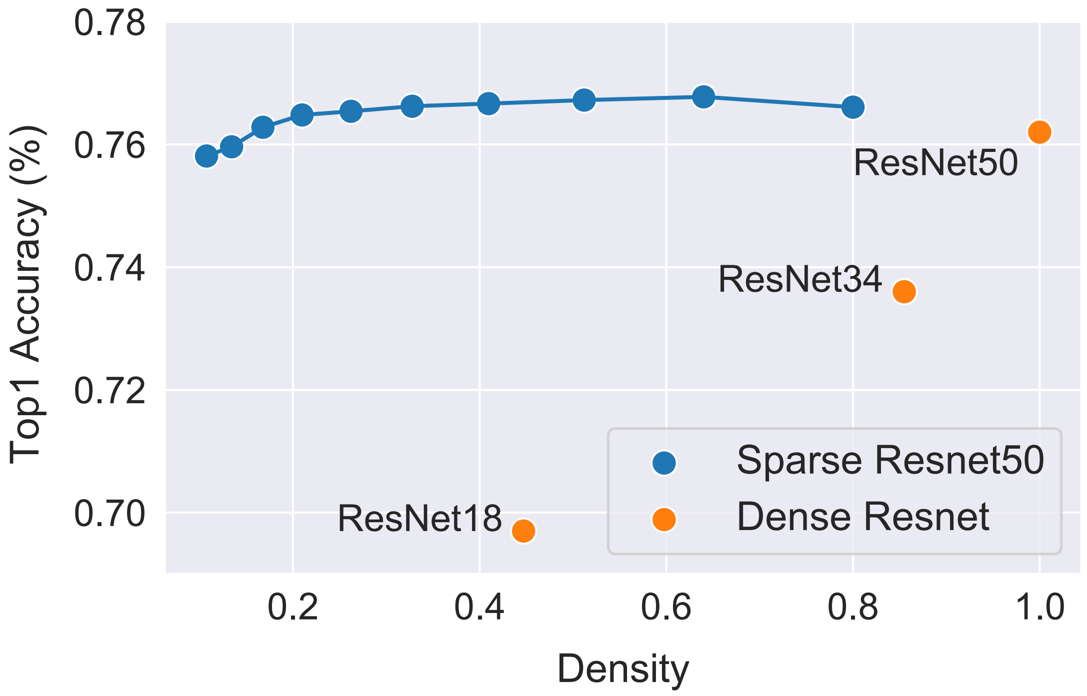
- Magnitude-based, element-wise pruning, ResNet50 on ImageNet

- **Element-Wise** Pruning.

  ↑ 90% sparse, no accuracy drop

  ↑ +6% accuracy w.r.t to dense model w/i same parameters

  ↓ Sparse format overhead not included



Renda, Alex, Jonathan Frankle, and Michael Carbin. "Comparing fine-tuning and rewinding in neural network pruning." *International Conference on Learning Representations.* 2020.

Research Question

- Pruning is a very **effective** compression technique, but

- **RQ1**. Is there any more **principled** and effective heuristic than magnitude?
- **RQ2**. What is the relationship between **learning** and **sparsity**?
- **RQ3**. Can we train sparse network from scratch?
- And many more..

Frankle, Jonathan, and Michael Carbin. "The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks." *International Conference on Learning Representations*. 2018

Quantization

Quantization

- **Classical** Computer Science problem

- **Large input** values set -> **small output** values set

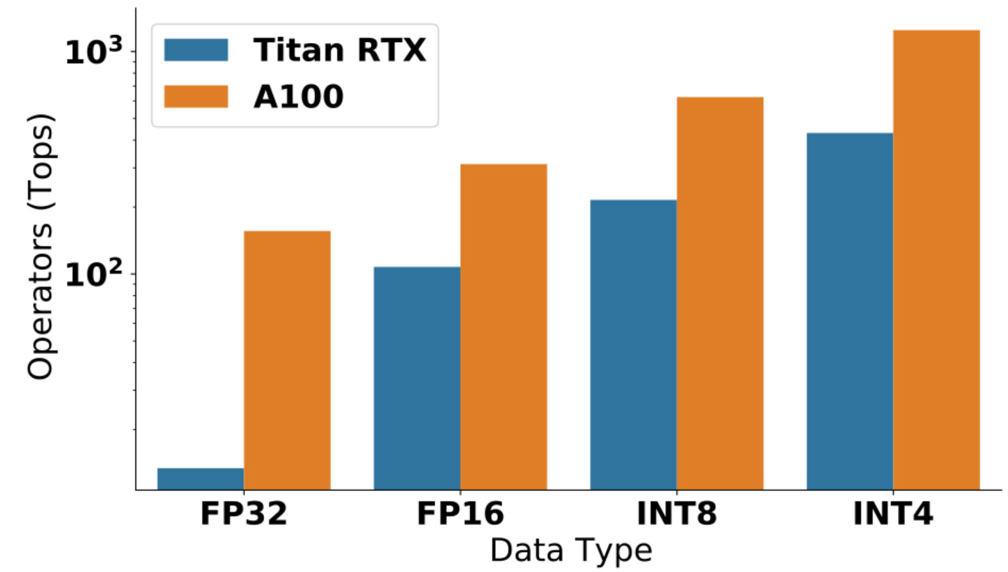
- Specific features of **neural quantization**
  - Heavily **over-parametrized** model
  - **Decoupling** between training and inference

2, 09	7,48	2,92	4, 16
8,25	3,59	1,04	4,66
10,62	5,32	2,63	4,34
0,58	5,08	1,40	8,58

2	7	3	4
7	4	1	5
7	5	3	4
1	5	1	7

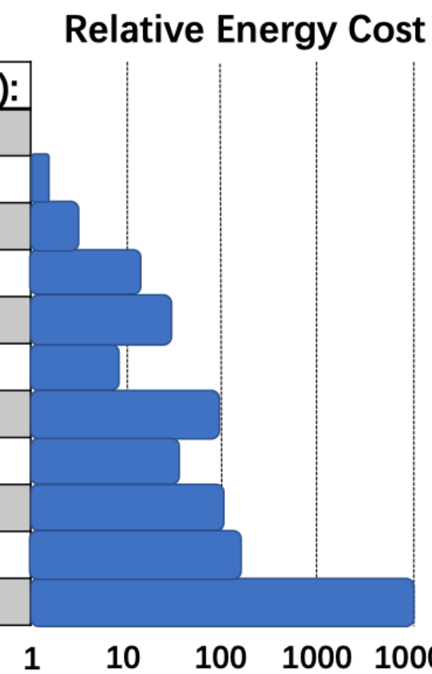
Why Quantization?

