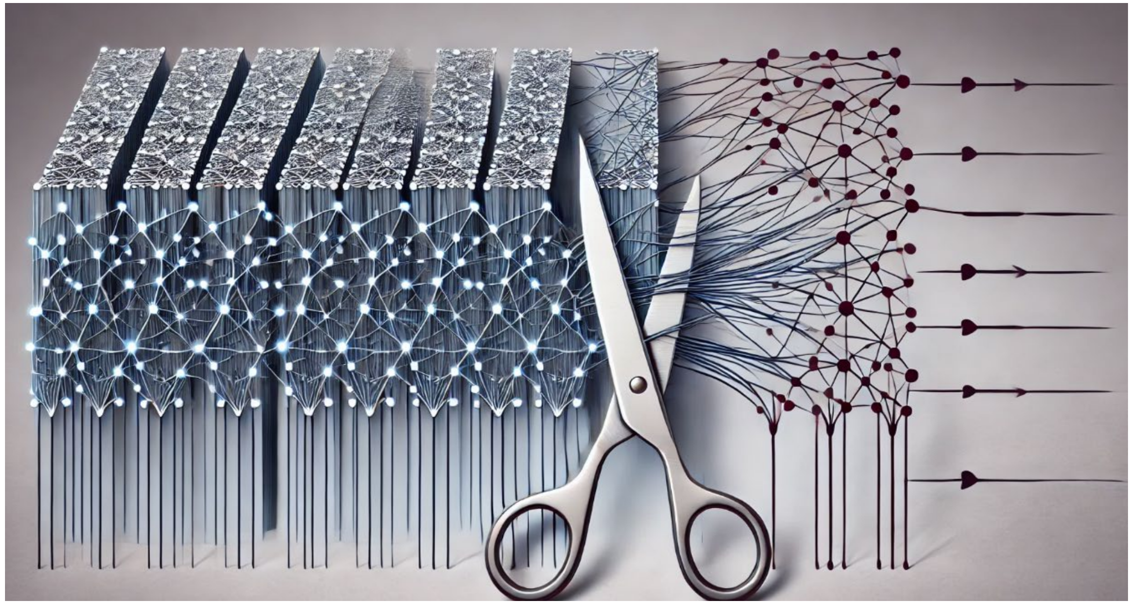
SAMSUNG

ICML

International Conference On Machine Learning

COPAL: Continual Pruning in Large Language Generative Models

Srikanth Malla, Joon Hee Choi, Chiho Choi



Introduction

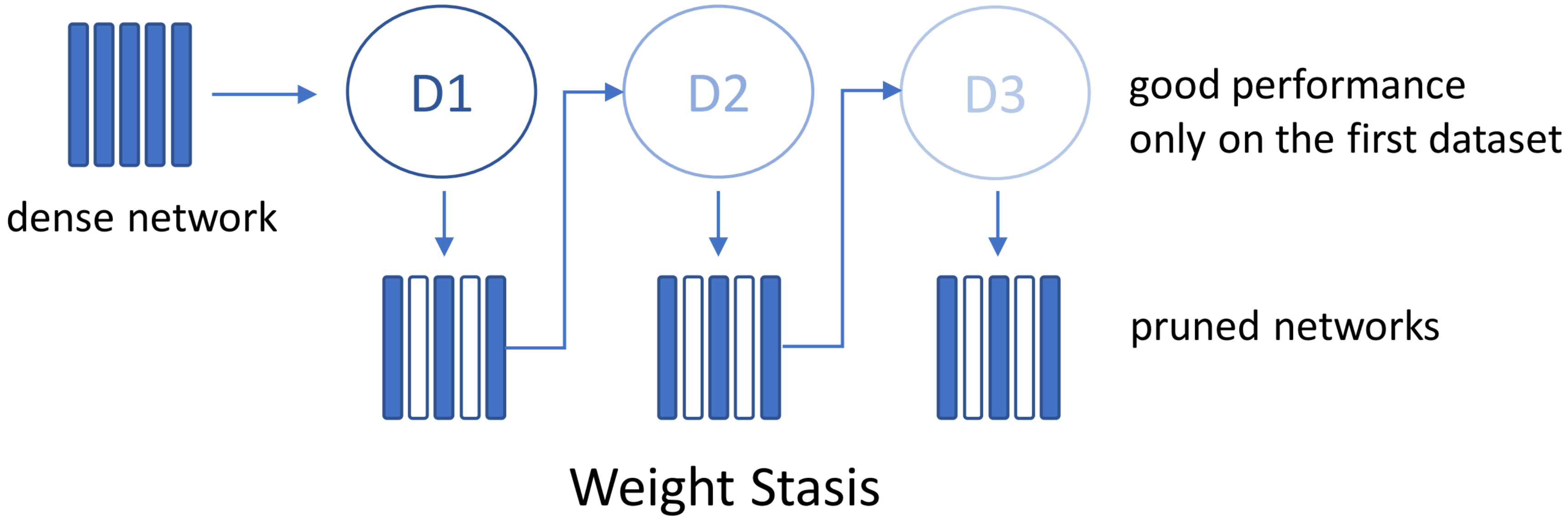
- Adapting pre-trained large language models to different domains in natural language processing requires two key considerations: high computational demands and model’s inability to continual adaptation.

