

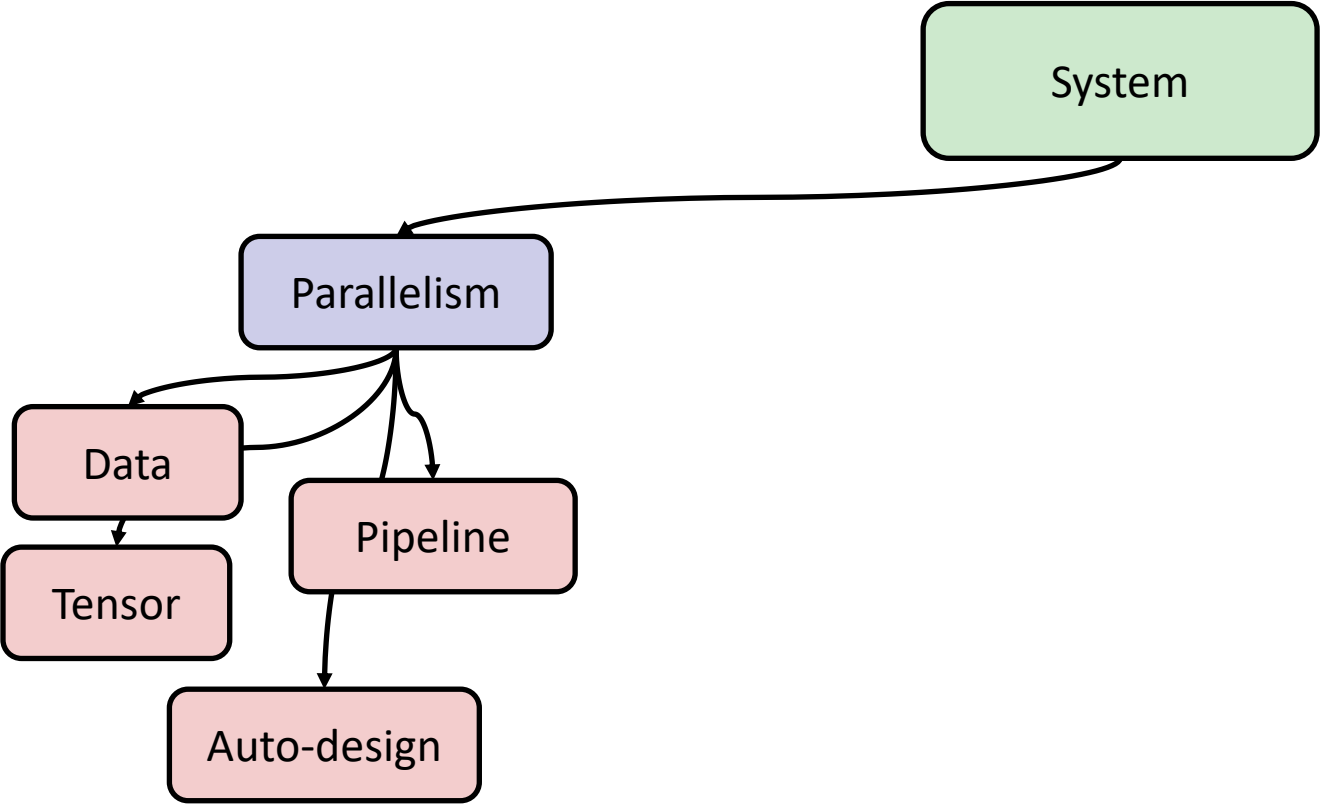
CS 598

AI Efficiency: Systems and Algorithms

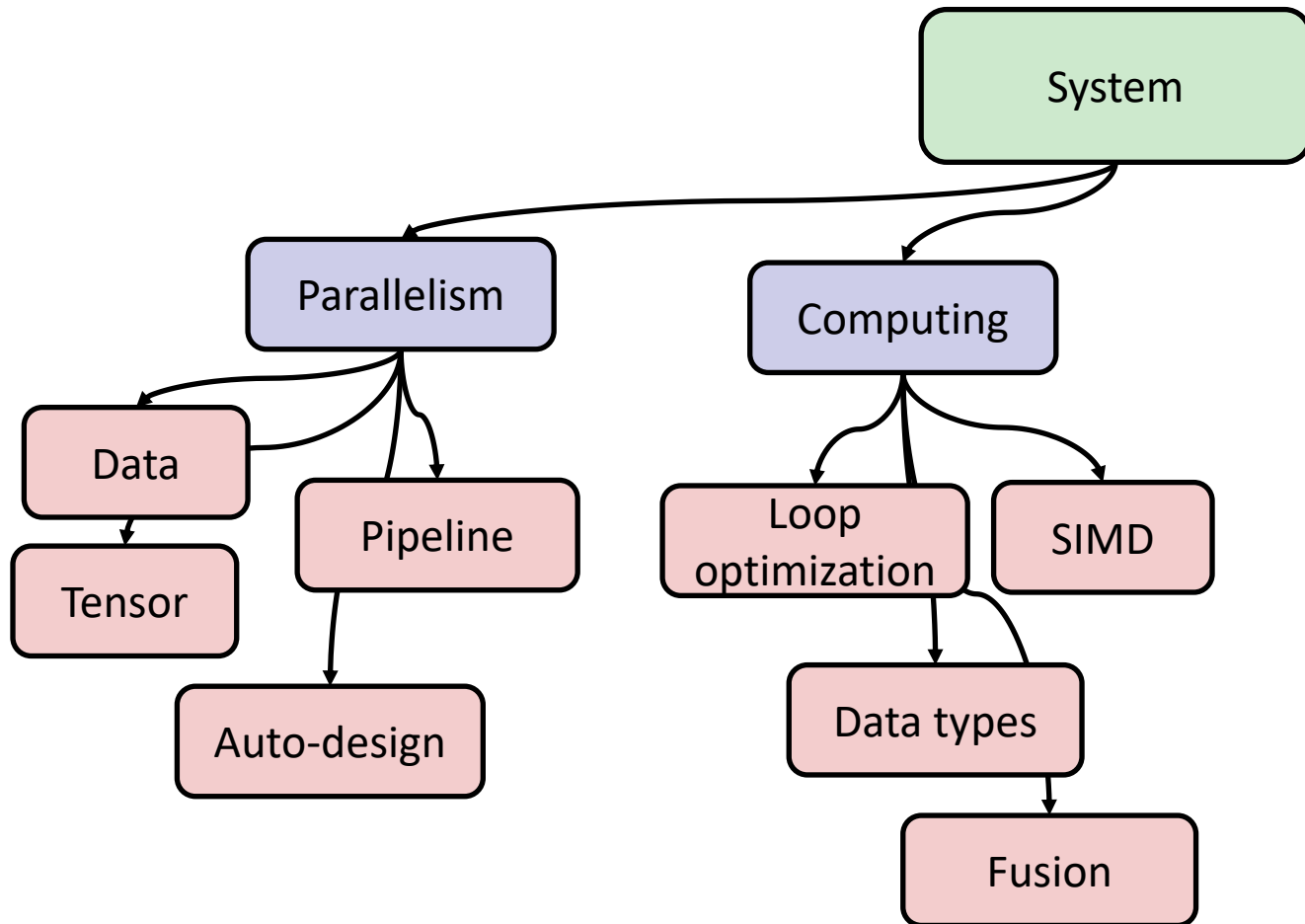
Minjia Zhang

Computer Science Department
University of Illinois at Urbana-Champaign

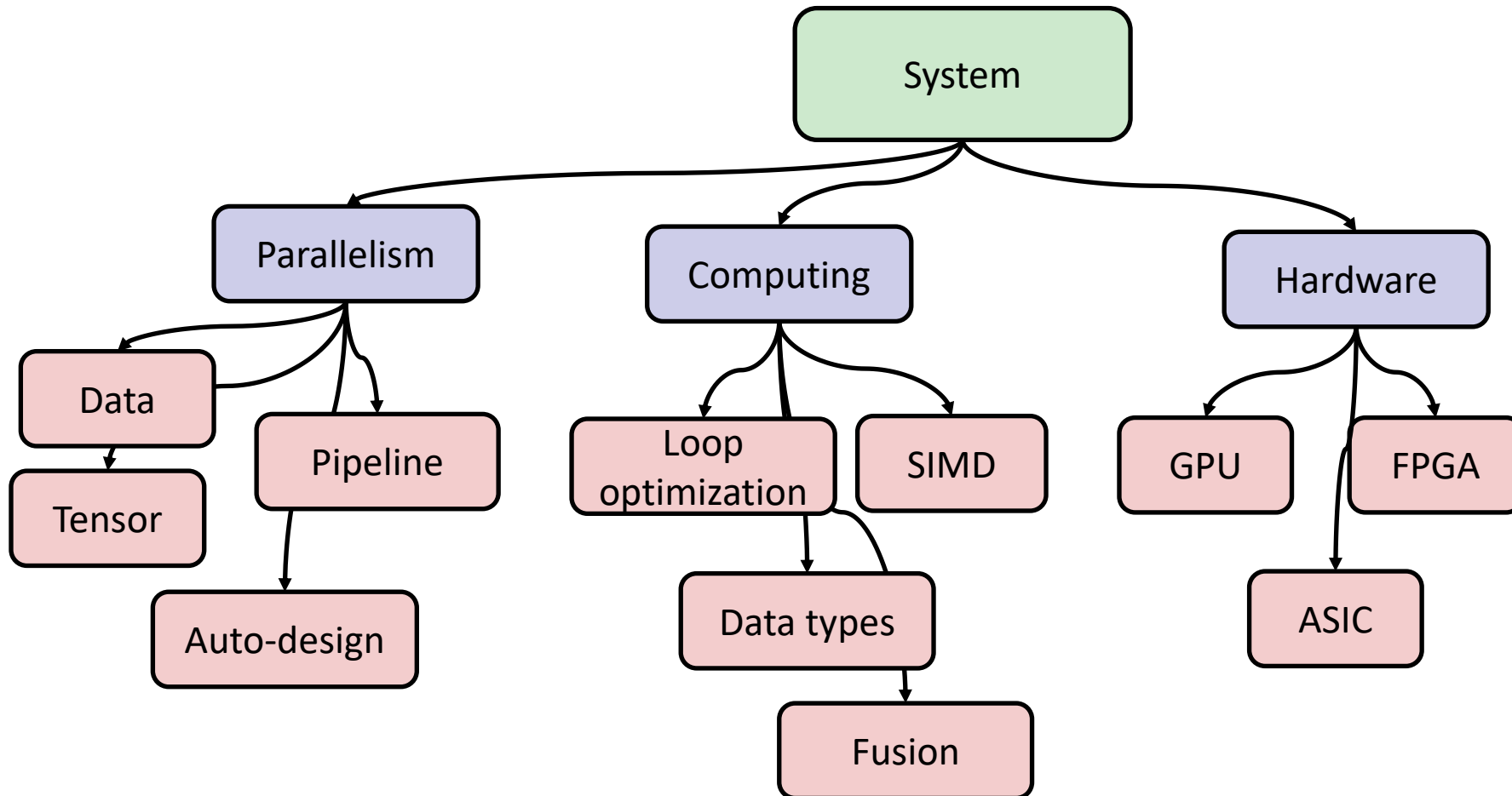
Systems for AI Efficiency



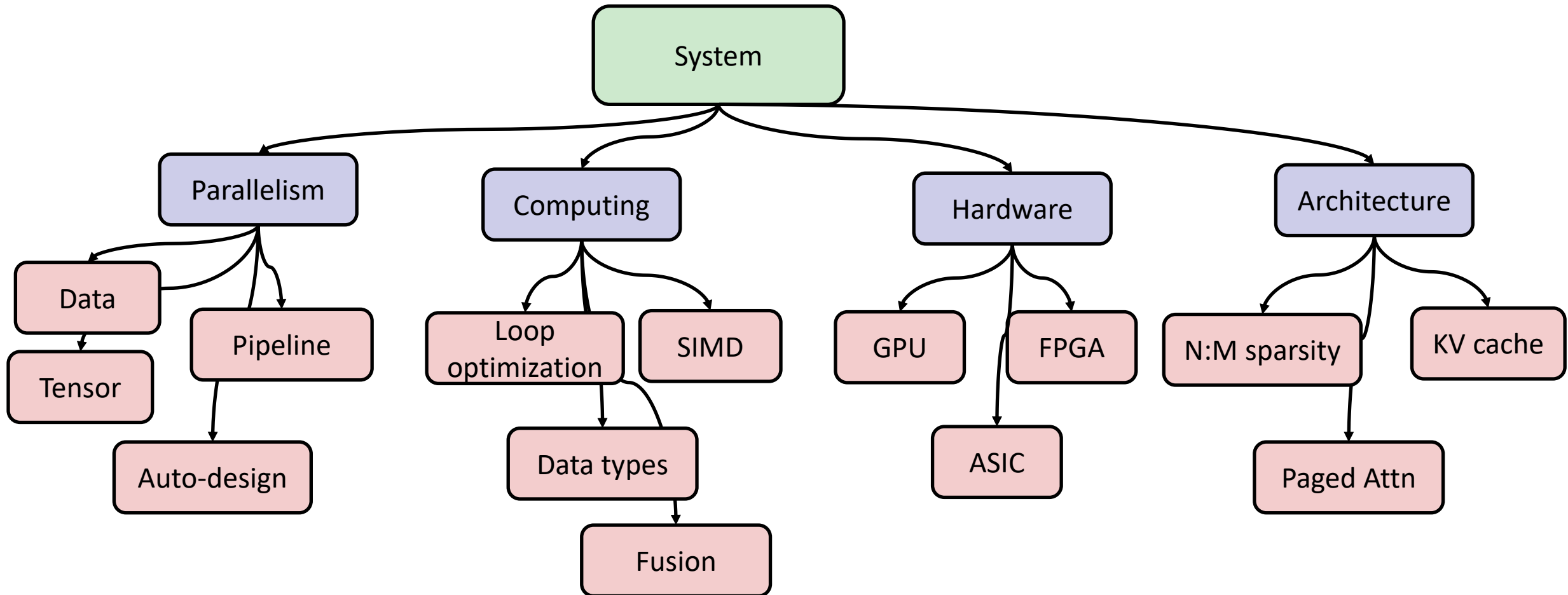
Systems for AI Efficiency



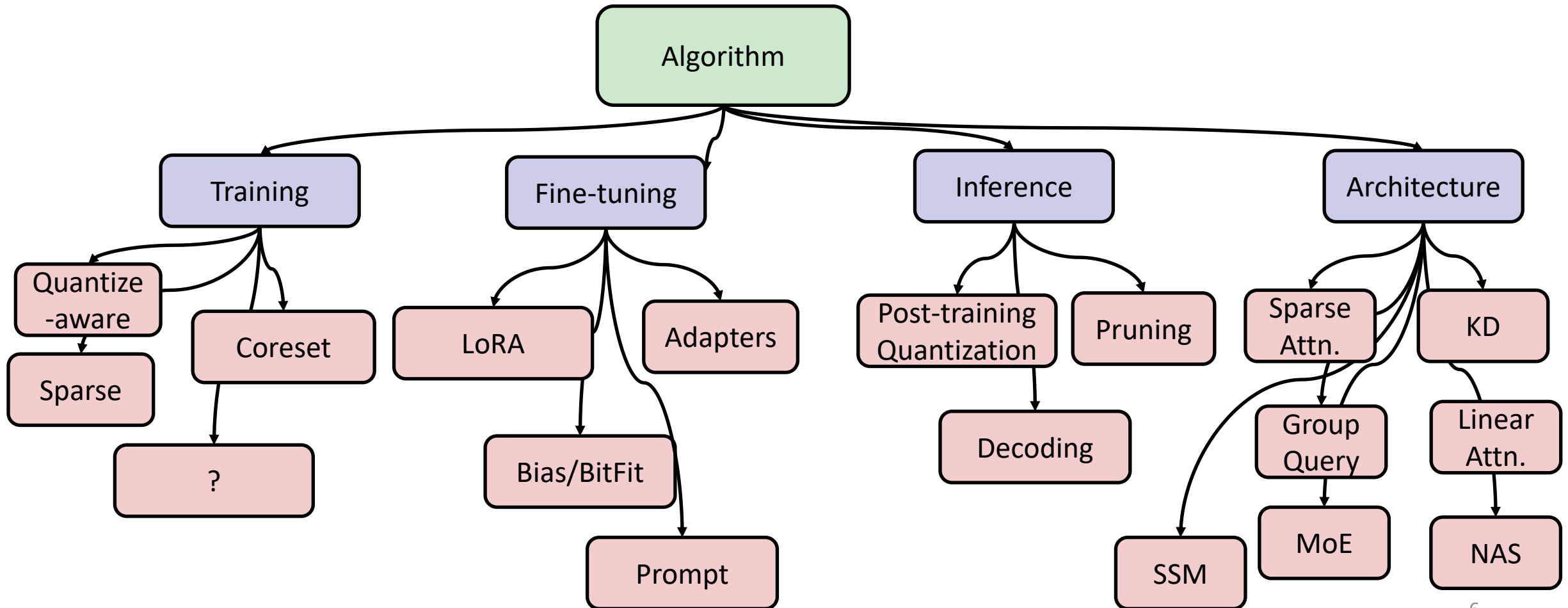
Systems for AI Efficiency



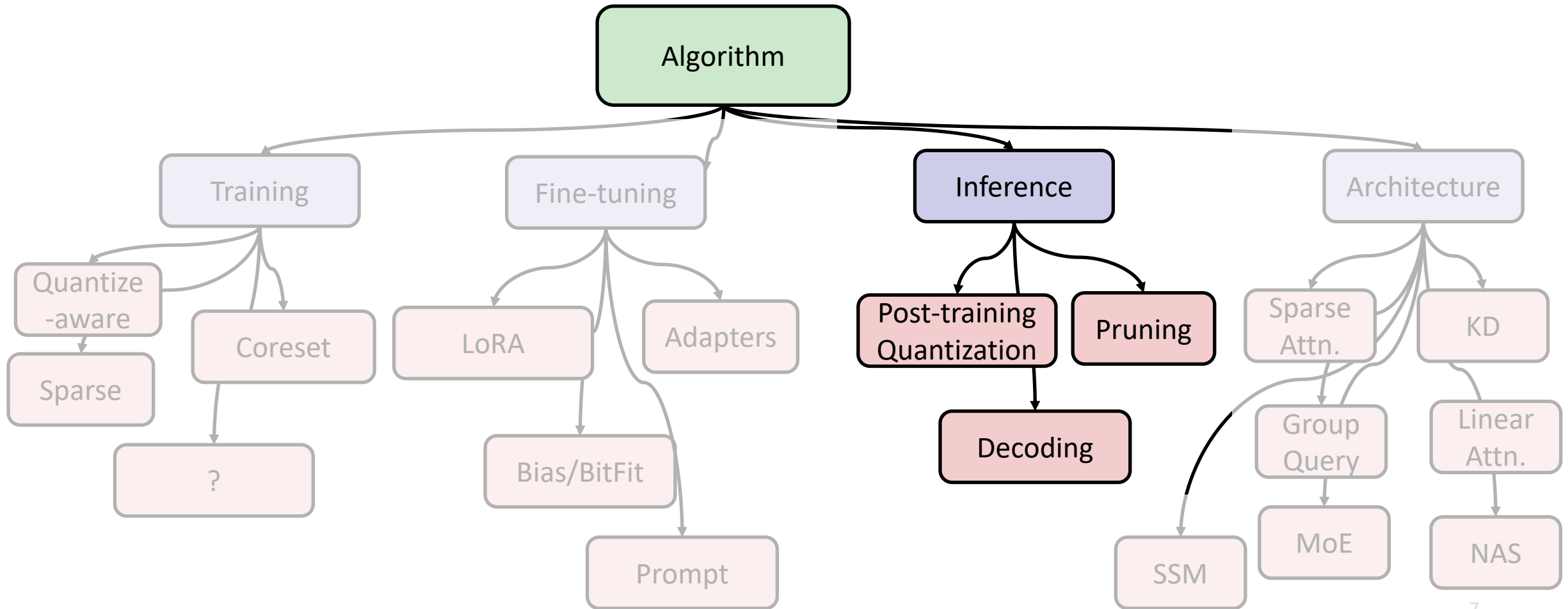
Systems for AI Efficiency



Algorithms for AI Efficiency



Algorithms for AI Efficiency



Algorithms for Efficient AI Inference

Inference Scenario 1: Online Q&A



Web Images Videos Maps News

3,330,000 RESULTS Any time ▾

William Shakespeare Quotes - BrainyQuote

Brainyquote > quotes > authors > ... > william_shakespeare ▾

Enjoy the best **William Shakespeare Quotes** at BrainyQuote. Quotations by **William Shakespeare**, English Dramatist, Born April 23, 1564. Share with your friends.

Shakespeare Quotes: Top 50 Famous Shakespeare Quotes

Nosweatshakespeare > quotes > famous-shakespeare-quotes ▾

★★★★★ 4.3/5 · 19 ratings · By Ed

Read **Shakespeare's** most **famous quotes**. ... and can now offer what we believe to be the 50 most **famous Shakespeare quotes**. So, without further ado, ...

Shakespeare Quotes - eNotes.com

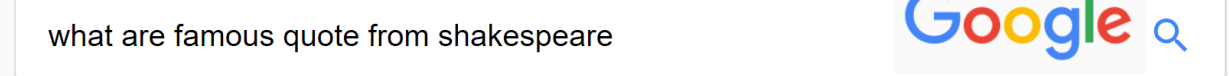
Enotes > shakespeare-quotes ▾

Shakespeare quotes with explanations at eNotes.com. eNotes Home; Homework Help; Study Guides; Study Tools ... and is one of **Shakespeare's** most **famous quotes**.

William Shakespeare Quotes at AbsoluteShakespeare.com

Absoluteshakespeare > trivia > quotes > quotes.htm ▾

William Shakespeare quotes such as "To be, ... Other **famous Shakespeare quotes** such as "I'll not budge an inch", "We have seen better ...



All Images News Shopping Videos More Settings Tools

About 7,370,000 results (0.88 seconds)

William Shakespeare / Quotes

