

ExCP: Extreme LLM Checkpoint Compression via Weight-Momentum Joint Shrinking

ICML 2024, Oral

Wenshuo Li¹, Xinghao Chen¹, Han Shu^{1,2}, Yehui Tang¹, Yunhe Wang¹

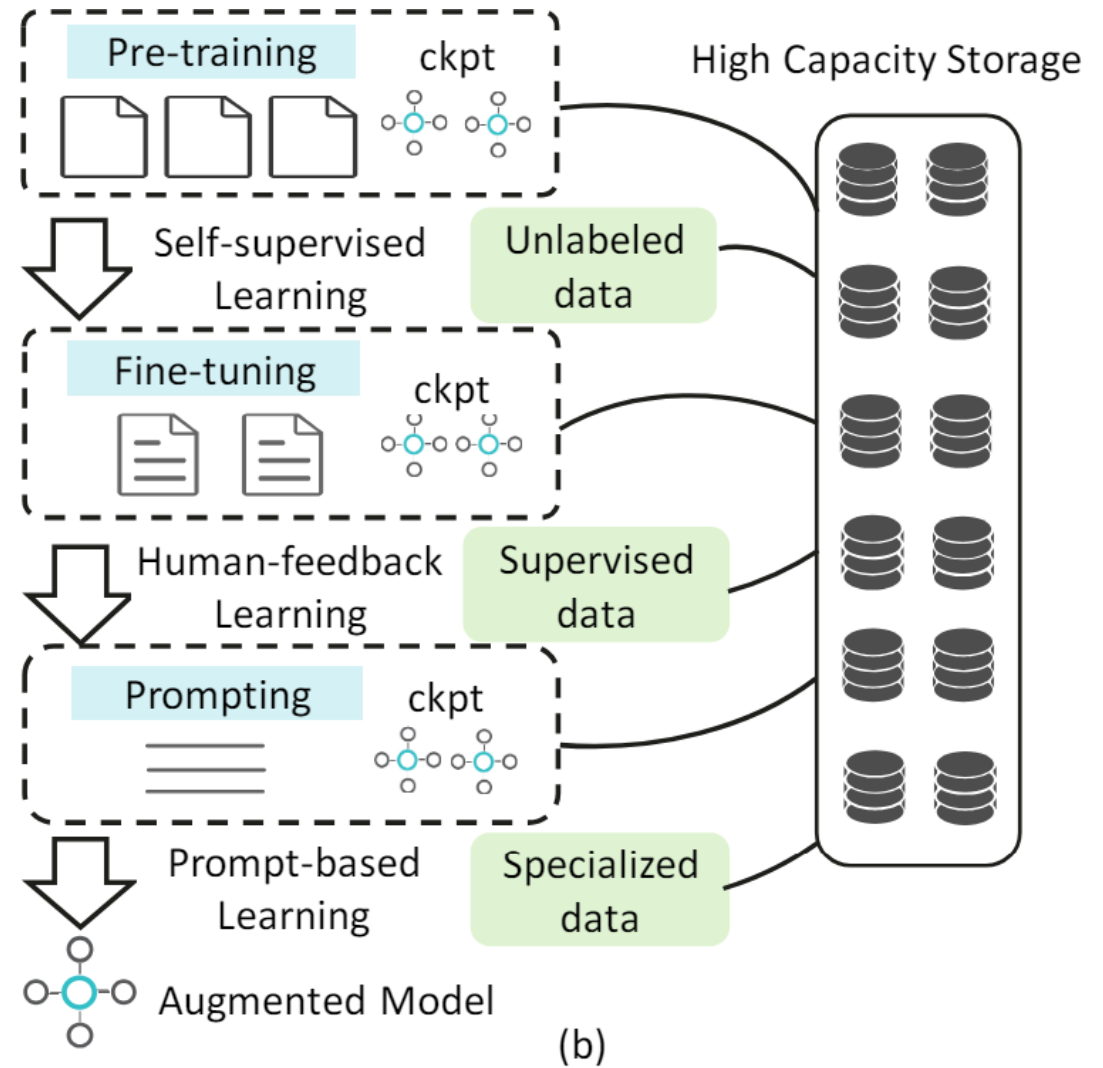
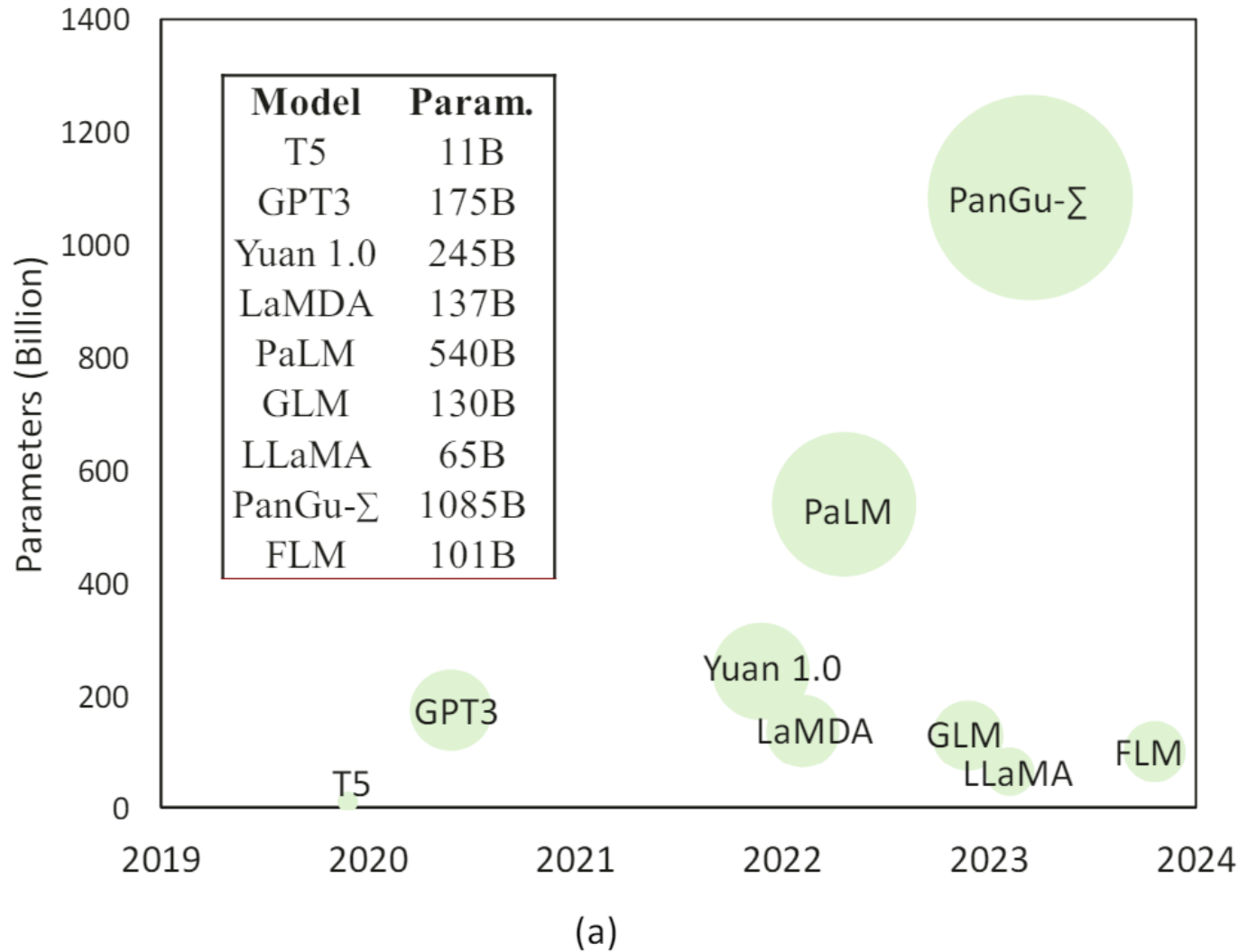


Huawei Noah's Ark Lab¹

USTC²

Motivation

Large language/vision models are consuming more and more storage resources.