Introduction

- Adapting pre-trained large language different domains in natural language processing requires two key considerations: high computational demands and model's inability to continual adaptation.

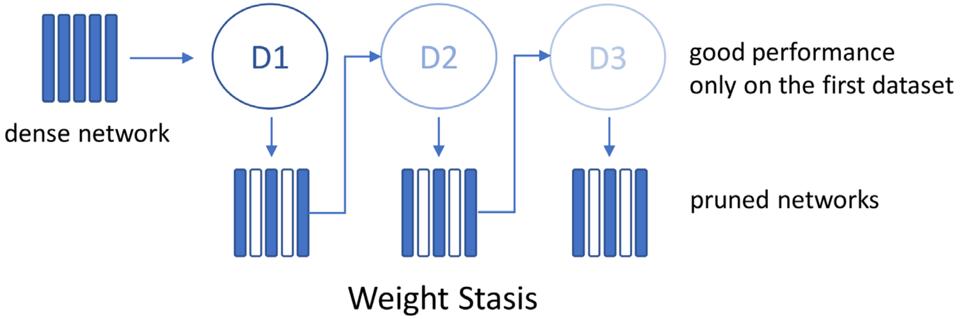
- To simultaneously address both issues, this paper presents "**CO**ntinual **P**runing in **A**daptive **L**anguage settings" (COPAL), an algorithm developed for pruning large language generative models in continual model adaptation settings.

Problem (Weight Stasis)



Problem (Weight Stasis)

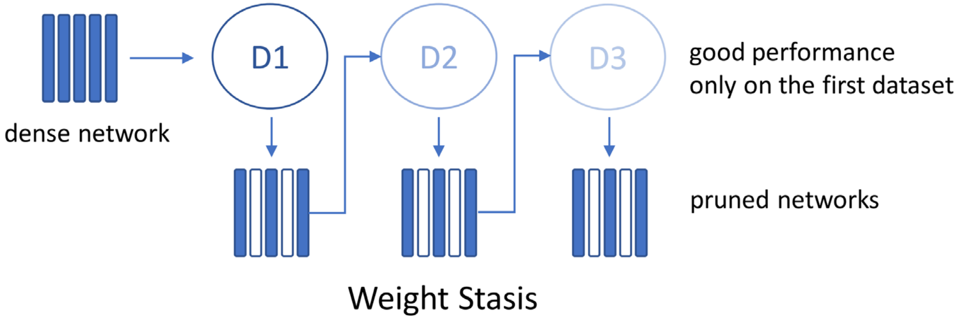
Importance of Weights $\mathbf{W}_i^* = |\mathbf{W}_i \cdot \mathcal{R}_i|$ (i is dataset index)



Problem (Weight Stasis)

Importance of Weights $\mathbf{W}_i^* = |\mathbf{W}_i \cdot \mathcal{R}_i|$ (i is dataset index)

Mask Computation $$\mathcal{M}_i = \mathcal{I}(\mathbf{W}_i^* < \mathcal{T}_s) = \begin{cases} 0 & \text{if } \mathbf{w}_i^* < \mathcal{T}_s, \mathbf{w}_i^* \in \mathbf{W}_i^* \\ 1 & \text{otherwise.} \end{cases},$$



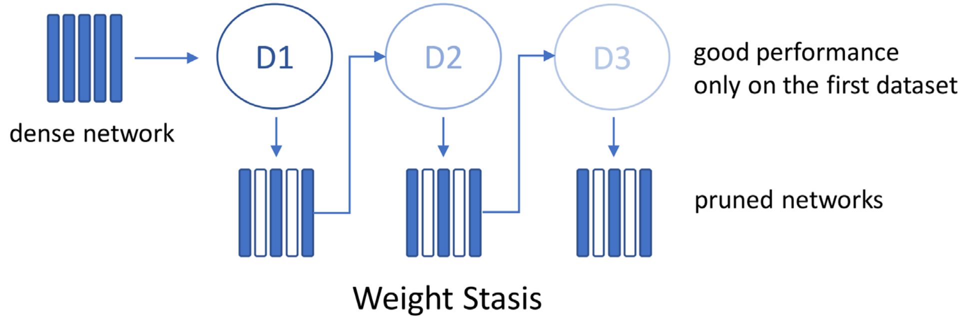
Problem (Weight Stasis)

Importance of Weights $\mathbf{W}_i^* = |\mathbf{W}_i \cdot \mathcal{R}_i|$

(i is dataset index)

Mask Computation $\mathcal{M}_i = \mathcal{I}(\mathbf{W}_i^* < \mathcal{T}_s) = \begin{cases} 0 & \text{if } \mathbf{w}_i^* < \mathcal{T}_s, \mathbf{w}_i^* \in \mathbf{W}_i^* \\ 1 & \text{otherwise.} \end{cases},$