Be not afraid of greatness: some are born great, some achieve greatness, and some have greatness thrust upon them.

To thine own self be true, and it must follow, as the night the day, thou canst not then be false to any man.

The course of true love never did run smooth.

Feedback

- Question and Answer Scenario
 - Direct answer not a list of webpages
 - Good quality of answer

Inference Scenario 2: Online ChatBot

What are some fun places to visit at Urbana Champaign?

✓ Searching for: **fun places to visit at Urbana Champaign**

✓ Generating answers for you...

There are many fun places to visit at Urbana Champaign, depending on your interests and preferences. Here are some of the most popular ones:

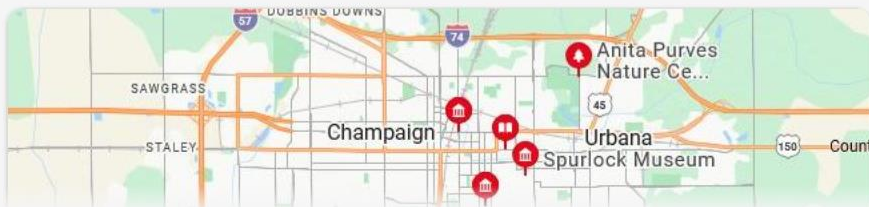
- If you love nature and animals, you might enjoy visiting the **Anita Purves Nature Center**¹, where you can explore trails, feed goats, and learn about local wildlife.
- If you are into art and culture, you might want to check out the **Krannert Art Museum**², which has a diverse collection of artworks from various regions and periods, as well as exhibitions and events.
- If you are looking for some entertainment and history, you might like the **Virginia Theatre**³, which is a restored historic venue that hosts concerts, movies, and shows.
- If you are feeling adventurous and sporty, you might have fun at the **University of Illinois Ice Arena**, where you can skate, play hockey, or watch games.

These are just some of the fun places to visit at Urbana Champaign. You can find more information and reviews on [Tripadvisor](#) or [Bing](#). I hope you have a great time exploring the city! 😊

Learn more ▾ 1 [experiencecu.org](#) 2 [kam.illinois.edu](#) 3 [bing.com](#)



2 of 30 ●



- ChatBot Scenario
 - Human-like responses with fewer hallucination

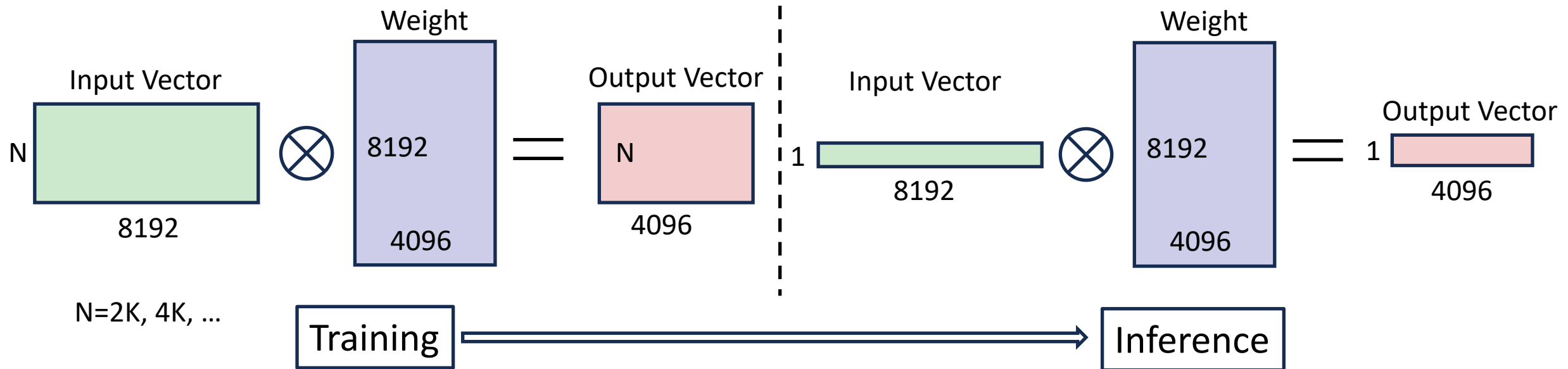
Inference Challenge 1: Long Latency

- Long serving latency blocks deployment
- Support advance models while meeting latency SLA and saving cost

