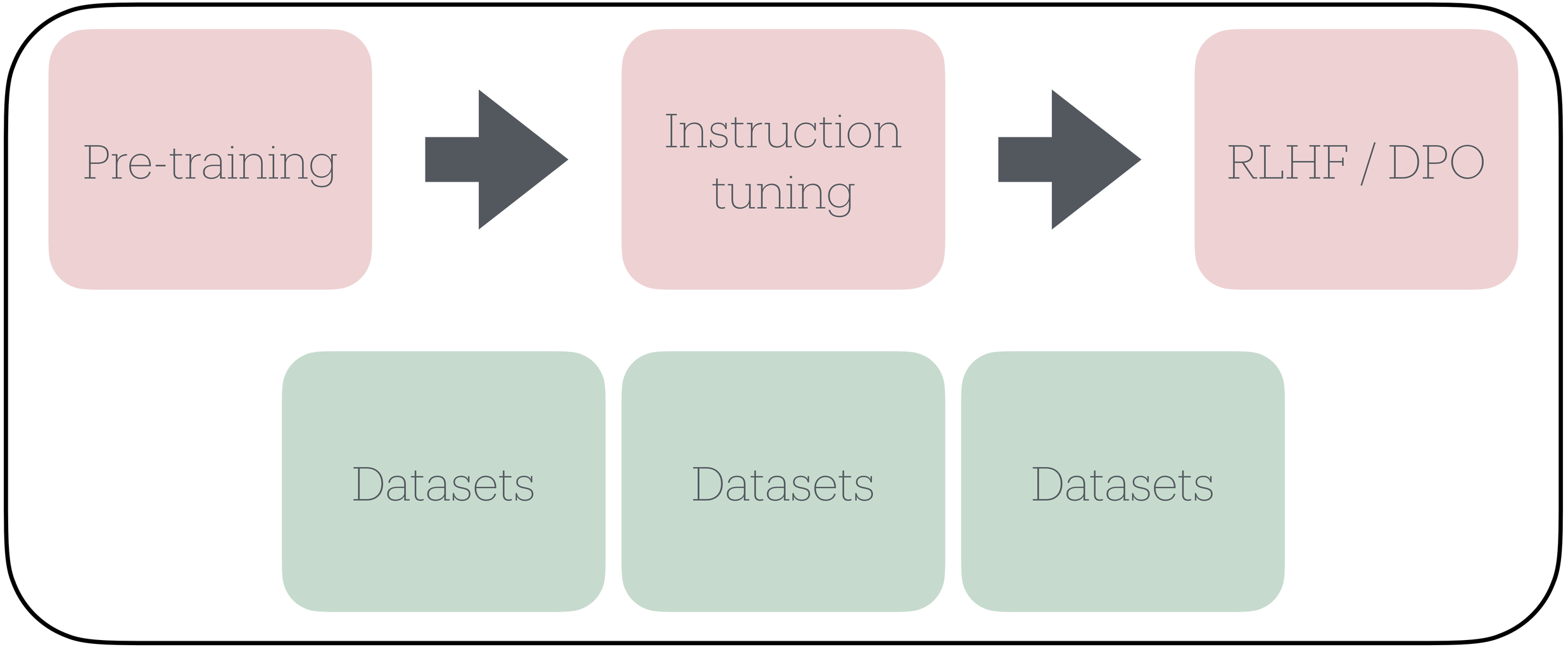
Efficient Training and Inference

Philipp Krähenbühl, UT Austin

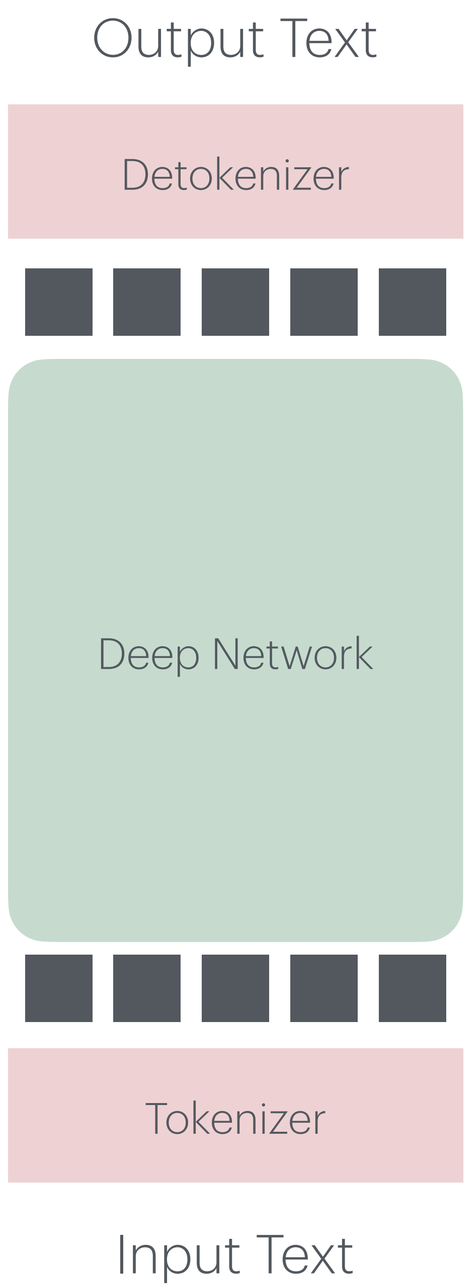
Full Picture

Basic LLM