Pruned Weights $\mathbf{W}_i^p = \mathbf{W}_i \cdot \mathcal{M}_i.$



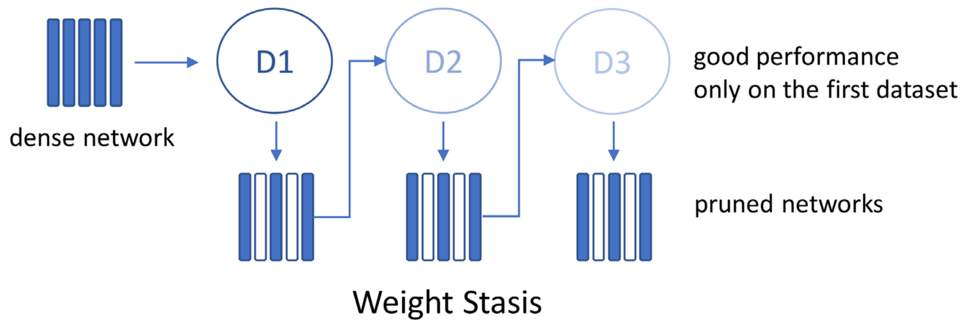
Problem (Weight Stasis)

Importance of Weights $\mathbf{W}_i^* = |\mathbf{W}_i \cdot \mathcal{R}_i|$ (i is dataset index)

Mask Computation $\mathcal{M}_i = \mathcal{I}(\mathbf{W}_i^* < \mathcal{T}_s) = \begin{cases} 0 & \text{if } \mathbf{w}_i^* < \mathcal{T}_s, \mathbf{w}_i^* \in \mathbf{W}_i^* \\ 1 & \text{otherwise.} \end{cases},$

Pruned Weights $\mathbf{W}_i^p = \mathbf{W}_i \cdot \mathcal{M}_i.$

Weights for Next Dataset $\mathbf{W}_{i+1} = \mathbf{W}_i^p$



Problem (Weight Stasis)

Importance of Weights $\mathbf{W}_i^* = |\mathbf{W}_i \cdot \mathcal{R}_i|$ (i is dataset index)

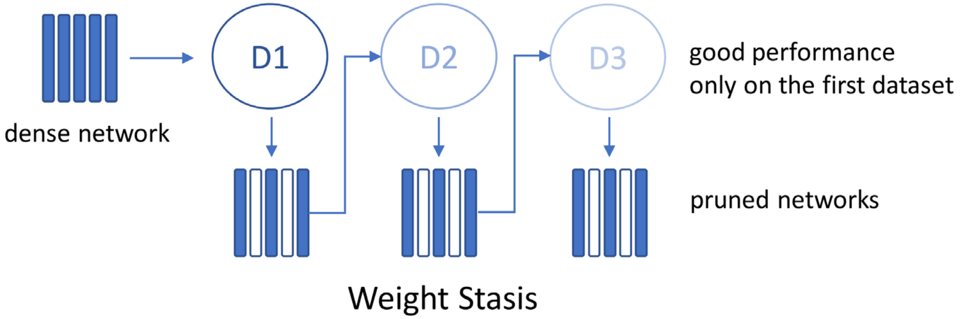
Mask Computation $\mathcal{M}_i = \mathcal{I}(\mathbf{W}_i^* < \mathcal{T}_s) = \begin{cases} 0 & \text{if } \mathbf{w}_i^* < \mathcal{T}_s, \mathbf{w}_i^* \in \mathbf{W}_i^* \\ 1 & \text{otherwise.} \end{cases},$

Pruned Weights $\mathbf{W}_i^p = \mathbf{W}_i \cdot \mathcal{M}_i.$

Weights for Next Dataset $\mathbf{W}_{i+1} = \mathbf{W}_i^p$

Mask Computation on Next Dataset

$$\begin{aligned} \mathcal{M}_{i+1} &= \mathcal{I}\left(\mathbf{W}_{i+1}^* < \mathcal{T}_s\right) \\ &= \mathcal{I}(|\mathbf{W}_{i+1} \cdot \mathcal{R}_{i+1}| < \mathcal{T}_s) \\ &= \mathcal{I}(|(\mathbf{W}_i \cdot \mathcal{M}_i) \cdot \mathcal{R}_{i+1}| < \mathcal{T}_s) \\ &= \mathcal{I}(|\mathbf{W}_i \cdot \mathcal{R}_{i+1}| \cdot |\mathcal{M}_i| < \mathcal{T}_s) \\ &\begin{cases} 0 & \text{if } m < \mathcal{T}_s, m \in |\mathbf{W}_i \cdot \mathcal{R}_{i+1}| \cdot |\mathcal{M}_i| \\ 1 & \text{otherwise.} \end{cases}, \\ &= \mathcal{M}_i, \end{aligned}$$



Problem (Weight Stasis)

(i is dataset index)