Motivation

Twice the size!

Weights

&

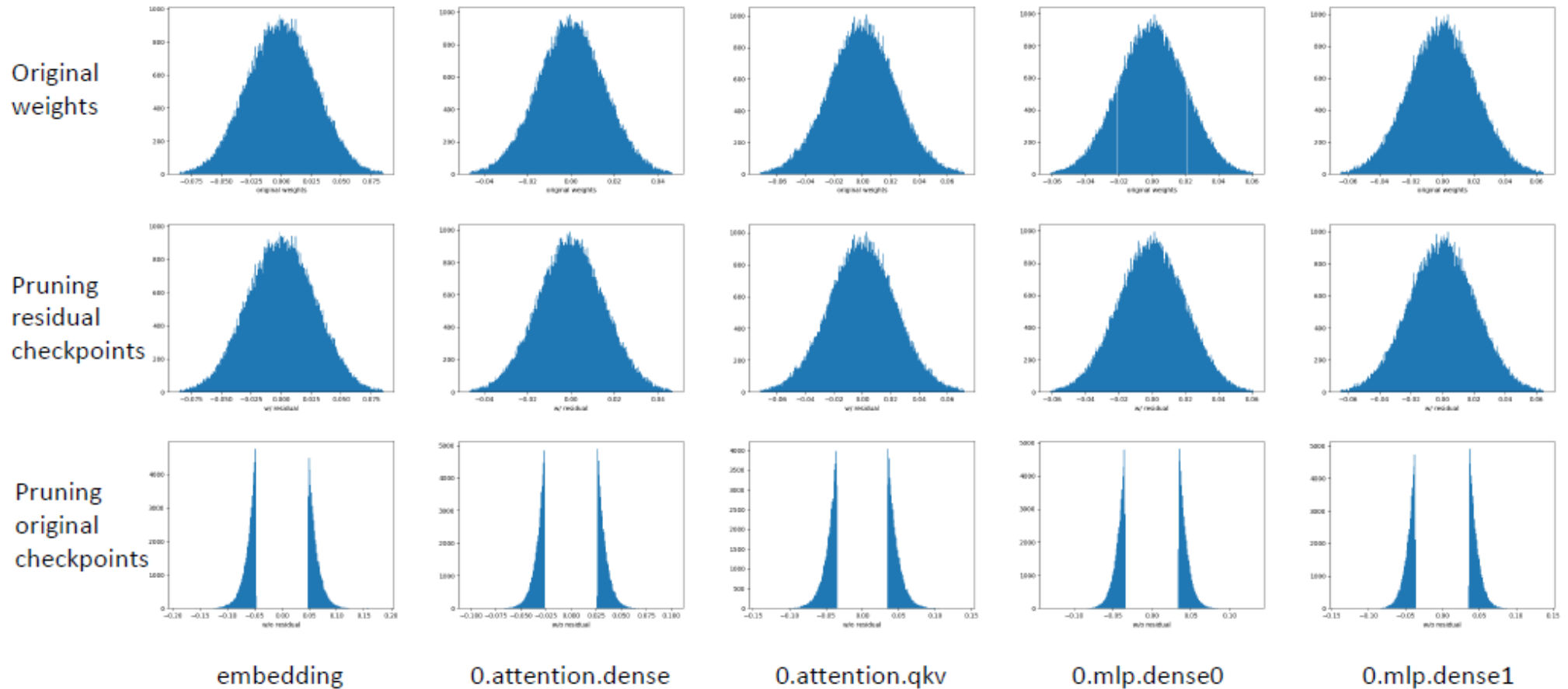
1st order Momentum

2nd order Momentum

Adam series optimizer

Model	Param.	Storage
GPT3 (Brown et al., 2020)	175B	2.3TB
PaLM (Chowdhery et al., 2023)	540B	~7TB
LLaMA-70B (Touvron et al., 2023)	75B	1.0TB
PanGu- π (Wang et al.)	7B	99GB