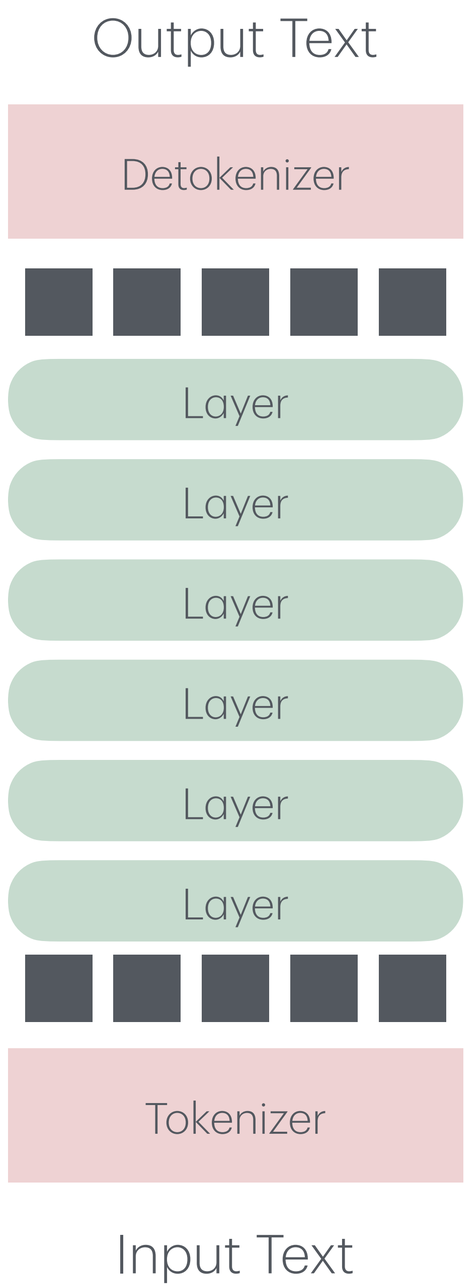
Training / Inference

- Step 1: Tokenize
- Training:
  - Step 2: Forward / Backward
  - Step 3: Step
- Inference:
  - Step 2: $N \times$ Forward
  - Step 3: Detokenize