- Quantization delivers benefits **both in training and inference**



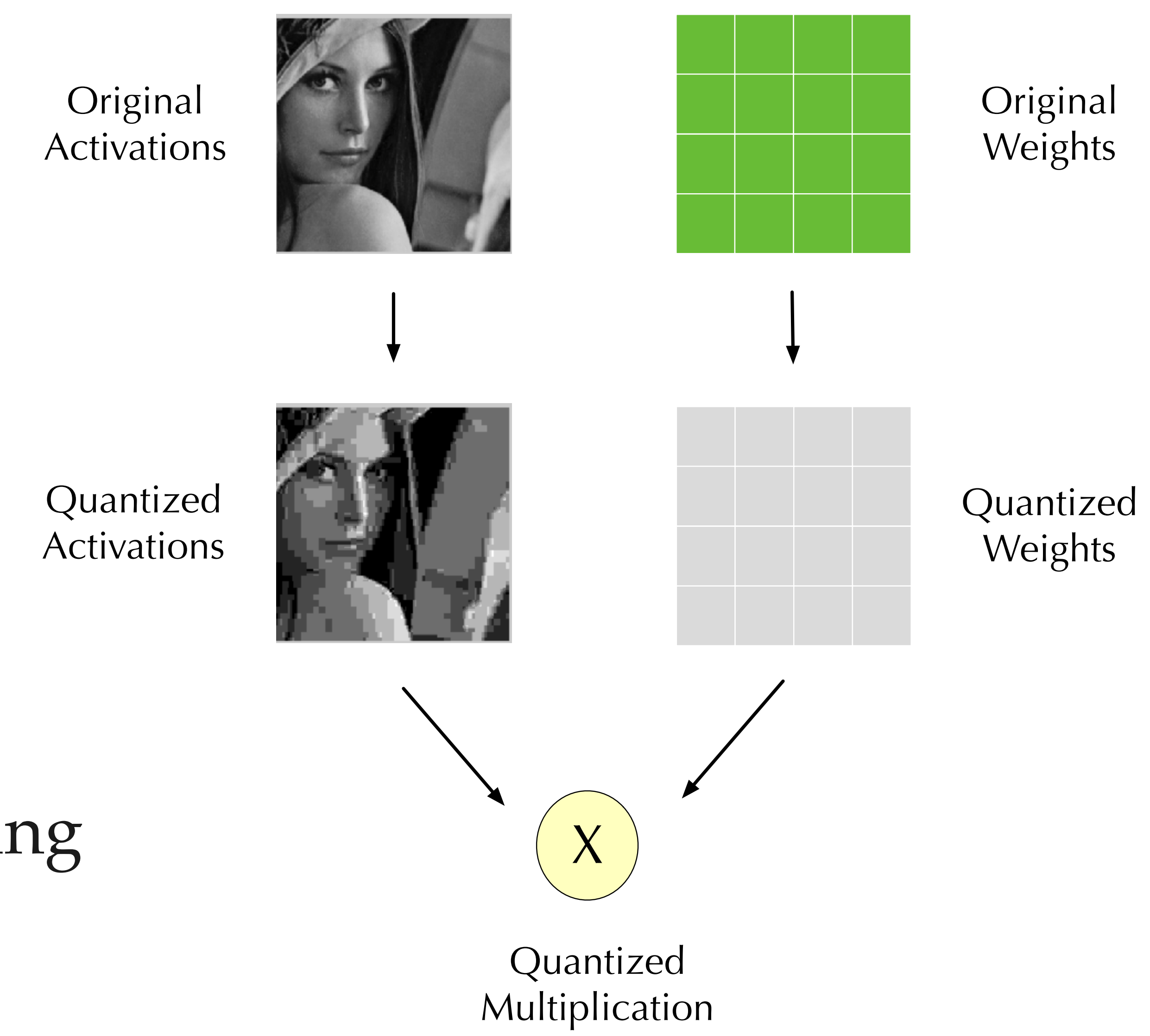
- Quantized models offers
  - Reduced **memory** impact
  - **Faster** operations
  - Reduced **energy** consumption

Operation:	Energy(pJ):
8b Add	0.03
16b Add	0.05
32b Add	0.1
16b FP Add	0.4
32b FP Add	0.9
8b Mult	0.2
32b Mult	3.1
16b FP Mult	1.1
32b FP Mult	3.7
32b SRAM Read (8kb)	5.0
32b DRAM Read	640



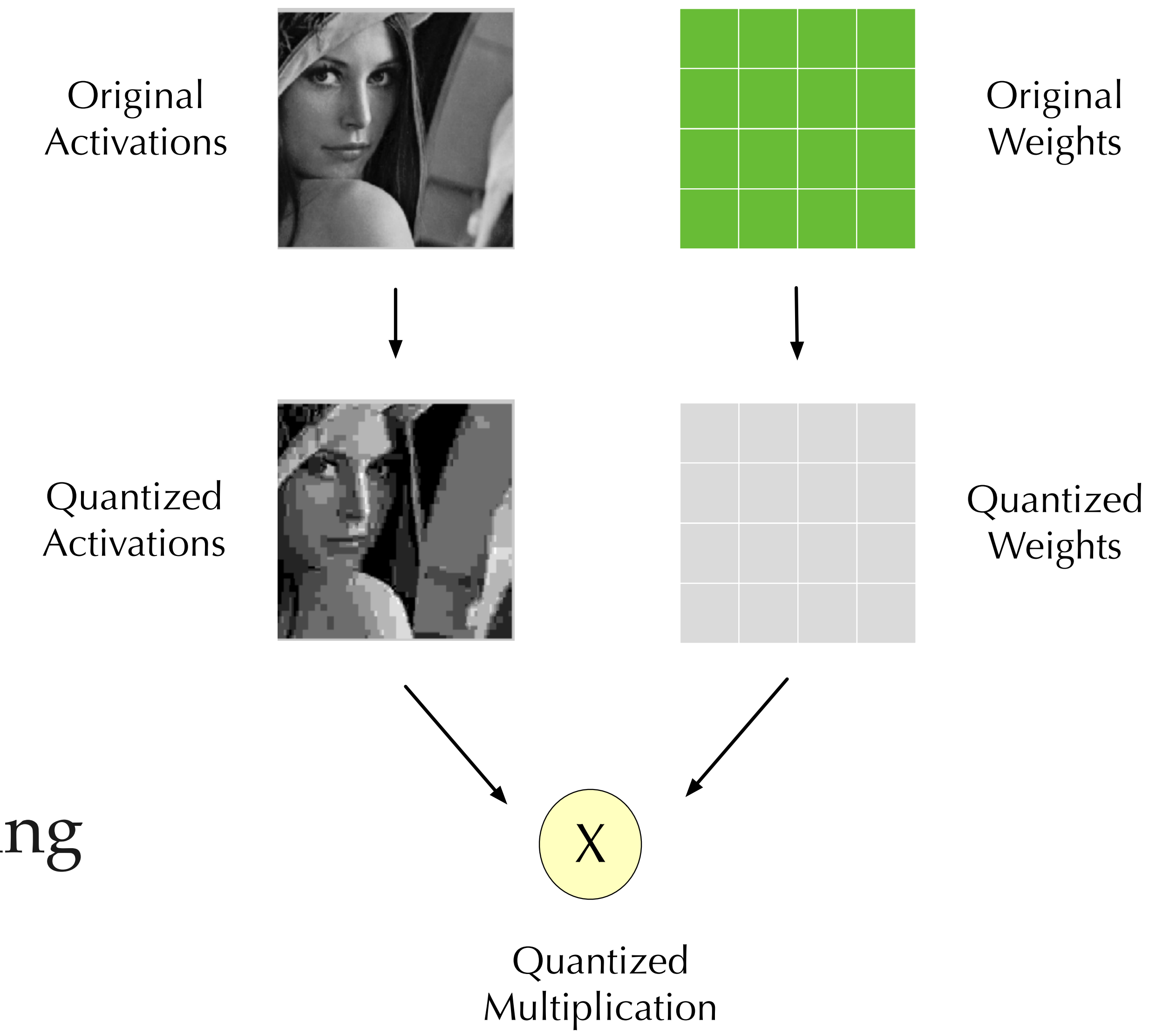
Weights and Activations

- **Quantize weights.**
  - Offline
  - Weights can be **optimized**
- **Quantize activations.**
  - Online (inference time) -> computing stats is **costly** (min, max,..)
  - No optimization