Importance of Weights $\mathbf{W}_i^* = |\mathbf{W}_i \cdot \mathcal{R}_i|$

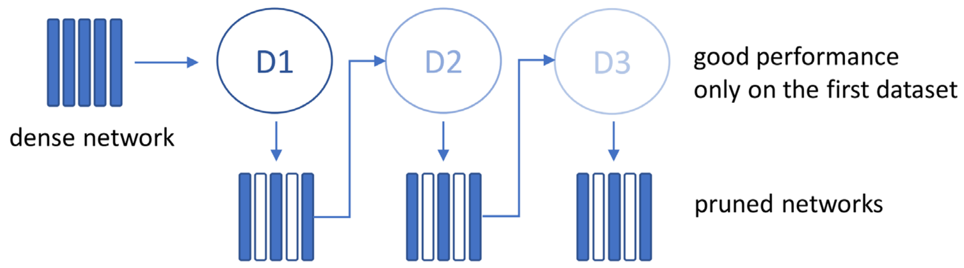
Mask Computation $\mathcal{M}_i = \mathcal{I}(\mathbf{W}_i^* < \mathcal{T}_s) = \begin{cases} 0 & \text{if } \mathbf{w}_i^* < \mathcal{T}_s, \mathbf{w}_i^* \in \mathbf{W}_i^* \\ 1 & \text{otherwise.} \end{cases},$

Pruned Weights $\mathbf{W}_i^p = \mathbf{W}_i \cdot \mathcal{M}_i.$

Weights for Next Dataset $\mathbf{W}_{i+1} = \mathbf{W}_i^p$

Mask Computation on Next Dataset

$$\begin{aligned}
\mathcal{M}_{i+1} &= \mathcal{I}\left(\mathbf{W}_{i+1}^* < \mathcal{T}_s\right) \\
&= \mathcal{I}(|\mathbf{W}_{i+1} \cdot \mathcal{R}_{i+1}| < \mathcal{T}_s) \\
&= \mathcal{I}(|(\mathbf{W}_i \cdot \mathcal{M}_i) \cdot \mathcal{R}_{i+1}| < \mathcal{T}_s) \\
&= \mathcal{I}(|\mathbf{W}_i \cdot \mathcal{R}_{i+1}| \cdot |\mathcal{M}_i| < \mathcal{T}_s) \\
& \begin{cases} 0 & \text{if } m < \mathcal{T}_s, m \in |\mathbf{W}_i \cdot \mathcal{R}_{i+1}| \cdot |\mathcal{M}_i| \\ 1 & \text{otherwise.} \end{cases}, \\
&= \mathcal{M}_i,
\end{aligned}$$

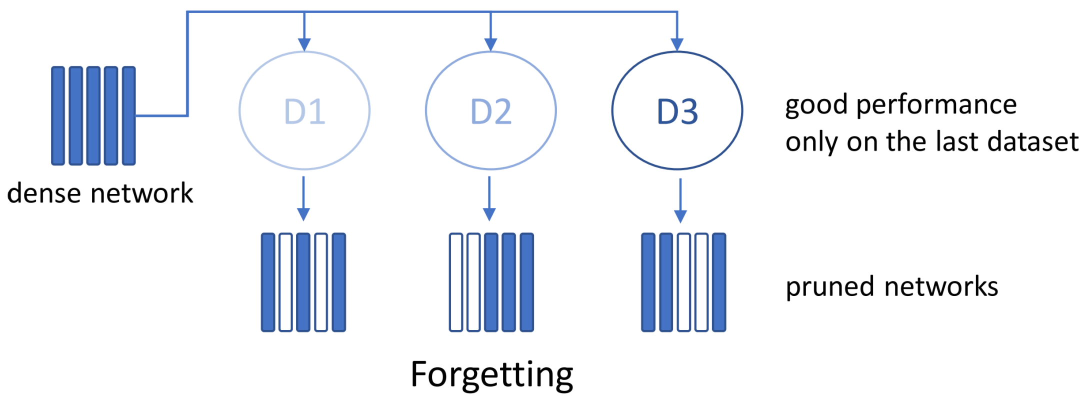


Weight Stasis

Weight Stasis Phenomenon $\mathbf{W}_{i+2} = \mathbf{W}_{i+1} = \mathbf{W}_i \cdot \mathcal{M}_i.$

Problem (Forgetting)

- The post training continual pruning performs better on the last dataset that we used calibrated dataset from, under global initialization of weights. Whereas the sequential initialization of weights suffers with weight stasis as shown before.



Sensitivity Analysis

Layer in a neural network $\mathbf{y}_j^i = f(\mathbf{x}_j^i, \mathbf{W})$ (j is sample index in dataset index i)

Sensitivity Analysis

Layer in a neural network $\mathbf{y}_j^i = f(\mathbf{x}_j^i, \mathbf{W})$ (j is sample index in dataset index i)

Output sensitivity $d\mathbf{y}_j^i = \frac{\partial f}{\partial \mathbf{x}_j^i} d\mathbf{x}_j^i + \frac{\partial f}{\partial \mathbf{W}} d\mathbf{W}$

Sensitivity Analysis

Layer in a neural network $\mathbf{y}_j^i = f(\mathbf{x}_j^i, \mathbf{W})$ (j is sample index in dataset index i)

Output sensitivity $d\mathbf{y}_j^i = \frac{\partial f}{\partial \mathbf{x}_j^i} d\mathbf{x}_j^i + \frac{\partial f}{\partial \mathbf{W}} d\mathbf{W}$