Motivation



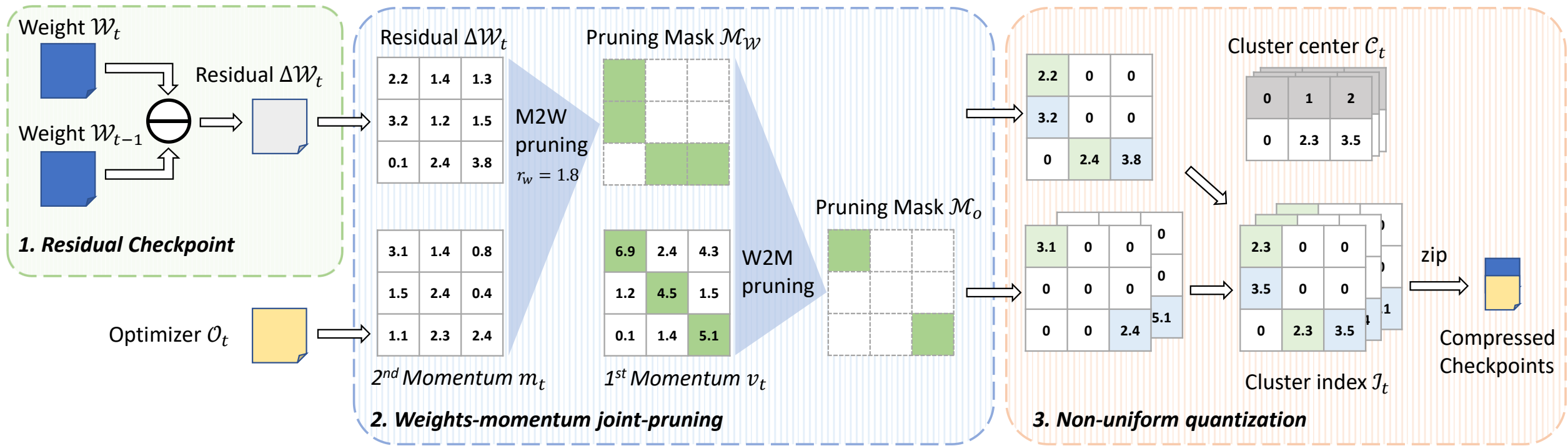
Residual Checkpoint

Pruning the residual checkpoint has almost no impact on the parameter distribution. This helps us to further prune the parameters.

Motivation

- **Previous work mainly concentrated on inference speed or the size of final checkpoints.** During the training process of LLMs or VLMs, large storage is required to store the checkpoints. The storage of optimizer states should also be considered.
- **Similarity of adjacent checkpoints should also be considered in the compression.** Compression of residual checkpoints achieves better results.