DL Scenarios	Original Latency	Latency Target
Turing Prototype 2	~100ms	< 10ms
Turing Prototype 3	~107ms	< 10ms
Deep Query Document Similarity	10~12ms for [query, 1 doc] x 33 docs	< 6ms
Malta Click Features	10ms for [query, 1 passage] x 150 passages	< 5ms
Ads seq2seq model for query rewriting	~51ms	< 5ms

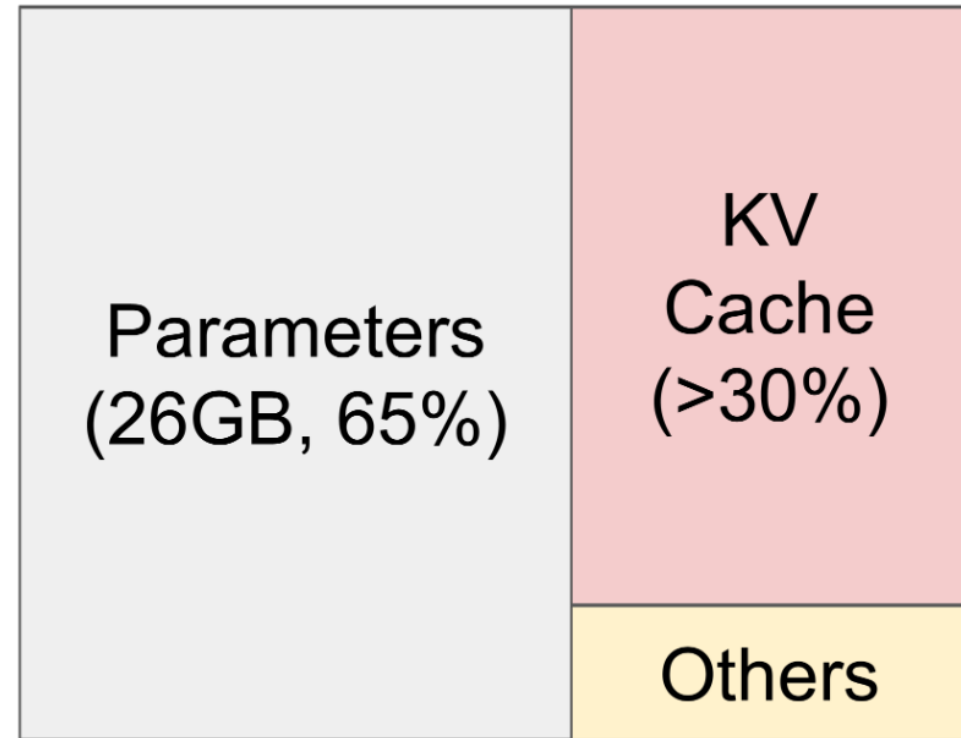
Inference Challenge 2: Limited Parallelism

- Small batch size \implies Low data reuse
- Autoregressive generation \implies Sequential dependency



Inference Challenge 3: Large Memory Footprint

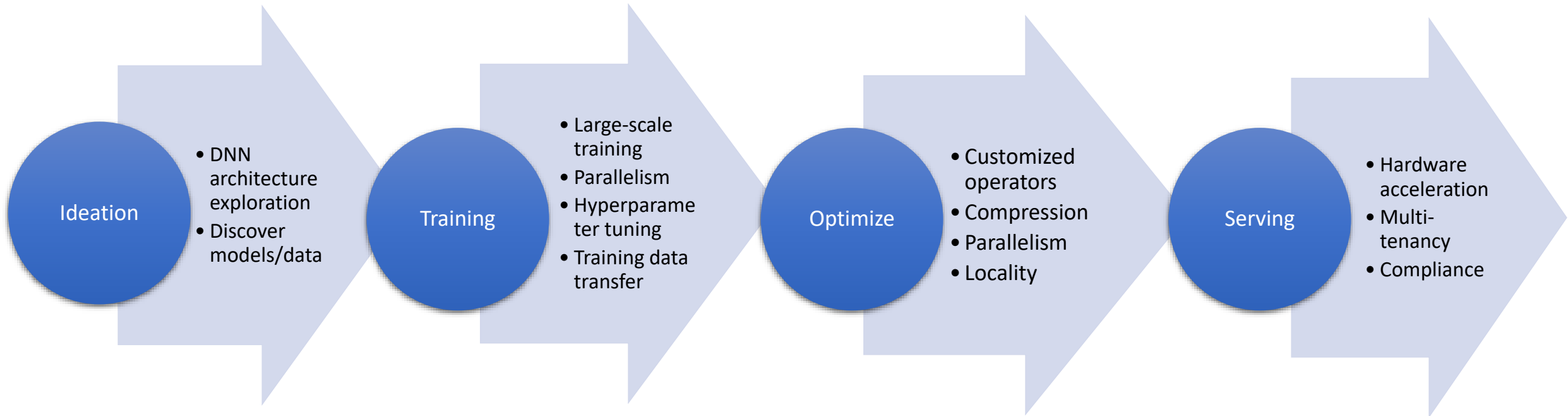
- Model parameters
 - # Layers
 - # Hidden dim
- KV cache
 - Batch size
 - Sequence length
 - # Layers
 - # Hidden
- Activation and others



OPT-13B on A100 40 GB

[Efficient Memory Management for Large Language Model Serving with PagedAttention](#), by Kwon et al., 2023

Inference Challenge 4: Putting DNN Models Into Production Takes Long Time



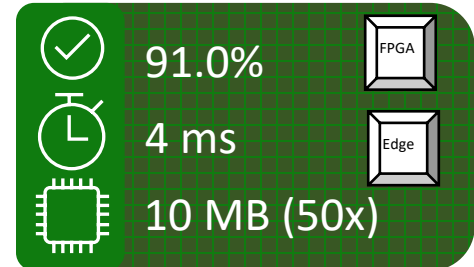
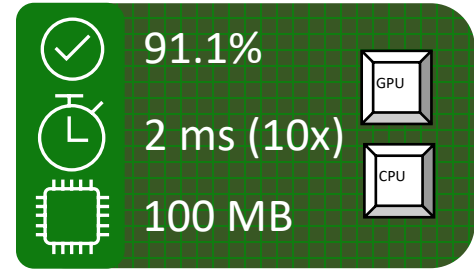
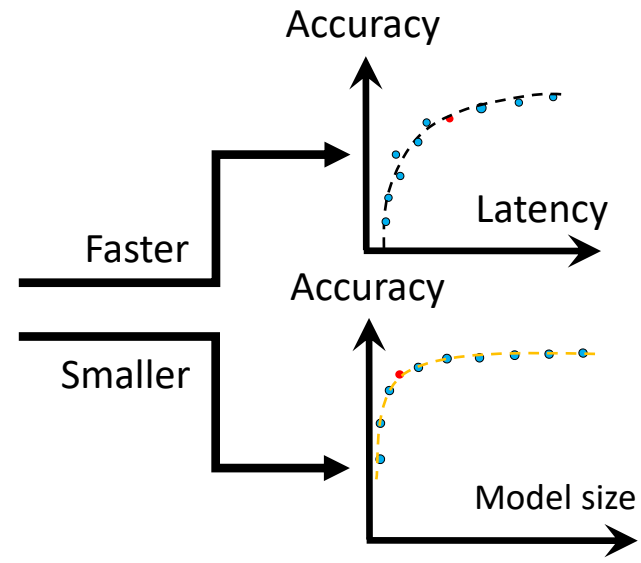
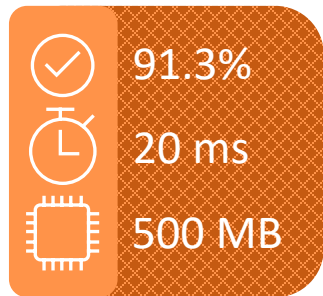
Inference Optimization Goals

- Reduce the inference latency to satisfy latency SLA
- Improve the inference throughput to save cost
- Reduce the memory footprint of the model by using fewer GPU devices and less GPU memory
- Improve agility from DNN prototype to deployment

Model Compression

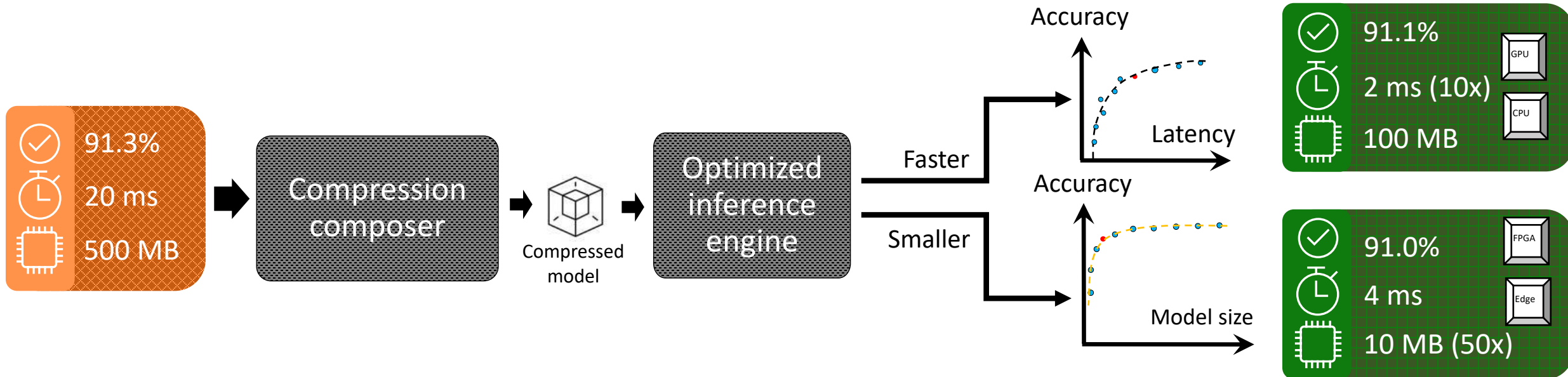
Compression Engine (DeepSpeed Compression)

Goal (**3Ss**): Make inference speed faster, model size smaller, while dramatically shortening the compression time



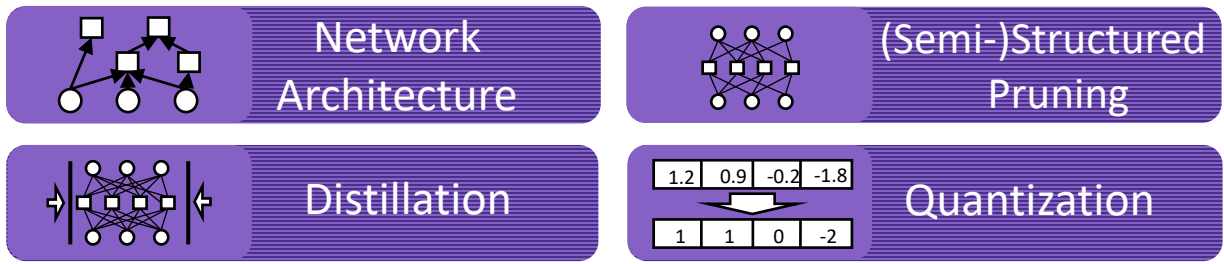
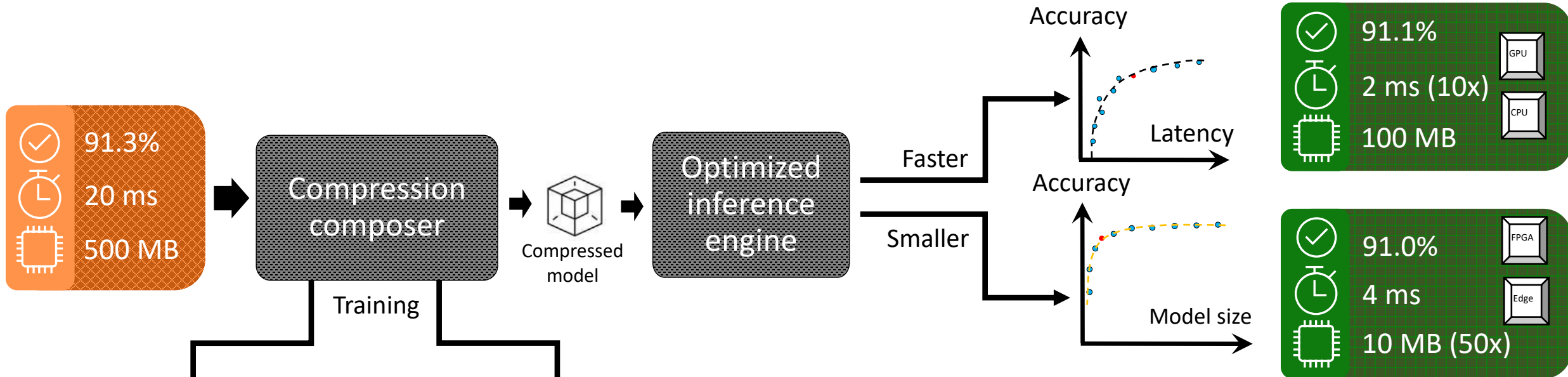
Compression Engine

Goal (**3Ss**): Make inference speed faster, model size smaller, while dramatically shortening the compression time