Sensitivity measures

$$S_{\mathbf{W}}^{ij} = f(\mathbf{W} + \Delta\mathbf{W}, \mathbf{x}_j^i) - \mathbf{y}_j^i,$$
$$S_{\mathbf{x}}^{ij} = f(\mathbf{W}, \mathbf{x}_j^i + \Delta\mathbf{x}_j^i) - \mathbf{y}_j^i.$$

Sensitivity Analysis

Layer in a neural network $\mathbf{y}_j^i = f(\mathbf{x}_j^i, \mathbf{W})$ (j is sample index in dataset index i)

Output sensitivity $d\mathbf{y}_j^i = \frac{\partial f}{\partial \mathbf{x}_j^i} d\mathbf{x}_j^i + \frac{\partial f}{\partial \mathbf{W}} d\mathbf{W}$

Sensitivity measures

$$S_{\mathbf{W}}^{ij} = f(\mathbf{W} + \Delta\mathbf{W}, \mathbf{x}_j^i) - \mathbf{y}_j^i,$$
$$S_{\mathbf{x}}^{ij} = f(\mathbf{W}, \mathbf{x}_j^i + \Delta\mathbf{x}_j^i) - \mathbf{y}_j^i.$$

Combined sensitivity $d\mathbf{y}_j^i = S_{\mathbf{W}}^{ij} + S_{\mathbf{x}}^{ij}$

Sensitivity Analysis

Layer in a neural network $\mathbf{y}_j^i = f(\mathbf{x}_j^i, \mathbf{W})$ (j is sample index in dataset index i)

Output sensitivity $d\mathbf{y}_j^i = \frac{\partial f}{\partial \mathbf{x}_j^i} d\mathbf{x}_j^i + \frac{\partial f}{\partial \mathbf{W}} d\mathbf{W}$

Sensitivity measures

$$S_{\mathbf{W}}^{ij} = f(\mathbf{W} + \Delta\mathbf{W}, \mathbf{x}_j^i) - \mathbf{y}_j^i,$$
$$S_{\mathbf{x}}^{ij} = f(\mathbf{W}, \mathbf{x}_j^i + \Delta\mathbf{x}_j^i) - \mathbf{y}_j^i.$$

Combined sensitivity $d\mathbf{y}_j^i = S_{\mathbf{W}}^{ij} + S_{\mathbf{x}}^{ij}$

Loss function $\mathcal{L}_j^i = \left\| d\mathbf{y}_j^i \right\|_2^2$

Sensitivity Analysis

Layer in a neural network $\mathbf{y}_j^i = f(\mathbf{x}_j^i, \mathbf{W})$ (j is sample index in dataset index i)

Output sensitivity $$d\mathbf{y}_j^i = \frac{\partial f}{\partial \mathbf{x}_j^i} d\mathbf{x}_j^i + \frac{\partial f}{\partial \mathbf{W}} d\mathbf{W}$$

Sensitivity measures $$S_{\mathbf{W}}^{ij} = f(\mathbf{W} + \Delta\mathbf{W}, \mathbf{x}_j^i) - \mathbf{y}_j^i,$$
$$S_{\mathbf{x}}^{ij} = f(\mathbf{W}, \mathbf{x}_j^i + \Delta\mathbf{x}_j^i) - \mathbf{y}_j^i.$$

Combined sensitivity $$d\mathbf{y}_j^i = S_{\mathbf{W}}^{ij} + S_{\mathbf{x}}^{ij}$$

Loss function $$\mathcal{L}_j^i = \left\| d\mathbf{y}_j^i \right\|_2^2$$

Derivative w.r.t perturbations in W $$\nabla_{d\mathbf{W}} \mathcal{L}_j^i = 2 d\mathbf{y}_j^i \frac{\partial f}{\partial \mathbf{W}}$$

Sensitivity Analysis

Layer in a neural network $\mathbf{y}_j^i = f(\mathbf{x}_j^i, \mathbf{W})$ (j is sample index in dataset index i)

Output sensitivity $d\mathbf{y}_j^i = \frac{\partial f}{\partial \mathbf{x}_j^i} d\mathbf{x}_j^i + \frac{\partial f}{\partial \mathbf{W}} d\mathbf{W}$

Sensitivity measures
$$S_{\mathbf{W}}^{ij} = f(\mathbf{W} + \Delta\mathbf{W}, \mathbf{x}_j^i) - \mathbf{y}_j^i,$$
$$S_{\mathbf{x}}^{ij} = f(\mathbf{W}, \mathbf{x}_j^i + \Delta\mathbf{x}_j^i) - \mathbf{y}_j^i.$$

Combined sensitivity $d\mathbf{y}_j^i = S_{\mathbf{W}}^{ij} + S_{\mathbf{x}}^{ij}$

Loss function $\mathcal{L}_j^i = \left\| d\mathbf{y}_j^i \right\|_2^2$

Derivative w.r.t perturbations in W $\nabla_{d\mathbf{W}} \mathcal{L}_j^i = 2 d\mathbf{y}_j^i \frac{\partial f}{\partial \mathbf{W}}$