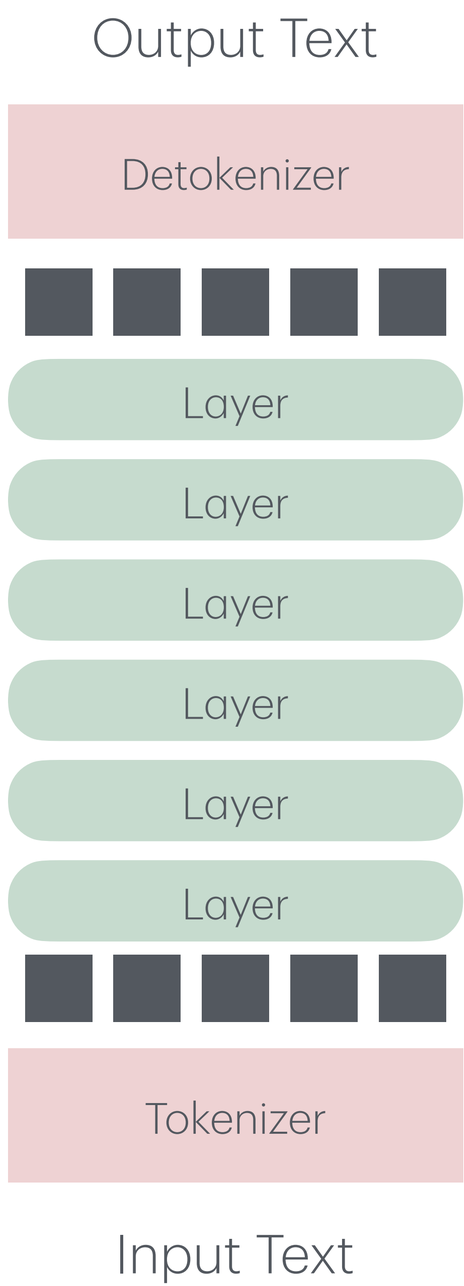
Training / Inference

- Step 1: Tokenize
- Training:
  - Step 2: Forward / Backward
  - Step 3: Step
- Inference:
  - Step 2: $N \times$ Forward
  - Step 3: Detokenize



Training / Inference

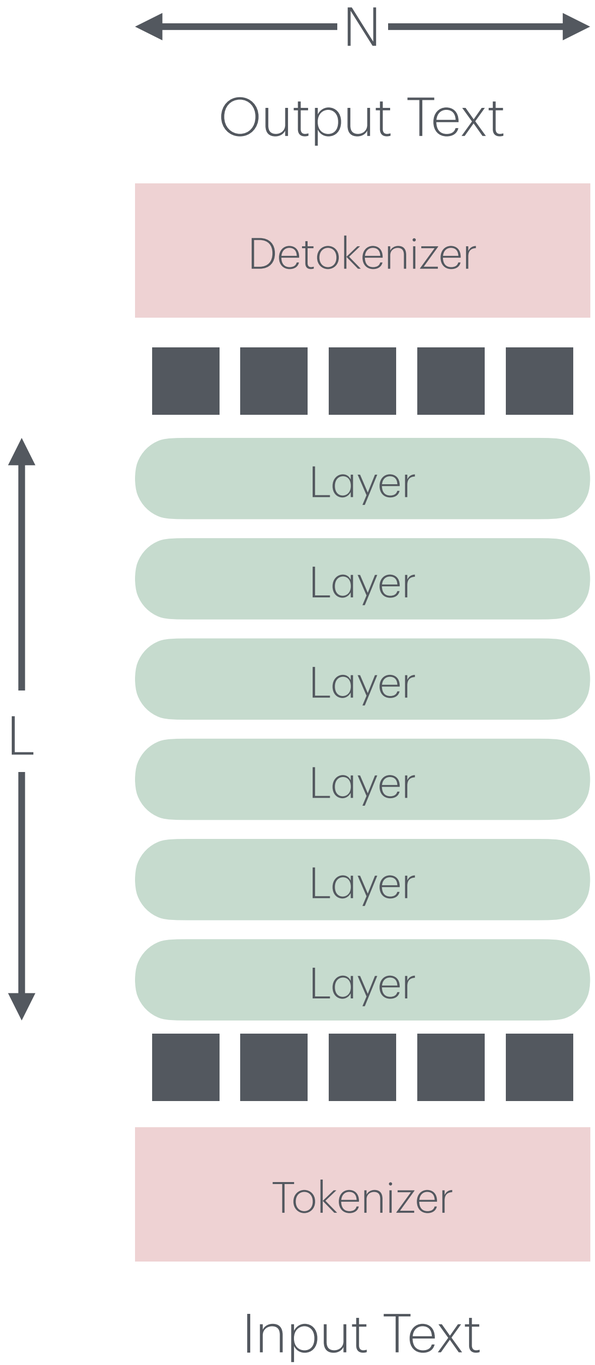
- Step 1: Tokenize
- Training:
  - Step 2: Forward / Backward
  - Step 3: Step
- Inference:
  - Step 2: $N \times$ Forward
  - Step 3: Detokenize