Framework of ExCP

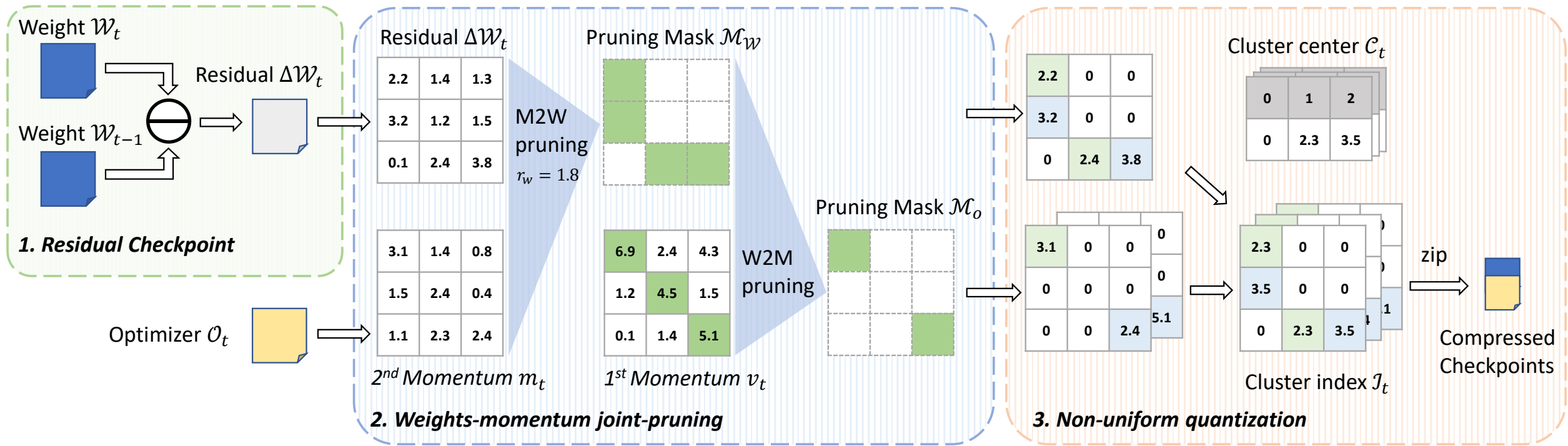


Residual Checkpoint

The difference between adjacent model weights is mostly to be sparse, which is more suitable for compression. The residual checkpoint $\Delta\mathcal{P}_t$ is defined as

$$\Delta\mathcal{P}_t = \{\Delta\mathcal{W}_t, \mathcal{O}_t\} = \{\mathcal{W}_t - \mathcal{W}_{t-1}, \mathcal{O}_t\}.$$

Framework of ExCP



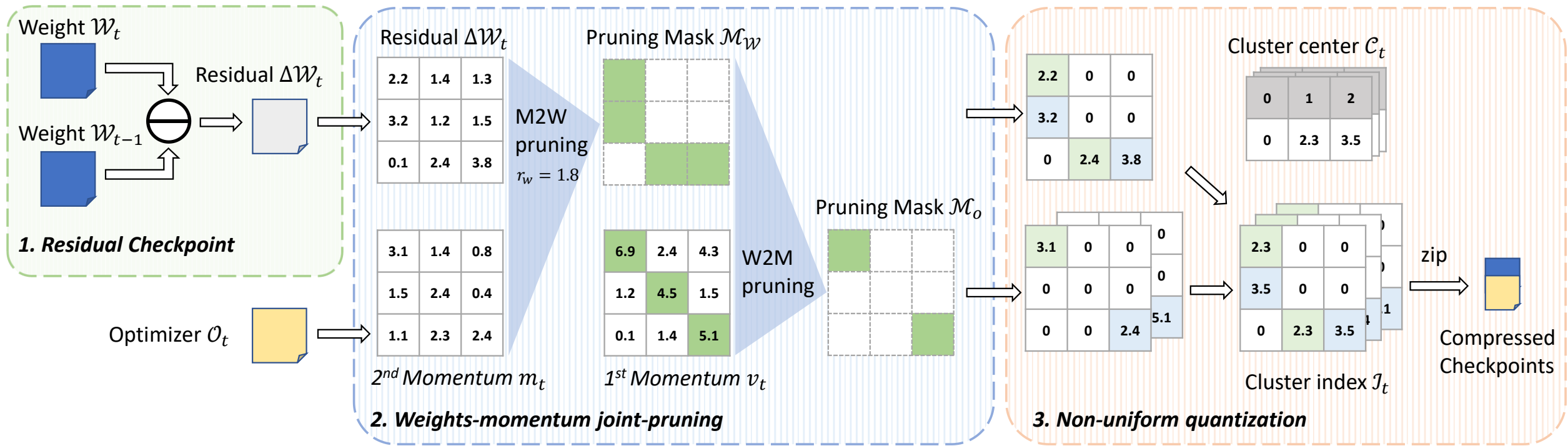
Joint Weight-Momentum Pruning

Joint pruning leads to better performance for checkpoint compression.

Weight pruning:
$$r_w = \frac{\alpha}{\sqrt{m_t}} \times \text{median}(\mathcal{W}), \mathcal{M}_w(i) = \mathbb{1}_{w_t(i) > r_w}.$$

Use the second-order momentum of gradients of weights as an indicator, since they can represent the **statistical average of the weight change** during training.