Absolute of derivative
$$\begin{aligned}
\nabla'_{d\mathbf{W}} \mathcal{L}^k &= \sum_{i=0}^{k} \sum_j \left| \nabla_{d\mathbf{W}} \mathcal{L}_j^i \right| \\
&= \sum_j 2 \left| d\mathbf{y}_j^k \frac{\partial f}{\partial \mathbf{W}} \right| + \sum_{i=0:k-1} \sum_j 2 \left| d\mathbf{y}_j^i \frac{\partial f}{\partial \mathbf{W}} \right| \\
&= \sum_j 2 \left| d\mathbf{y}_j^k \right| \left| \frac{\partial f}{\partial \mathbf{W}} \right| + \nabla'_{d\mathbf{W}} \mathcal{L}_{k-1} \\
&= \nabla'_{d\mathbf{W}} \tilde{\mathcal{L}}^k + \nabla'_{d\mathbf{W}} \mathcal{L}^{k-1},
\end{aligned}$$

Sensitivity Analysis

Layer in a neural network $\mathbf{y}_j^i = f(\mathbf{x}_j^i, \mathbf{W})$ (j is sample index in dataset index i)

Output sensitivity $d\mathbf{y}_j^i = \frac{\partial f}{\partial \mathbf{x}_j^i} d\mathbf{x}_j^i + \frac{\partial f}{\partial \mathbf{W}} d\mathbf{W}$

Sensitivity measures

$$S_{\mathbf{W}}^{ij} = f(\mathbf{W} + \Delta\mathbf{W}, \mathbf{x}_j^i) - \mathbf{y}_j^i,$$
$$S_{\mathbf{x}}^{ij} = f(\mathbf{W}, \mathbf{x}_j^i + \Delta\mathbf{x}_j^i) - \mathbf{y}_j^i.$$

Combined sensitivity $d\mathbf{y}_j^i = S_{\mathbf{W}}^{ij} + S_{\mathbf{x}}^{ij}$

Loss function $\mathcal{L}_j^i = \left\| d\mathbf{y}_j^i \right\|_2^2$

Derivative w.r.t perturbations in W $\nabla_{d\mathbf{W}} \mathcal{L}_j^i = 2 d\mathbf{y}_j^i \frac{\partial f}{\partial \mathbf{W}}$

Absolute of derivative

$$\begin{aligned}
\nabla'_{d\mathbf{W}} \mathcal{L}^k &= \sum_{i=0}^{k} \sum_{j} \left| \nabla_{d\mathbf{W}} \mathcal{L}_j^i \right| \\
&= \sum_j 2 \left| d\mathbf{y}_j^k \frac{\partial f}{\partial \mathbf{W}} \right| + \sum_{i=0:k-1} \sum_j 2 \left| d\mathbf{y}_j^i \frac{\partial f}{\partial \mathbf{W}} \right| \\
&= \sum_j 2 \left| d\mathbf{y}_j^k \right| \left| \frac{\partial f}{\partial \mathbf{W}} \right| + \nabla'_{d\mathbf{W}} \mathcal{L}_{k-1} \\
&= \nabla'_{d\mathbf{W}} \tilde{\mathcal{L}}^k + \nabla'_{d\mathbf{W}} \mathcal{L}^{k-1},
\end{aligned}$$

Important weights (Directional derivative)

$$\begin{aligned}
\mathbf{W}_k^* &= \sum_{i=0:k} \sum_j \left| D_{\mathbf{W}} \mathcal{L}_j^i \right| \\
&= \sum_{i=0:k} \sum_j \left| \mathbf{W} \cdot \nabla_{d\mathbf{W}} \mathcal{L}_j^i \right| \\
&= |\mathbf{W}| \cdot \sum_{i=0:k} \sum_j \left| \nabla_{d\mathbf{W}} \mathcal{L}_j^i \right| \\
&= |\mathbf{W}| \cdot \nabla'_{d\mathbf{W}} \mathcal{L}^k \\
&= |\mathbf{W} \cdot| \left(\nabla_{d\mathbf{W}} \tilde{\mathcal{L}}^k + \nabla'_{d\mathbf{W}} \mathcal{L}_{k-1} \right) \\
&= \sum_j \left| \mathbf{W} \cdot \nabla_{d\mathbf{W}} \mathcal{L}_j^k \right| + \mathbf{W}_{k-1}^*.
\end{aligned}$$

Our Approach

Algorithm 1 COPAL

Input: Weights $\mathbf{W}$, Sparsity ratio s,
Datasets $\mathcal{D}_1, \cdots, \mathcal{D}_k$,
using j-th input data from dataset i the input and output feature of the layer f are $\left(\mathbf{x}_j^i, \mathbf{y}_j^i\right)$

Output: Pruned weights $\mathbf{W}_k^p$

Initialize: $\mathbf{W}_0^* = 0$

for i = 1:k **do**

$S_{\mathbf{W}}^{ij} \leftarrow f(\mathbf{W} + \Delta\mathbf{W}, \mathbf{x}_j^i) - \mathbf{y}_j^i$

$S_{\mathbf{x}}^{ij} \leftarrow f(\mathbf{W}, \mathbf{x}_j^i + \Delta\mathbf{x}_j^i) - \mathbf{y}_j^i$