Weights and Activations

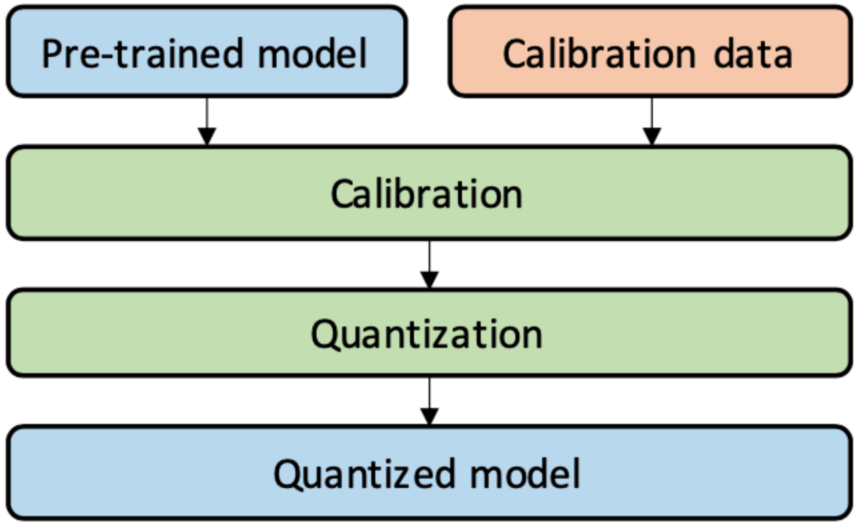
- **Quantize weights.**
  - Offline
  - Weights can be **optimized**
- **Quantize activations.**
  - Online (inference time) -> computing stats is **costly** (min, max,..)
  - No optimization



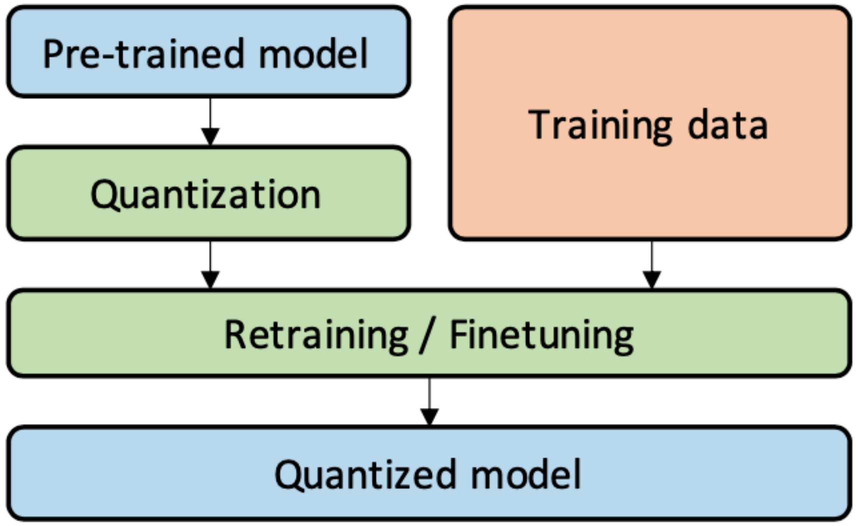
Quantizing activations has a huge impact on accuracy

Fine-Tuning

- **Post-training Quantization (PTQ).**
  - ↑ No re-training (~)
  - ↓ Reduced precision



- **Quantization-Aware Training (QAT)**
  - ↑ High precision
  - ↓ Costly re-training phase



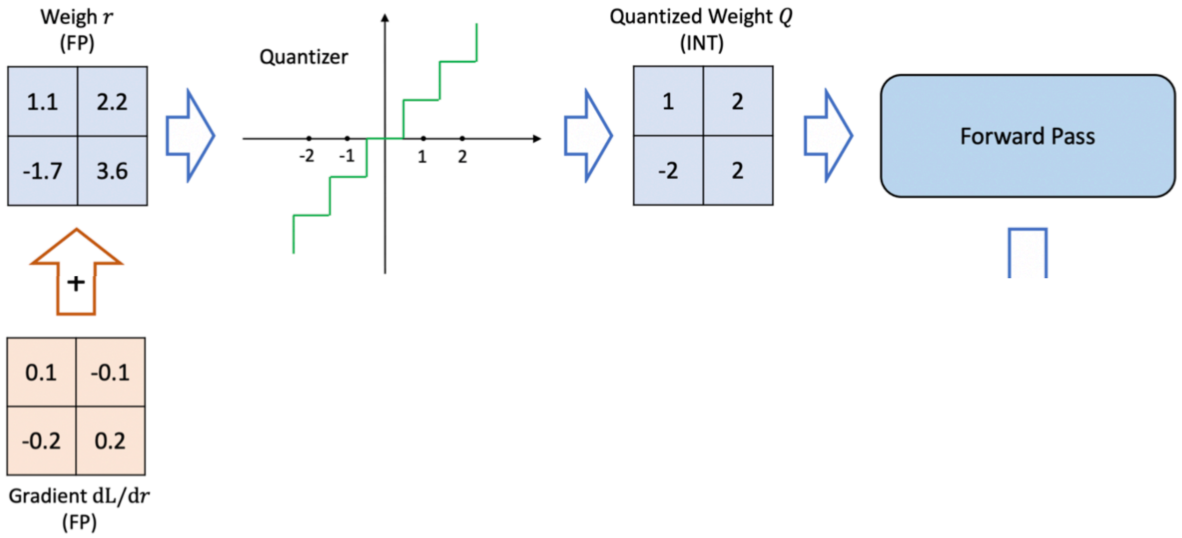
Quantization-Aware Training

- **Methodology**. Weights quantized **after** each gradient update

- **Requirements**. Backward and gradient update in **full-precision** for numerical reasons