Framework of ExCP



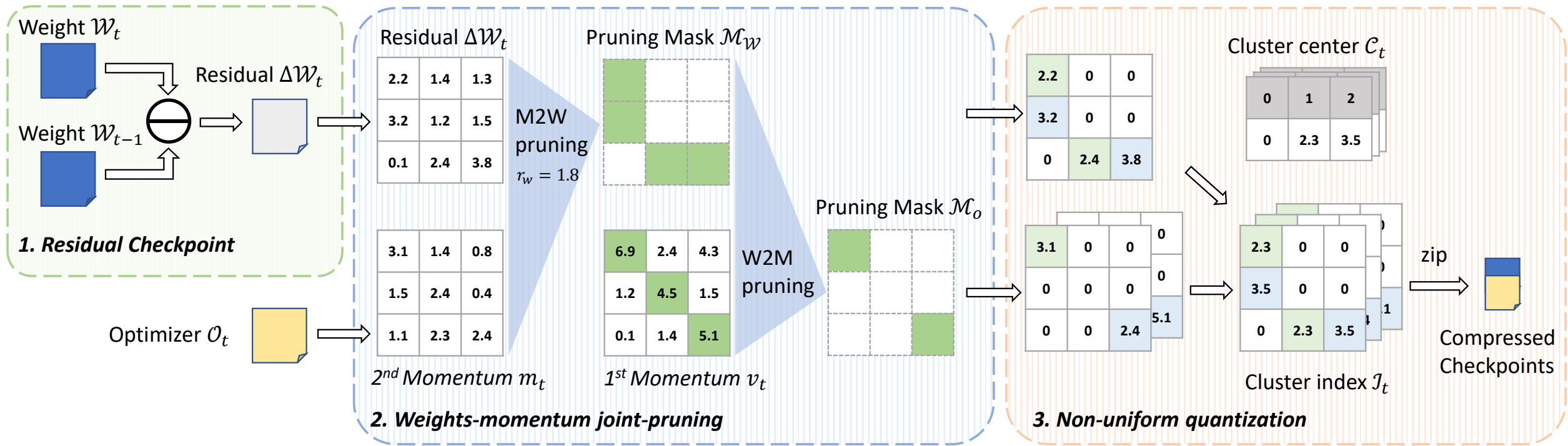
Joint Weight-Momentum Pruning

Joint pruning leads to better performance for checkpoint compression.

Momentum pruning: $r_o = \beta \times \text{mean}(v_t)$, $\mathcal{M}_o(i) = \mathbb{1}_{v_t(i) > r_o}$ and $\mathcal{M}_w(i) = 1$.

If a specific location of weights is pruned, intuitively it is not important to preserve the corresponding momentum states.

Framework of ExCP



Non-uniform quantization

Leave the pruned weights or momentum states to zero, and apply K-means algorithm on other weights or momentum states to cluster them to $2^n - 1$ cluster centers.

Compressing and Reconstructing

Algorithm 1 Compressing process

Require: last reconstructed weight checkpoint \hat{W}_{t-1} , original weight checkpoint W_t , original optimizer checkpoint

O_t

$$\Delta W_t \leftarrow W_t - \hat{W}_{t-1}$$

$$\Delta W_t^*, O_t^* \leftarrow \text{joint_prune}(\Delta W_t, O_t)$$

$$\mathcal{I}_t^W, \mathcal{C}_t^W \leftarrow \text{quantize}(\Delta W_t^*)$$

$$\mathcal{I}_t^O, \mathcal{C}_t^O \leftarrow \text{quantize}(\Delta O_t^*)$$

$$\mathcal{P}_t' \leftarrow \text{zip}(\mathcal{I}_t^W, \mathcal{C}_t^W, \mathcal{I}_t^O, \mathcal{C}_t^O)$$

save \mathcal{P}_t'

