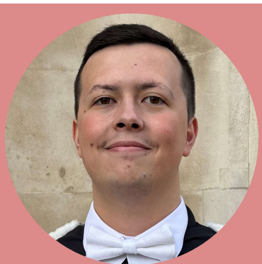
DAGER: Exact Gradient Inversion for Large Language Models


Ivo Petrov


Dimitar I. Dimitrov


Maximilian Baader

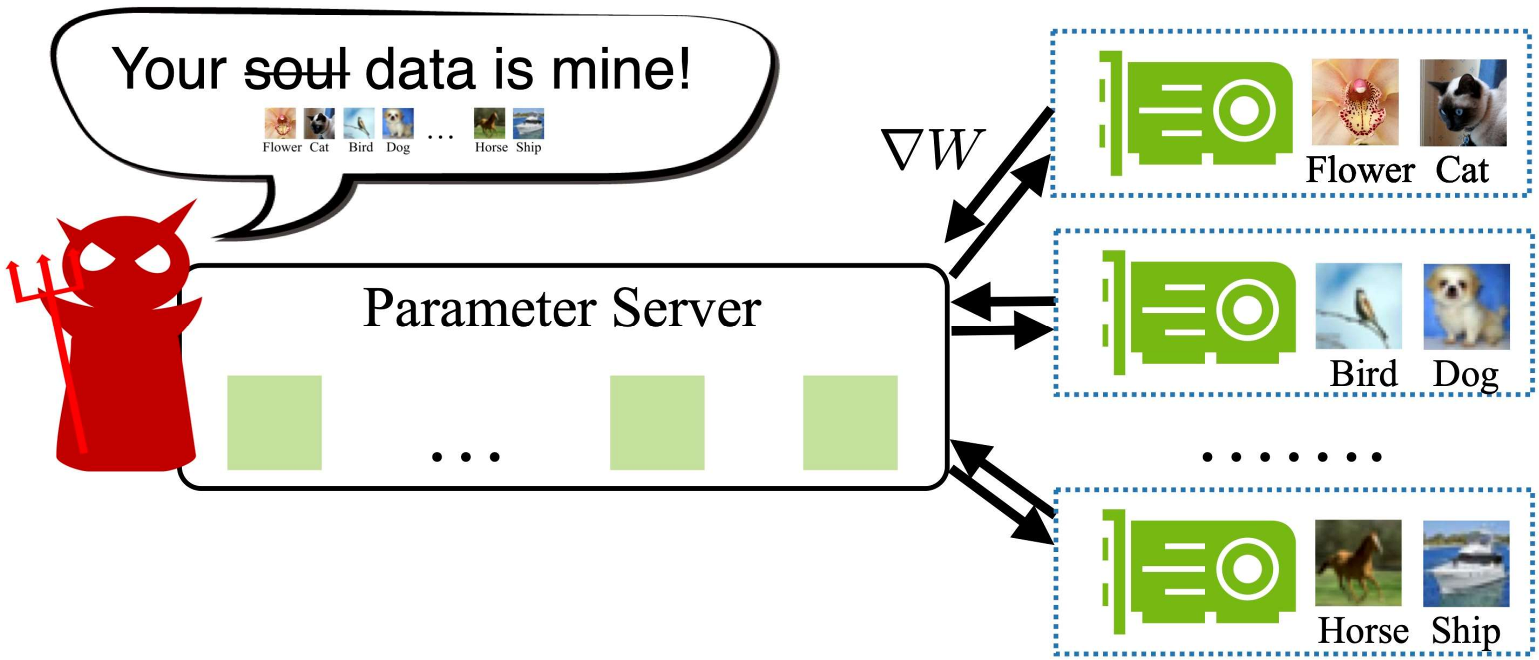

Mark Müller


Martin Vechev

INSAIT ETH zürich SRILAB

NeurIPS 2024 page:

Gradient Inversion



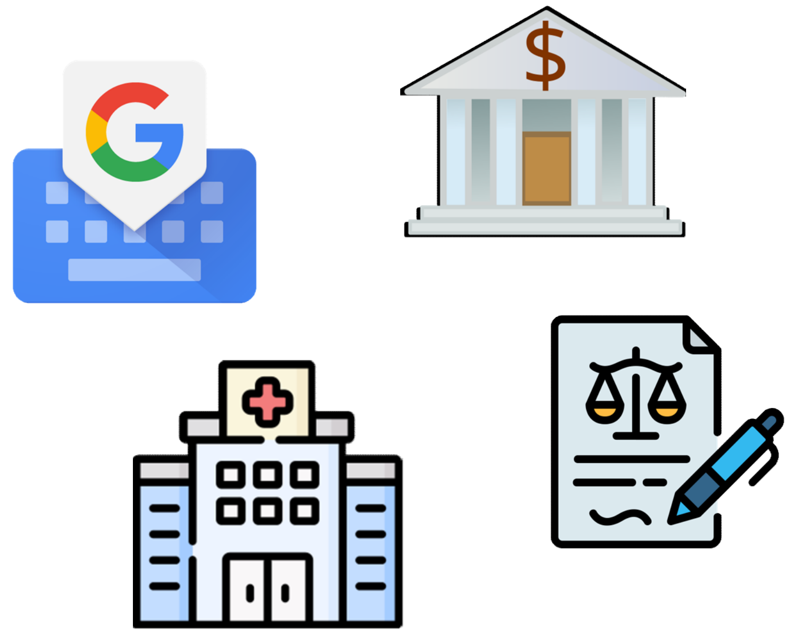
Zhu et al., "Deep leakage from gradients." *Advances in neural information processing systems* 32 (2019).

LLMs (and transformers) in Federated Learning

LLMs are becoming increasingly popular (and powerful!)

Uses of LLMs in FL include:

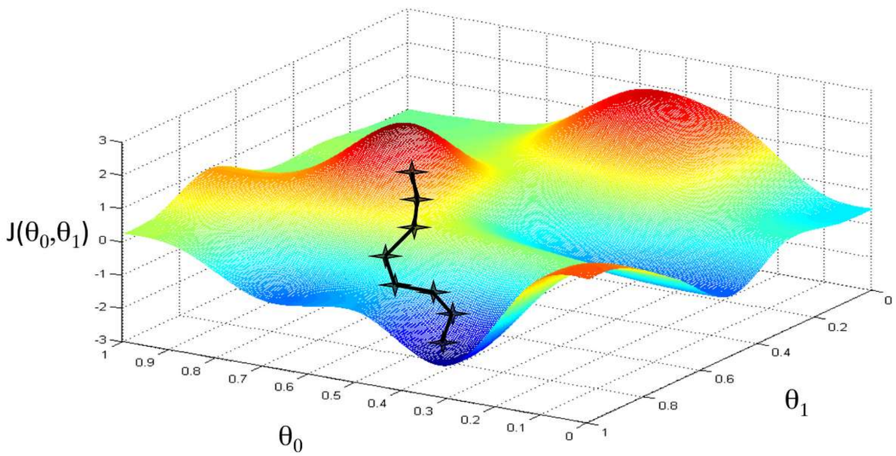
- Keyboard suggestions
- Fraud detection
- Healthcare diagnostics
- Legal Document Analysis



Jaleel Adejumo, Gradient Descent From Scratch- Batch Gradient Descent, *Medium*

Gradient Inversion – Prior Work

State-of-the-art attacks utilize a continuous optimization approach

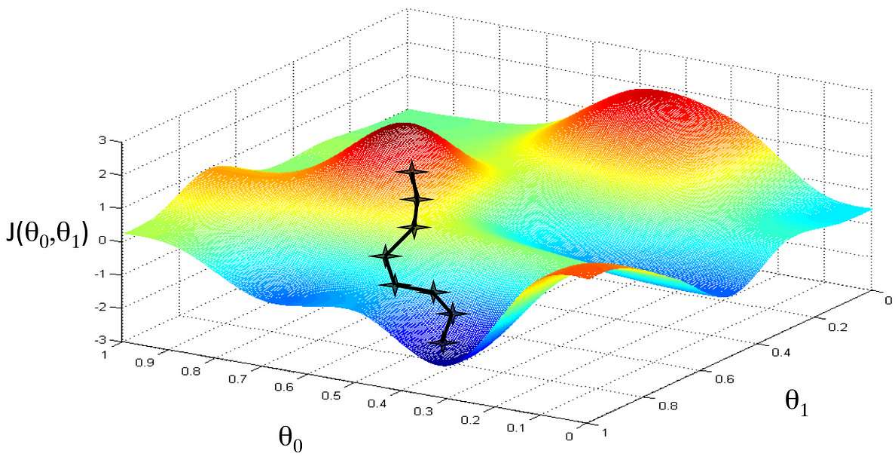


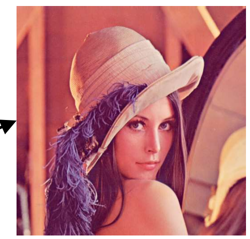
$$\mathbf{X}, y = \underset{\mathbf{X}, y}{\operatorname{argmin}} \, \| \frac{\partial \mathcal{L}_\theta(X, y)}{\partial \mathbf{W}} - \nabla \mathbf{W} \|^2$$