- **Problem**. Quantizer gradient is **zero** almost everywhere

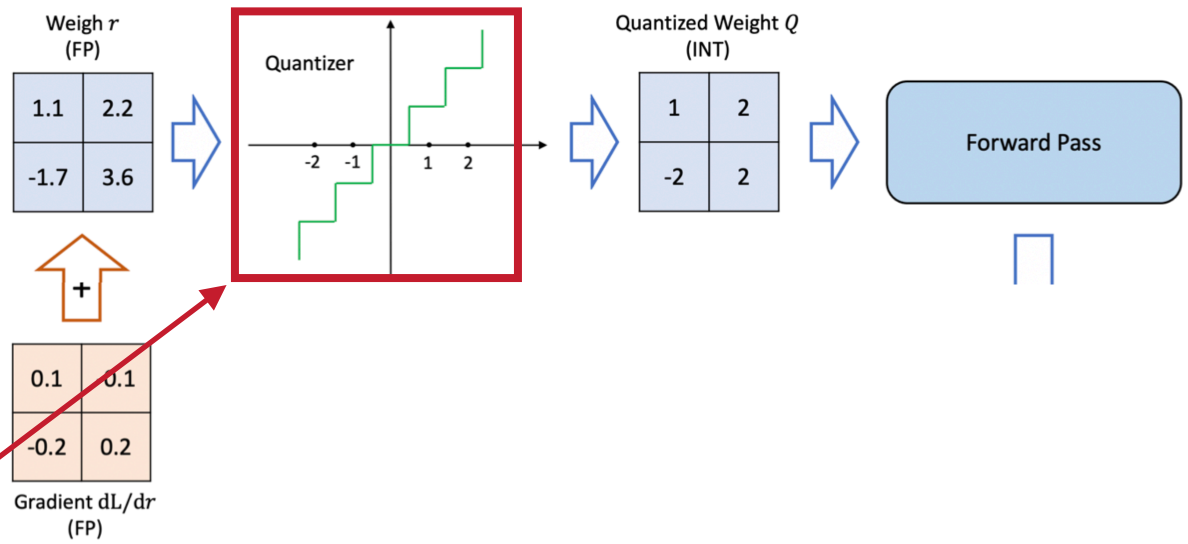
- **Solution**. Straight-Through Estimator (STE)



Quantization-Aware Training

- **Methodology**. Weights quantized **after** each gradient update

- **Requirements**. Backward and gradient update in **full-precision** for numerical reasons



- **Problem**. Quantizer gradient is **zero** almost everywhere

- **Solution**. Straight-Through Estimator (STE)

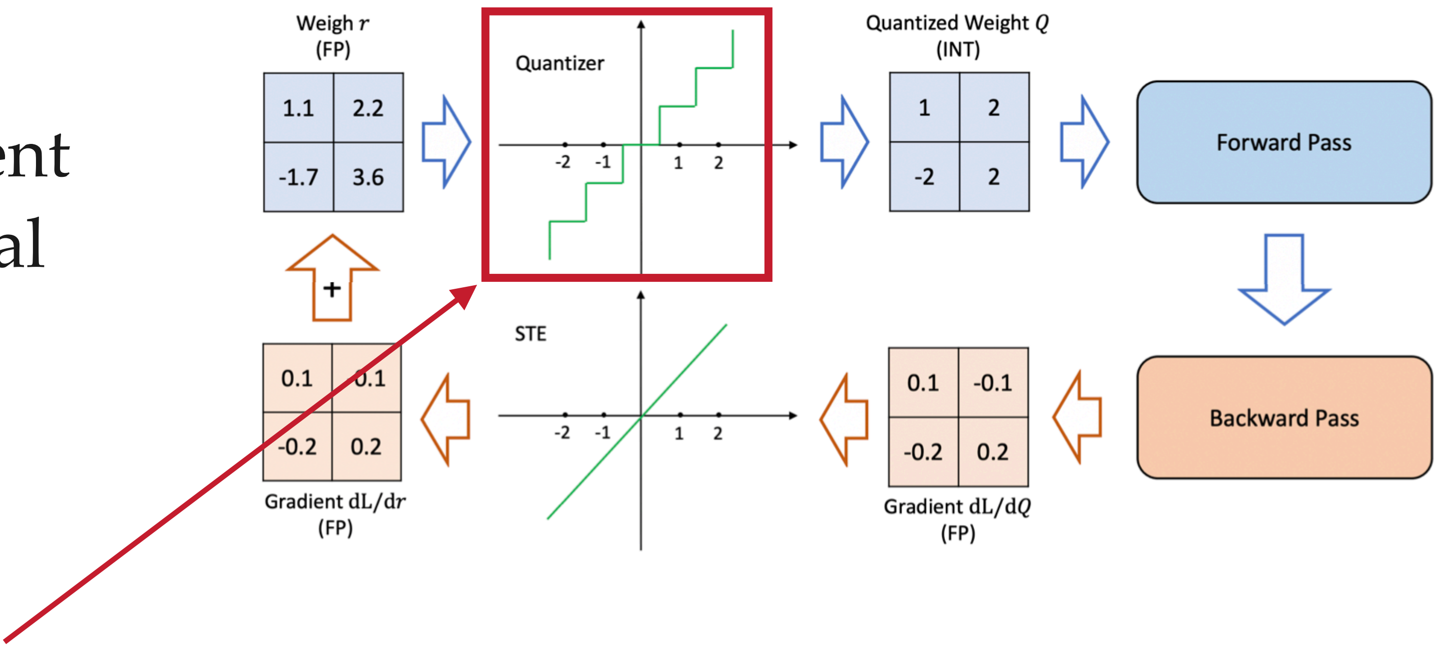
Quantization-Aware Training

- **Methodology**. Weights quantized **after** each gradient update

- **Requirements**. Backward and gradient update in **full-precision** for numerical reasons

- **Problem**. Quantizer gradient is **zero** almost everywhere

- **Solution**. Straight-Through Estimator (STE)