Compression Engine

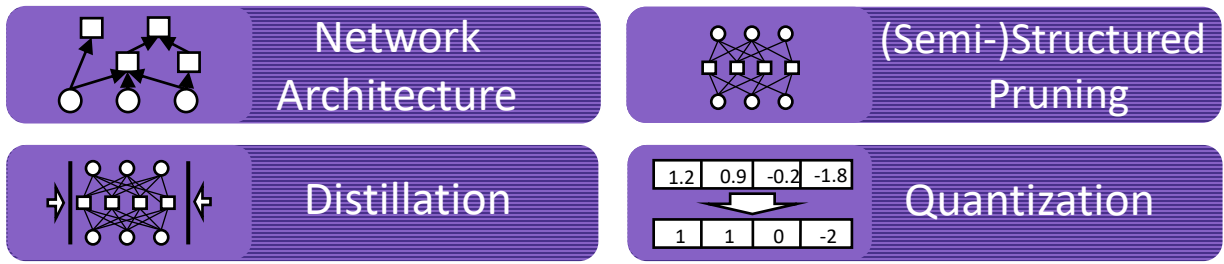
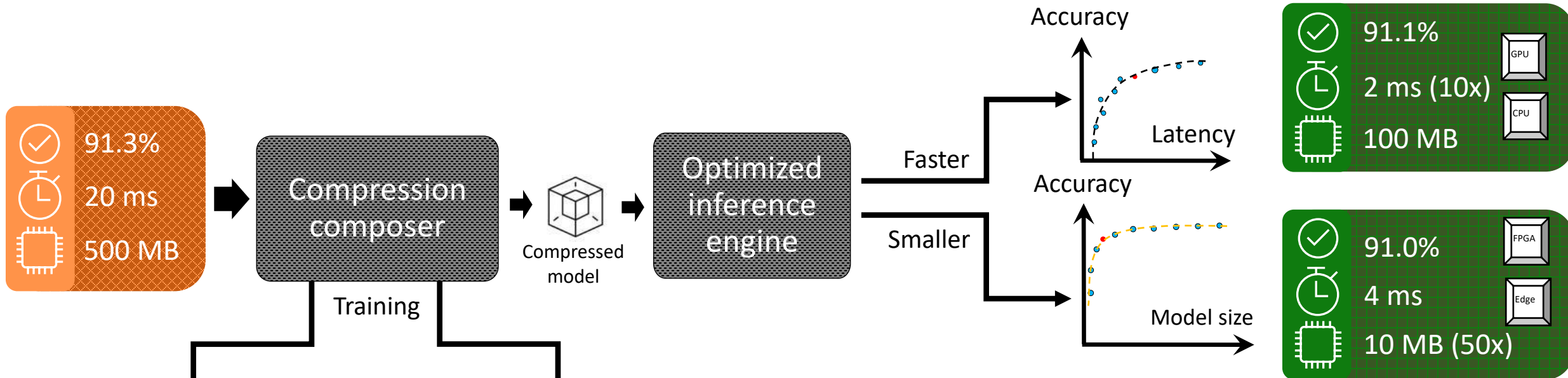
Goal (**3Ss**): Make inference **speed faster**, model size **smaller**, while dramatically **shortening the compression time**



- XTC: Extreme compression (NeurIPS'22 Oral)
 - Great model size reduction (>50x) via layer reduction and weight binarization techniques
- ZeroQuant: zero-cost quantization (NeurIPS'22 Spotlight)
 - Quantizing models with >5000x compression cost reduction and no training data
- AD²: Adversarial knowledge distillation (AAA'22)
 - 2x latency reduction via deep knowledge distillation and adversarial data augmentation
- NxMTransformer: Semi-structured sparsification (NeurIPS'21)
 - 2x latency reduction via N:M semi-structured sparsity and ADMM
- ...

Compression Engine

Goal (**3Ss**): Make inference **speed faster**, model size **smaller**, while dramatically **shortening the compression time**



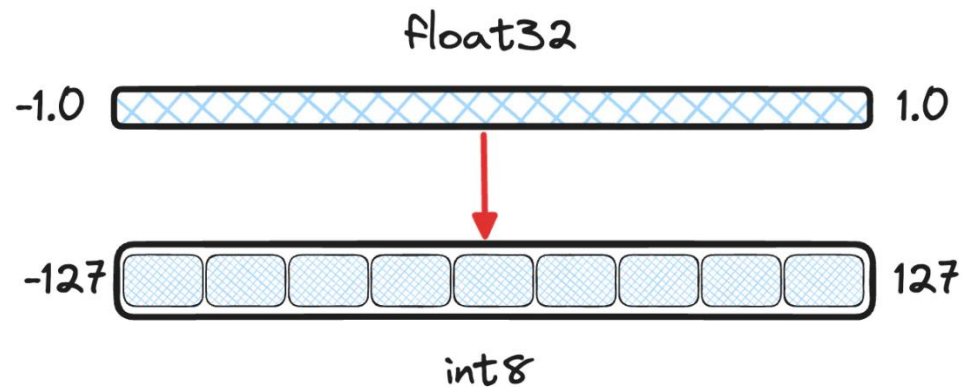
- **XTC: Extreme compression (NeurIPS'22 Oral)**
 - Great model size reduction (>50x) via layer reduction and weight binarization techniques
- **ZeroQuant: zero-cost quantization (NeurIPS'22 Spotlight)**
 - Quantizing models with >5000x compression cost reduction and no training data
- **AD²: Adversarial knowledge distillation (AAA'22)**
 - 2x latency reduction via deep knowledge distillation and adversarial data augmentation
- **NxMTransformer: Semi-structured sparsification (NeurIPS'21)**
 - 2x latency reduction via N:M semi-structured sparsity and ADMM
- ...

XTC: Extreme Compression

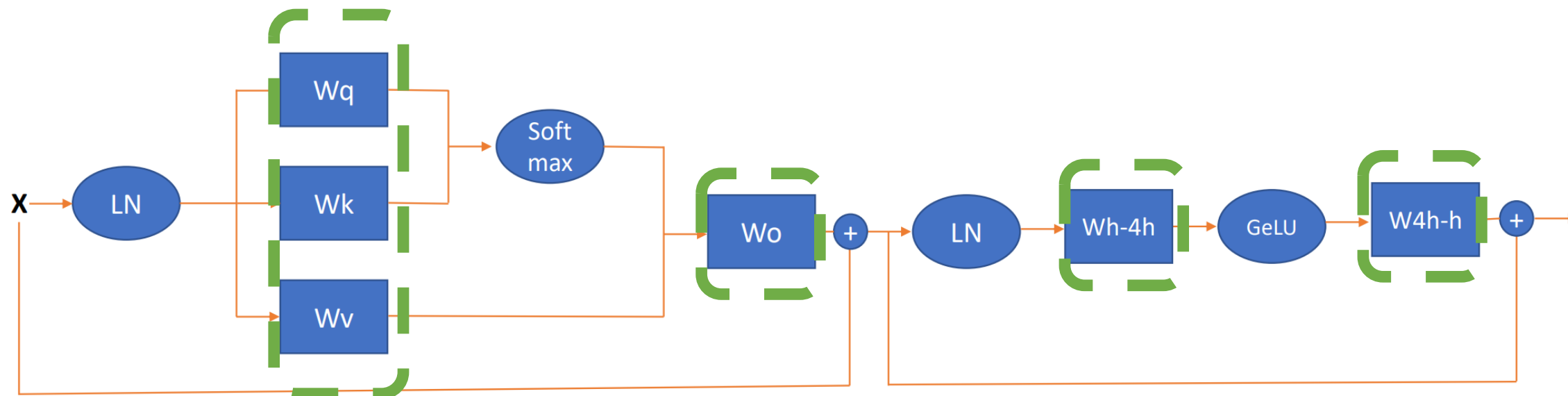
A technique that achieves great model size reduction (>50x) via layer reduction and weight binarization techniques

Quantization: Quick Recap

- Reduce the bits per weight, saving memory consumption
- Accelerate inference speed on supporting hardware



8-bit Weight Quantization



- 8-bit weight quantization

$$x_{quantize} = \text{round} \left(\text{clamp} \left(\frac{x}{S}, -2^{bit-1}, 2^{bit-1} - 1 \right) \right)$$

FP32 weight matrix

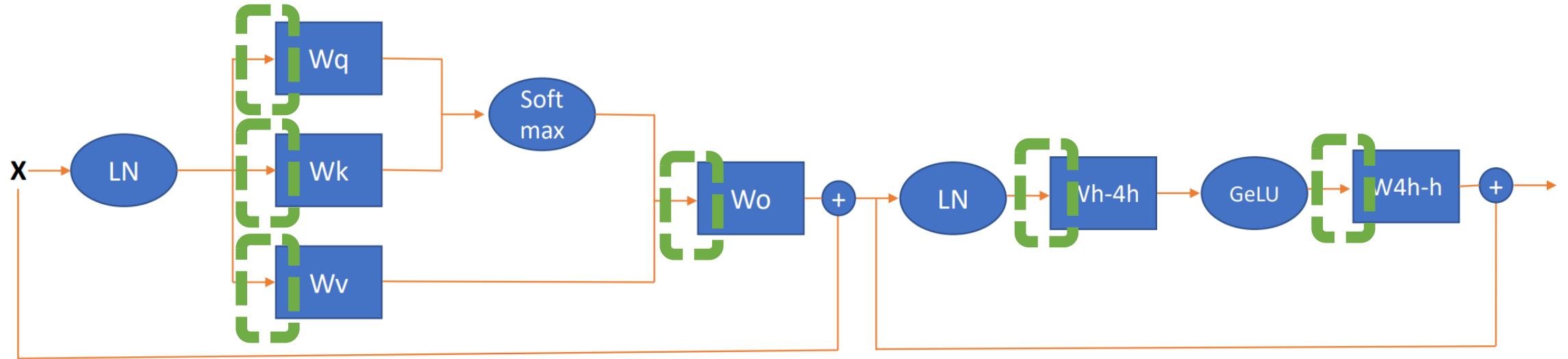
1.1	2.2	0.1	-0.1	-5.5	-6.6
...					
...					
...					
...					
1.1	2.1	0.1	-0.1	-4.8	-6.6

Scaling
Factor
 $1/S$
 $\approx 0.05^*$

8-bit quantization

21	42	2	-2	-106	-127
...					
...					
...					
...					
21	40	2	-2	-92	-127

8-bit Activation Quantization



- 8-bit activation
(Input to the linear layer)

$$x_{quantize} = \text{round} \left(\text{clamp} \left(\frac{x}{S}, -2^{bit-1}, 2^{bit-1} - 1 \right) \right)$$

FP32 input matrix

1.1	2.2	0.1	-0.1	-5.5	-6.6
...					
...					
...					
...					
1.1	2.1	0.1	-0.1	-4.8	-6.6

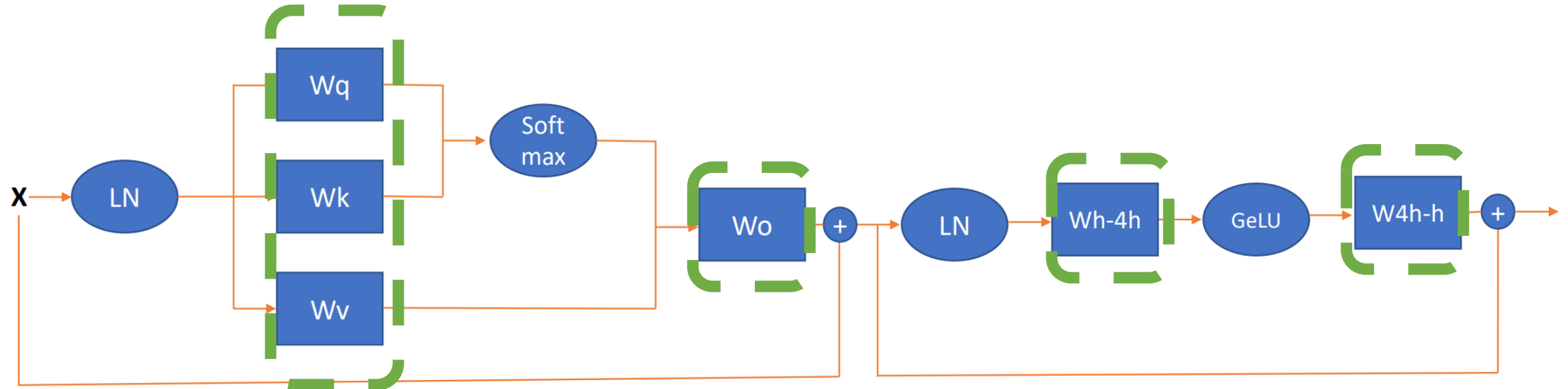
Scaling
Factor
 $1/S$

$\approx 0.05^*$

8-bit quantization

21	42	2	-2	-106	-127
...					
...					
...					
...					
21	40	2	-2	-92	-127

Weight Ternarization



- Ternarization (weight)**

W : weight matrix, FP32.

$Q(W)$: Quantization mapping, 2-bit.