Jaleel Adejumo, Gradient Descent From Scratch- Batch Gradient Descent, *Medium*

Gradient Inversion – Prior Work

State-of-the-art attacks utilize a continuous optimization approach





✓

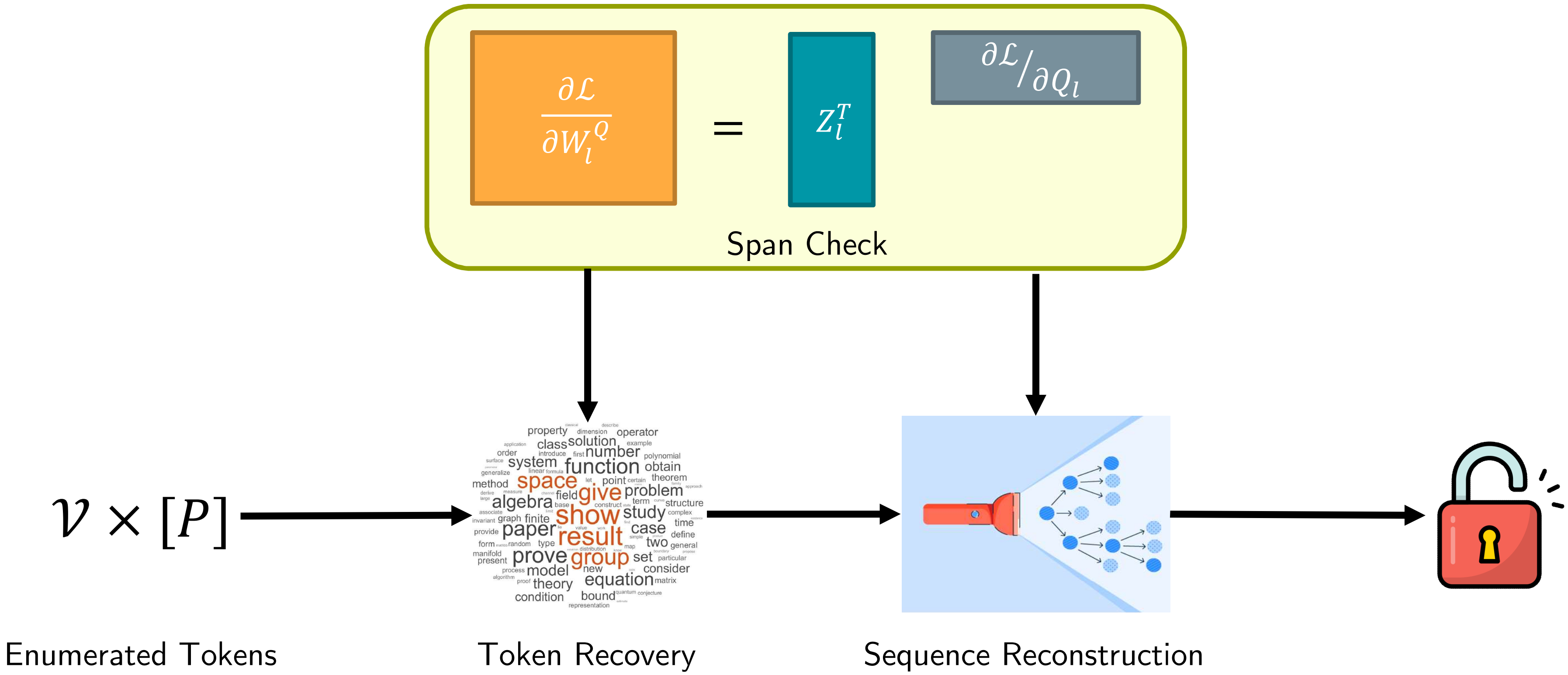
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec non lectus lobortis purus egestas rhoncus quis eu erat.

✗

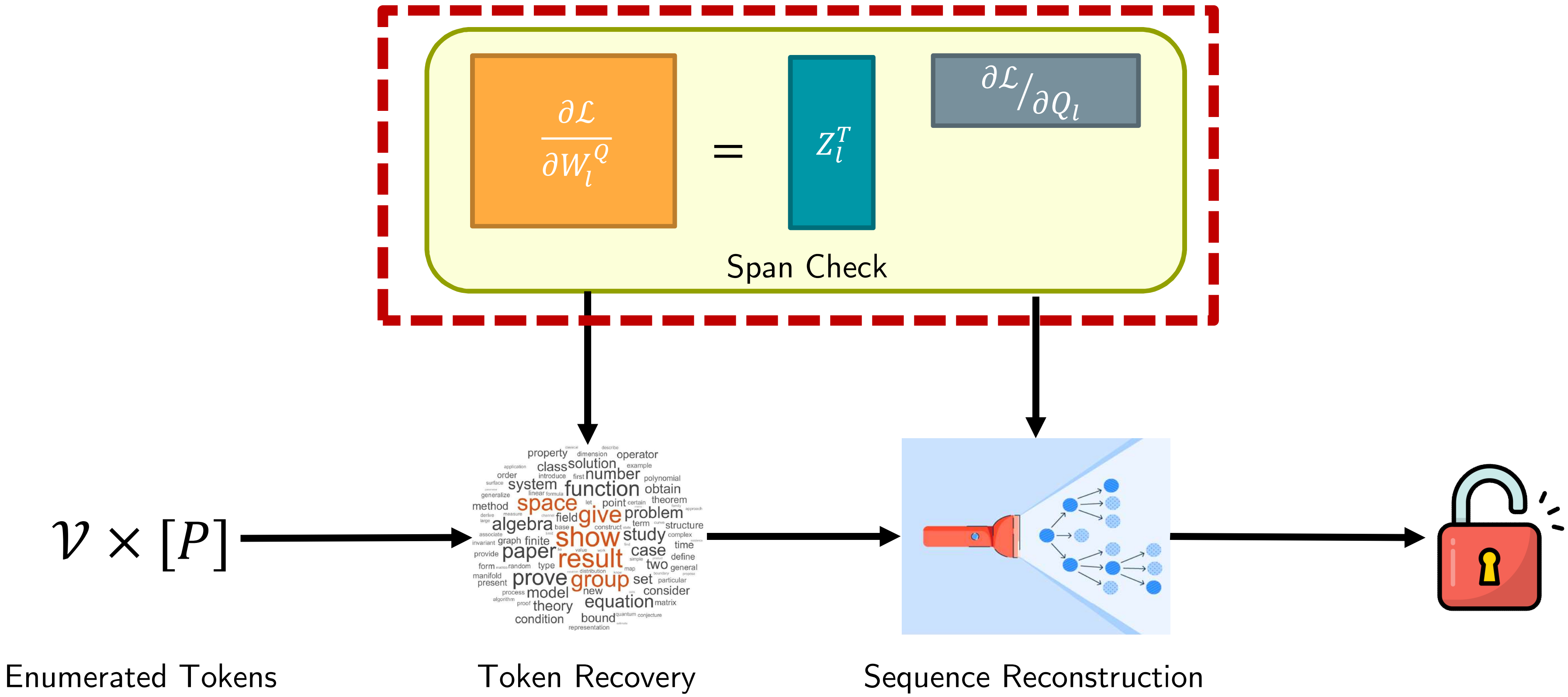
$$\mathbf{X}, y = \underset{\mathbf{X},y}{\operatorname{argmin}} \| \frac{\partial \mathcal{L}_\theta(X,y)}{\partial \mathbf{W}} - \nabla \mathbf{W} \|^2$$

Jaleel Adejumo, Gradient Descent From Scratch- Batch Gradient Descent, *Medium*

The DAGER Pipeline



The DAGER Pipeline

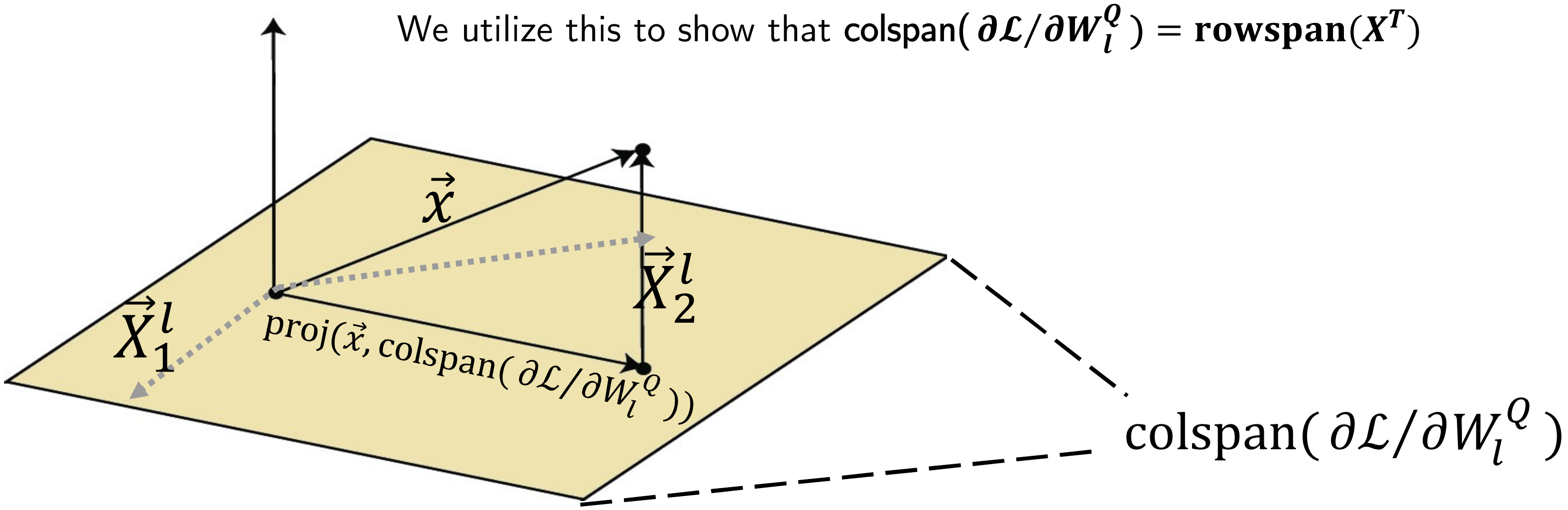


The Span Check Filter

Dimitrov et al. [1] showed that for a linear layer $Y = XW$ that:

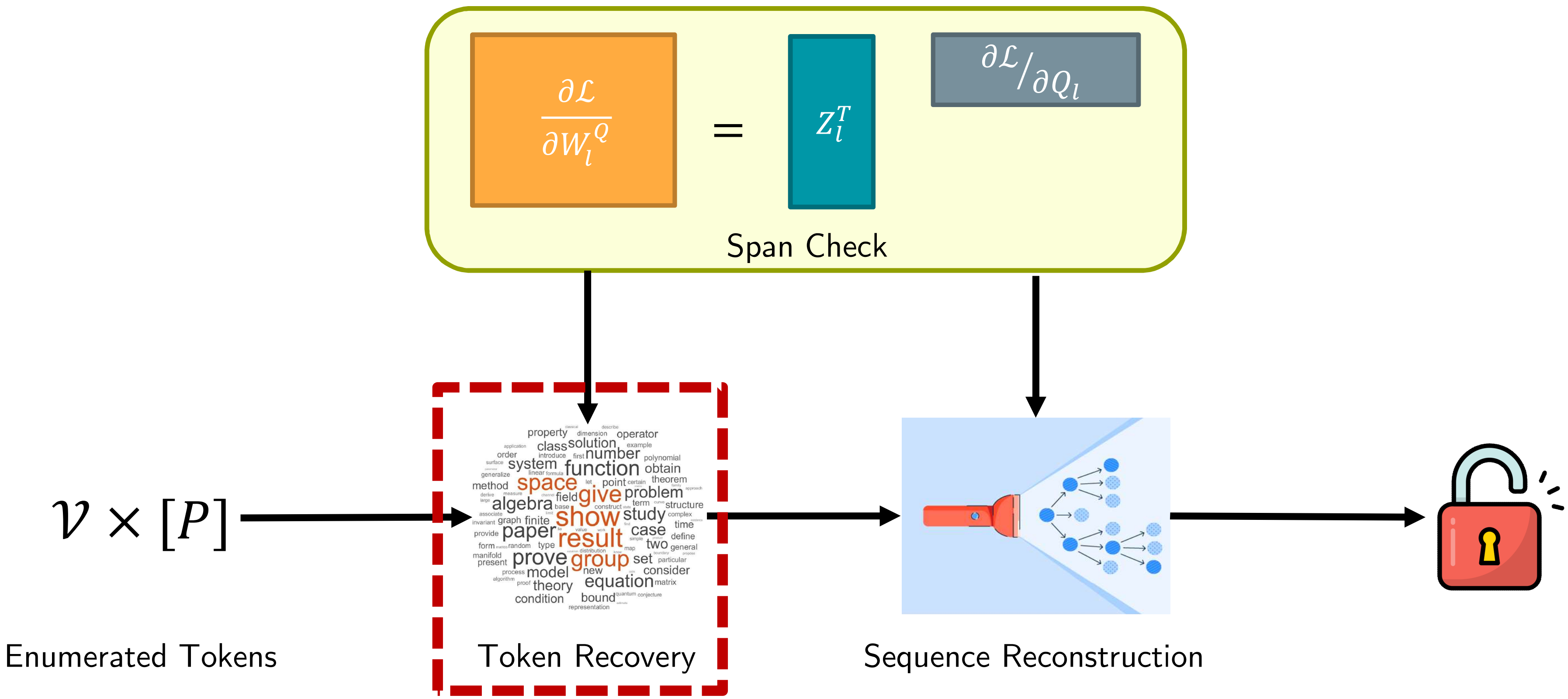
$$\frac{\partial \mathcal{L}}{\partial W} = X^T \frac{\partial \mathcal{L}}{\partial Y}$$

We utilize this to show that **colspan(** $\boldsymbol{\partial \mathcal{L} / \partial W_l^Q}$ **)** $=$ $\mathbf{rowspan}(\boldsymbol{X^T})$

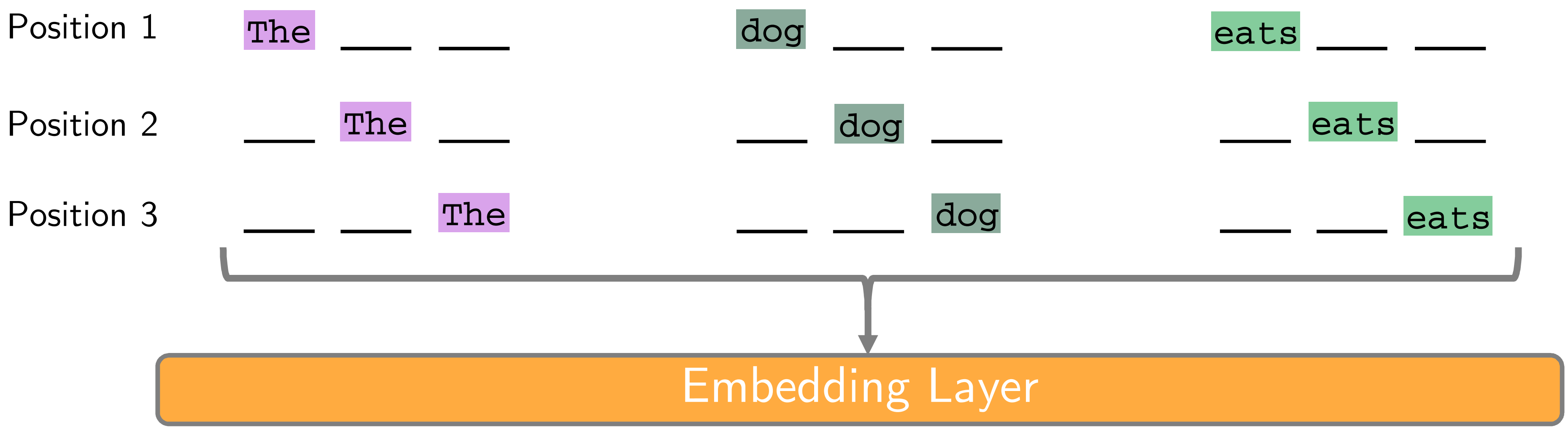


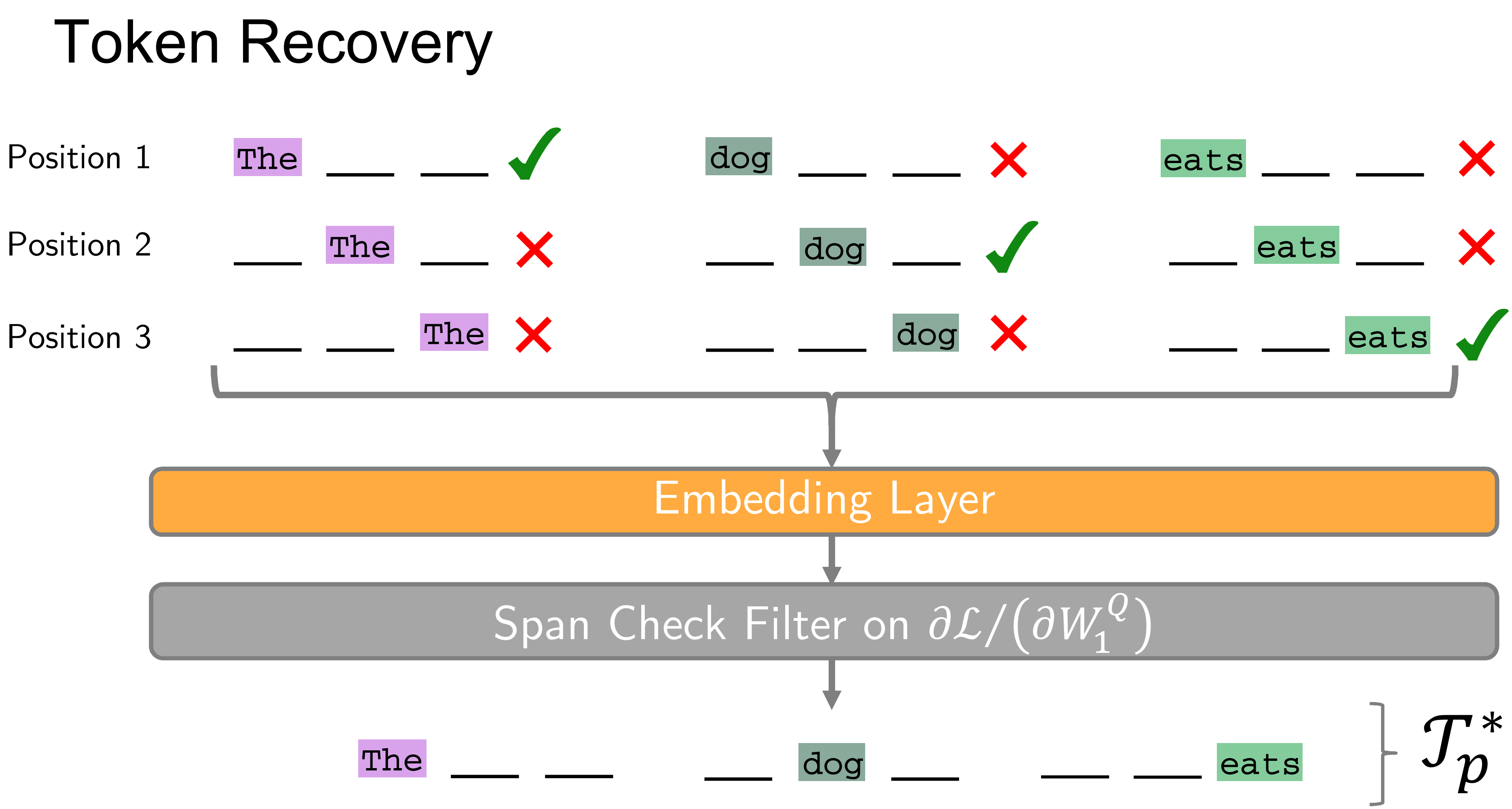
[1] Dimitrov, Dimitar I., et al. "Spear: Exact gradient inversion of batches in federated learning."