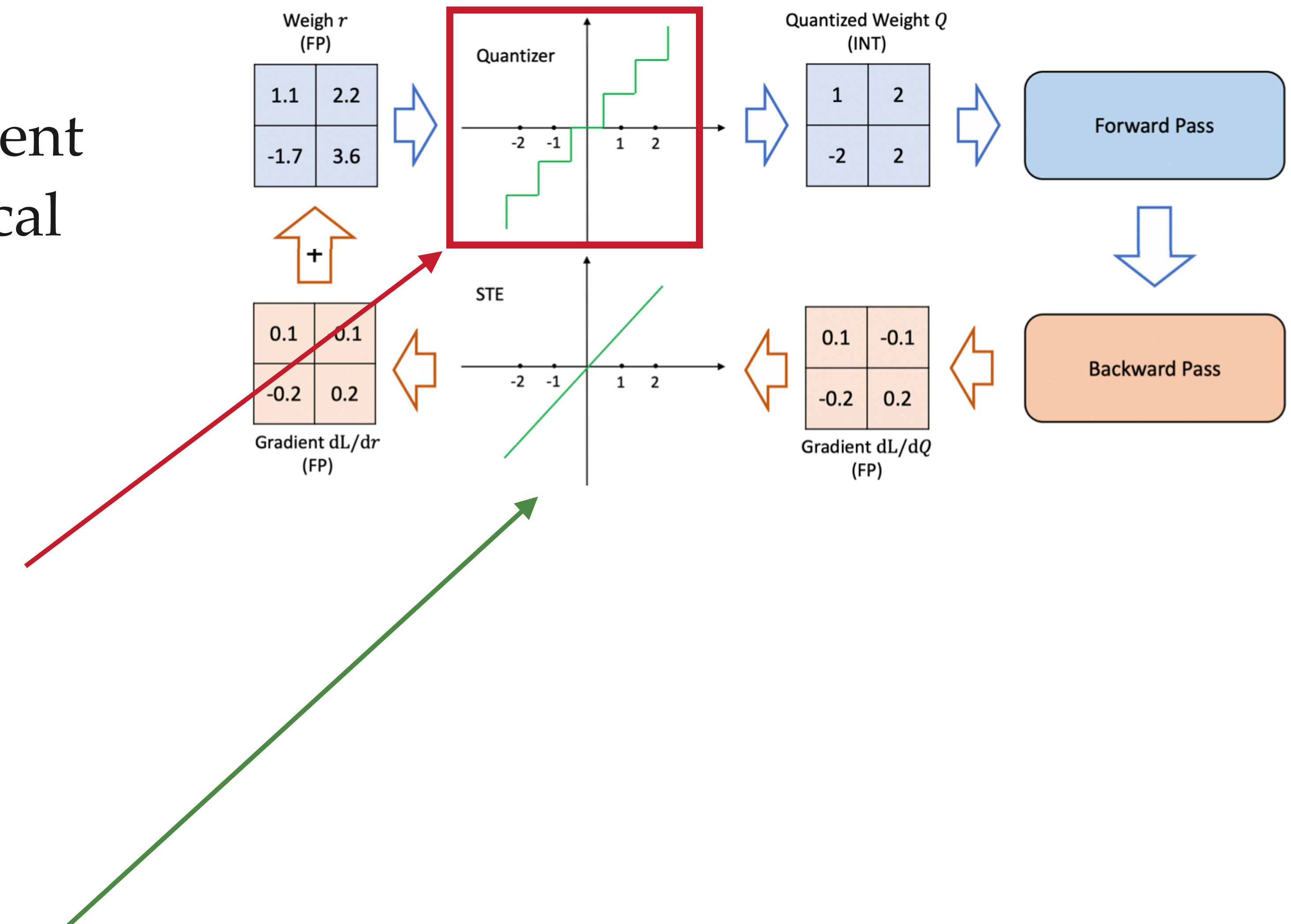
Quantization-Aware Training

- **Methodology**. Weights quantized **after** each gradient update

- **Requirements**. Backward and gradient update in **full-precision** for numerical reasons

- **Problem**. Quantizer gradient is **zero** almost everywhere

- **Solution**. Straight-Through Estimator (STE)



Quantization Performance

- **Fully-quantized** training

Optimizer	Task	Model	Metric	Time	Mem saved
32-bit Momentum	MoCo v2	ResNet-50	67.3	30 days	0.0 GB
8-bit Momentum	MoCo v2	ResNet-50	**67.4**	**28** days	**0.1GB**
32-bit Adam	LM	Transformer-1.5B	9.0	308 days	0.0 GB
8-bit Adam	LM	Transformer-1.5B	9.0	**297** days	**8.5GB**
32-bit Adam	LM	GPT3-Medium	10.62	795 days	0.0 GB
8-bit Adam	LM	GPT3-Medium	10.62	**761days**	**1.7GB**

- **PTQ vs QAT** - ResNet18 on Imagenet
  - PTQ ~0.1 training budget w.r.t. QAT
  - QAT lossless quantization up to 3/3

W/A	Approach	Top1
Baseline	PTQ	71.1
	QAT	69.9
4/4	PTQ	69.1
	QAT	70.6
3/3	PTQ	65.6
	QAT	69.7
2/2	PTQ	51.1
	QAT	67.0

Quantization Performance

- **Fully-quantized** training

Optimizer	Task	Model	Metric	Time	Mem saved
32-bit Momentum	MoCo v2	ResNet-50	67.3	30 days	0.0 GB
8-bit Momentum	MoCo v2	ResNet-50	**67.4**	**28** days	**0.1GB**
32-bit Adam	LM	Transformer-1.5B	9.0	308 days	0.0 GB
8-bit Adam	LM	Transformer-1.5B	9.0	**297** days	**8.5GB**
32-bit Adam	LM	GPT3-Medium	10.62	795 days	0.0 GB
8-bit Adam	LM	GPT3-Medium	10.62	**761days**	**1.7GB**

Dettmers, Tim, et al. "8-bit Optimizers via Block-wise Quantization." *International Conference on Learning Representations*. 2022.

- **PTQ vs QAT** - ResNet18 on Imagenet
  - PTQ ~0.1 training budget w.r.t. QAT
  - QAT lossless quantization up to 3/3

W/A	Approach	Top1
Baseline	PTQ	71.1
	QAT	69.9
4/4	PTQ	69.1
	QAT	70.6
3/3	PTQ	65.6
	QAT	69.7
2/2	PTQ	51.1
	QAT	67.0

Wei, Xiuying, et al. "QDrop: Randomly Dropping Quantization for Extremely Low-bit Post-Training Quantization." *International Conference on Learning Representations*. 2021.
Lee, Junghyup, Dohyung Kim, and Bumsub Ham. "Network quantization with element-wise gradient scaling." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021

Research Question

- Quantization is an extremely effective solution

- **RQ1**. Can we produce extreme low-bits models as effective as full-precision ones?
- **RQ2**. Can we go beyond STE?
- **RQ3.** Can we use FPGA and ASIC to fully leverage the benefit of quantization?
- And many more..