Training - An analysis

Vanilla Training

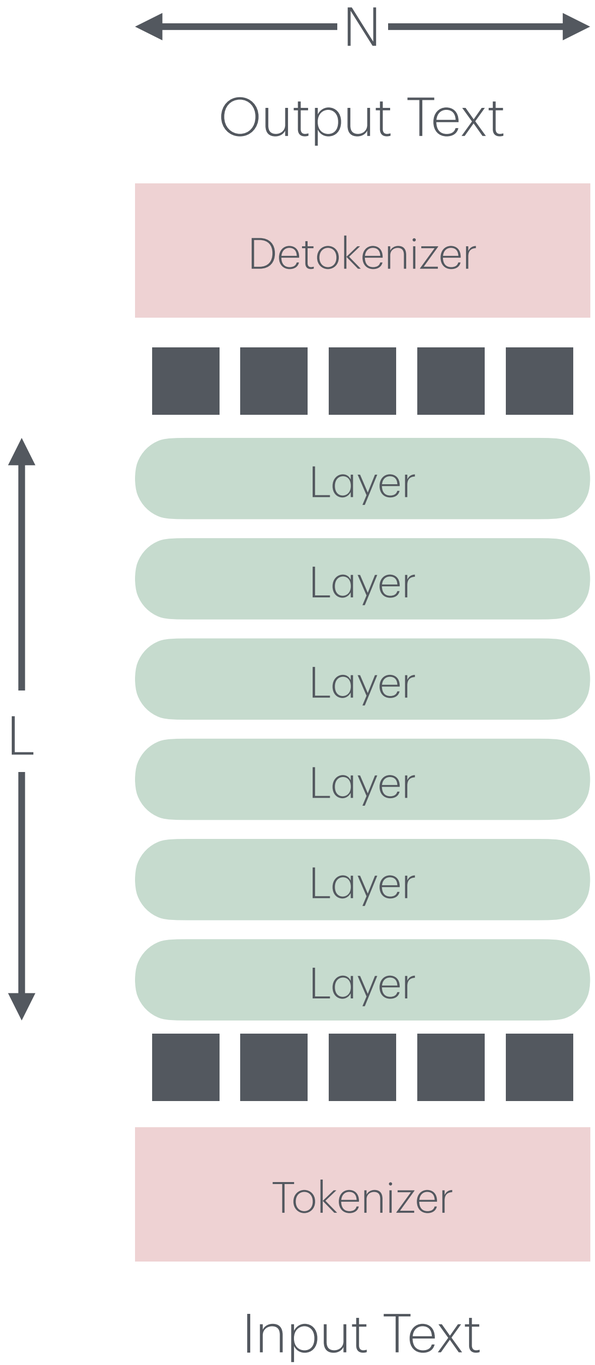
Peak Memory	
Runtime	



Training - An analysis

Vanilla Training

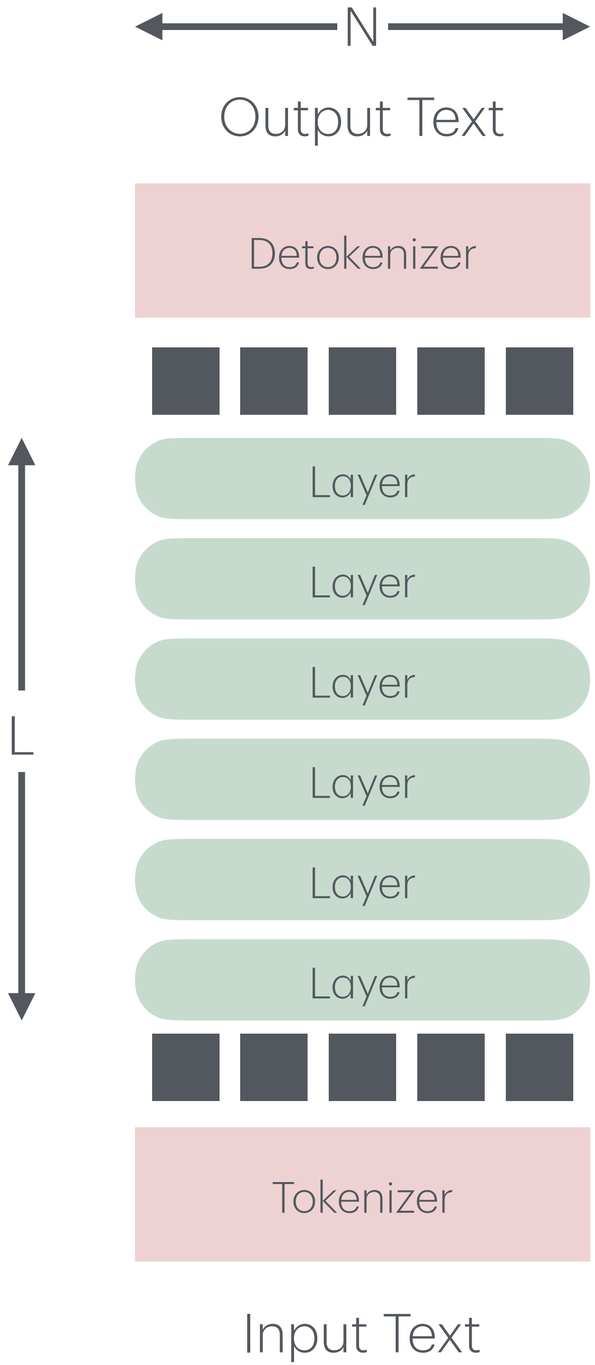
Peak Memory	O(NL)
Runtime	$O(N^2L)$



Training - An analysis