The DAGER Pipeline



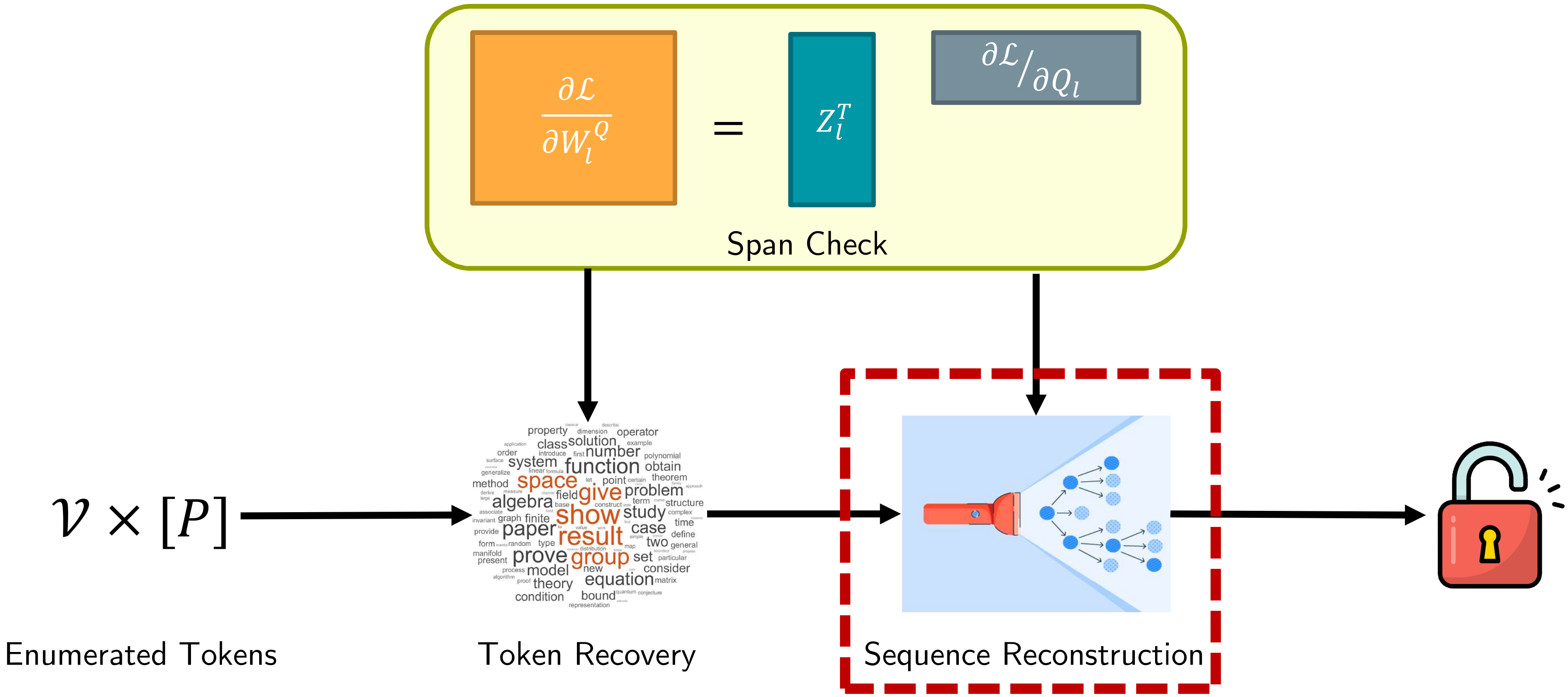
Token Recovery



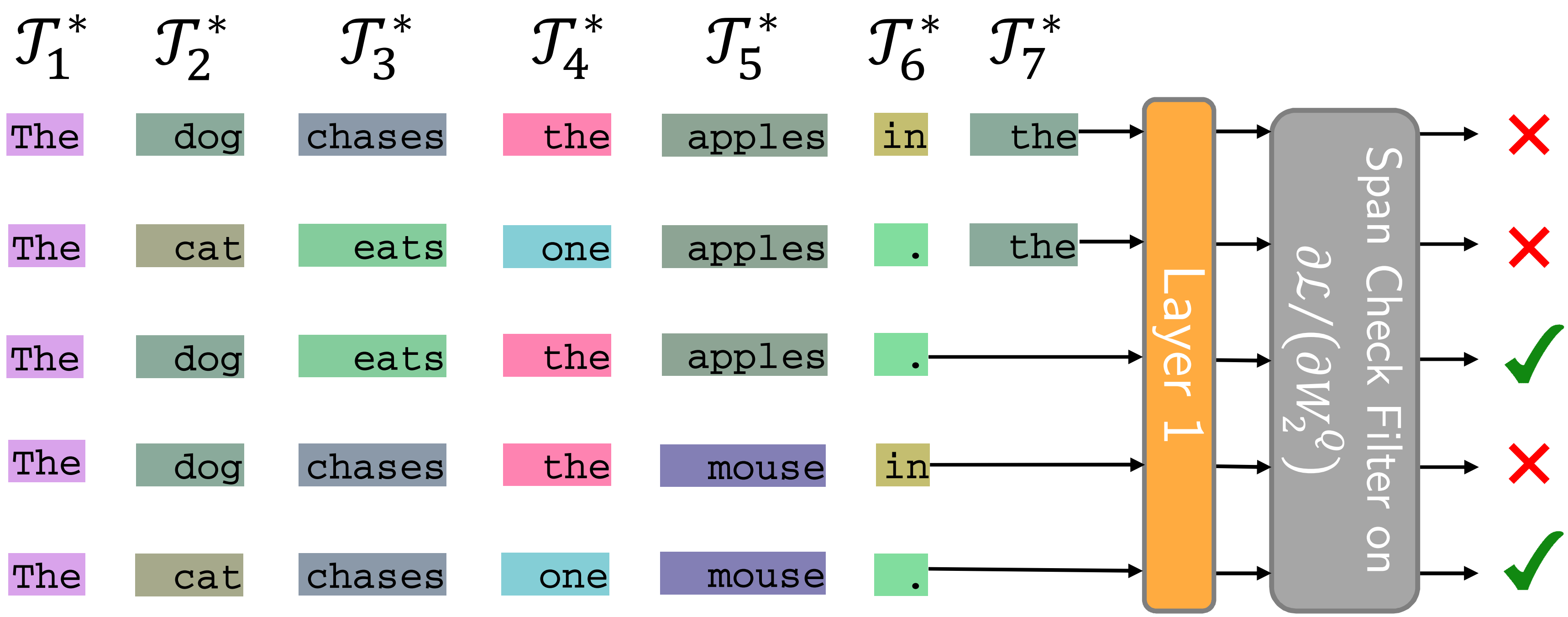


Token Recovery
Position 1
Position 2
Position 3
The
dog
eats
Embedding Layer
Span Check Filter on $\partial\mathcal{L}/(\partial W_1^Q)$
$\mathcal{T}_p^*$