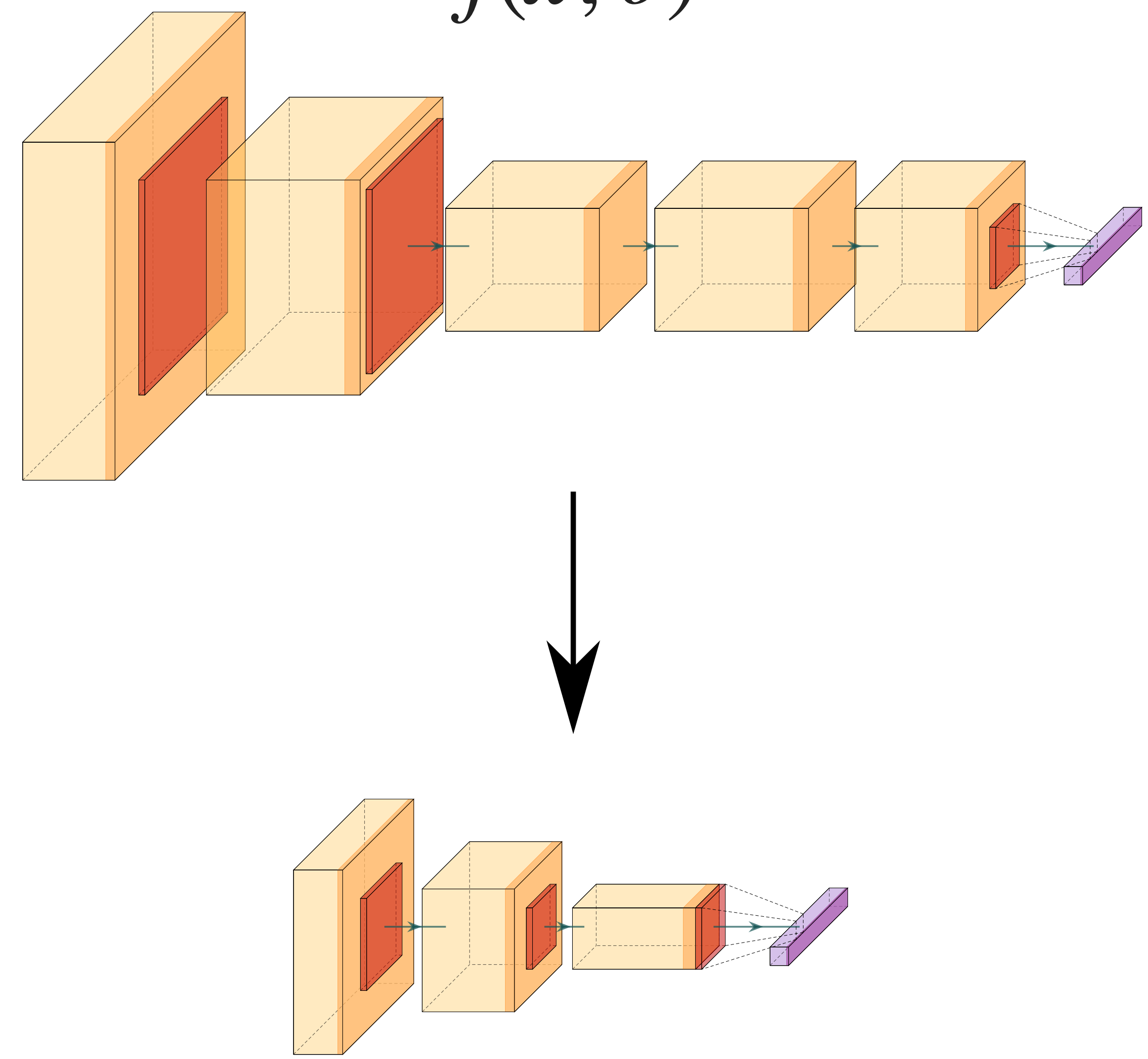
Frankle, Jonathan, and Michael Carbin. "The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks." *International Conference on Learning Representations*. 2018

Knowledge Distillation

Knowledge Distillation

- Training **paradigm** that involves
  - **Student**: the model to be trained. Small, shallow and deployment oriented
  - **Teacher**: pre-trained. Deep and effective
- The student cannot learn the same function $f(\boldsymbol{x}, \theta)$ as the teacher **extrapolating** it from the examples
- It could by **mimicking** its outputs on the samples

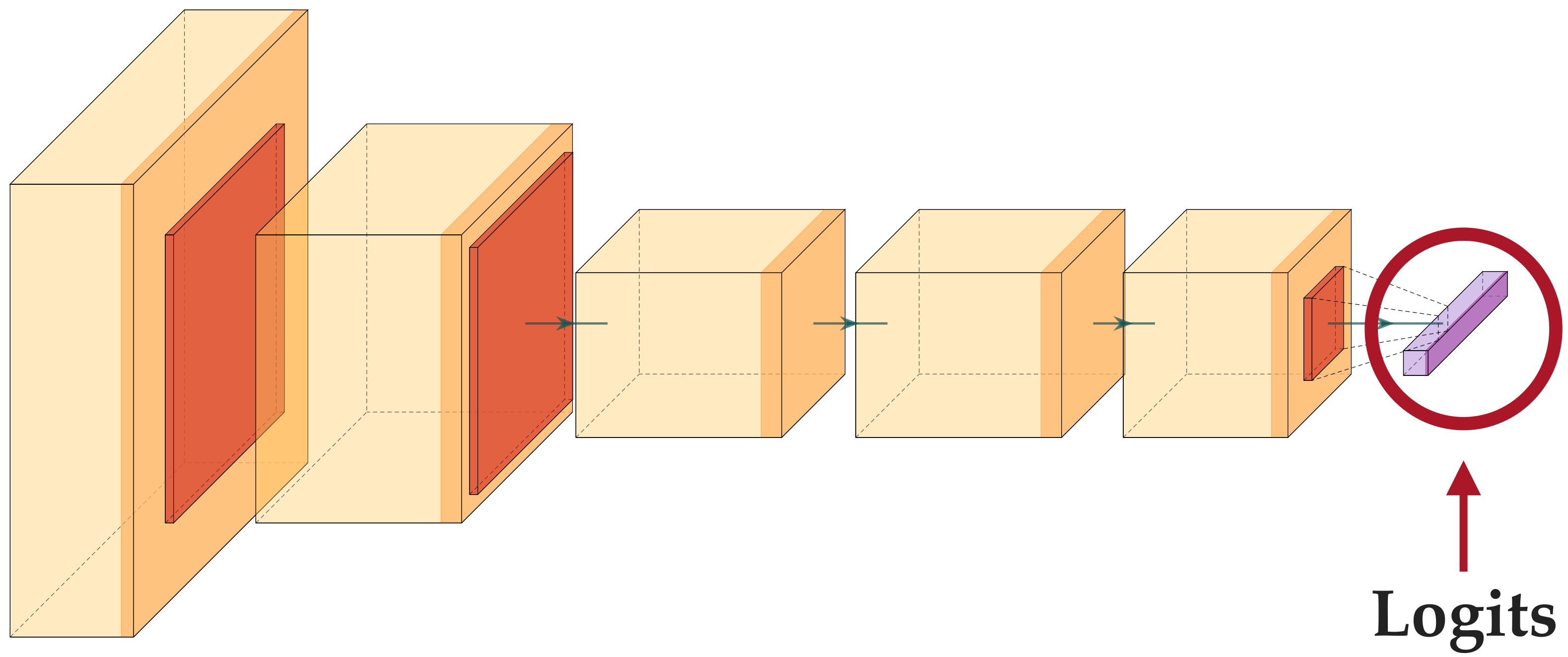
$f(\boldsymbol{x}, \theta)$



$f(\boldsymbol{x}, \theta')$

$$\boxed{f(x, \theta') \sim f(x, \theta)}$$

Logits

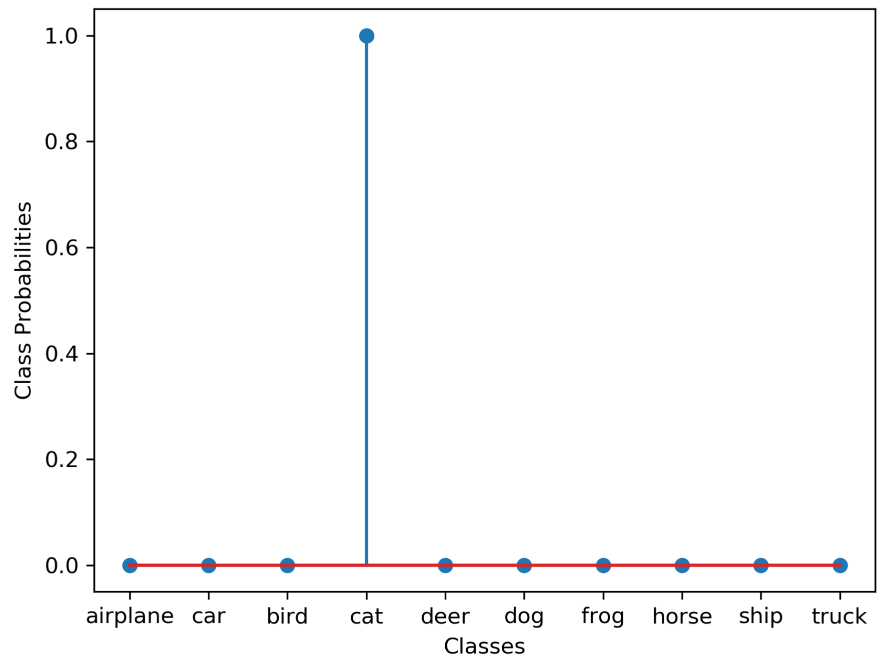


- **Logits.** $z \in R^c$, with c number of classes.