Compressing process

Algorithm 2 Reconstructing process

Require: last reconstructed weight checkpoint \hat{W}_{t-1} , compressed checkpoint \mathcal{P}_t

$$\mathcal{I}_t^W, \mathcal{C}_t^W, \mathcal{I}_t^O, \mathcal{C}_t^O \leftarrow \text{unzip}(\mathcal{P}_t)$$

$$\Delta W_t^{Q*} \leftarrow \mathcal{C}_t^W[\mathcal{I}_t^W]$$

$$\hat{O}_t \leftarrow \mathcal{C}_t^O[\mathcal{I}_t^O]$$

$$\hat{W}_t \leftarrow \hat{W}_{t-1}/W_{t-1} + \Delta W_t^{Q*}$$

Algorithm 3 Reconstructing arbitrary checkpoints

Require: random seed s , compressed checkpoints \mathcal{P}_i , required iterations t

$$\hat{W}_0 = \text{init}(s)$$

while $i < t$ **do**

$$\hat{W}_i \leftarrow \text{recon}(\hat{W}_{i-1}, \mathcal{P}_i)$$

end while

Reconstructing process

Experimental results: Pythia-410M

Table 3. Results of Pythia-410M models. We achieve almost lossless results while the storage is reduce by $\sim 70\times$.

Model	Method	Size	Tasks						
			hellaswag	arc-e	piqa	C3	cs1	lambada	Avg
Pythia-410M	Original model	4.53G	32.52	35.80	62.13	37.21	53.75	37.22	43.11
	Residual+7Zip	3.40G	32.52	35.80	62.13	37.21	53.75	37.22	43.11
	ExCP (Ours)	0.06G	31.95	37.04	62.62	36.22	52.50	37.24	42.93

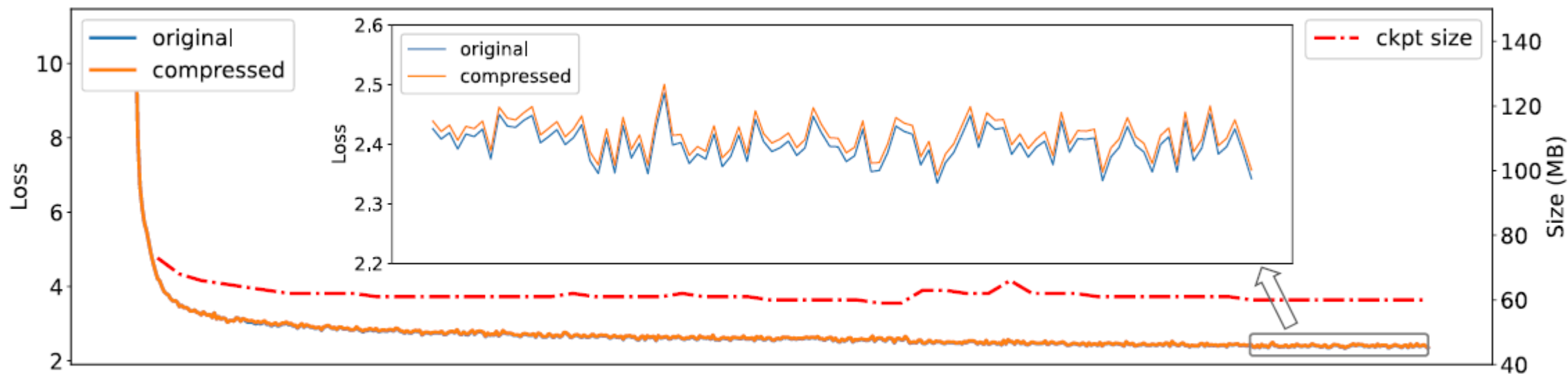


Figure 4. Comparison of training loss and checkpoint size between original models and our methods.