The DAGER Pipeline

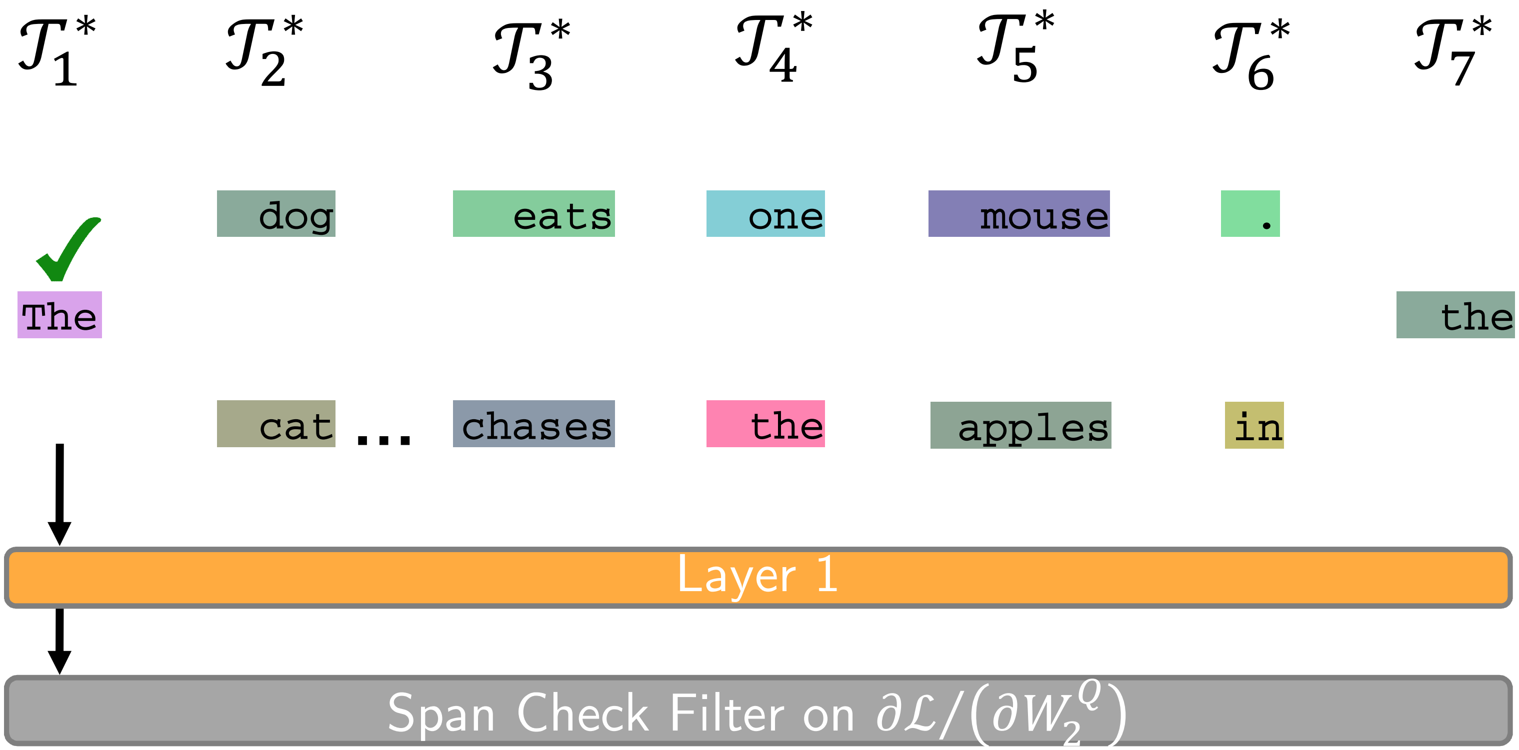


Sequence Reconstruction – Encoder-only models



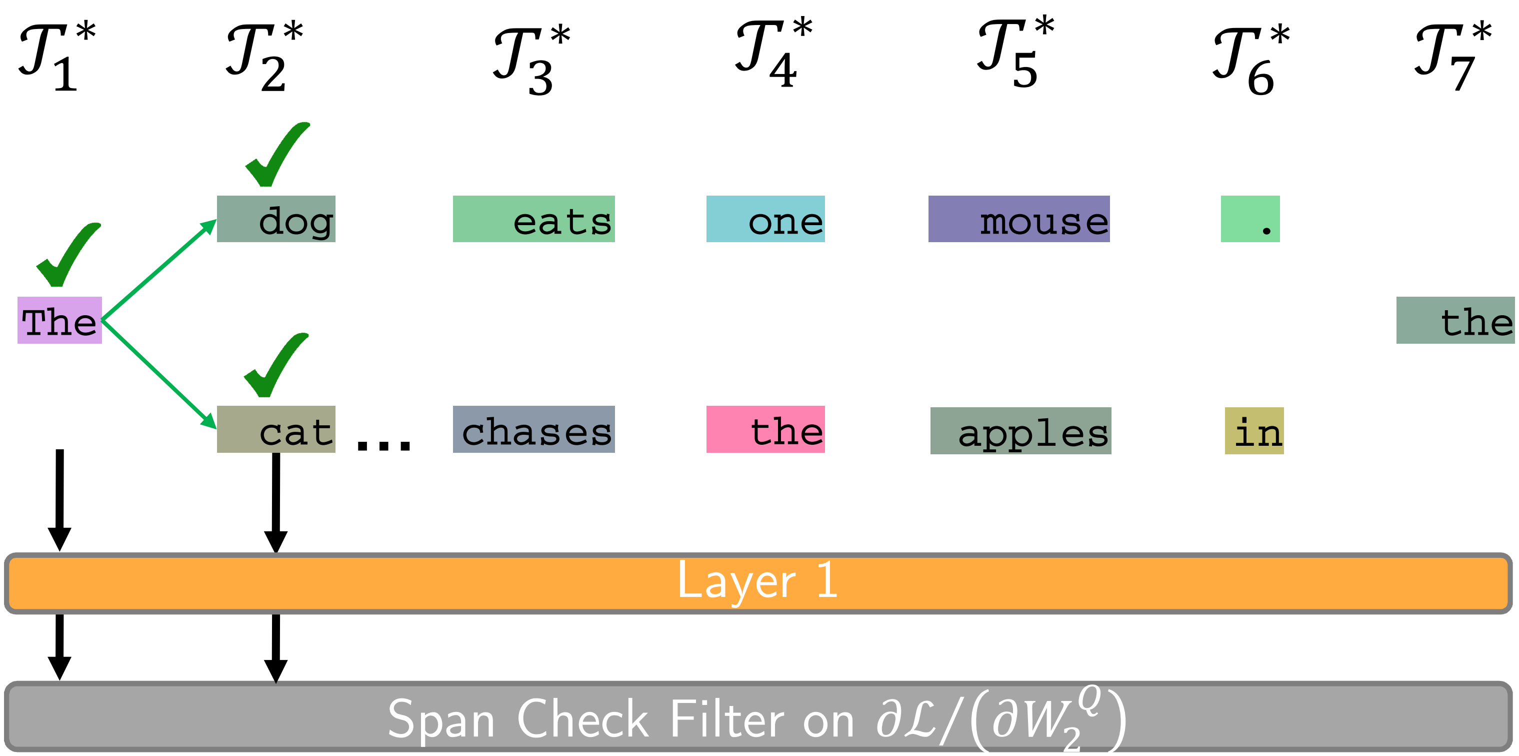
Exhausting all token combinations is possible for **encoder-only** models, but is computationally expensive.

Sequence Reconstruction – Decoder-only models

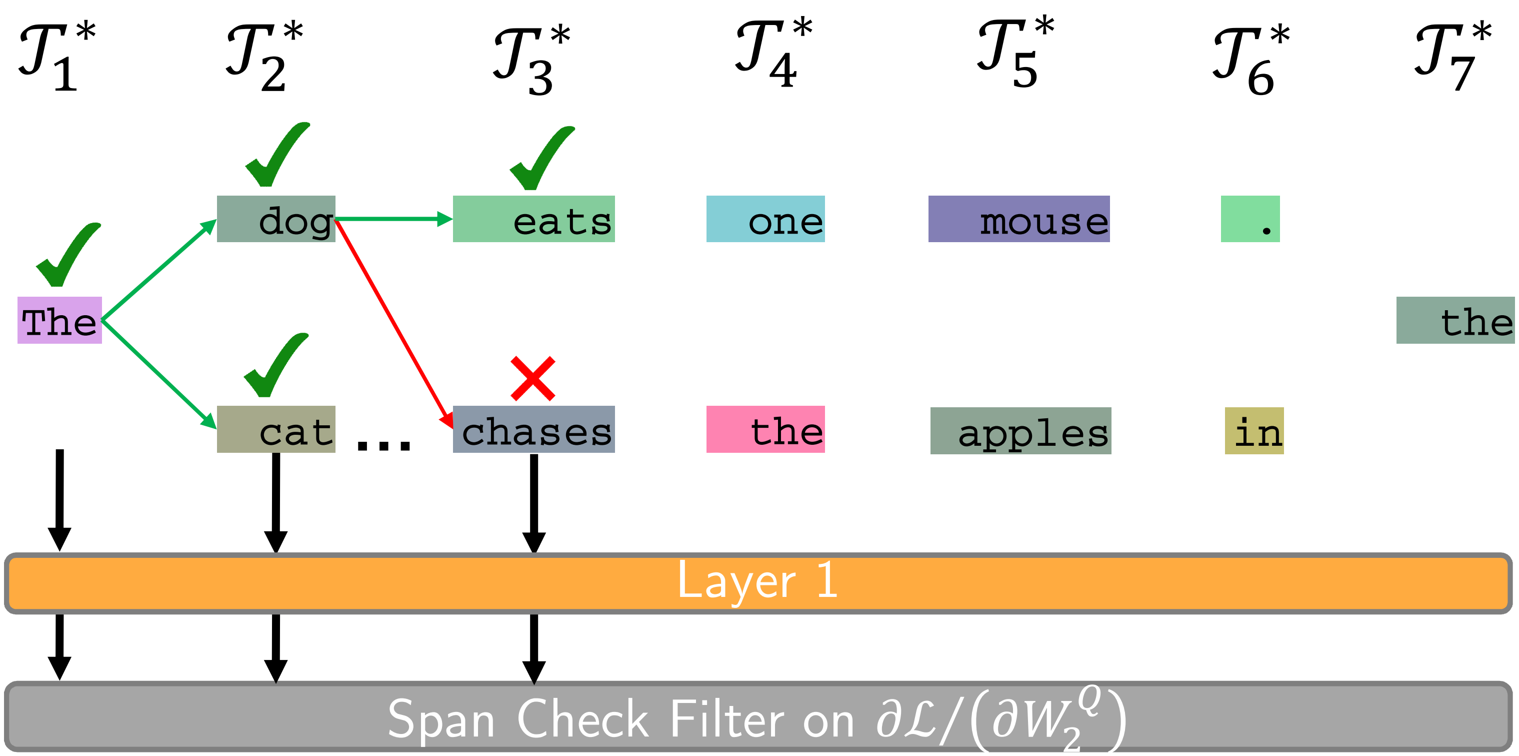


Key Observation: The outputs at position t only depend on inputs of tokens up to t-1