- **Class Probabilities.** $p_i = \textbf{softmax}(z_i)$

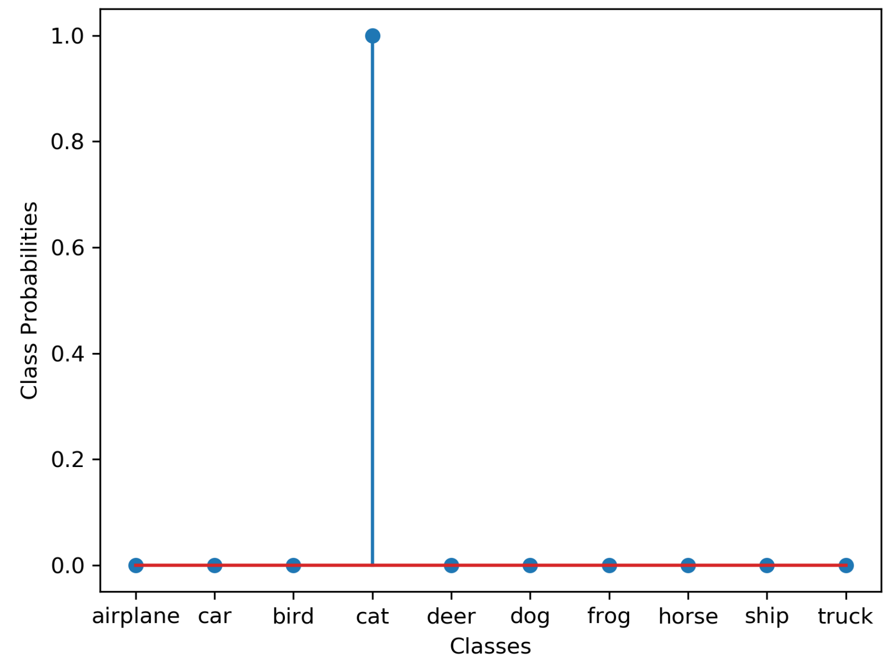
Logits Approximation

- **One-hot encoded** label
  - Single class information

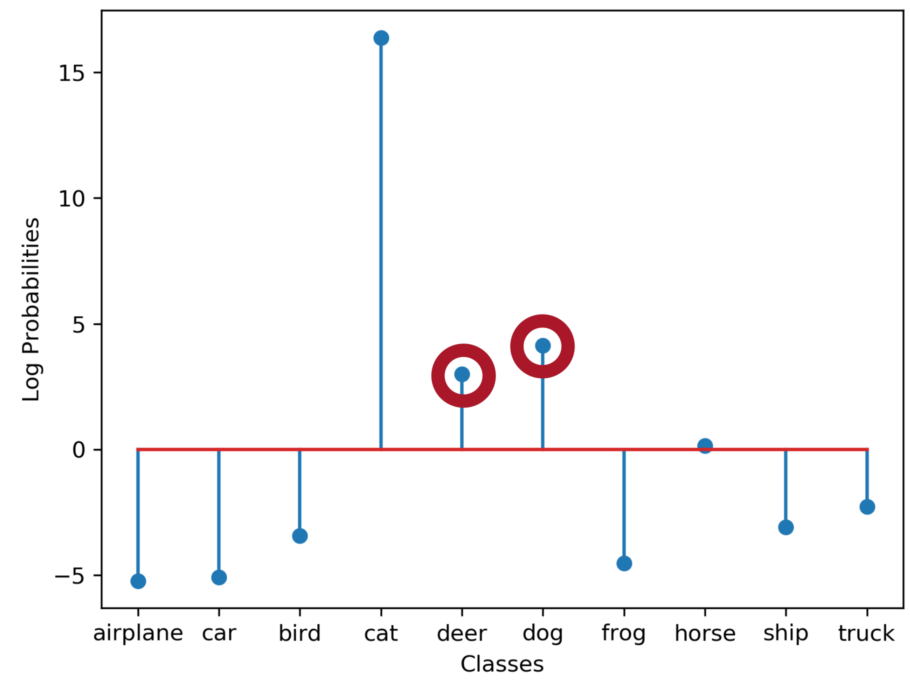


Logits Approximation

- **One-hot encoded** label
  - Single class information

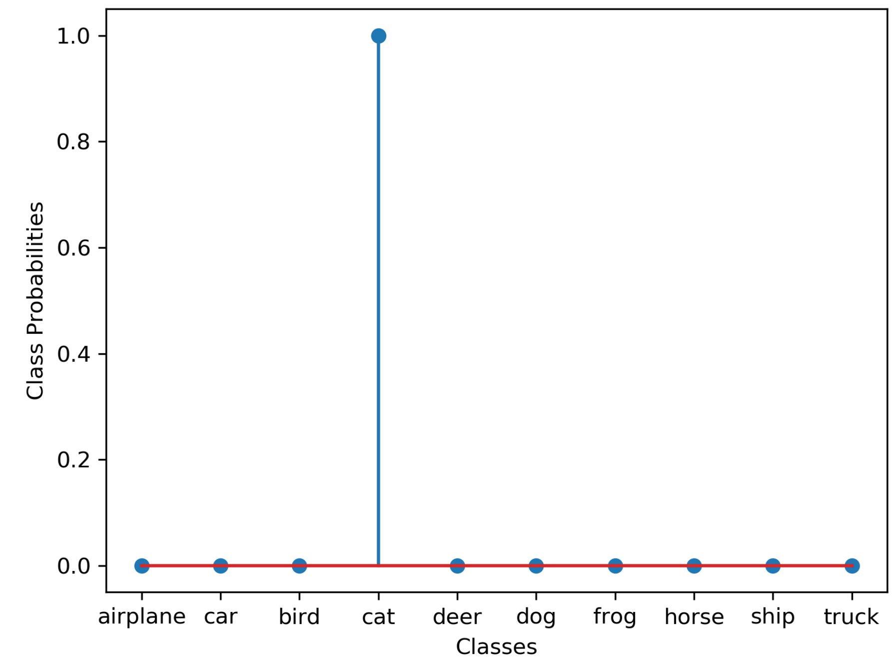


- Teacher **logits.**
  - **Multi-class** and **intra-class** information

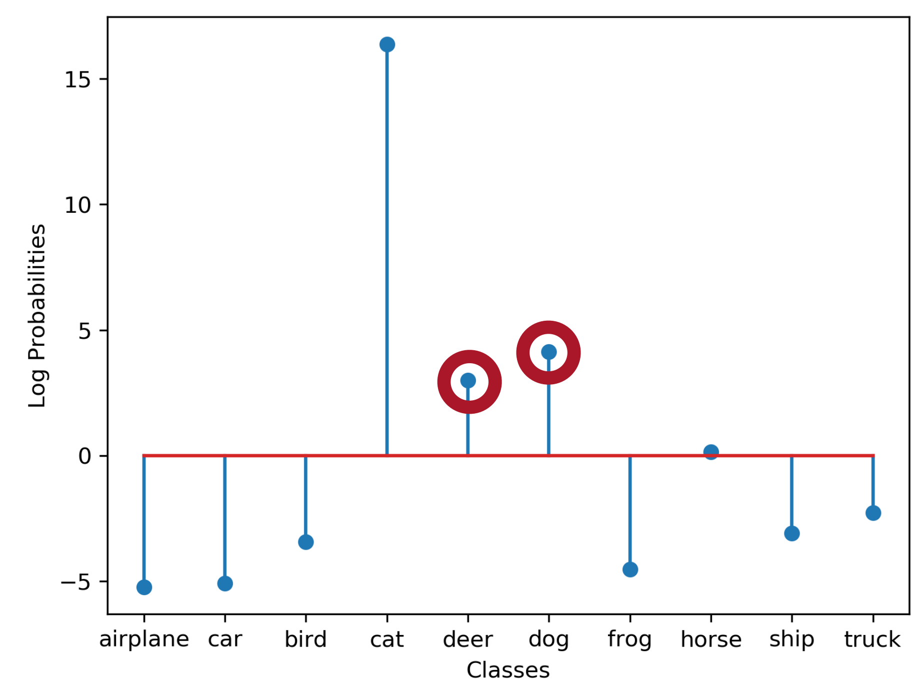


Logits Approximation

- **One-hot encoded** label
  - Single class information



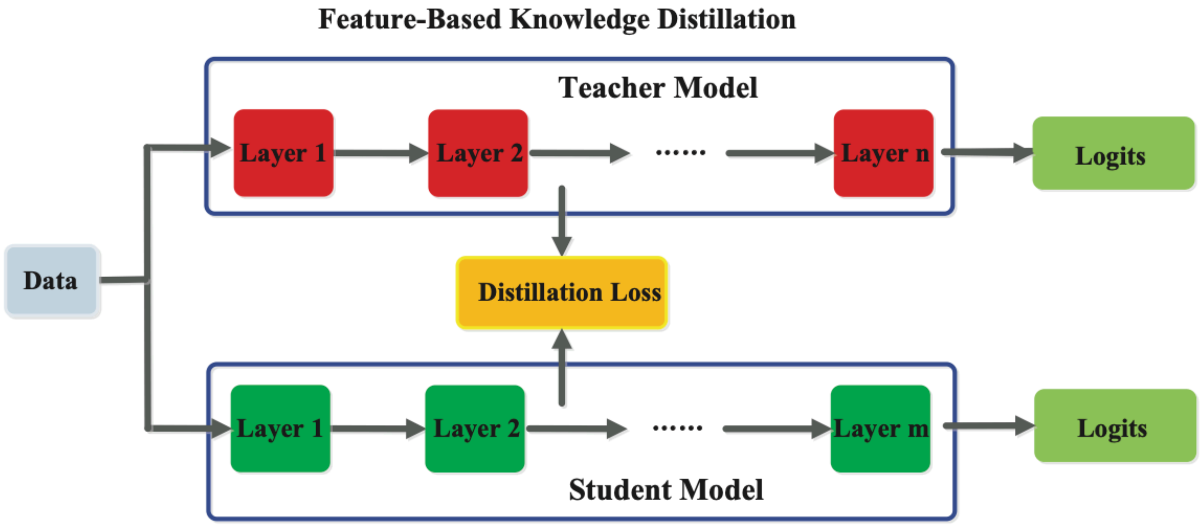
- Teacher **logits.**
  - **Multi-class** and **intra-class** information



Train the student to approximate the logits of the teacher

Feature Approximation

- Features **representation** encodes **inner** knowledge of the teacher

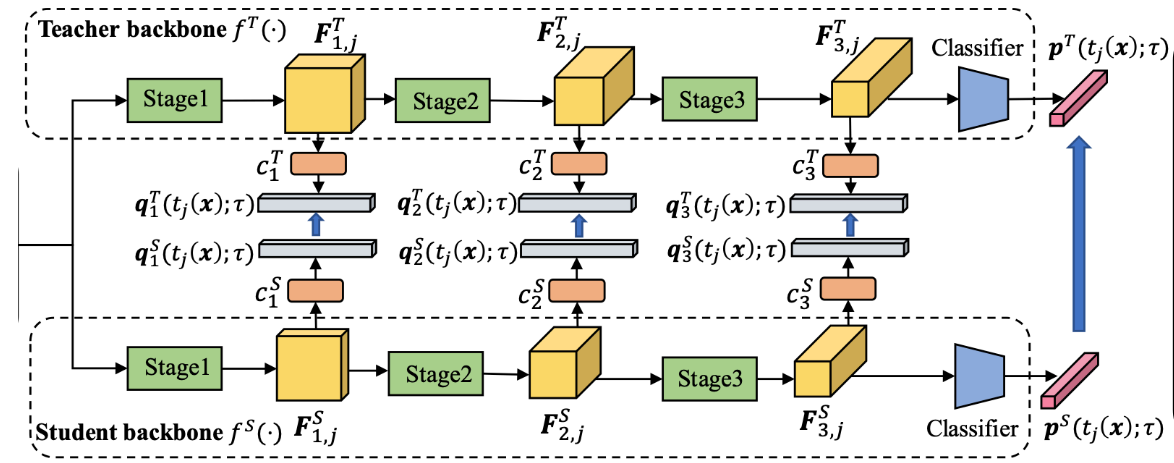


- Forcing the **student activations** to be similar to the teacher ones

- $L_{\text{tot}} = \underbrace{\boxed{H(x, y)}}_{\text{Classical Loss}} + \underbrace{\boxed{L(\phi_t(x), \phi_s(x))}}_{\text{Hint Loss}}$

Knowledge Distillation Performance

- **Multi-level** distillation



- Performance on ImageNet
  - + 2.6 % Top1 w.r.t to standard training
  - No **inference overhead**

Model	Top1
Student	69.8
Teacher	73.3
Student + KD	72.4

Yang, Chuanguang, et al. "Hierarchical Self-supervised Augmented Knowledge Distillation", International Joint Conference on Artificial Intelligence 2021.

Research Question

- Knowledge Distillation is effective but..

- **RQ1**. Poor theoretical basis
- **RQ2**. Knowledge distillation vs label smoothing?
- **RQ3**. Combinations with other compression methods?

- And many more..

Research Question

Thanks for the attention!

cosimo.rulli@phd.unipi.it