With $\alpha = \|W\|_1/n$, for some scalar s

$$Q(W_{ij}) = \begin{cases} \alpha \cdot \text{sign}(W_{ij}) & \text{when } |W_{ij}| > s \\ 0 & \text{when } |W_{ij}| < s \end{cases}$$

FP32 weight matrix

1.1	2.2	0.1	-0.1	-5.5	-6.6
...					
...					
...					
...					
1.1	2.1	0.1	-0.1	-4.8	-6.0

Scaling

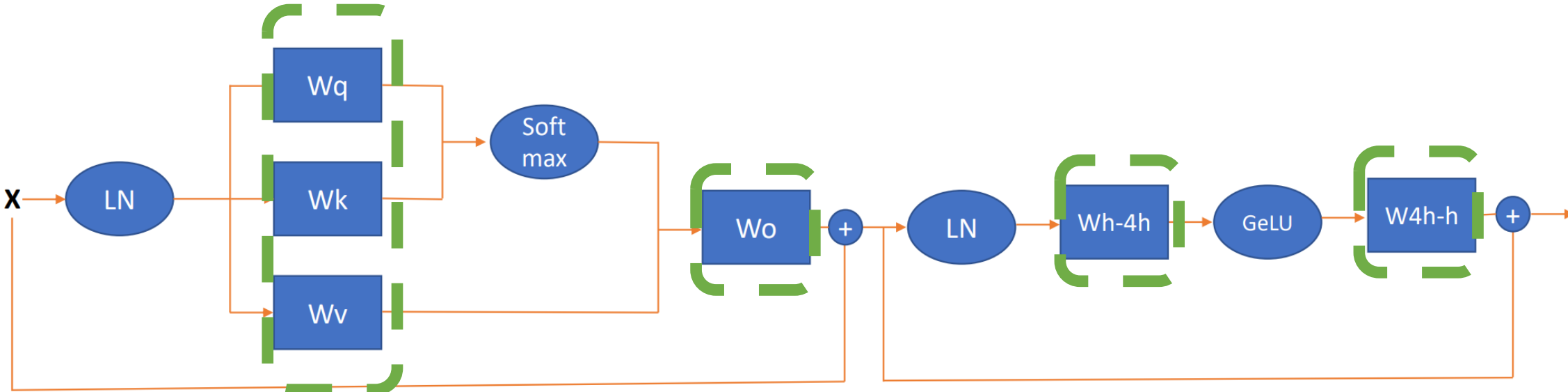
Factor α

$\approx 2.06 *$

2-bit quantization

1	1	0	0	-1	-1
...					
...					
...					
...					
1	1	0	0	-1	-1

Weight Binarization



- Binarization (weight)
 - W : weight matrix, FP32.
 - $Q(W)$: Quantization mapping, 1-bit.
 - With $\alpha = \|W\|_1/n$
 - $Q(W_{ij}) = \alpha \cdot \text{sign}(W_{ij})$

FP32 weight matrix

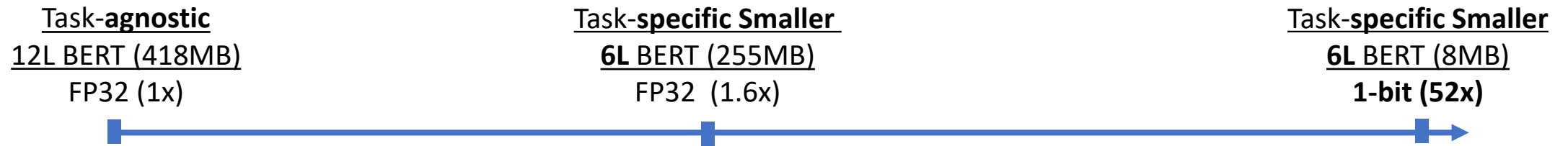
1.1	2.2	0.1	-0.1	-5.5	-6.6
...					
...					
...					
...					
1.1	2.1	0.1	-0.1	-4.8	-6.0

1-bit quantization

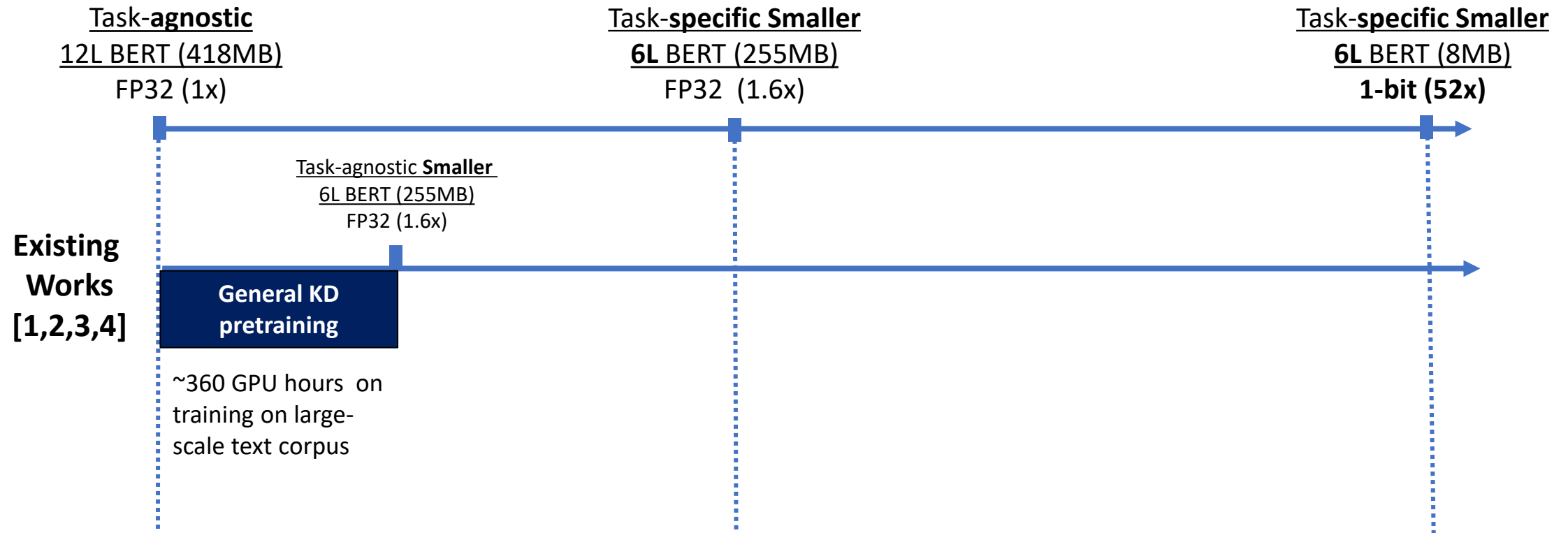
$\approx 1.4 * \text{Scaling Factor } \alpha$

1	1	1	-1	-1	-1
...					
...					
...					
...					
1	1	1	-1	-1	-1

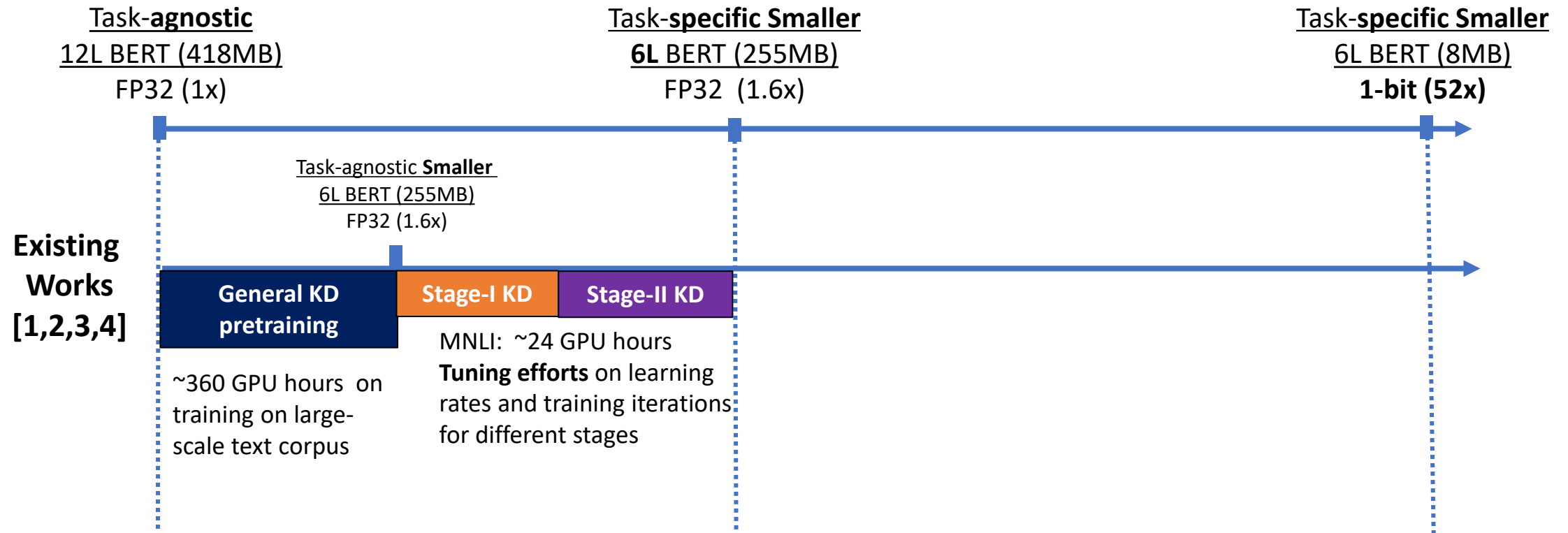
How Does Extreme Compression Work?



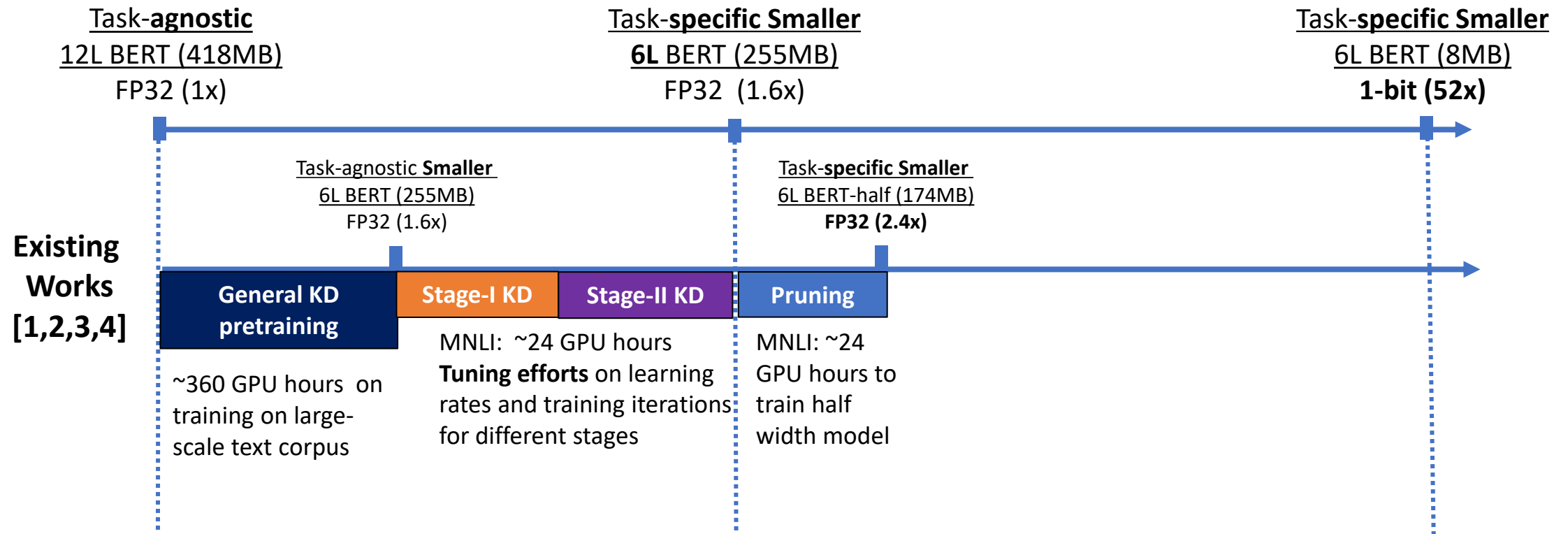
How Does Extreme Compression Work?



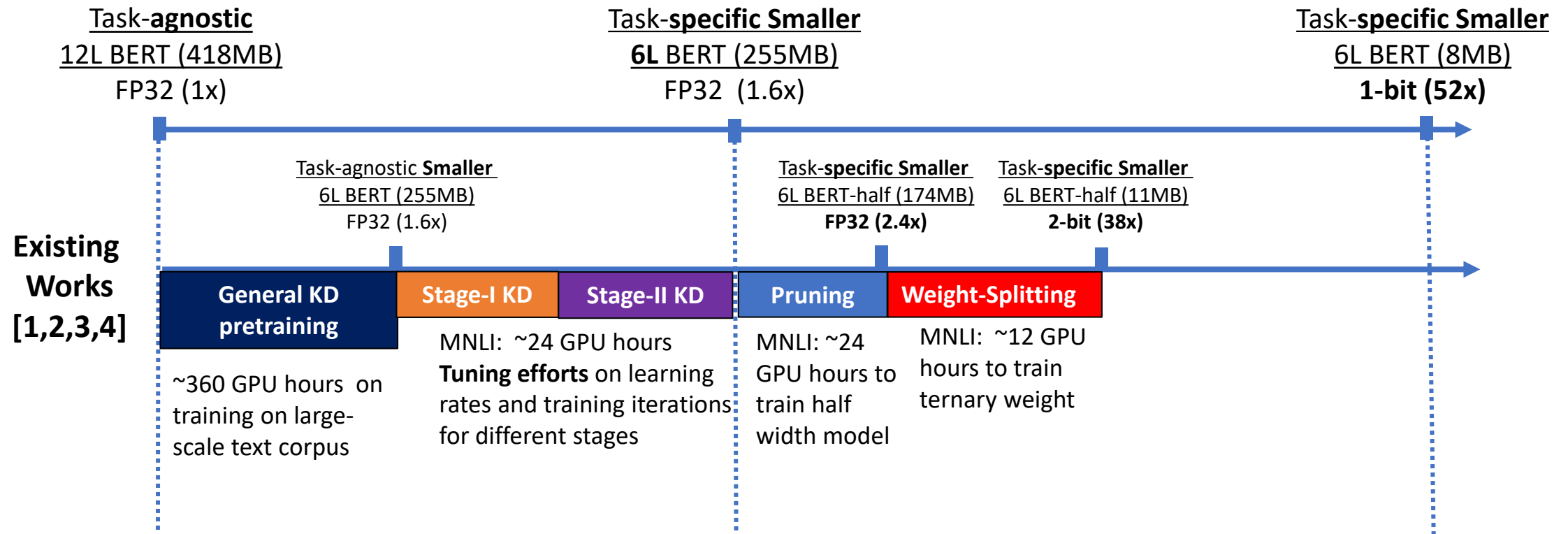
How Does Extreme Compression Work?