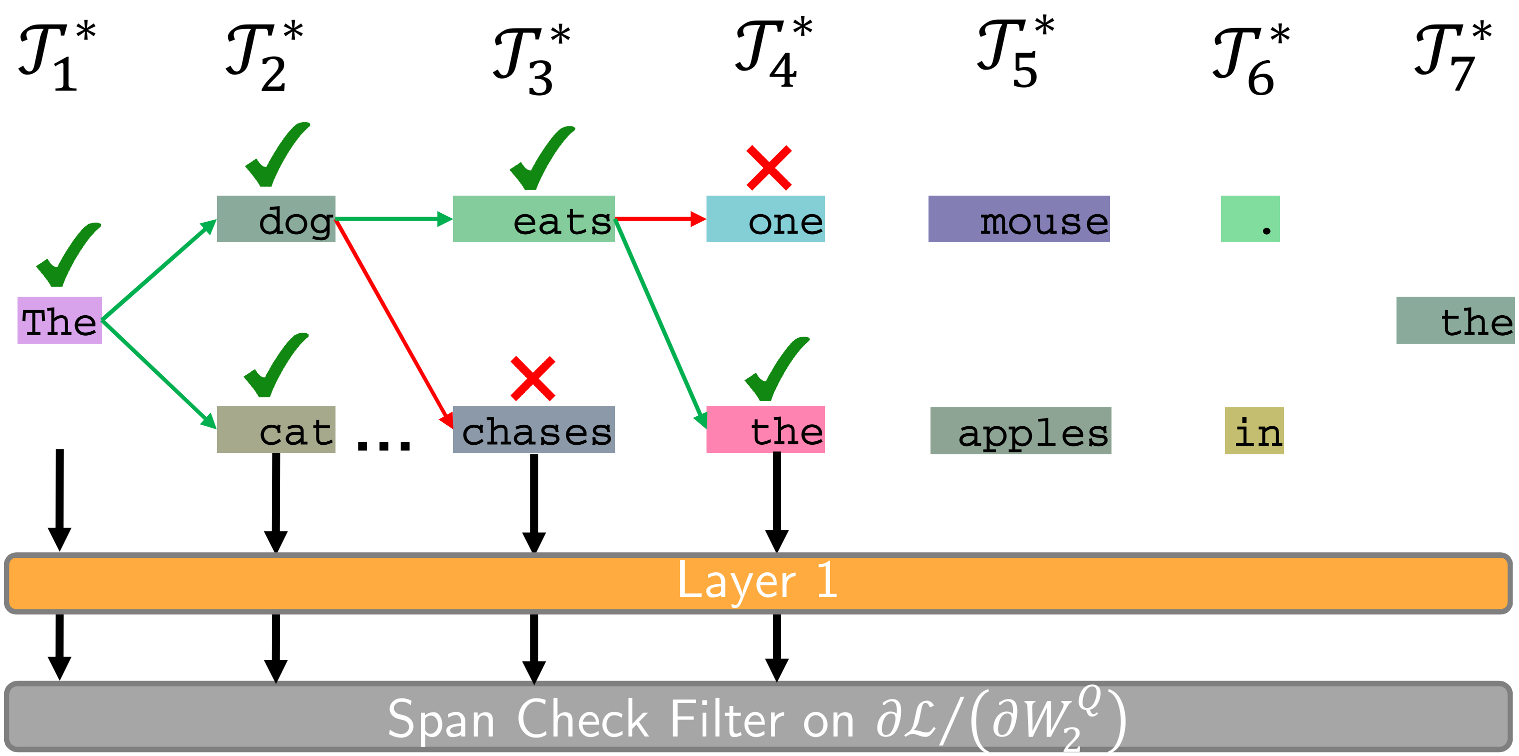
Sequence Reconstruction – Decoder-only models



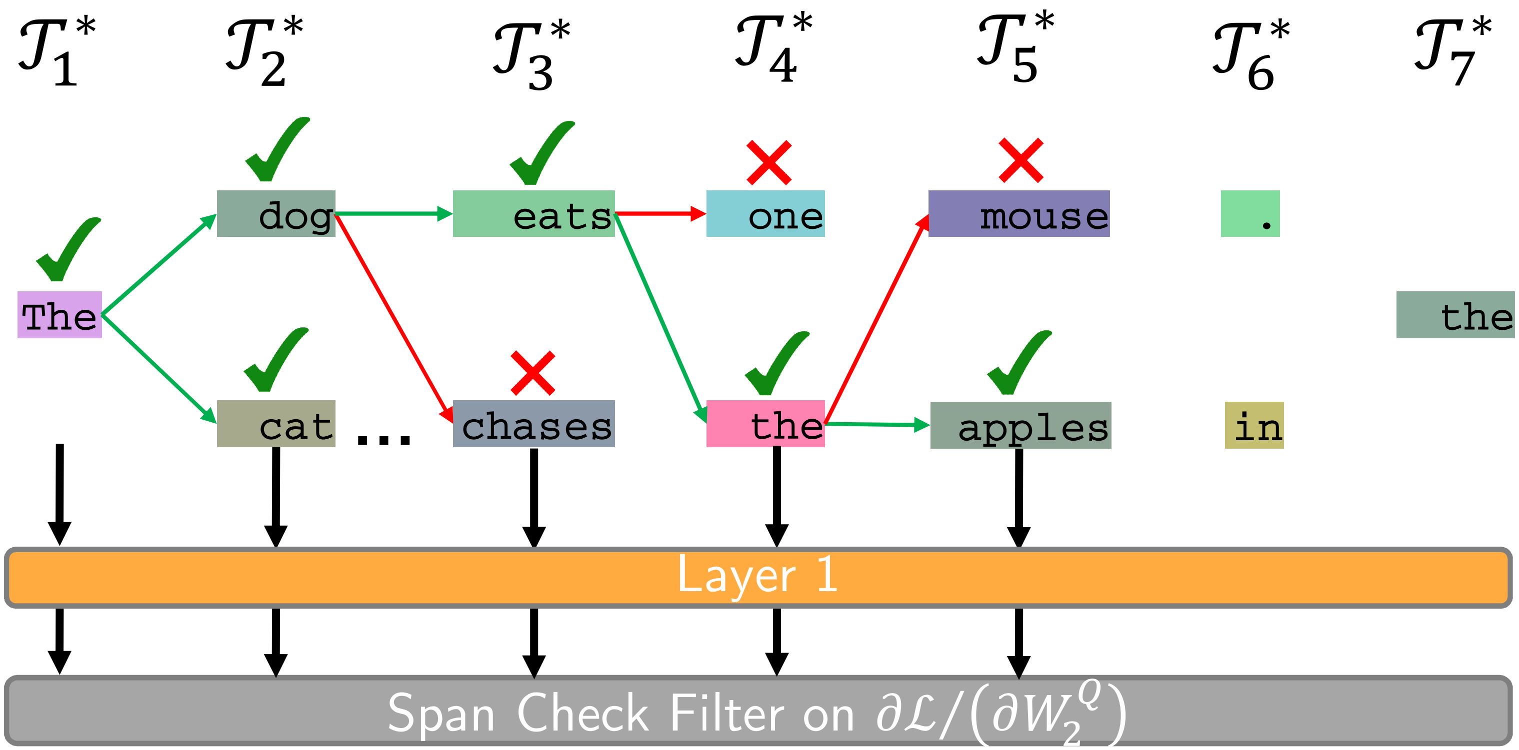
Sequence Reconstruction – Decoder-only models