$d\mathbf{y}_j^i \leftarrow S_{\mathbf{W}}^{ij} + S_{\mathbf{x}}^{ij}$

$$\frac{\partial f}{\partial \mathbf{W}} \leftarrow \begin{cases} S_{\mathbf{W}}^{ij} \Delta\mathbf{W}^{+}, & \text{if } f \text{ is non-linear layer} \\ x, & \text{if } f \text{ is linear layer} \end{cases}$$

$\nabla_{d\mathbf{W}} \mathcal{L}_j^i \leftarrow 2 d\mathbf{y}_j^i \frac{\partial f}{\partial \mathbf{W}}$

$\mathbf{W}_i^* \leftarrow \sum_j \left| \mathbf{W} \cdot \nabla_{d\mathbf{W}} \mathcal{L}_j^i \right| + \mathbf{W}_{i-1}^*$

$N \leftarrow$ total number of elements in $\mathbf{W}_i^*$

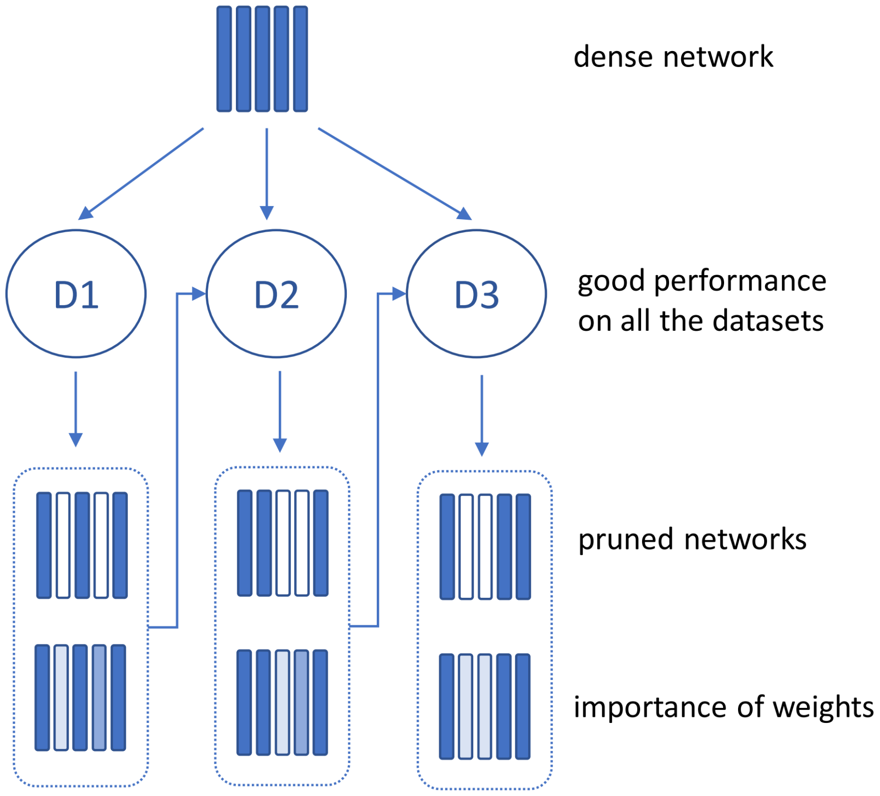
$\mathcal{T}_s \leftarrow$ Sorted $\mathbf{W}_i^* \, [\lceil (1 - s/100) \times N \rceil]$

$$\mathcal{M}_i \leftarrow \begin{cases} 0, & \text{if } w^i < \mathcal{T}_s, \quad w^i \in \mathbf{W}_i^* \\ 1, & \text{otherwise} \end{cases}$$