Activation checkpointing

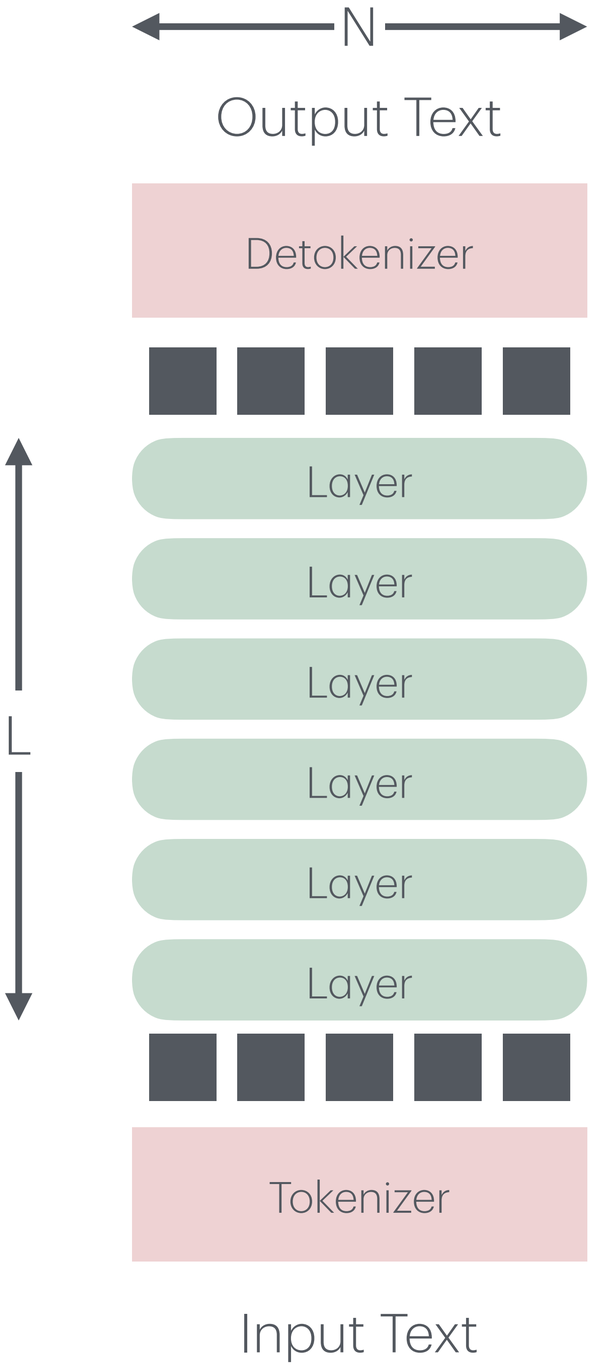
Peak Memory	
Runtime	
# forward calls	



Training - An analysis

Activation checkpointing

Peak Memory	$O(NL^{½})$
Runtime	$O(2\ N^2L)$

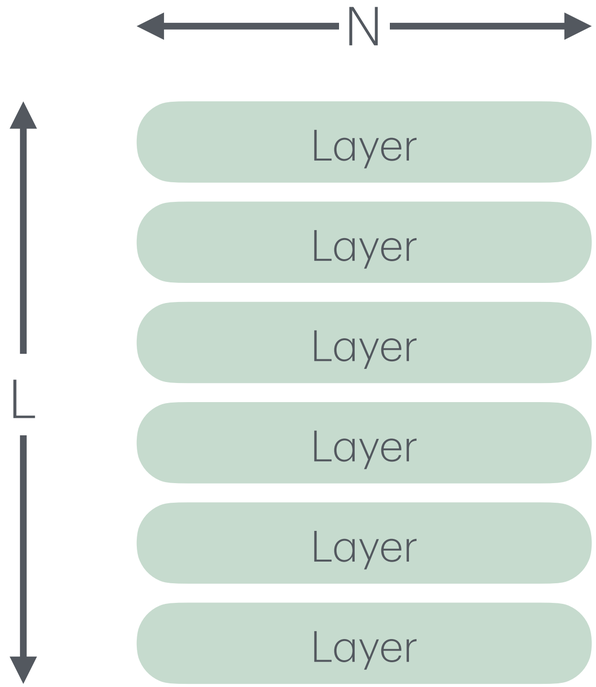


Training - An analysis