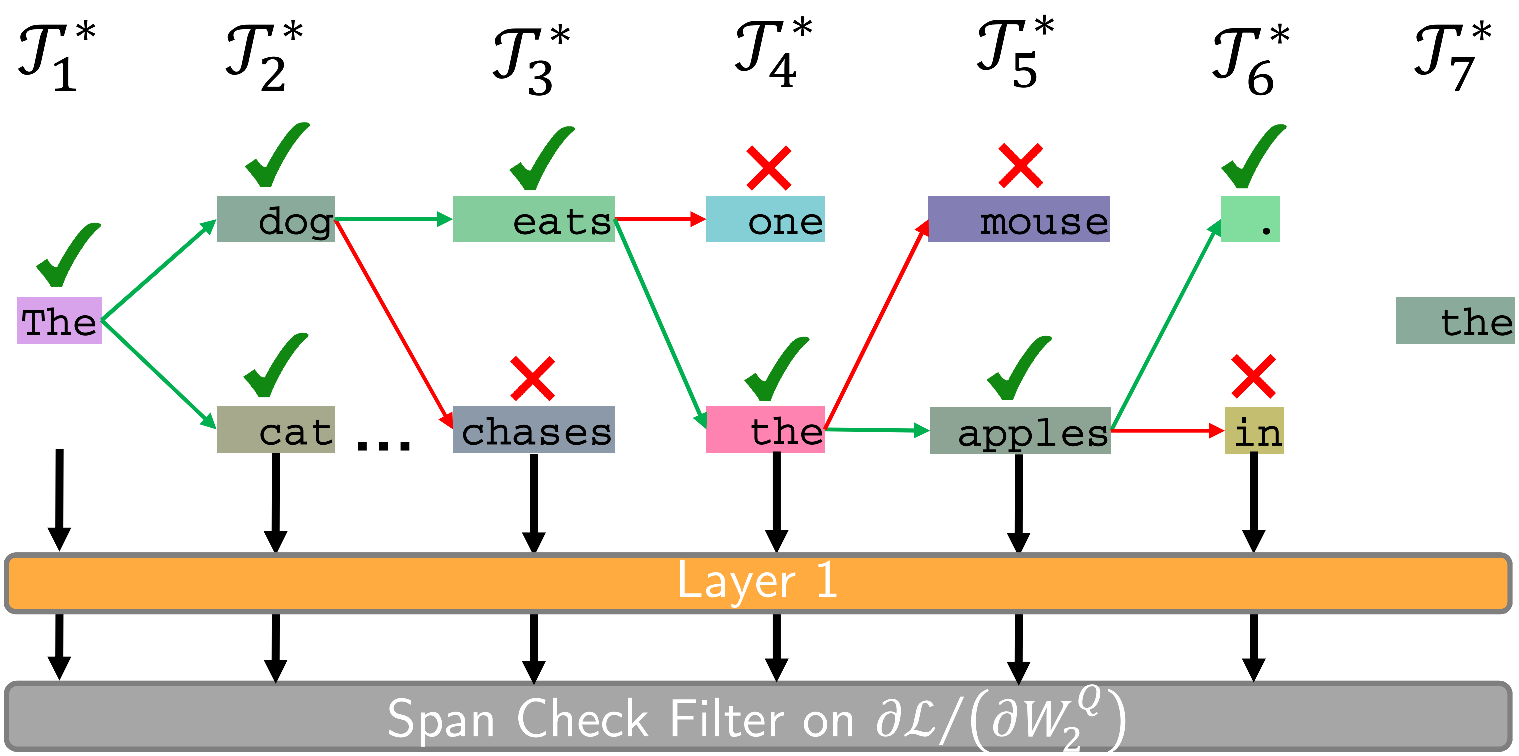
Sequence Reconstruction – Decoder-only models



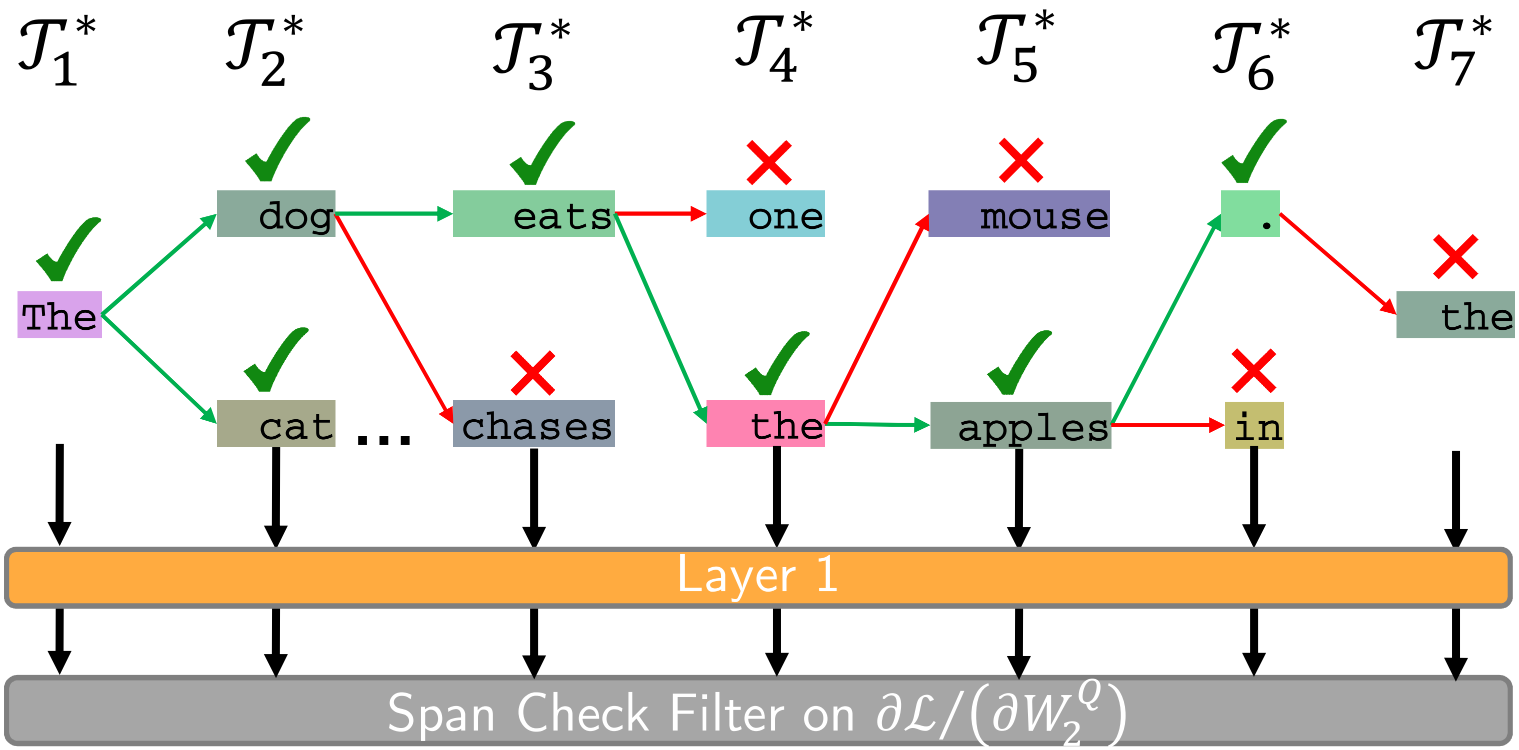
Sequence Reconstruction – Decoder-only models



Sequence Reconstruction – Decoder-only models

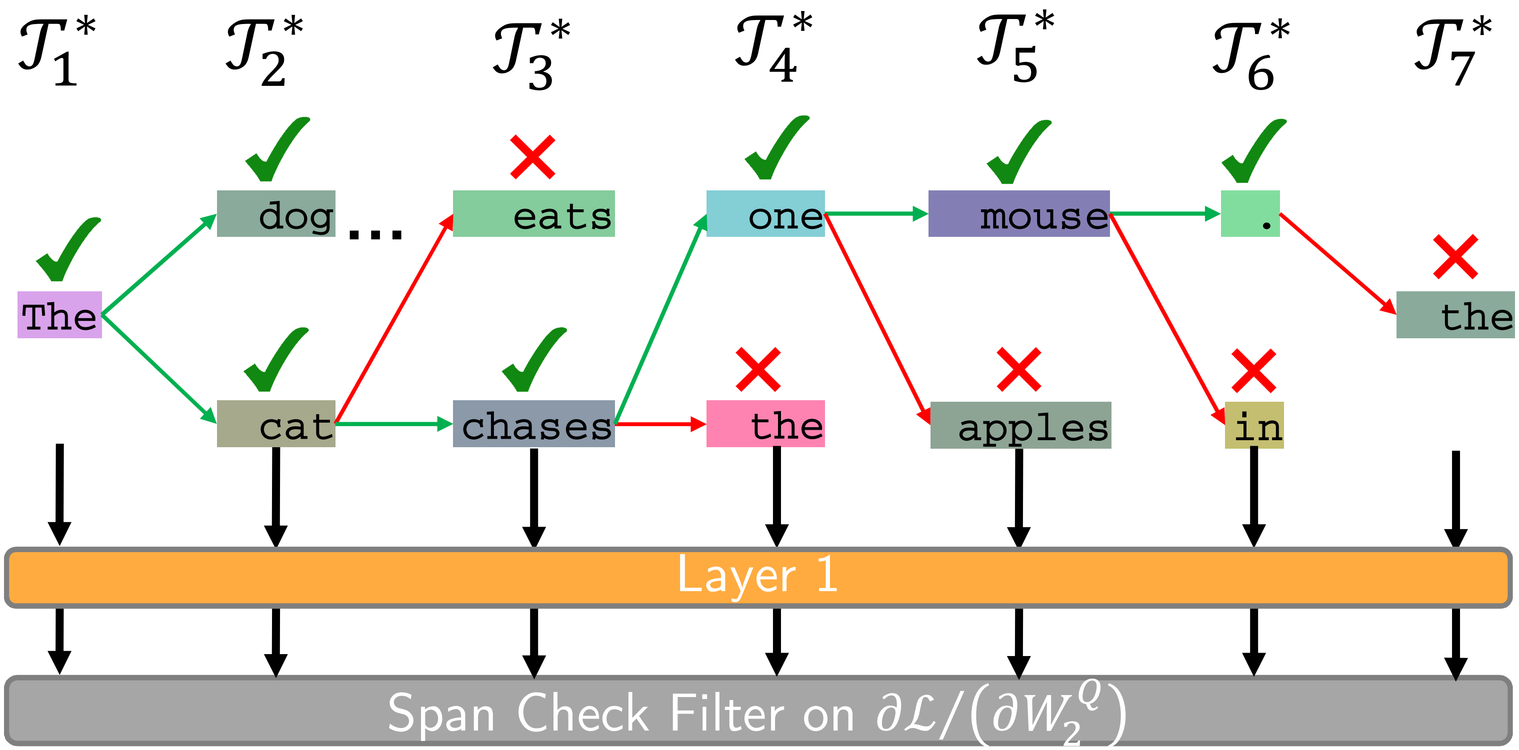


Sequence Reconstruction – Decoder-only models



We can iteratively reconstruct **entire sentences exactly**

Sequence Reconstruction – Decoder-only models



We can recover **entire batches** of text by following different beams.