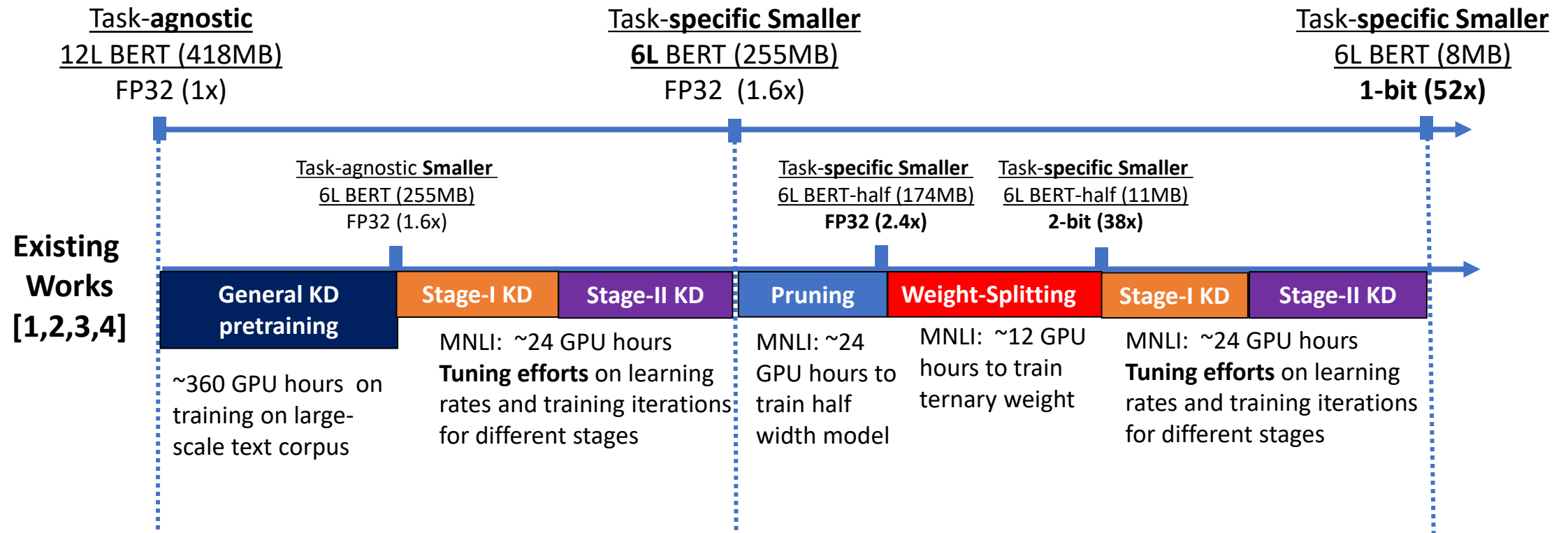
How Does Extreme Compression Work?



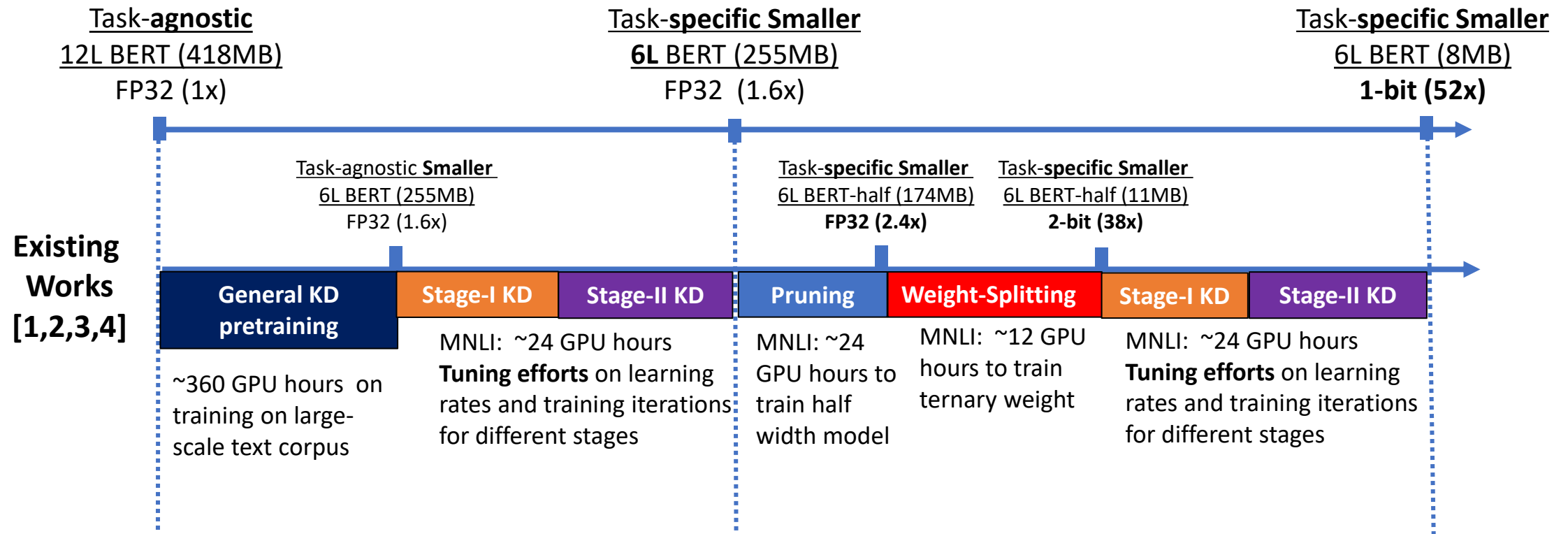
How Does Extreme Compression Work?



How Does Extreme Compression Work?



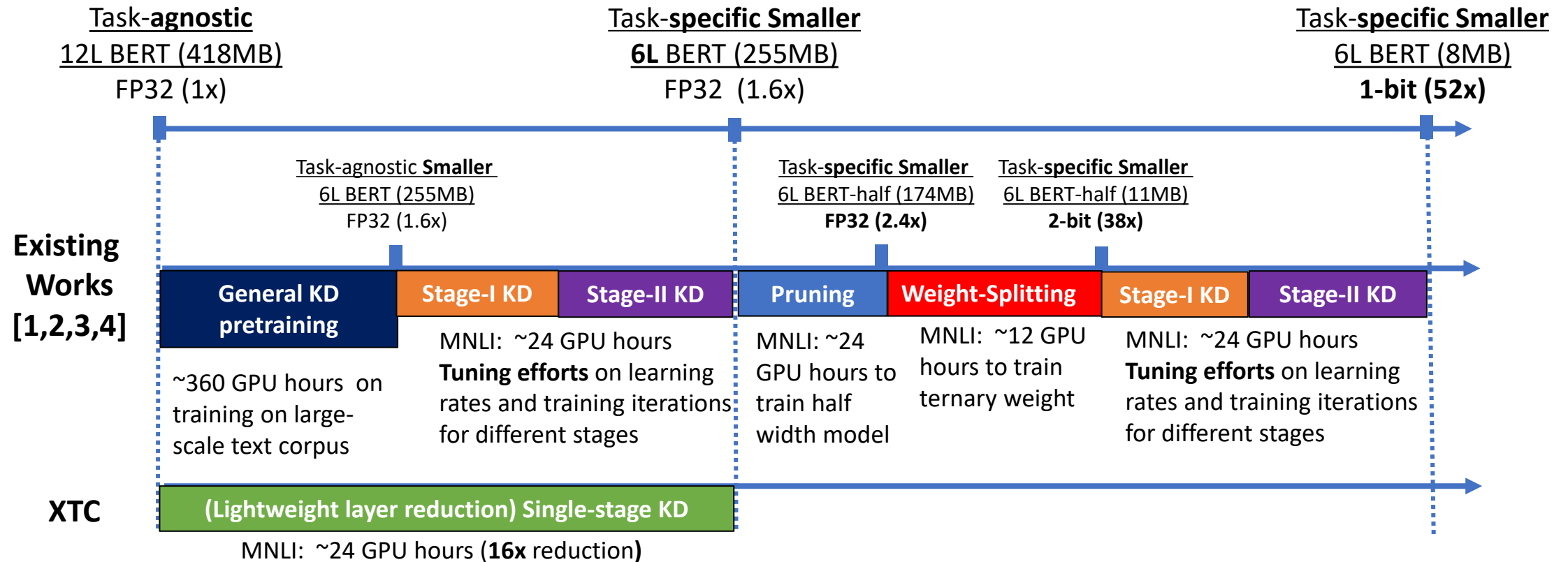
How Does Extreme Compression Work?



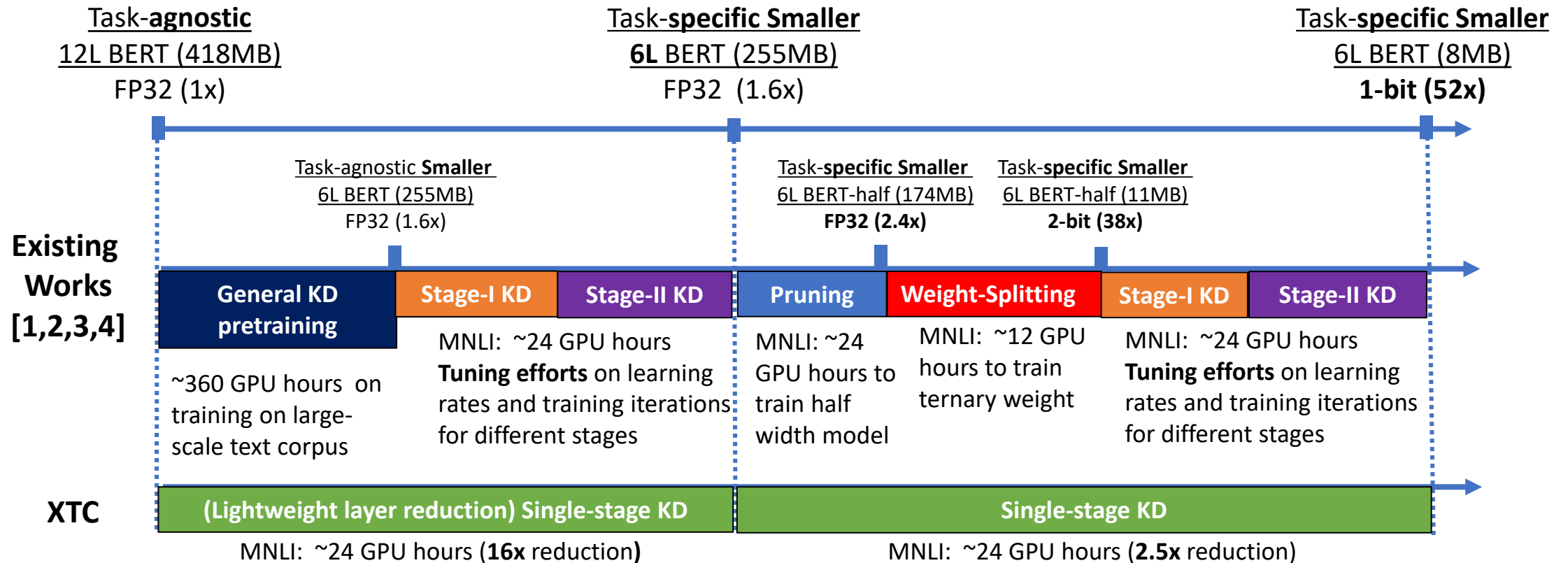
Limitations:

- High compression cost and long training time in practice
- Too many stages lead to complicated engineering and tuning effort