Experimental results: ViT-L32 and ablation study

Method	M2W pruning	W2M pruning	Top-1 Accuracy(%)	CR(Weights)	CR(Weights & Momentum)
baseline			71.36	1	1
CNR+			71.57	7.82	1.41*
QD+			71.24	16.31	1.45*
DynaQuant			71.82	26.19	1.47*
ExCP(Ours)			71.51	-	19.88
ExCP(Ours)	✓		71.53	-	25.54
ExCP(Ours)		✓	71.80	-	22.76
ExCP(Ours)	✓	✓	71.69	-	35.21

method			Size	Avg Acc
residual	prune	quant		
			4070M	43.11
✓			3484M	43.11
	✓		324M	29.95
		✓	492M	40.17
✓	✓		276M	42.92
✓		✓	493M	42.94
✓	✓	✓	61M	42.93

- Pruning without residual would harm the accuracy.
- Joint weight-momentum pruning achieve the best results.

Experimental results: examples

Question: What kind of exercise do you suggest I do indoors?

Original PanGu- π -1B: If you are looking for a way to do some exercise in the comfort of your own home, there are several great exercises you can try. Some suggestions are: Squats: This is a great exercise for all levels, from beginner to advanced. You can do them by sitting on a chair with a squat rack at the back. Slowly lower your hips until your thighs are parallel to the ground, then push it back up. 2. Plank: Planks are a great core-strengthening exercise that can help improve your posture and lower your heart rate. Start by getting into a pushup, then a plank, and finally a layover. 3. Mountain climbers: Mountain climbers are a fun and challenging exercise that can help improve your strength and endurance. Start by bringing a heavy object, like a bag or bottle, into a vertical position. Climb the object up to the ceiling, then bring it back down to the ground. 4. Mountain climbers: Mountain climbers are a challenging and exciting exercise that can help improve your upper body strength. Start by bringing a heavy object, like a bag or bottle, into a vertical position. Climb the object up to the ceiling, then bring it back down to the ground. 5. Yoga: Yoga is a great way to improve your flexibility, strength, and balance. There are many online videos and apps that you can use to learn. Remember, it's important to start slowly and gradually increase the intensity of your workouts. If you feel uncomfortable with any of these exercises, you can always turn back to doing them at your own pace.

Compressed PanGu- π -1B: There are a lot of great exercises you can do indoors to help improve your fitness, such as: 1. Yoga: Yoga is a great way to improve flexibility, strength, and balance. There are many online videos and classes available that can help you learn the basics. 2. Bodyweight exercises: You can do bodyweight exercises such as push-ups, squats, lunges, and planks to build strength and improve your overall fitness. 3. Dancing: Dancing is a fun way to improve your cardiovascular fitness, coordination, and balance. There are many online tutorials and classes available that can help you learn the basics. 4. Yoga for beginners: If you're new to yoga, there are many beginner classes available that can help you learn the basics. 5. Pilates: Pilates is a great way to improve your core strength, flexibility, and balance. There are many online videos and classes available that can help you learn the basics. Remember to start slow and gradually increase the intensity and duration of your workouts as you get stronger and more confident.

Thanks

liwenshuo@huawei.com



paper



code

SLAB: Efficient Transformers with Simplified Linear Attention and Progressive Re-parameterized Batch Normalization

Jialong Guo^{1*} Xinghao Chen^{1*} Yehui Tang¹ Yunhe Wang¹

ICML 2024 Poster

Rethinking Optimization and Architecture for Tiny Language Models

Yehui Tang¹ Kai Han¹ Fangcheng Liu¹ Yunsheng Ni¹ Yuchuan Tian² Zheyuan Bai¹ Yi-Qi Hu³
Sichao Liu³ Shangling Jui⁴ Yunhe Wang¹

ICML 2024 Poster

DiJiang: Efficient Large Language Models through Compact Kernelization

Hanting Chen^{*1} Zhicheng Liu^{*1} Xutao Wang¹ Yuchuan Tian² Yunhe Wang¹
{chenhanting,yunhe.wang}@huawei.com;

ICML 2024 Oral

Memory-Space Visual Prompting for Efficient Vision-Language Fine-Tuning

Shibo Jie¹ Yehui Tang² Ning Ding^{1,2} Zhi-Hong Deng^{1,3} Kai Han² Yunhe Wang²

ICML 2024 Poster