Activation checkpointing

Peak Memory	$O(NL^{½})$
Runtime	$O(2\ N^2L)$

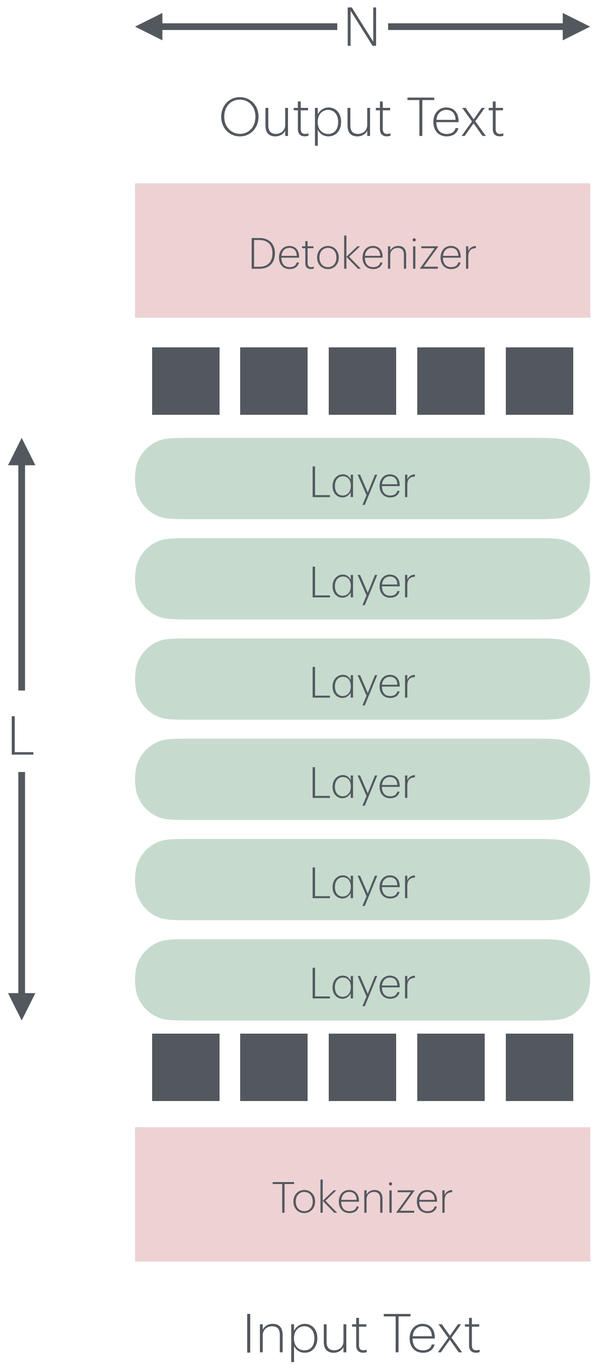
- What is N is large?
  - One GPU no longer has enough memory for activation checkpoints
  - Solution: Sequence parallelism (next)