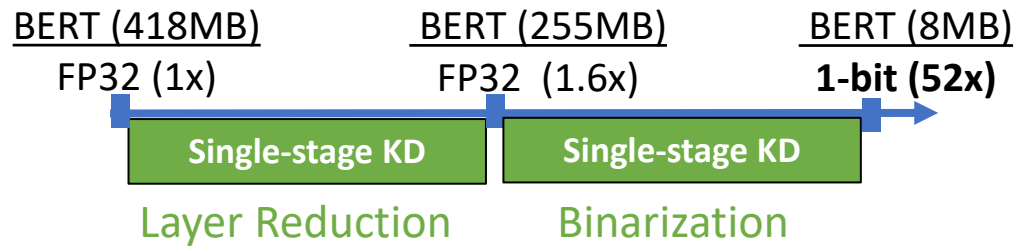
XTC: Effective Extreme Compression



XTC: Effective Extreme Compression



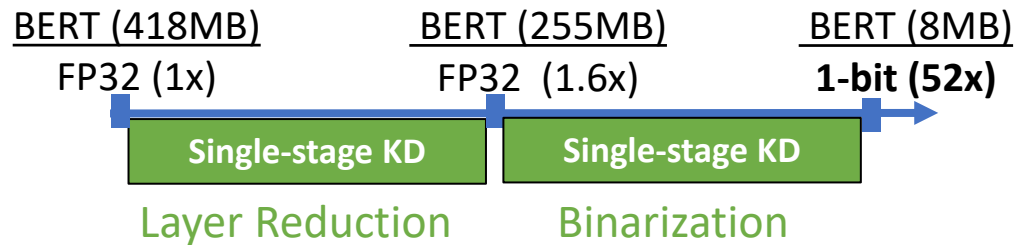
XTC: Effective Extreme Compression



😊 reduce the compression time, make compression easy to implement, and save enormous amount of engineering tuning efforts

- Great model size reduction (>50x)

XTC: Effective Extreme Compression

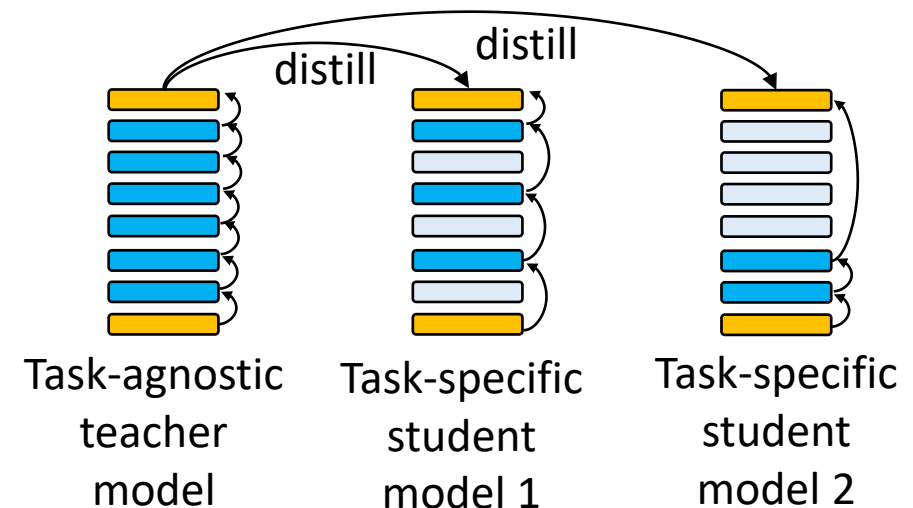
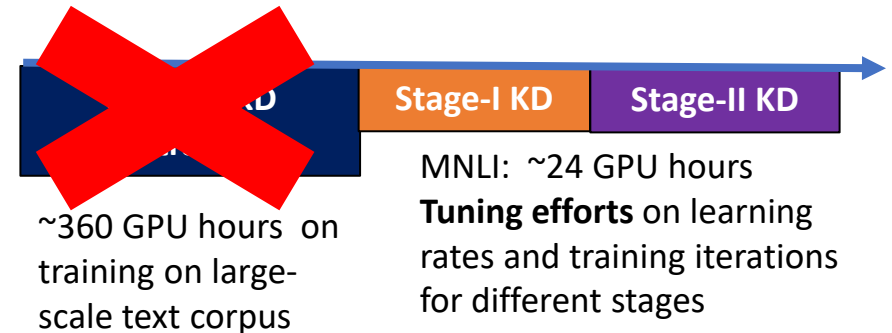


- Great model size reduction (>50x)

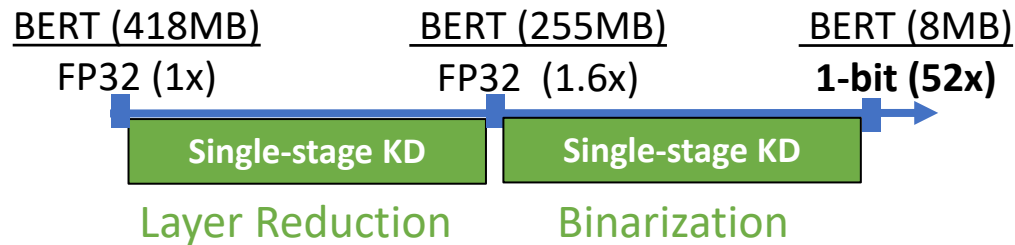
- Layer Reduction (~2x)

- No pre-training data needed

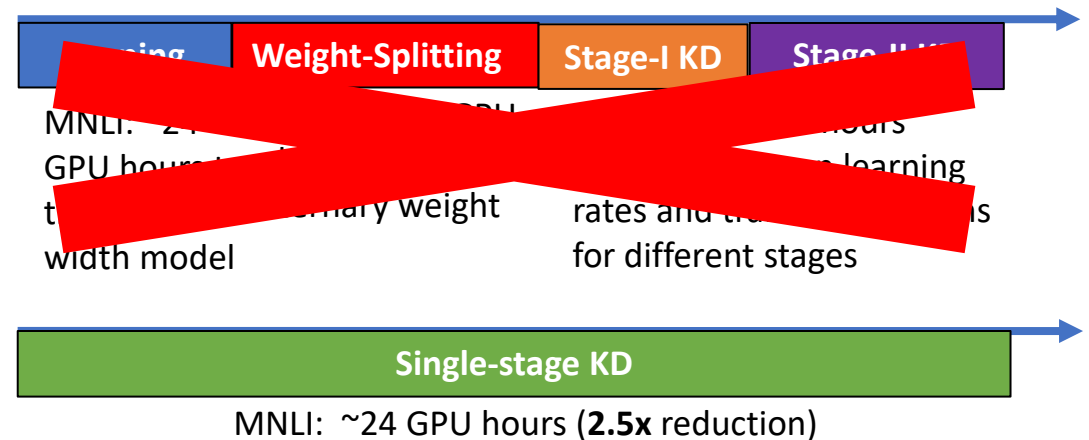
- Flexible to choose any layers of teacher model



XTC: Effective Extreme Compression



- Great model size reduction (>50x)
 - Layer Reduction (~2x)
 - No pre-training data needed
 - Flexible to choose any layers of teacher model
 - Weight Binarization (32x)
 - Train longer (+ learning rate decay)
 - Deep knowledge distillation
 - Mimic logits, feature map, attention map
 - Data augmentation
 - Word-level replacement



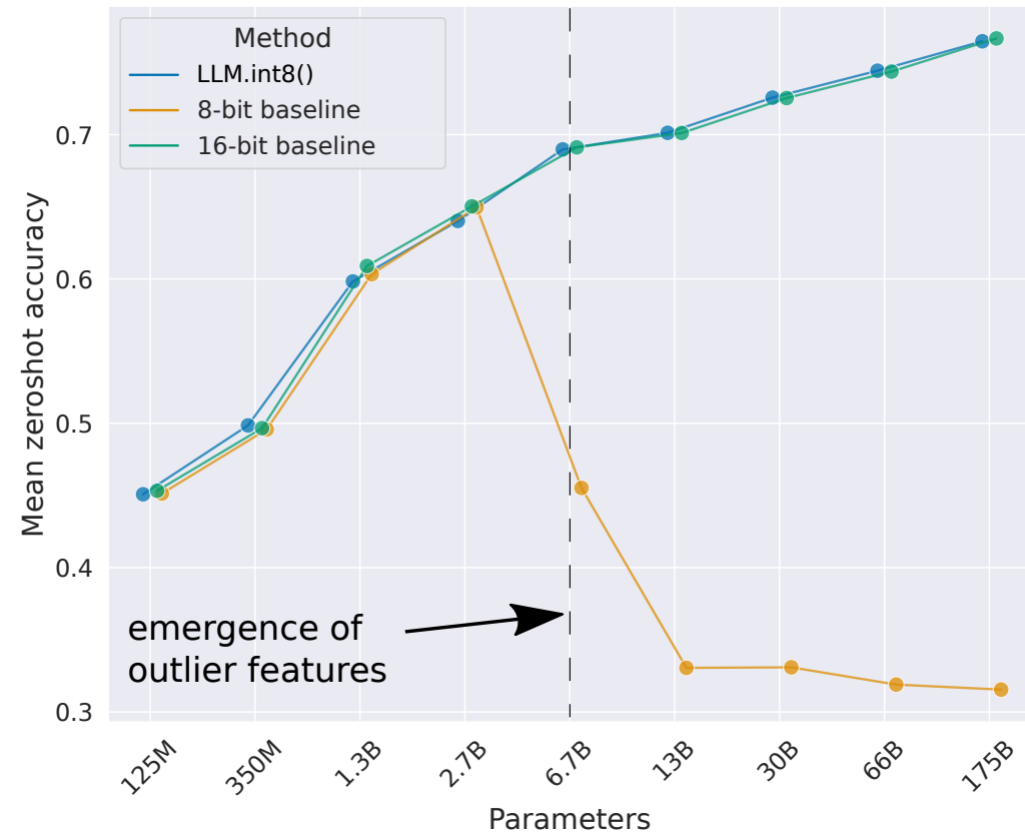
Lessons from Extreme Compression

- Simple schedule, e.g., longer training iterations with learning rate decay, is highly preferred for closing the accuracy gap of extreme quantization.
- Single-stage training, e.g., single-stage knowledge distillation, save enormous amount of engineering tuning efforts.
- Avoid pre-training compression as much as possible due to its expensive cost.