$\mathbf{W}_i^p \leftarrow \mathbf{W}_i \cdot \mathcal{M}_i$

end for

Return: $\mathbf{W}_k^p$



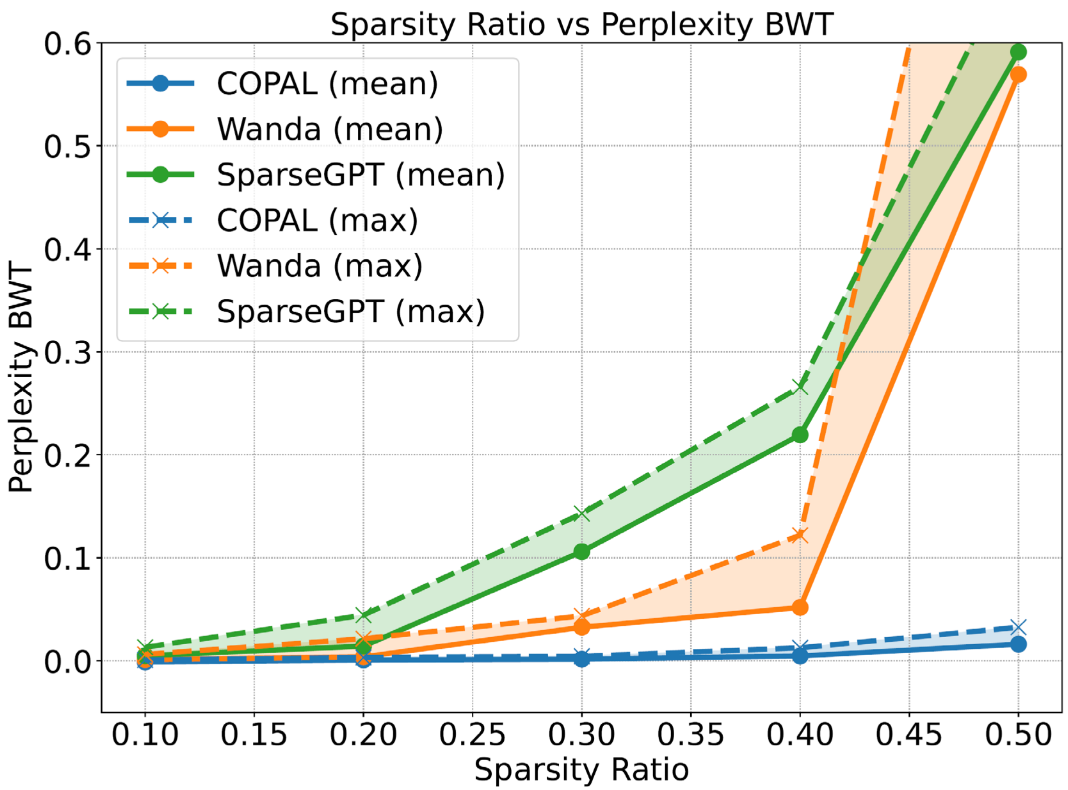
COPAL Framework

Results

	LLAMA-7B				LLAMA-30B			
	A-BWT	M-BWT	A-PPL	M-PPL	A-BWT	M-BWT	A-PPL	M-PPL
DENSE (NO PRUNING)	-	-	7.714	10.120	-	-	6.131	8.159
UNSTRUCTURED								
MAGNITUDE	WS	WS	30.246	49.670	WS	WS	10.958	14.638
SPARSEGPT	0.591	0.690	10.166	13.253	0.395	0.730	7.452	9.520
WANDA	0.569	1.072	9.991	13.626	0.132	0.192	7.261	9.231
COPAL (OURS)	**0.016**	**0.032**	**9.728**	**12.585**	**0.007**	**0.025**	**7.240**	**9.081**
SEMI STRUCTURED 2:4								
MAGNITUDE	WS	WS	131.653	303.710	WS	WS	13.757	19.139
SPARSEGPT	4.365	6.391	15.744	21.771	1.436	3.015	9.592	12.657
WANDA	1.667	3.192	16.154	23.266	0.493	1.079	9.363	12.183
COPAL (OURS)	**0.009**	**0.075**	**15.335**	**21.159**	**0.036**	**0.100**	**9.274**	**11.478**
SEMI STRUCTURED 4:8								
MAGNITUDE	WS	WS	32.105	56.652	WS	WS	12.998	16.881
SPARSEGPT	1.929	3.045	11.924	15.884	0.838	1.670	8.351	10.790
WANDA	0.771	1.645	11.929	16.631	0.231	0.486	8.094	10.353
COPAL (OURS)	**0.038**	**0.075**	**11.734**	**15.532**	**0.012**	**0.050**	**8.032**	**9.982**

	LLAMA-13B				LLAMA-65B			
	A-BWT	M-BWT	A-PPL	M-PPL	A-BWT	M-BWT	A-PPL	M-PPL
DENSE (NO PRUNING)	-	-	6.990	9.081	-	-	6.139	8.878
UNSTRUCTURED								
MAGNITUDE	WS	WS	28.935	41.368	WS	WS	9.399	13.701
SPARSEGPT	0.606	1.123	8.467	11.094	0.334	0.564	7.235	9.874
WANDA	0.203	0.298	8.570	10.896	0.172	0.628	7.584	11.354
COPAL (OURS)	**0.029**	**0.078**	**8.354**	**10.818**	**0.001**	**0.208**	**6.791**	**8.839**
SEMI-STRUCTURED (2:4)								
MAGNITUDE	WS	WS	28.702	44.072	WS	WS	10.544	14.704
SPARSEGPT	2.670	5.978	11.970	17.386	2.128	6.979	9.333	16.037
WANDA	0.871	1.698	13.209	18.920	0.184	0.389	9.230	12.665
COPAL (OURS)	**-0.007**	**0.340**	**11.455**	**17.155**	**0.038**	**0.258**	**9.161**	**11.233**
SEMI-STRUCTURED (4:8)								
MAGNITUDE	WS	WS	20.476	29.055	WS	WS	9.247	12.568
SPARSEGPT	1.248	2.905	9.773	13.456	0.907	2.992	8.363	13.358
WANDA	0.375	0.793	9.946	13.281	0.172	0.610	8.315	11.746
COPAL (OURS)	**-0.019**	**0.160**	**9.402**	**12.338**	**0.019**	**0.260**	**8.291**	**10.807**

Results of continual pruning on wikitext2, ptb, c4 datasets with all permutations.



LLAMA-7B

Thank you

- Please visit our poster on 24th Wed 2024 Jul 4:30 a.m. PDT — 6 a.m. PDT (Hall C 4-9)
- Link to the Arxiv paper.