Generation - An analysis

Vanilla Generation

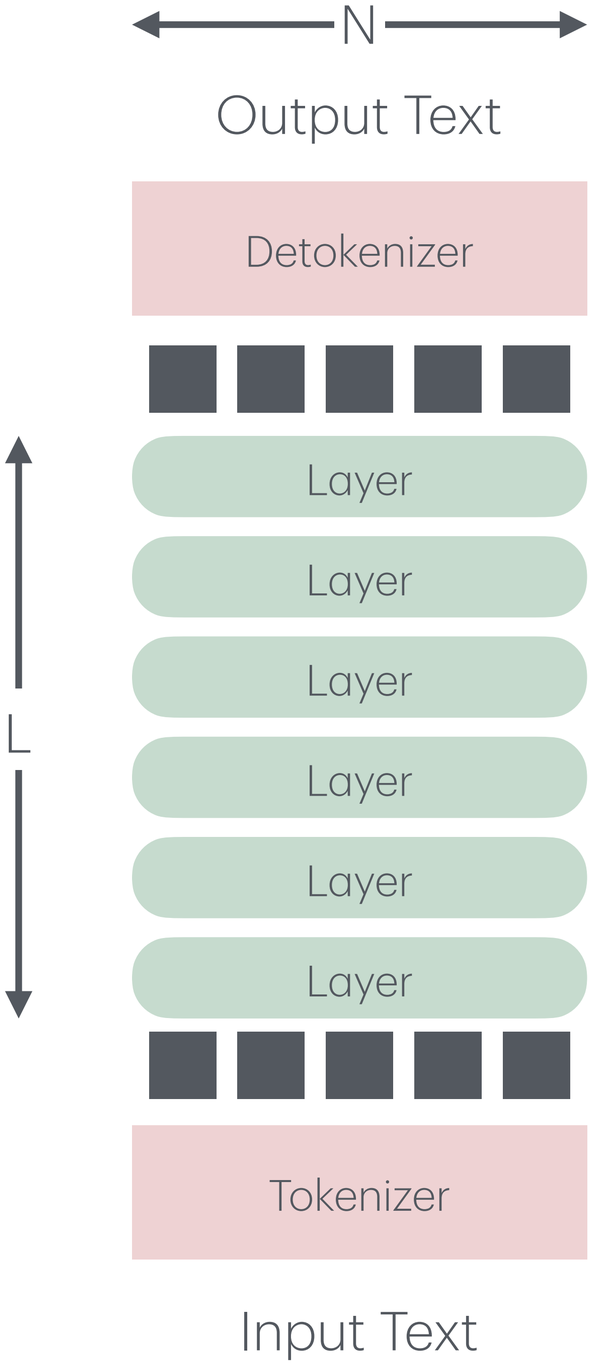
Peak Memory	
Runtime	
# forward calls	



Generation - An analysis

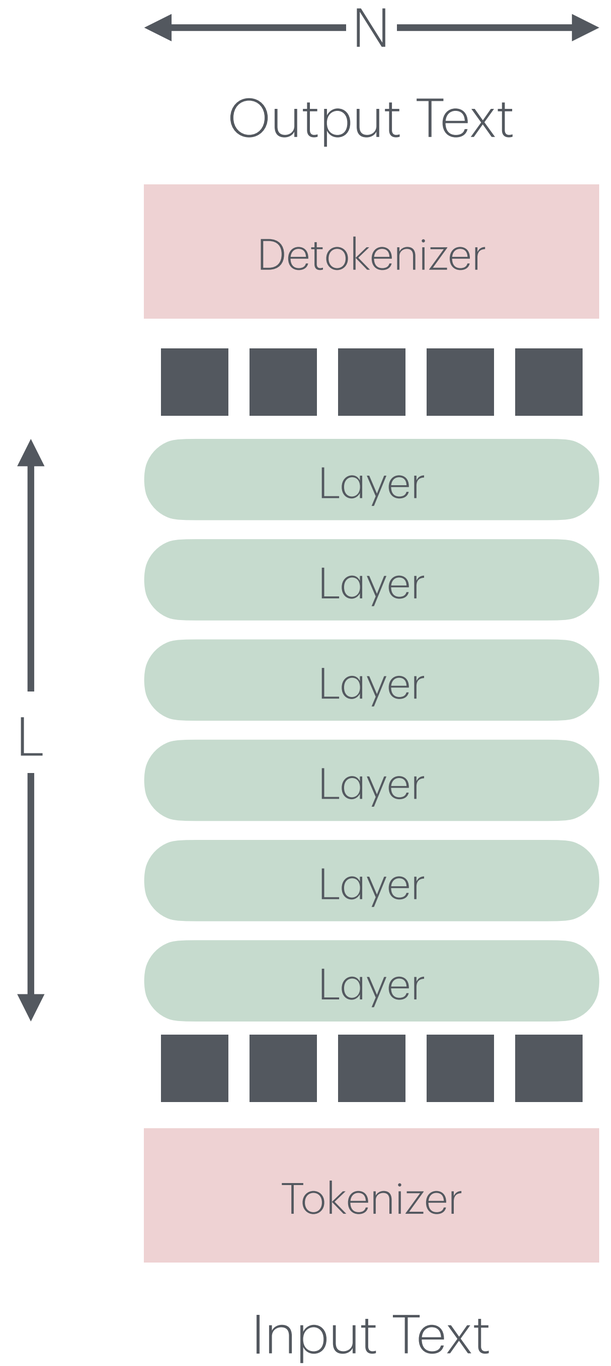
Vanilla Generation

Peak Memory	O(N)
Runtime	N x $O(N^2L) = O(N^3L)$



Training and Generation

	Training	Training - Checkpointing	Generation
Peak Memory	O(NL)	O(NL½)	O(N)
Runtime	O(N²L)	O(2 N²L)	O(N³L)
# forward calls	1	2	N



Generation - An analysis

Vanilla Generation

- Vanilla generation is
  - SLOW
  - Solution: Caching (later in section)

	Training	Training - Checkpointing	Generation
Peak Memory	O(NL)	$O(NL^{½})$	O(N)
Runtime	$O(N^2L)$	$O(2\ N^2L)$	$O(N^3L)$