New Challenges to Compress Generative LLMs

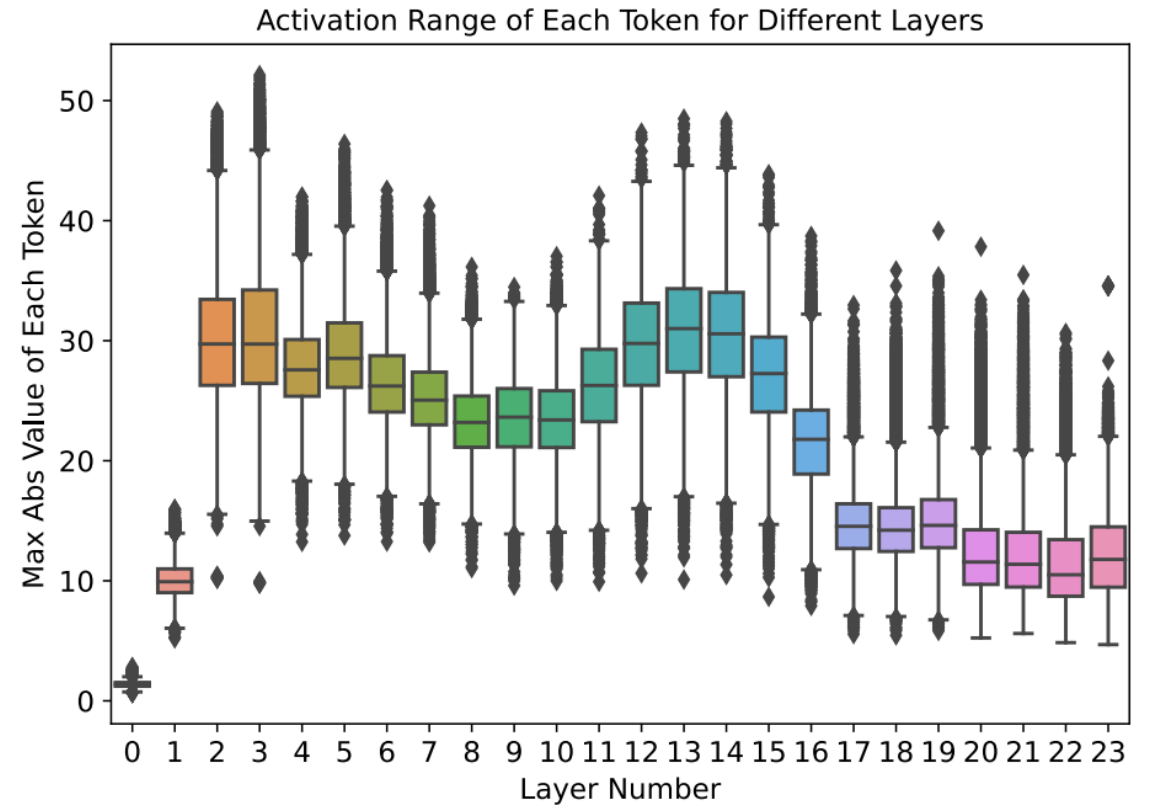
Challenges to Quantize LLMs

- Standard quantization strategy leads to catastrophic accuracy drop



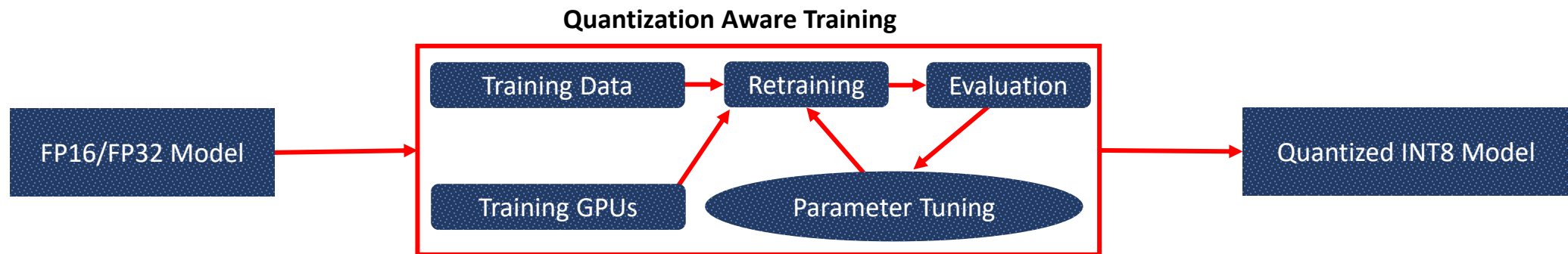
Challenges to Quantize LLMs

- High dynamic ranges of activation, leading to large quantization errors



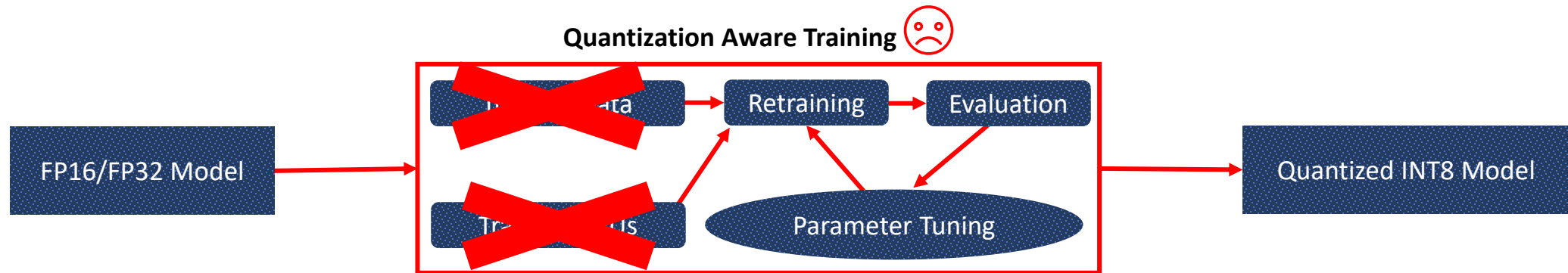
Challenges to Quantize LLMs

- Difficult to quantize large-scale models within limited time/resource budget



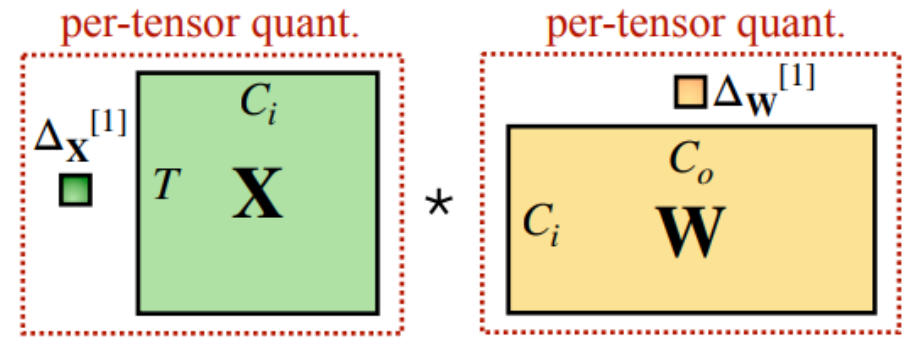
Challenges to Quantize LLMs

- Difficult to quantize large-scale models within limited time/resource budget

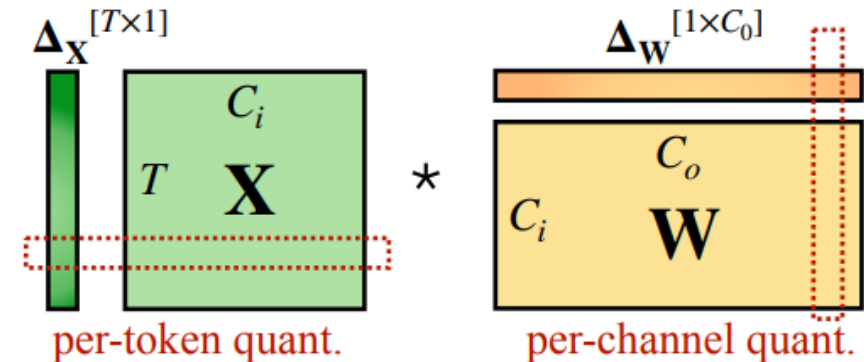


Fine-grained Quantization

- Per-tensor quantization
 - Low accuracy
 - Fast to quantize/dequantize
- Per-token/channel quantization
 - High accuracy
 - Slower to quantize/dequantize
 - Custom kernels required



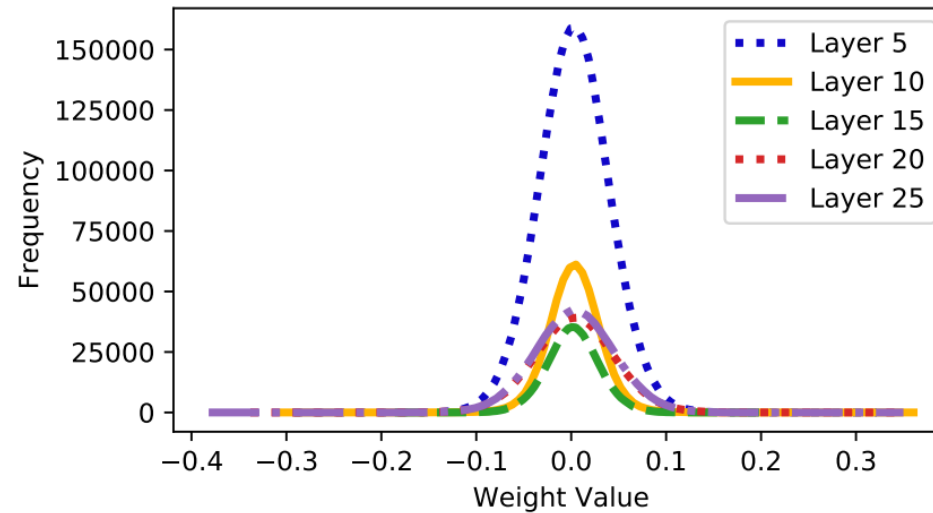
(a) per-tensor quantization



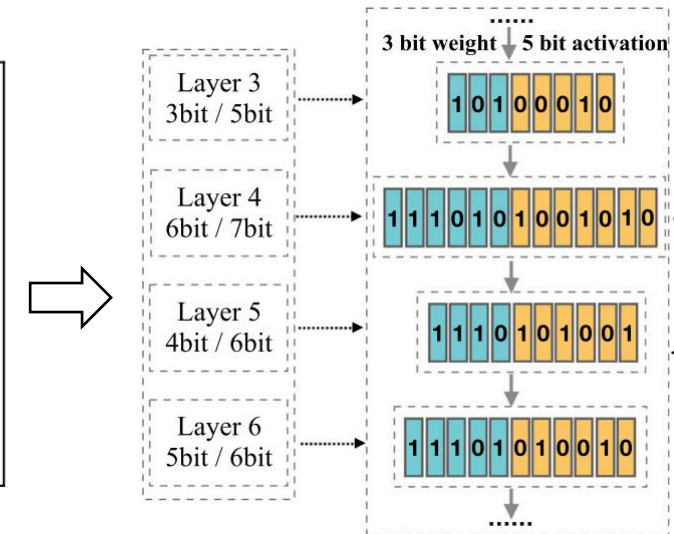
(b) per-token + per-channel quantization

Mixed Precision Quantization

- Weights follow Gaussian distribution
- Outliers remain in original form, quantize the rest of the values
- Different bits for different layers

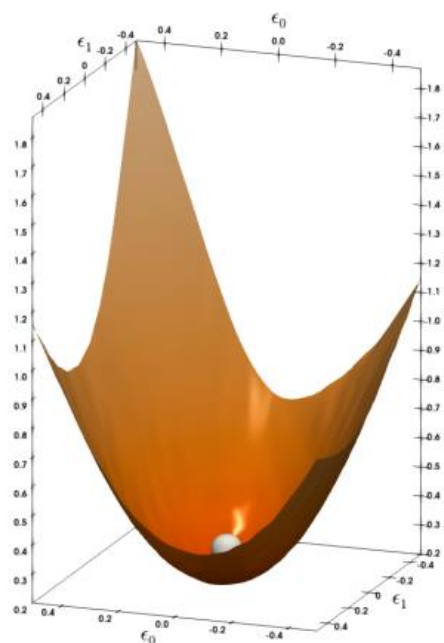


Per-layer weight distribution of BERT model

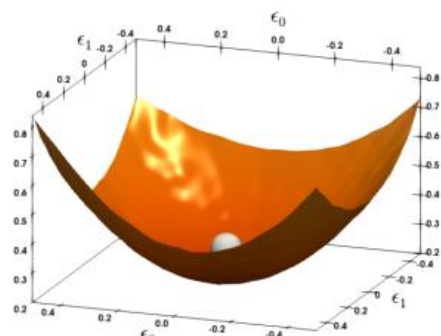


Mixed Precision + Second Order Information

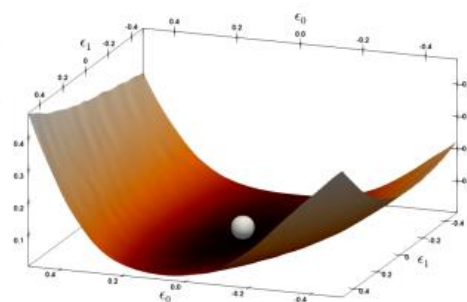
- Analyze the loss curvature (Hessian matrices) to help identify layer sensitivity



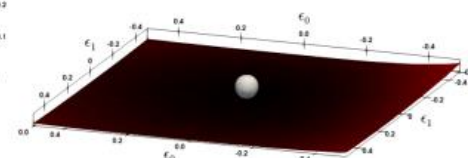
(a) MNLI 4th layer



(b) MNLI 10th layer



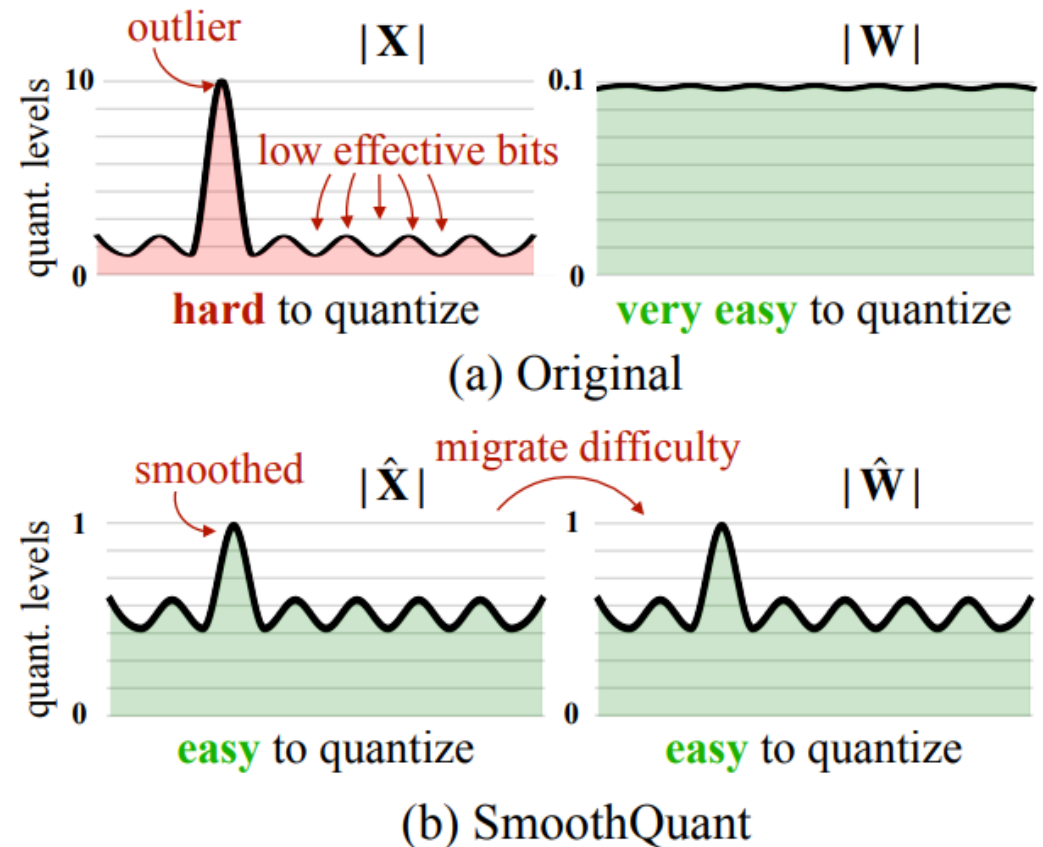
(c) CoNLL-03 4th layer



(d) CoNLL-03 11th layer

Outlier Smoothing

- Activation X is hard to quantize because outliers stretch the quantization range, leaving few effective bits for most values.
- One can migrate the scale variance from activations to weights W during offline to reduce the quantization difficulty of activations.
- The smoothed activation \hat{X} and the adjusted weight \hat{W} are both easy to quantize.



Algorithms for AI Efficiency