Evaluation

Baseline Comparison – Encoder-only Models

			$B=1$		$B=2$		$B=4$		$B=8$	
			R-1	R-2	R-1	R-2	R-1	R-2	R-1	R-2
BERT	CoLA	TAG	78.9 ± 4.4	10.3 ± 3.0	68.9 ± 4.2	7.7 ± 1.7	56.3 ± 3.4	6.8 ± 1.4	45.9 ± 1.9	3.9 ± 0.6
		LAMP	89.6 ± 2.5	51.9 ± 6.7	77.8 ± 3.6	31.5 ± 4.6	66.2 ± 3.4	21.8 ± 1.7	52.9 ± 2.2	13.1 ± 1.9
		DAGER	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{94.0 \pm 2.0}$	$\mathbf{89.9 \pm 3.1}$	$\mathbf{67.8 \pm 2.3}$	$\mathbf{48.8 \pm 4.5}$
	SST-2	TAG	75.4 ± 4.3	19.0 ± 6.9	71.8 ± 3.6	16.0 ± 3.9	61.0 ± 3.4	12.3 ± 2.8	50.4 ± 2.4	9.2 ± 1.6
		LAMP	88.8 ± 3.0	56.8 ± 7.9	82.4 ± 3.6	45.7 ± 6.0	69.5 ± 3.6	32.5 ± 4.4	56.9 ± 2.6	19.1 ± 2.8
		DAGER	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{99.3^{+0.7}_{-2.0}}$	$\mathbf{99.0^{+0.8}_{-2.1}}$	$\mathbf{95.6 \pm 2.2}$	$\mathbf{93.0 \pm 3.3}$	$\mathbf{74.1 \pm 3.3}$	$\mathbf{59.8 \pm 2.9}$
	Rotten Tomatoes	TAG	60.1 ± 4.4	3.3 ± 1.2	49.2 ± 3.5	3.0 ± 0.9	33.7 ± 2.5	1.6 ± 0.7	25.4 ± 1.2	0.9 ± 0.4
		LAMP	64.7 ± 4.4	16.5 ± 3.9	46.4 ± 3.7	7.6 ± 2.0	35.1 ± 2.7	4.2 ± 1.3	27.3 ± 1.4	2.0 ± 0.6
		DAGER	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{98.1 \pm 1.2}$	$\mathbf{96.5 \pm 1.8}$	$\mathbf{66.8 \pm 3.2}$	$\mathbf{50.1 \pm 4.4}$	$\mathbf{37.1 \pm 1.2}$	$\mathbf{11.4 \pm 1.3}$

*we observe non-perfect R-2 scores on the SST-2 dataset due to an artifact of our metric library that assigns a R-2 score of 0 to single-word sequences.

Baseline Comparison – Decoder-only Models

			$B=1$		$B=2$		$B=4$		$B=8$	
			R-1	R-2	R-1	R-2	R-1	R-2	R-1	R-2
GPT-2	CoLA	TAG	7.0 ± 2.5	0.54 ± 0.54	8.0 ± 2.0	1.4 ± 1.3	7.8 ± 1.2	0.8 ± 0.5	5.3 ± 0.7	0.4 ± 0.2
		LAMP	73.3 ± 4.5	43.3 ± 7.0	26.8 ± 2.8	11.0 ± 3.0	13.4 ± 1.4	3.9 ± 1.2	8.9 ± 1.2	1.9 ± 0.6
		DAGER	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$
	SST-2	TAG	5.3 ± 0.5	0.0 ± 0.0	6.0 ± 1.7	0.5 ± 0.4	6.1 ± 1.2	0.6 ± 0.6	4.4 ± 0.6	$0.2^{+0.6}_{-0.1}$
		LAMP	62.2 ± 6.9	31.8 ± 8.4	21.4 ± 3.1	9.2 ± 3.1	9.8 ± 2.0	2.7 ± 1.3	8.1 ± 1.1	0.7 ± 0.4
		DAGER	$\mathbf{100.0 \pm 0.0}$	$\mathbf{86.0 \pm 7.0}$*	$\mathbf{100.0 \pm 0.0}$	$\mathbf{89.5 \pm 4.1}$*	$\mathbf{100.0 \pm 0.0}$	$\mathbf{92.8 \pm 2.4}$*	$\mathbf{100.0 \pm 0.0}$	$\mathbf{92.9 \pm 1.6}$*
	Rotten Tomatoes	TAG	7.1 ± 1.8	$0.1^{+0.4}_{-0.1}$	7.0 ± 1.2	$0.1^{+0.2}_{-0.1}$	6.2 ± 0.8	$0.1^{+0.2}_{-0.1}$	6.1 ± 0.5	0.1 ± 0.1
		LAMP	31.4 ± 4.4	9.3 ± 3.6	11.2 ± 1.2	0.9 ± 0.42	6.3 ± 1.1	0.9 ± 0.6	6.8 ± 0.7	$0.3^{+0.2}_{-0.1}$
		DAGER	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{99.3^{+0.7}_{-1.7}}$	$\mathbf{99.3^{+0.7}_{-1.8}}$	$\mathbf{100.0^{+0.0}_{-0.1}}$	$\mathbf{99.9^{+0.1}_{-0.6}}$

		$B=16$		$B=32$		$B=64$		$B=128$	
		R-1	R-2	R-1	R-2	R-1	R-2	R-1	R-2
CoLA	GPT-2	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	30.3 ± 1.0	14.6 ± 0.9
	LLaMa-2 (7B)	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$99.9^{+0.0}_{-0.1}$	$99.9^{+0.0}_{-0.1}$	$\mathbf{99.5 \pm 0.2}$	$\mathbf{99.3 \pm 0.3}$
SST-2	GPT-2	$\mathbf{100.0 \pm 0.0}$	94.6 ± 1.1	$\mathbf{100.0^{+0.0}_{-0.1}}$	93.4 ± 1.0	92.9 ± 3.0	85.0 ± 3.5	13.7 ± 1.4	4.3 ± 0.5
	LLaMa-2 (7B)	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$99.9^{+0.0}_{-0.1}$	$\mathbf{99.9 \pm 0.1}$	$\mathbf{99.9 \pm 0.1}$	$\mathbf{99.9 \pm 0.1}$	$\mathbf{98.2 \pm 0.4}$	$\mathbf{97.8 \pm 0.4}$
Rotten Tomatoes	GPT-2	$\mathbf{100.0 \pm 0.0}$	$99.9^{+0.1}_{-0.3}$	98.0 ± 1.7	97.8 ± 1.8	2.8 ± 1.1	1.1 ± 0.4	0.0 ± 0.0	0.0 ± 0.0
	LLaMa-2 (7B)	$\mathbf{100.0^{+0.0}_{-0.1}}$	$\mathbf{100.0^{+0.0}_{-0.1}}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{97.9 \pm 0.5}$	$\mathbf{97.8 \pm 0.5}$	$\mathbf{99.7^{+0.1}_{-0.2}}$	$\mathbf{99.7^{+0.2}_{-0.3}}$

*we observe non-perfect R-2 scores on the SST-2 dataset due to an artifact of our metric library that assigns a R-2 score of 0 to single-word sequences.

DAGER under different settings

	$B=16$		$B=32$	
	R-1	R-2	R-1	R-2
GPT-2$_{\text{BASE}}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{99.9^{+0.1}_{-0.3}}$	98.0 ± 1.7	97.8 ± 1.8
GPT-2$_{\text{FineTuned}}$	$\mathbf{100.0 \pm 0.0}$	$99.8^{+0.1}_{-0.3}$	96.4 ± 2.3	96.0 ± 2.5
GPT-2$_{\text{NextToken}}$	99.9 ± 0.0	$99.7^{+0.2}_{-0.3}$	$99.6^{+0.3}_{-0.9}$	$99.4^{+0.3}_{-0.9}$
GPT-2$_{\text{LARGE}}$	$\mathbf{100.0 \pm 0.0}$	$99.8^{+0.1}_{-0.3}$	$\mathbf{100.0 \pm 0.0}$	$\mathbf{99.9^{+0.1}_{-0.2}}$

	LLaMa-3 70B ($B=1$)	**LoRA** ($r=256$)
R-1	$99.9^{+0.1}_{-0.2}$	94.8
R-2	$99.9^{+0.1}_{-0.2}$s	94.2 ± 0.7

DAGER under different settings

E	R-1	R-2
2	98.4 ± 0.9	98.0 ± 1.0
5	97.3 ± 1.2	96.8 ± 1.3
10	95.4 ± 1.6	94.7 ± 1.7
20	96.0 ± 1.4	95.3 ± 1.6

η	R-1	R-2
10^{-5}	$100.0^{+0.0}_{-0.2}$	$99.8^{+0.2}_{-0.4}$
5×10^{-5}	$99.8^{+0.2}_{-0.5}$	$99.6^{+0.3}_{-0.7}$
10^{-4}	95.4 ± 1.6	94.7 ± 1.7
5×10^{-4}	84.2 ± 1.8	82.2 ± 1.9

B_{mini}	R-1	R-2
2	93.2 ± 1.7	92.3 ± 1.9
4	95.4 ± 1.6	94.7 ± 1.7
8	$98.6^{+0.5}_{-0.9}$	$98.2^{+0.7}_{-1.0}$
16	100.0 ± 0.0	$99.8^{+0.2}_{-0.3}$

Further details can be found in the paper.

NeurIPS 2024 page:

OpenReview: