Beyond Multi-Token Prediction: Pretraining LLMs with Future Summaries

Divyat Mahajan[1], **Sachin Goyal**[2], **Badr Youbi Idrissi**[3], **Mohammad Pezeshki**[3], **Ioannis Mitliagkas**[1], **David Lopez-Paz**[3], **Kartik Ahuja**[3]

[1]Mila, Université de Montréal, [2]Carnegie Mellon University, [3]FAIR at Meta

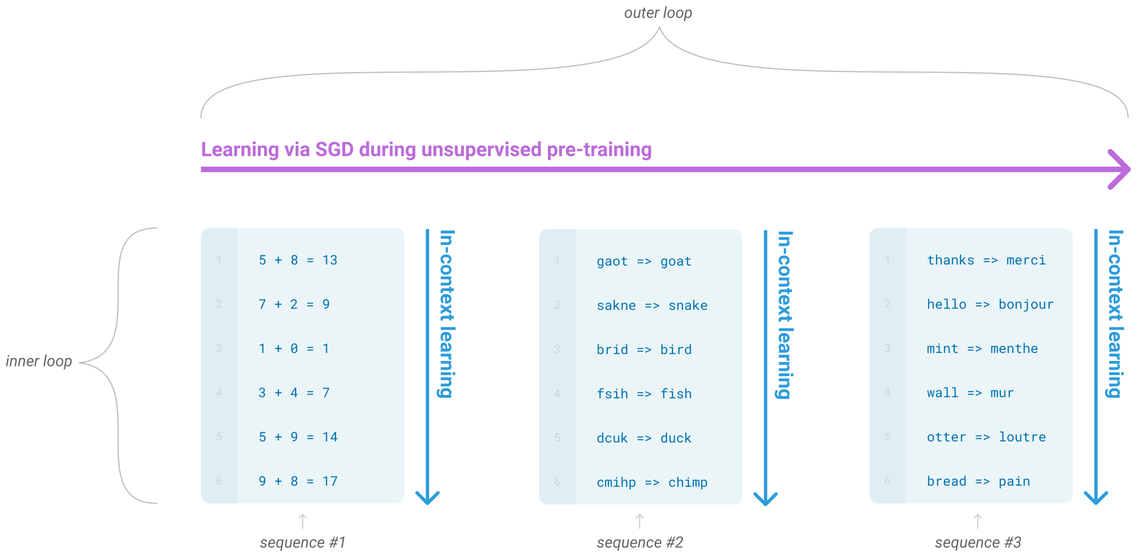
International Conference on Learning Representations (ICLR) 2026

Outline

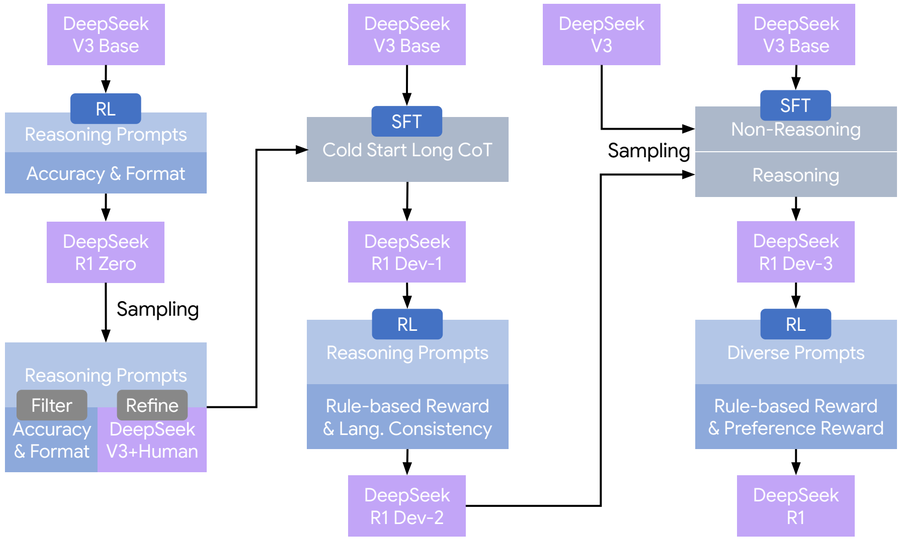
- Importance of Pretraining
- Background on Multi-Token Prediction (MTP)
  - Highlight failure of Next-Token Prediction (NTP) via controlled synthetic tasks
- Future Summary Prediction (FSP): Going beyond MTP
  - Scalable approach towards long horizon future prediction
- Future Work

Importance of Pretraining

Strong pretrained models lead to better post-training



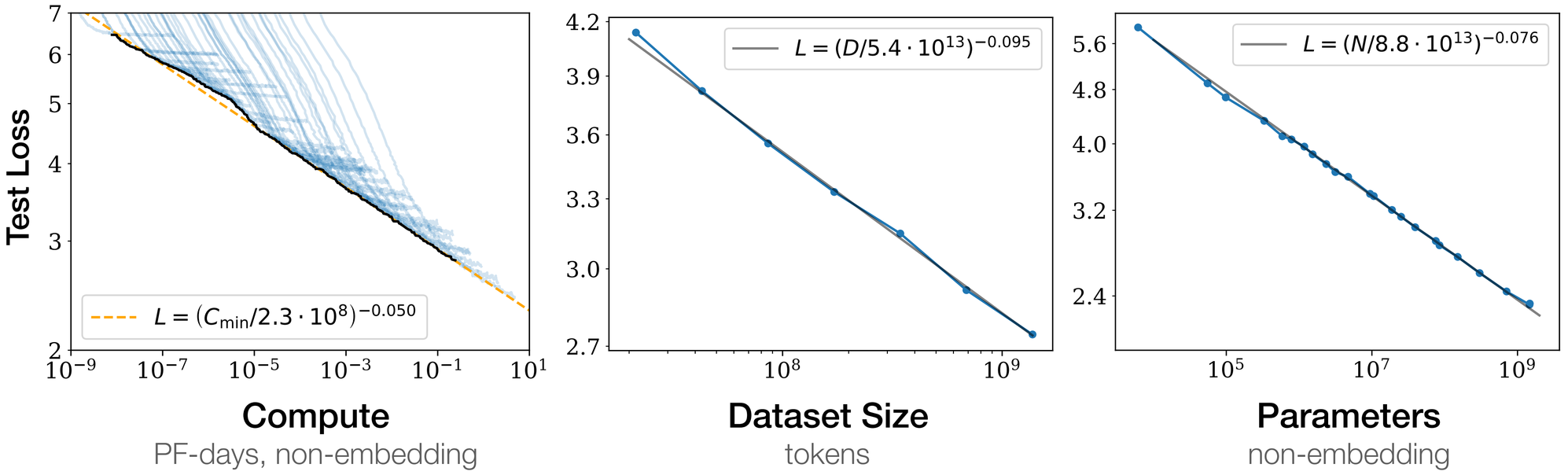
Language Models are Few-Shot Learners— Brown et al., 2020



DeepSeek-R1, 2026

LLMs Scale Predictably

Scaling compute via parameters or data provides steady improvements



Scaling Laws for Neural Language Models— Kaplan et al., 2020

Challenges Ahead with Scaling

Need new recipes as scaling data is not sustainable

Pre-training as we know it will end

Compute is growing:

- Better hardware
- Better algorithms
- Larger clusters

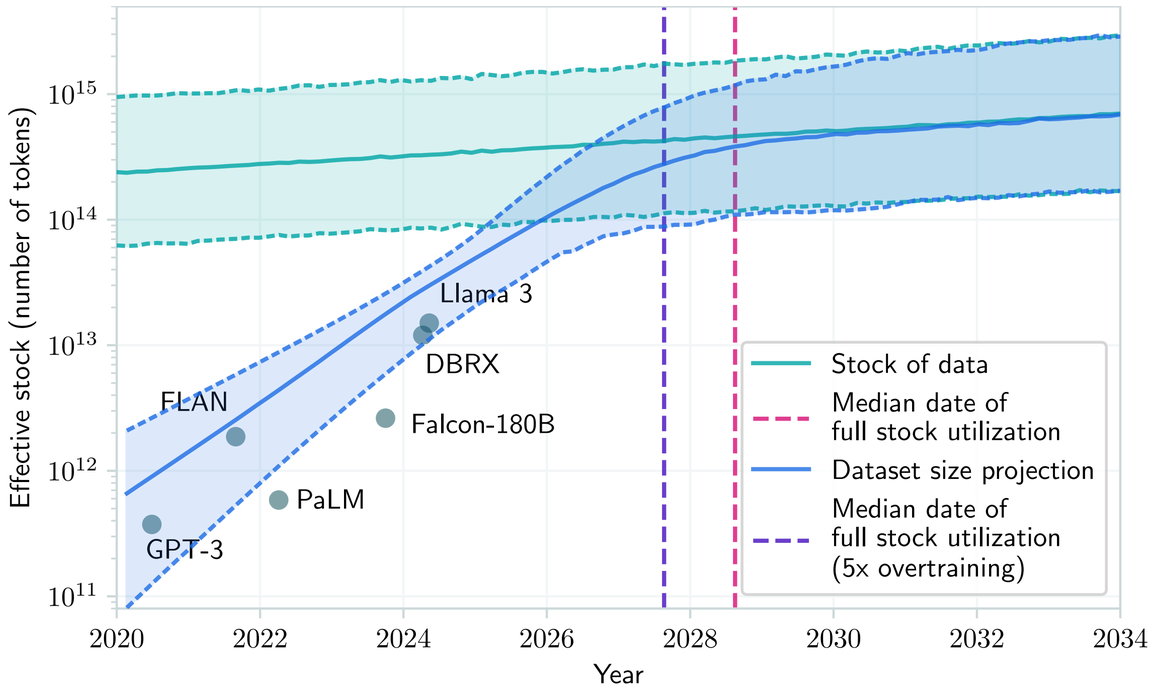
Data is not growing:

- We have but one internet
- **The fossil fuel of AI**

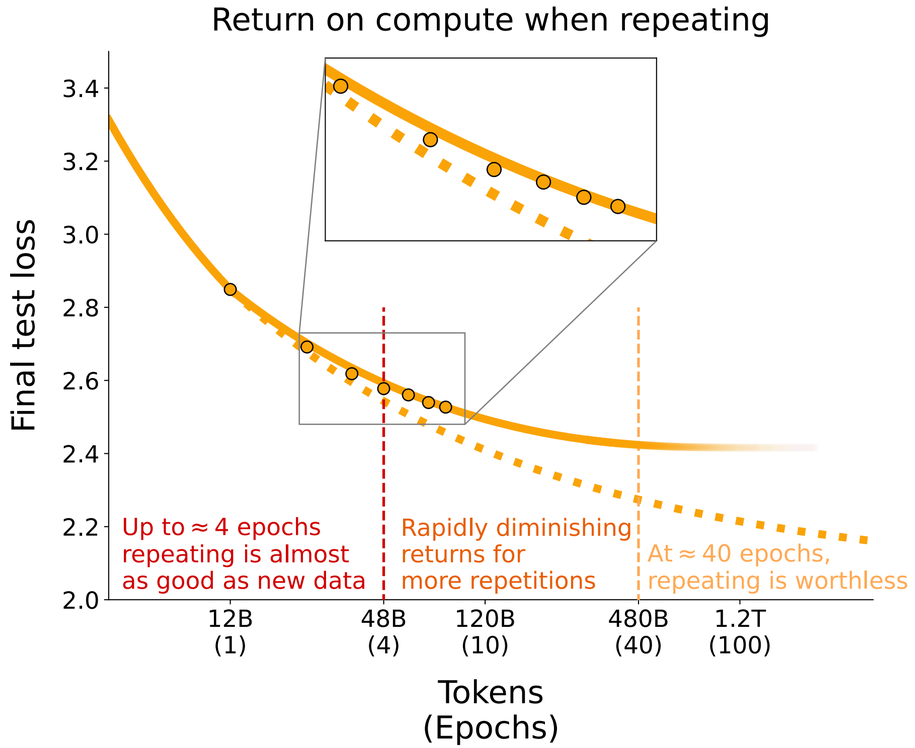
Ilya Sutskever— Test of time talk, NeurIPS 2024

Challenges Ahead with Scaling

Available internet text data grows much slower than compute



Limits of LLM scaling based on human-generated data— Villalobos et al., 2024



Scaling Data-Constrained Neural Language Models— Muennighoff et al., 2025

Directions in Pretraining

- Data (Data Mixture, Deduplication, synthetic data)
- Architecture (Attention Variants, Mixture of Experts)
- Optimization (Muon, Dion)
- **Learning Objective** (Multi-Token Prediction, Masked Language Modeling)

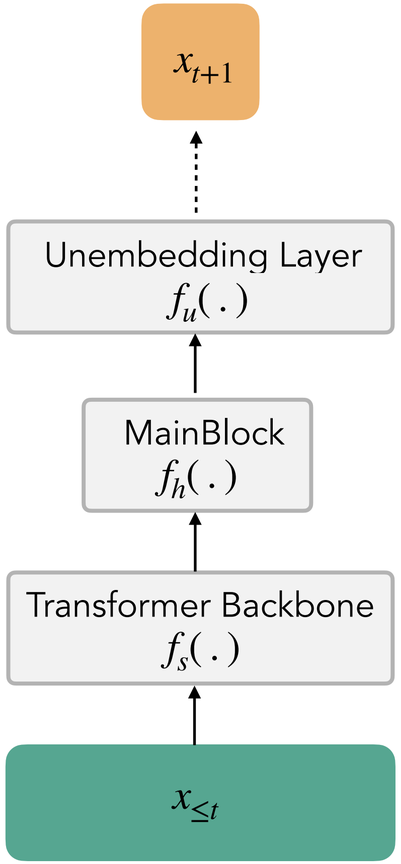
Design new objectives to extract more information from the same data

Background

Going beyond Next-token Prediction

Next-Token Prediction (NTP)

NTP can in principle learn any distribution

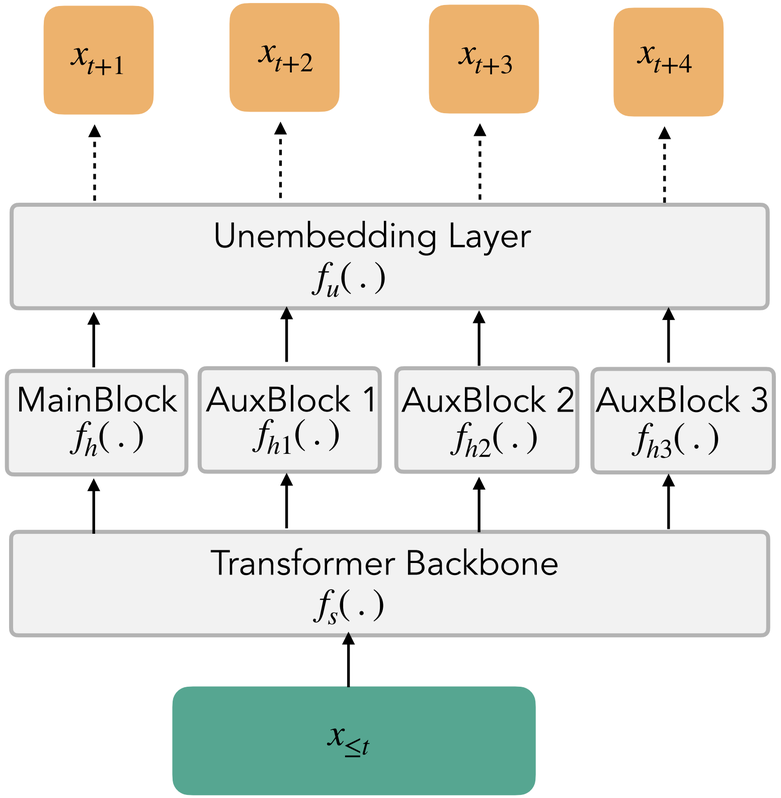


$$P_\theta(x_{t+1} \mid x_{\leq t}) = Softmax(f_u \circ f_h \circ f_s(x_{\leq t}))$$

$$L_{NTP}(x, P_\theta) = -\sum_{t=1}^{T-1} \log P_\theta(x_{t+1} \mid x_{\leq t})$$

Multi-token Prediction (MTP)

MTP provides a richer learning objective than NTP



$$Softmax(f_u \circ f_h \circ f_s(x_{\leq t})) \quad Softmax(f_u \circ f_{h1} \circ f_s(x_{\leq t})) \quad Softmax(f_u \circ f_{h2} \circ f_s(x_{\leq t})) \quad Softmax(f_u \circ f_{h3} \circ f_s(x_{\leq t}))$$

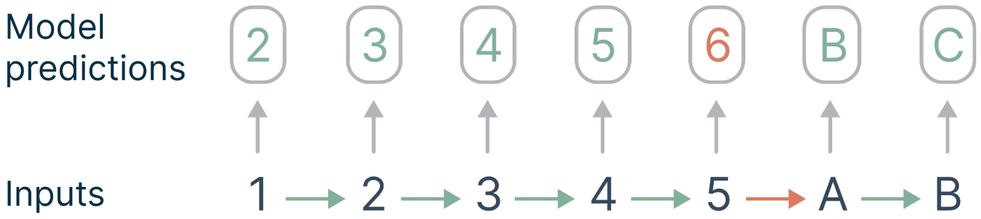
$$L_{MTP}(x, P_\theta) = -\sum_{t=1}^{T-1} \log P_\theta(x_{t+1} \mid x_{\leq t}) \; -\sum_{t=1}^{T-2} \log P_\theta(x_{t+2} \mid x_{\leq t}) \; -\sum_{t=1}^{T-3} \log P_\theta(x_{t+3} \mid x_{\leq t}) \; -\sum_{t=1}^{T-4} \log P_\theta(x_{t+4} \mid x_{\leq t})$$

Better & Faster Language Models via Multi-token Prediction— Gloeckle et al., 2024

Intuition behind MTP

Exposure bias due to teacher forcing in auto-regressive training

Training

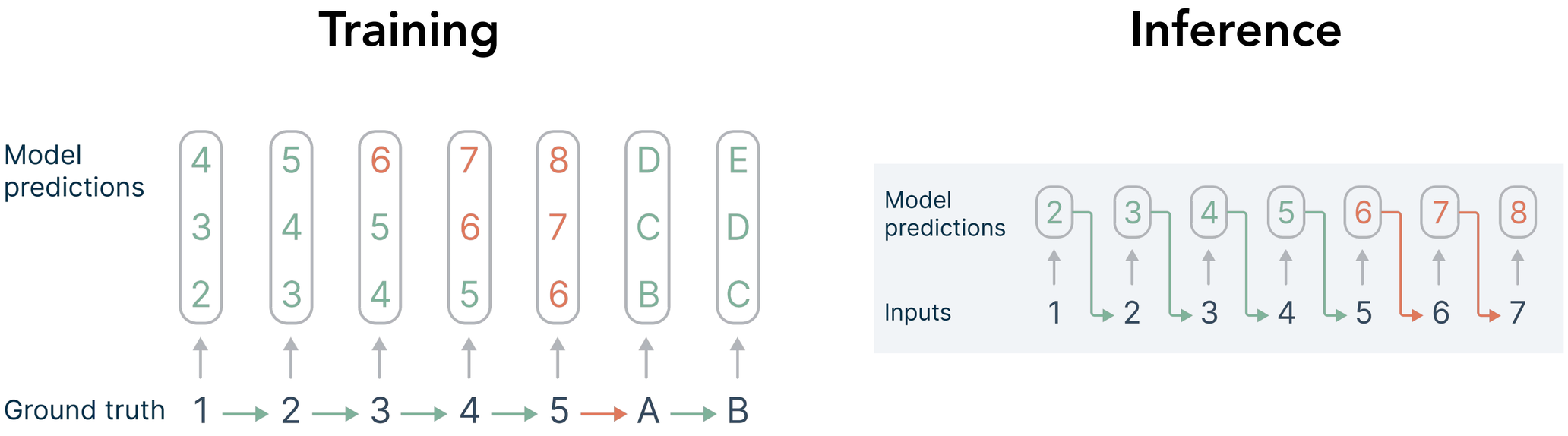


Inference

Better & Faster Language Models via Multi-token Prediction— Gloeckle et al., 2024

Intuition behind MTP

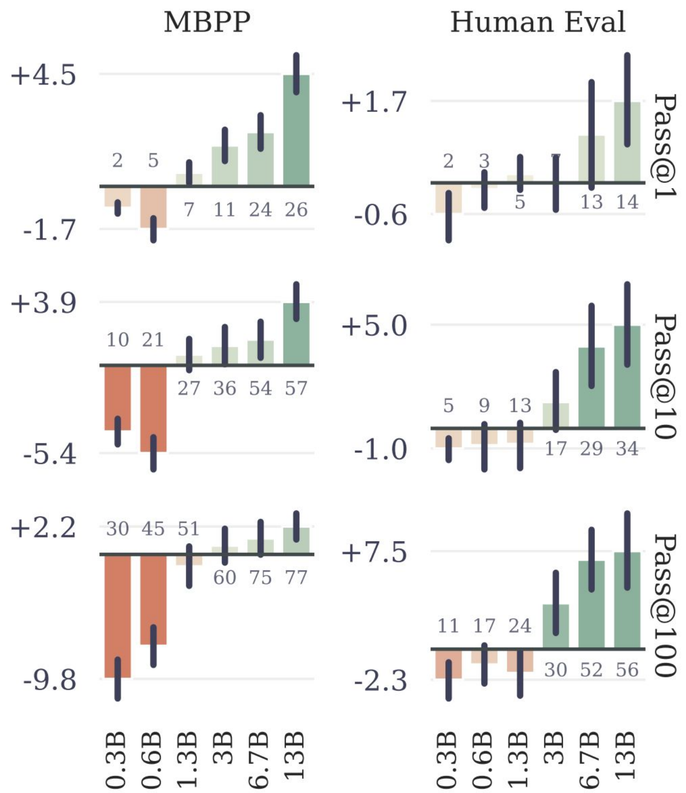
MTP reduces teacher forcing & promotes better generalization



Better & Faster Language Models via Multi-token Prediction— Gloeckle et al., 2024

Intuition behind MTP

Better performance than NTP on reasoning tasks

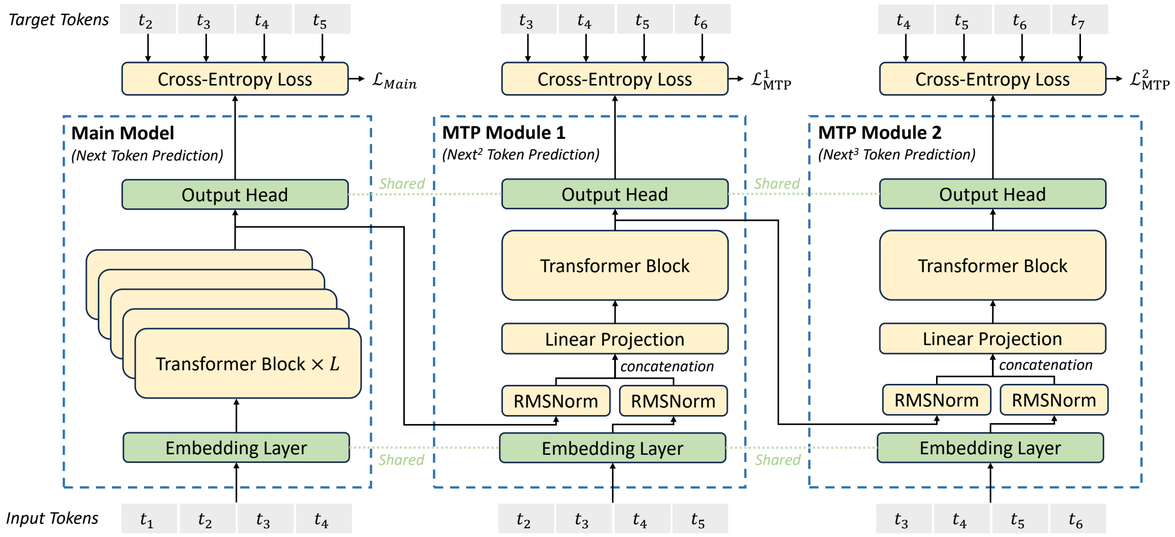


Training data	Vocabulary	n	MBPP			HumanEval			APPS/Intro		
			@1	@10	@100	@1	@10	@100	@1	@10	@100
313B bytes (0.5 epochs)	bytes	1	19.3	42.4	64.7	18.1	28.2	47.8	0.1	0.5	2.4
		8	**32.3**	**50.0**	**69.6**	**21.8**	**34.1**	**57.9**	**1.2**	**5.7**	**14.0**
		16	28.6	47.1	68.0	20.4	32.7	54.3	1.0	5.0	12.9
		32	23.0	40.7	60.3	17.2	30.2	49.7	0.6	2.8	8.8
200B tokens (0.8 epochs)	32k tokens	1	30.0	53.8	73.7	22.8	36.4	62.0	2.8	7.8	17.4
		2	30.3	55.1	76.2	22.2	38.5	62.6	2.1	9.0	21.7
		4	**33.8**	**55.9**	**76.9**	**24.0**	**40.1**	**66.1**	1.6	7.1	19.9
		6	31.9	53.9	73.1	20.6	38.4	63.9	**3.5**	**10.8**	**22.7**
		8	30.7	52.2	73.4	20.0	36.6	59.6	3.5	10.4	22.1
1T tokens (4 epochs)	32k tokens	1	40.7	65.4	83.4	31.7	57.6	83.0	**5.4**	**17.8**	**34.1**
		4	**43.1**	65.9	83.7	31.6	57.3	**86.2**	4.3	15.6	33.7

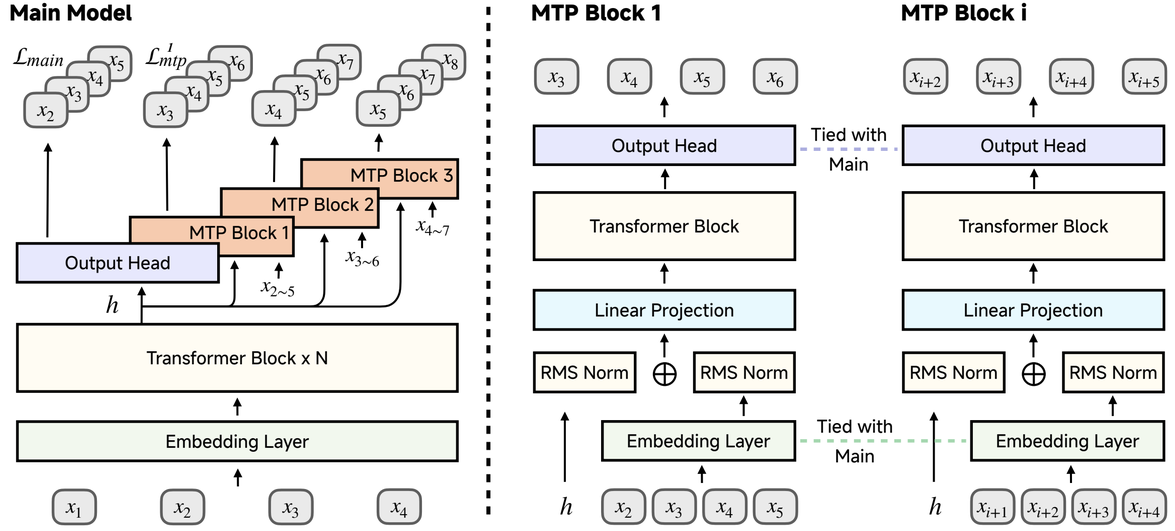
Better & Faster Language Models via Multi-token Prediction— Gloeckle et al., 2024

Multi-Token Prediction

Strong open source models are using MTP (variants)



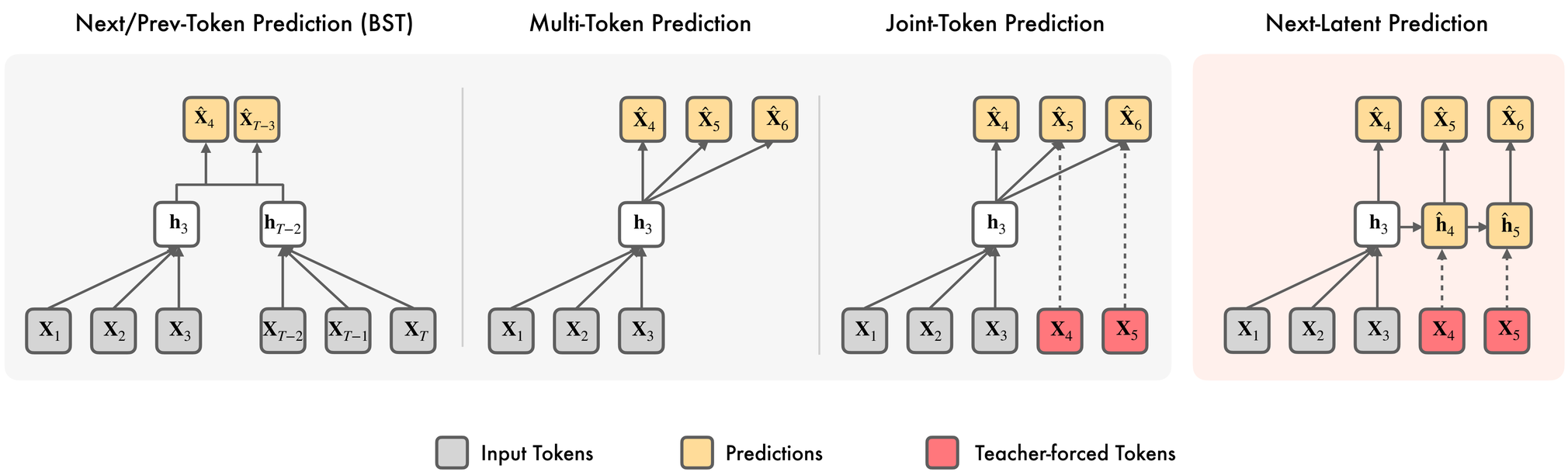
DeepSeek-V3 Technical Report— DeepSeekAI, 2025



MiMo: Unlocking reasoning potential of Language Model— Xiaomi., 2025

Recent advances on MTP

Learning belief-states via light teacher forcing on future tokens



Next-Latent Prediction Transformers Learn Compact World Models— Teoh et al., 2025

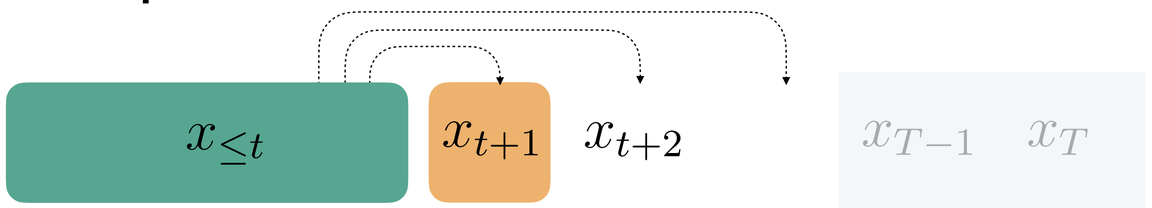
Future Summary Prediction

Going beyond Multi-Token Prediction

Issue with MTP: Scaling Prediction Horizon

Need auxiliary head for every additional future token

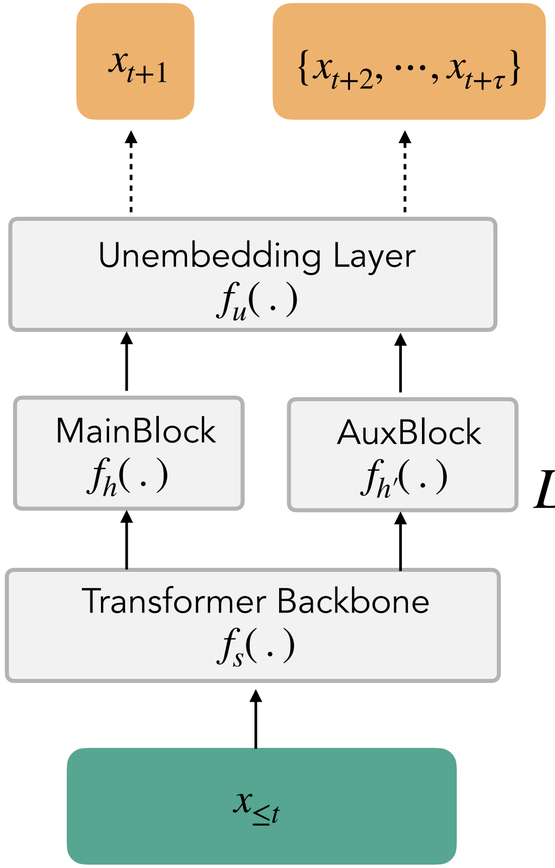
MTP: Uses multiple auxiliary heads, each predicting a specific future token



Instead of the entire future sequence, lets predict a future summary!

Future Summary Prediction: Bag-of-words (FSP-BoW)

Auxiliary target as bag-of-words summary of future (Single auxiliary head!)



$$Softmax(f_u \circ f_h \circ f_s(x_{\leq t})) \qquad Sigmoid(f_u \circ f_{h'} \circ f_s(x_{\leq t}))$$

$$L_{FSP-BoW}(x, P_\theta) = -\sum_{t=1}^{T-1} \log P_\theta(x_{t+1} \mid x_{\leq t}) + BCE\big(A_\theta(x_{\leq t}), \{x_{t+2}, \cdots, x_{t+\tau}\} \big)$$

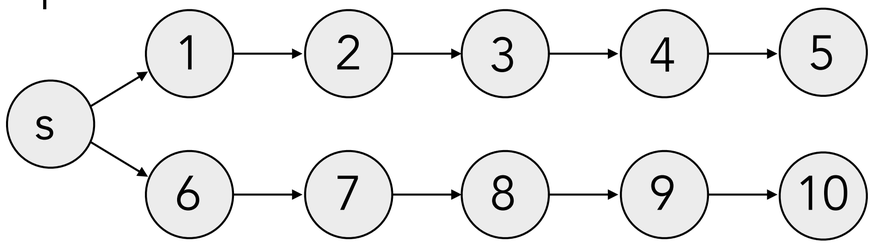
Binary Cross Entropy Loss

Path-star Graph: Long Horizon Prediction

Task: Predict path between the input start and end nodes

Example Graph:

$G(2,5)$



Example Sequence:

start/goal path

x = <graph> s 5 1 2 3 4 5

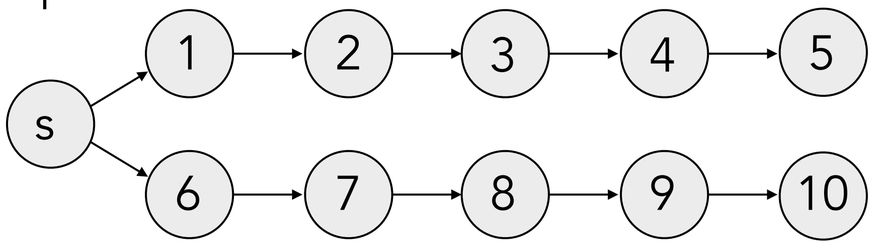
Pitfalls of next-token prediction— Bachmann et al., 2024

Path-star Graph: Long Horizon Prediction

NTP fails due to shortcut learning for intermediate nodes

Example Graph:

$G(2,5)$



NTP:

<graph> s 5 1 2 3 4 5

Example Sequence:

start/goal path

x = <graph> s 5 1 2 3 4 5

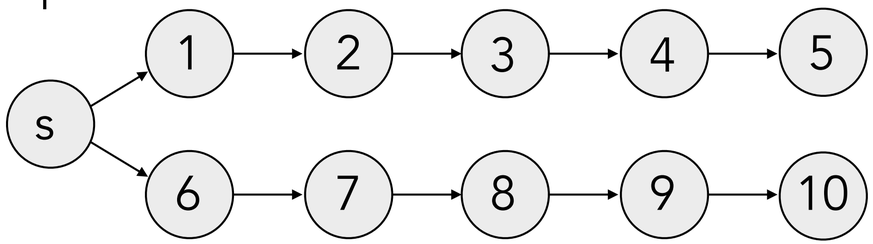
Multiple edges from start node

Path-star Graph: Long Horizon Prediction

NTP fails due to shortcut learning for intermediate nodes

Example Graph:

$G(2,5)$



NTP:

<graph> s 5 1 2 3 4 5
<graph> s 5 1 2 3 4 5

Example Sequence:

start/goal path

x = <graph> s 5 1 2 3 4 5

Only one edge from node 1

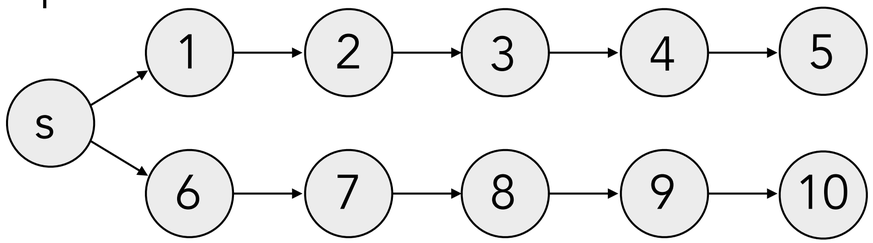
Lookup in graph and predict!

Path-star Graph: Long Horizon Prediction

NTP fails due to shortcut learning for intermediate nodes

Example Graph:

$G(2,5)$



Example Sequence:

start/goal path

x = <graph> s 5 1 2 3 4 5

NTP:

<graph> s 5 1 2 3 4 5
<graph> s 5 1 2 3 4 5
<graph> s 5 1 2 3 4 5

Only one edge from node 2

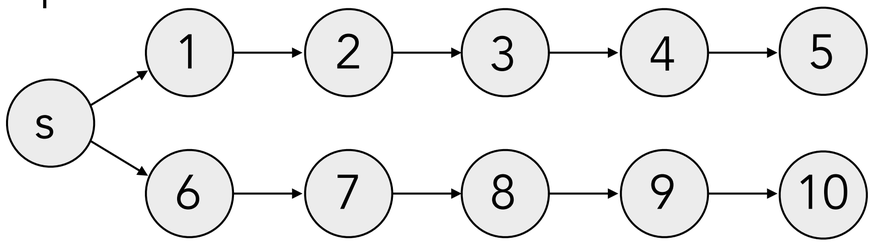
Lookup in graph and predict!

Path-star Graph: Long Horizon Prediction

NTP fails due to shortcut learning for intermediate nodes

Example Graph:

$G(2,5)$



Example Sequence:

start/goal path

x = <graph> s 5 1 2 3 4 5

NTP:

<graph> s 5 1 2 3 4 5
<graph> s 5 1 2 3 4 5
<graph> s 5 1 2 3 4 5
<graph> s 5 1 2 3 4 5

Only one edge from node 3

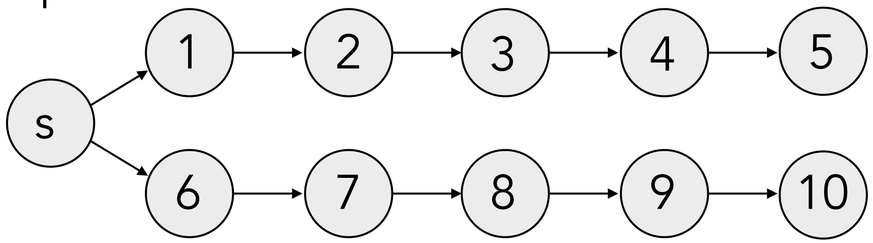
Lookup in graph and predict!

Path-star Graph: Long Horizon Prediction

NTP fails due to shortcut learning for intermediate nodes

Example Graph:

$G(2,5)$



Example Sequence:

start/goal path

x = <graph> s 5 1 2 3 4 5

NTP:

<graph> s 5 1 2 3 4 5
<graph> s 5 1 2 3 4 5
<graph> s 5 1 2 3 4 5
<graph> s 5 1 2 3 4 5
<graph> s 5 1 2 3 4 5

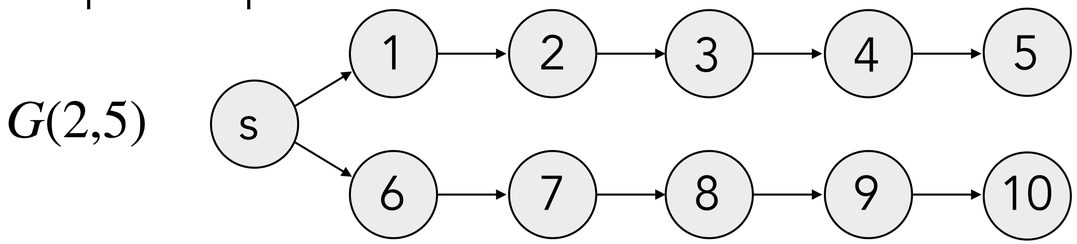
Only one edge from node 4

Lookup in graph and predict!

Path-star Graph: Long Horizon Prediction

NTP fails due to shortcut learning for intermediate nodes

Example Graph:



Example Sequence:

start/goal path

x = <graph> s 5 1 2 3 4 5

NTP:

<graph> s 5 1 2 3 4 5 → *Difficult prediction task*

<graph> s 5 1 2 3 4 5

<graph> s 5 1 2 3 4 5

<graph> s 5 1 2 3 4 5

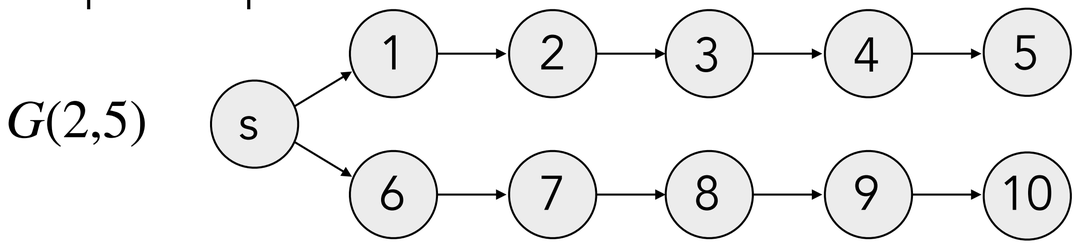
<graph> s 5 1 2 3 4 5 → *Easy prediction task*

Inference accuracy solely depends upon correct prediction of first node

Path-star Graph: Long Horizon Prediction

Predicting future tokens avoids shortcuts as it requires multiple lookups

Example Graph:



Example Sequence:

start/goal path

x = <graph> s 5 1 2 3 4 5

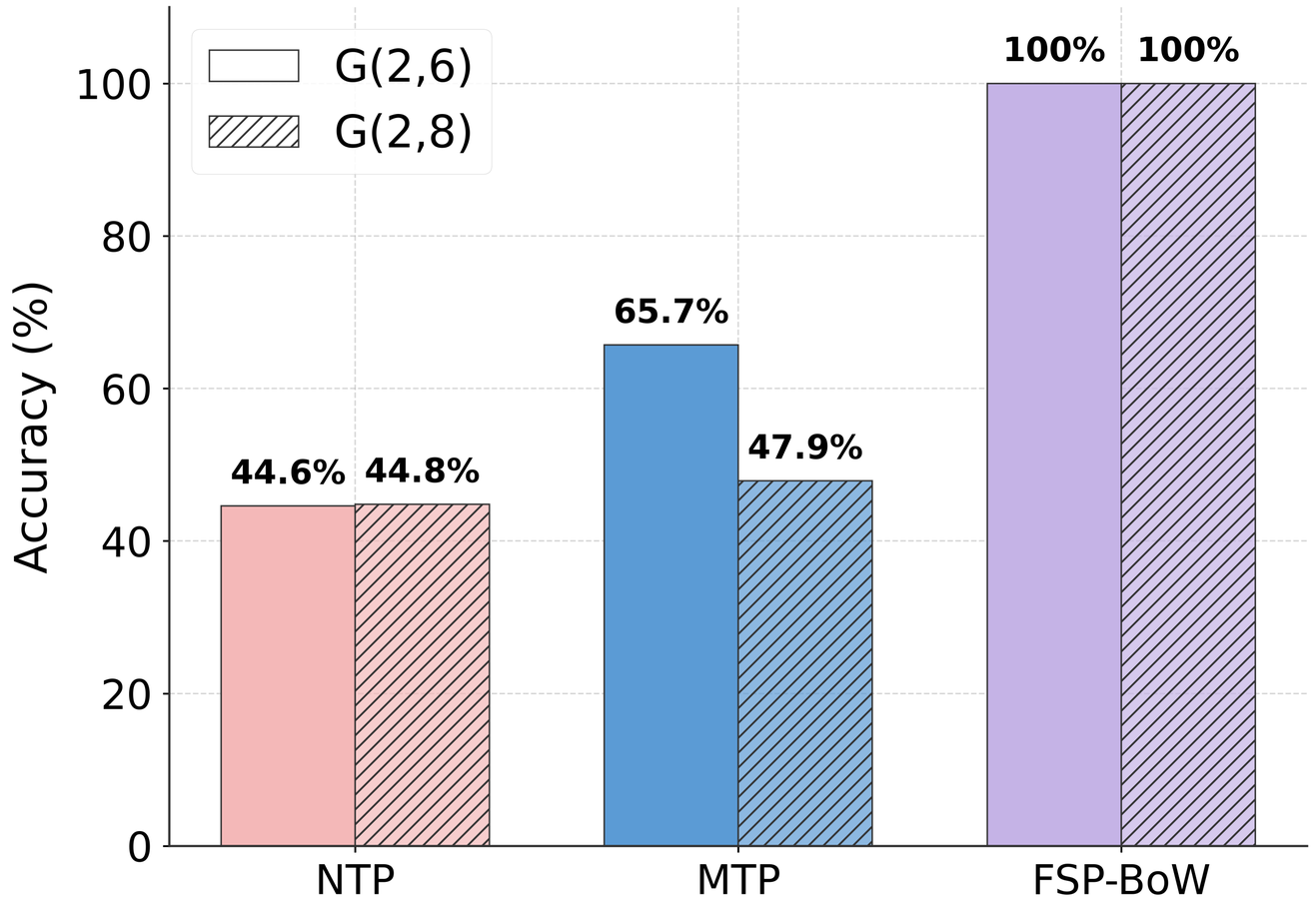
FSP-BoW:

<graph> s 5 1 2 3 4 5
<graph> s 5 1 2 3 4 5
<graph> s 5 1 2 3 4 5
<graph> s 5 1 2 3 4 5
<graph> s 5 1 2 3 4 5

Difficult prediction tasks

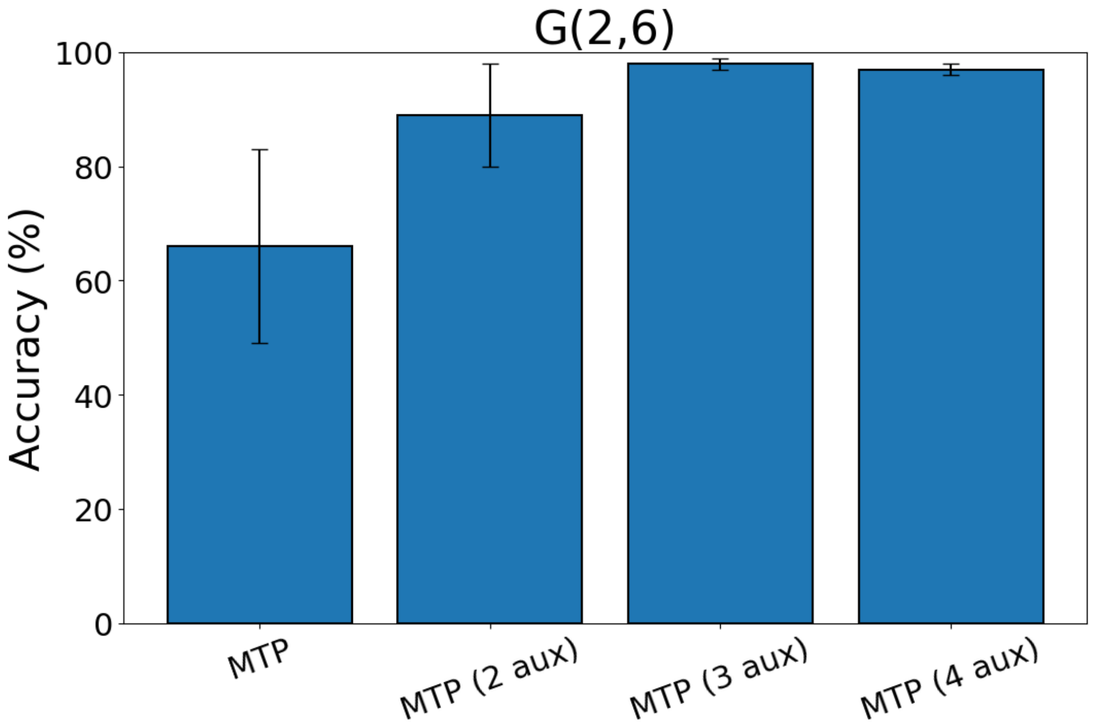
Path-star Graph: Long Horizon Prediction

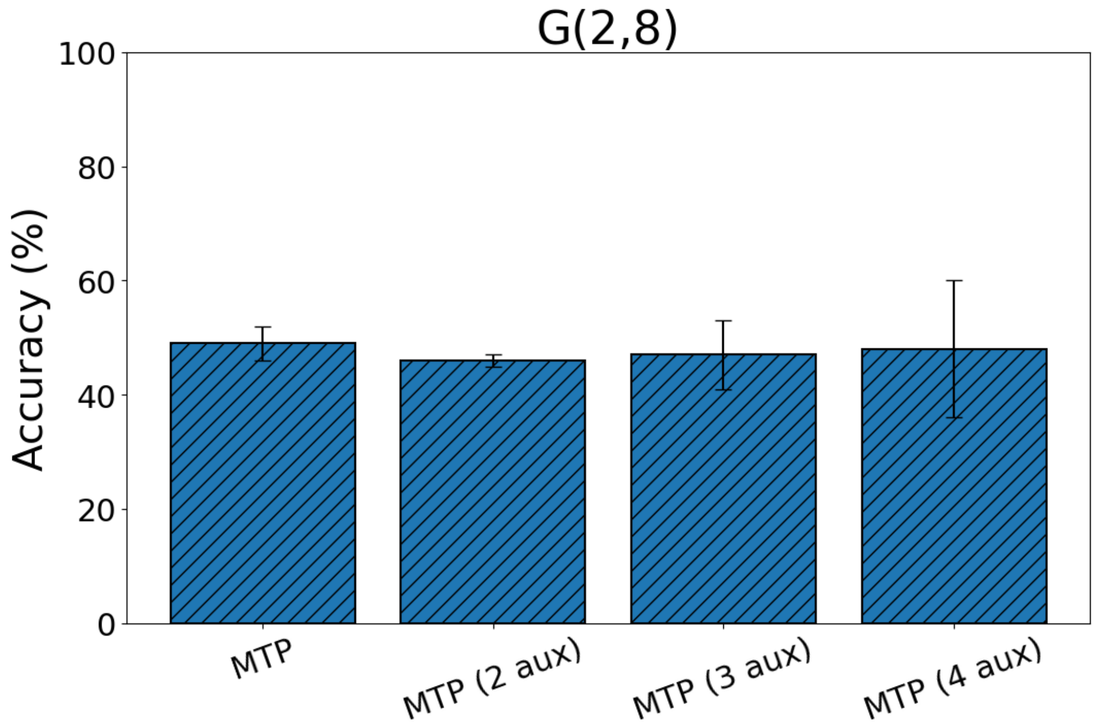
FSP-BoW with a single auxiliary head achieves perfect score



Path-star Graph: Long Horizon Prediction

MTP via scaling auxiliary heads becomes impractical for larger graphs

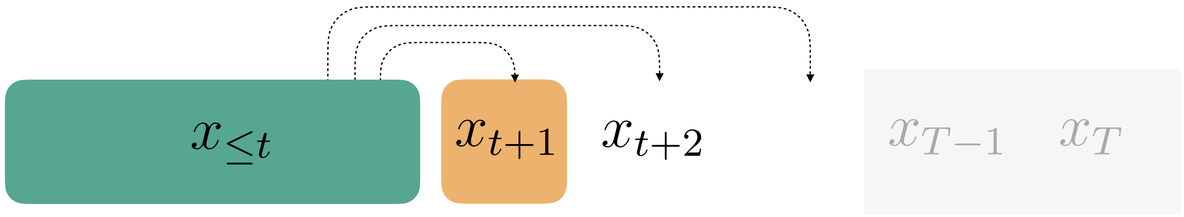




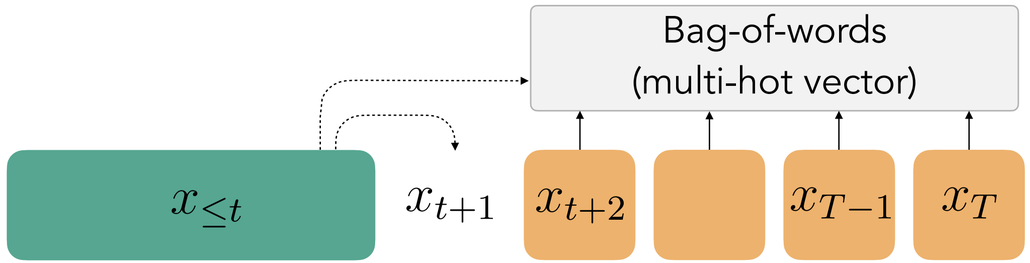
Issue with MTP: How much lookahead?

Future can have irrelevant or noise tokens

MTP: Uses multiple auxiliary heads, each predicting a specific future token



FSP-BoW: Predicts a "bag-of-tokens" summary

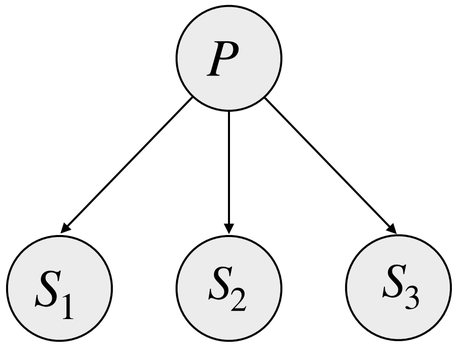


Need to learn a future summary instead of hand-crafted choices!

Sibling Discovery

Task: Generate valid sequences, i.e, siblings followed by their parent

Graph:



Example Sequence:

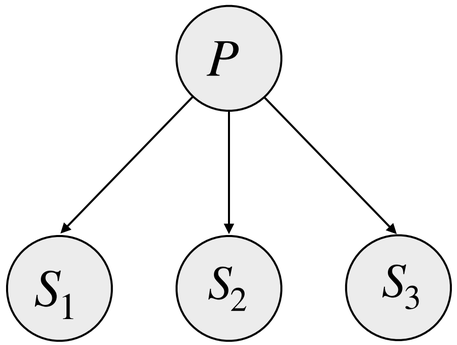
$$x = S_1 \ S_2 \ S_3 \ P$$

Going beyond the creative limits of next-token prediction— Nagarajan et al., 2025

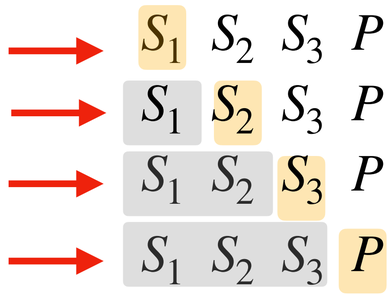
Sibling Discovery

Task: Generate valid sequences, i.e, siblings followed by their parent

Graph:



NTP:



Example Sequence:

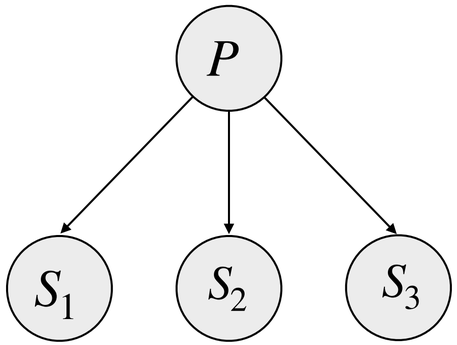
x = S_1 S_2 S_3 P

The model now sees the common parent between siblings

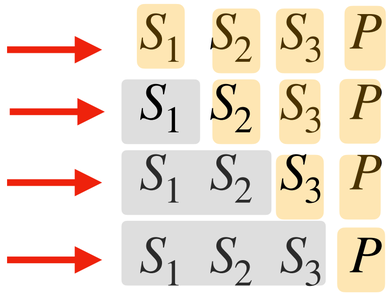
Sibling Discovery

Task: Generate valid sequences, i.e, siblings followed by their parent

Graph:



FSP-BoW:



Example Sequence:

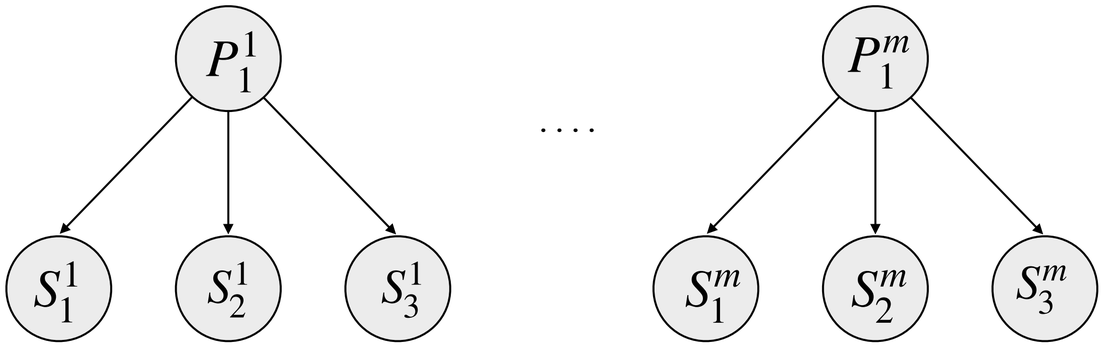
$$x = S_1 \ S_2 \ S_3 \ P$$

The model sees the common parent between siblings more often than NTP

Sibling Discovery Modified

Lets add multiple independent components (m) to the graph

Graph:



FSP-BoW:

x = S_1^1 S_2^1 S_3^1 P^1 S_1^2 S_2^2 S_3^2 P^2 S_1^3 S_2^3 S_3^1 P^3

If this is the prefix — *Relevant future* — *Not predictable from the prefix*

Example Sequence:

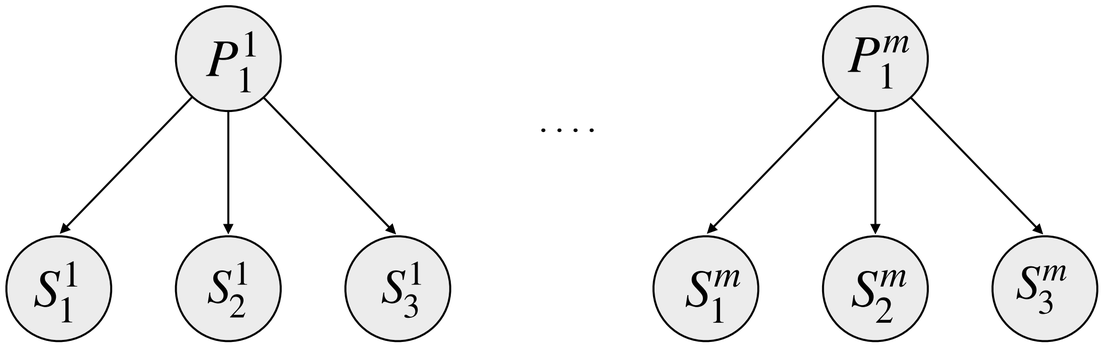
x = S_1^1 S_2^1 S_3^1 P^1 S_1^2 S_2^2 S_3^2 P^2 S_1^3 S_2^3 S_3^1 P^3

FSP-BoW would predict a lot of irrelevant information from the future!

Sibling Discovery Modified

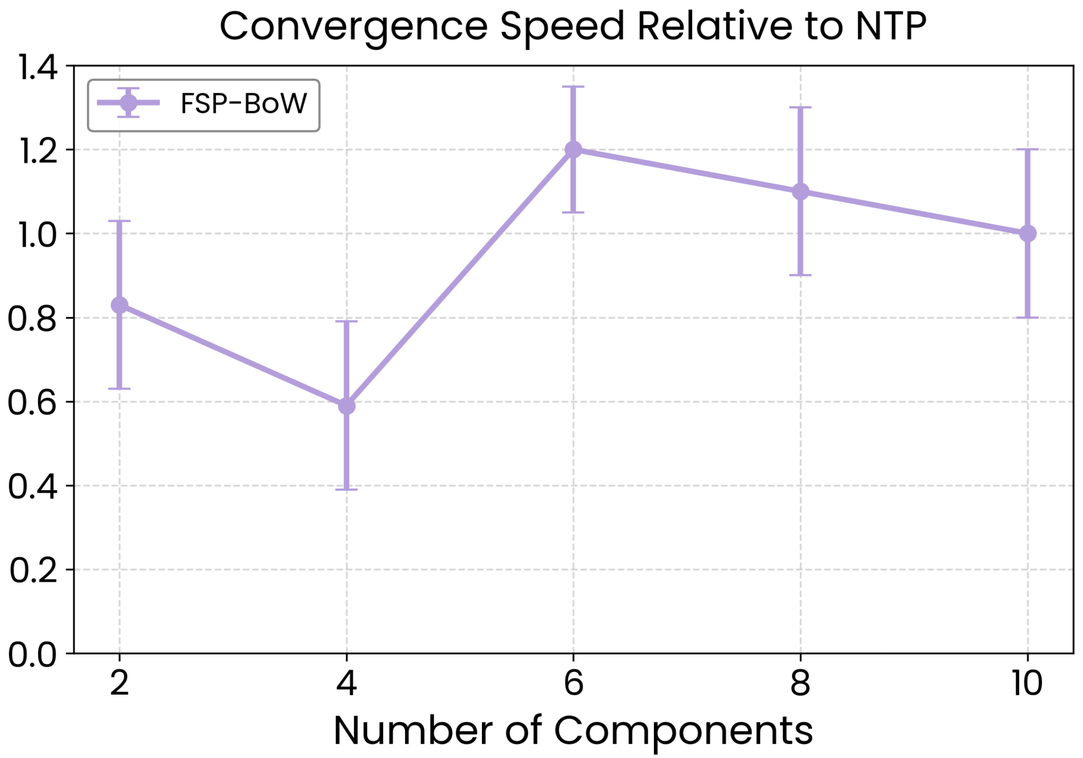
FSP-BoW only provide speed up over NTP for low component cases

Graph:



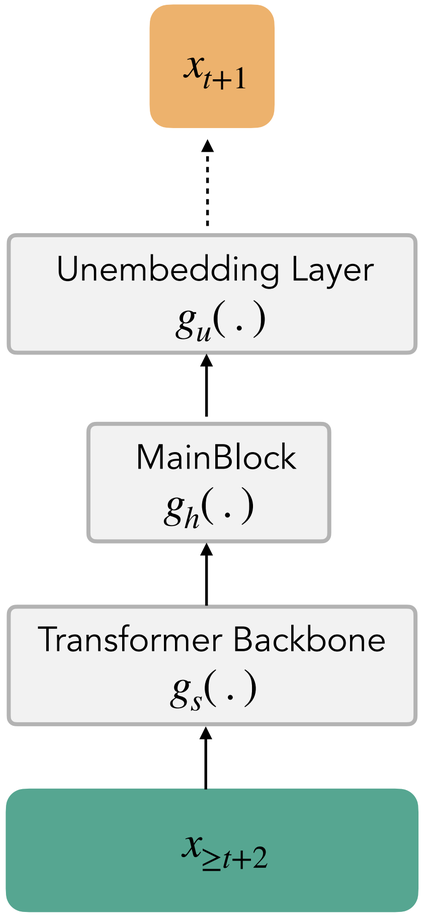
Example Sequence:

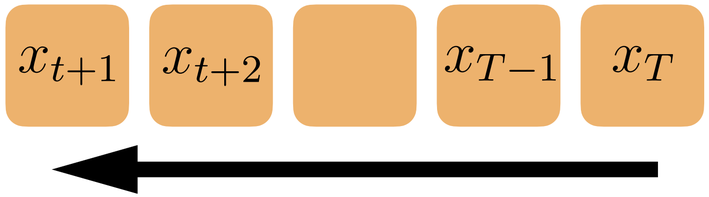
$$x = S_1^1 \; S_2^1 \; S_3^1 \; P^1 \; S_1^2 \; S_2^2 \; S_3^2 \; P^2 \; S_1^3 \; S_2^3 \; S_3^1 \; P^3$$



Future Summary Prediction: ReverseLM (FSP-RevLM)

Step 1. Train language model on reverse sequences (RevLM)





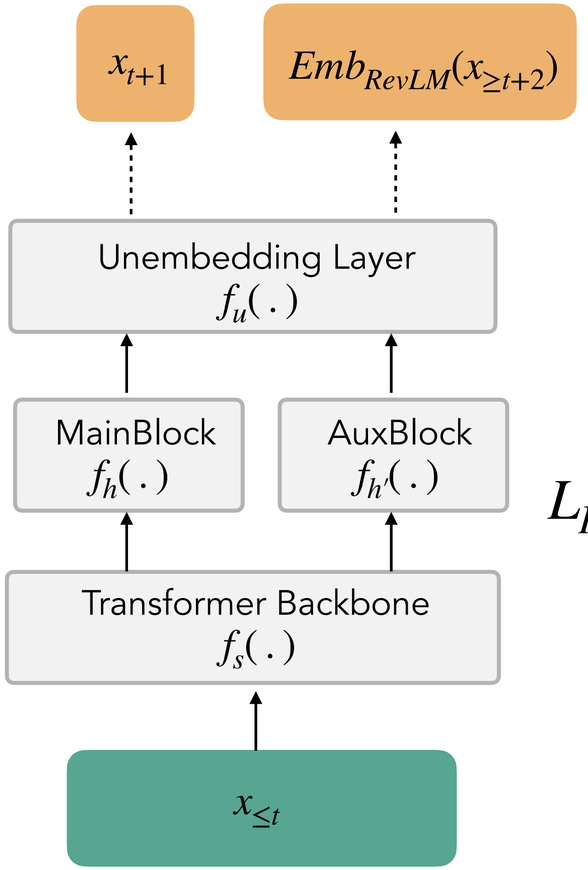
$$Q_\phi(x_{t+1} \,|\, x_{\geq t+2}) = Softmax(g_u \circ g_h \circ g_s(x_{\geq t+2}))$$

$$L_{RevLM}(x, Q_\phi) = -\sum_{t=0}^{T-2} \log Q_\phi(x_{t+1} \,|\, x_{\geq t+2})$$

Future Summary: $Emb_{RevLM}(x_{\geq t+2}) = g_h \circ g_s(x_{\geq t+2})$

Future Summary Prediction: ReverseLM (FSP-RevLM)

Step 2. Auxiliary target as a learned embedding of future (Single auxiliary head!)



$$Softmax(f_u \circ f_h \circ f_s(x_{\leq t})) \qquad f_{h'} \circ f_s(x_{\leq t})$$

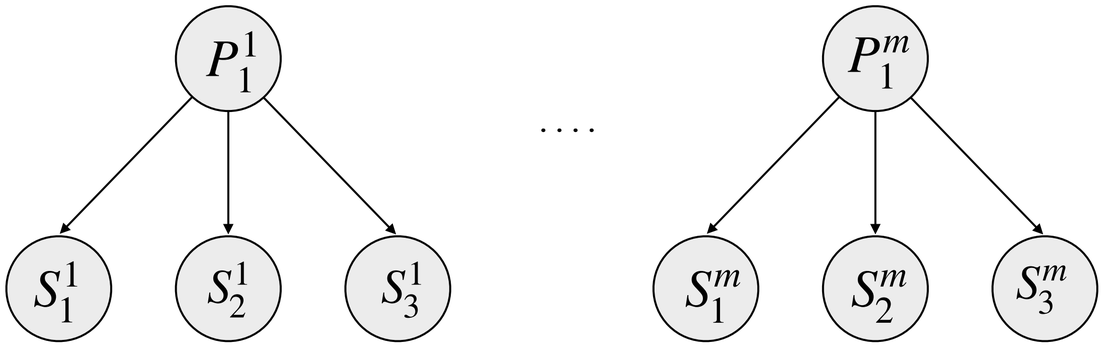
$$L_{FSP-RevLM}(x, P_\theta) = -\sum_{t=1}^{T-1} \log P_\theta(x_{t+1} \mid x_{\leq t}) + || A_\theta(x_{\leq t}), Emb_{RevLM}(x_{\geq t+2}) ||_2^2$$

Embedding Matching Loss

Sibling Discovery: Learned Future Summary

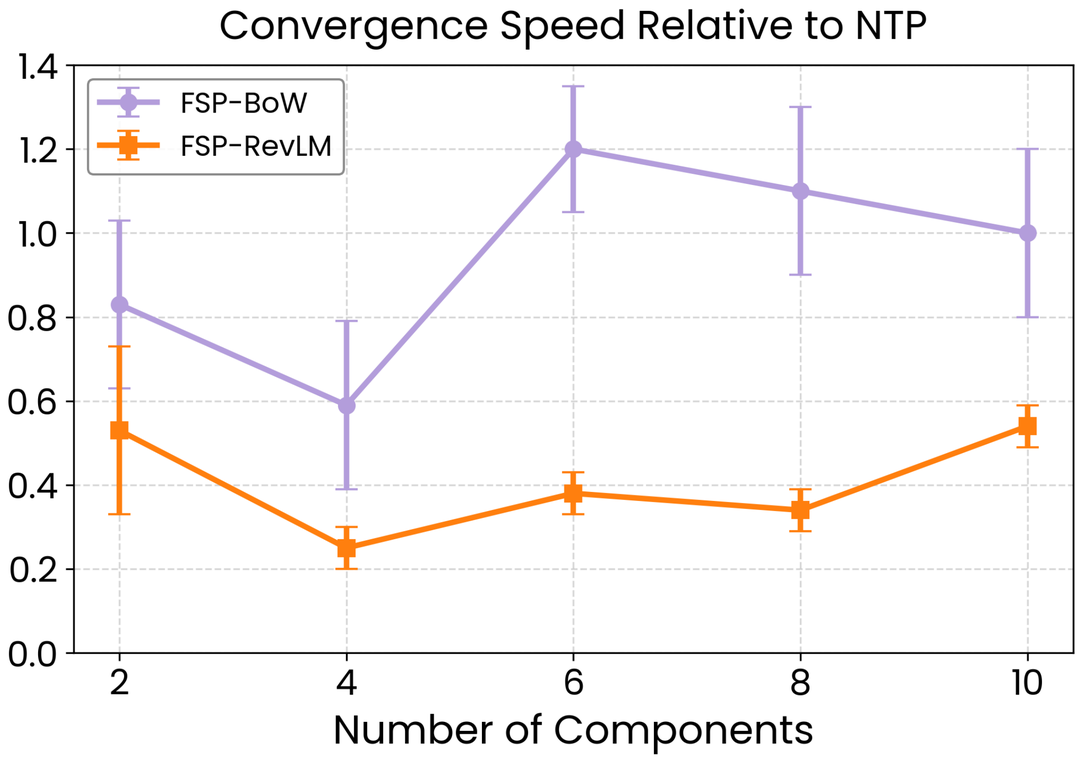
FSP-RevLM provides speed up over NTP even for high component cases

Graph:

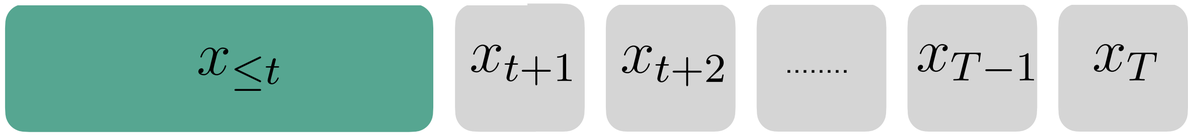


Example Sequence:

$$x = S_1^1 \; S_2^1 \; S_3^1 \; P^1 \; S_1^2 \; S_2^2 \; S_3^2 \; P^2 \; S_1^3 \; S_2^3 \; S_3^1 \; P^3$$



Future Summary Prediction



Future Summary Prediction

Real-world pretraining experiments

Experiment Setup

- Data (DCLM, Github, Proof Pile, etc.) & Architecture (LLaMA 3)

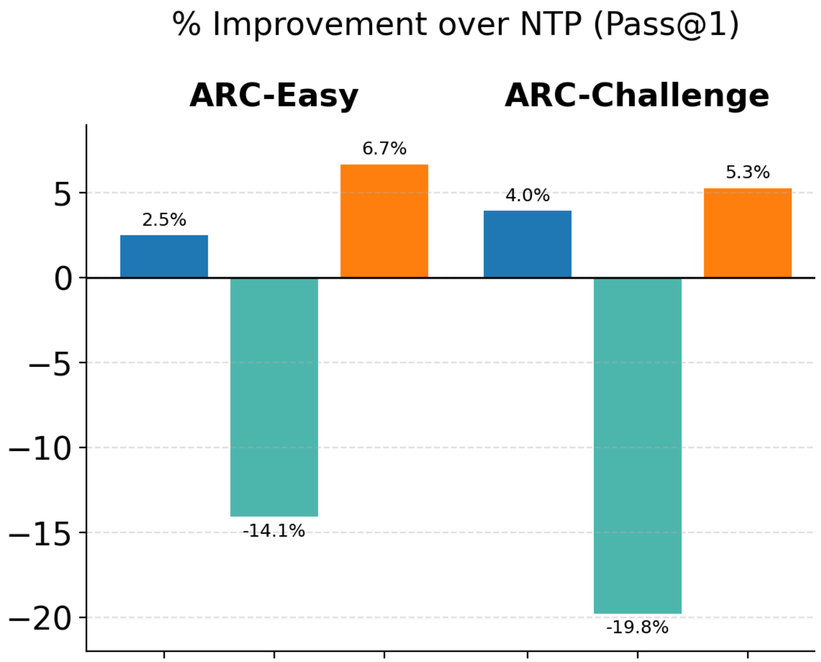
- Scale
  - 3B Parameters, 250B Tokens
  - 8B Parameters, 1T Tokens

- Auxiliary Heads
  - Training: Single auxiliary head for MTP & DS-MTP for fair comparison with FSP
  - Inference: Discard the auxiliary head and only use the next-token (main) head

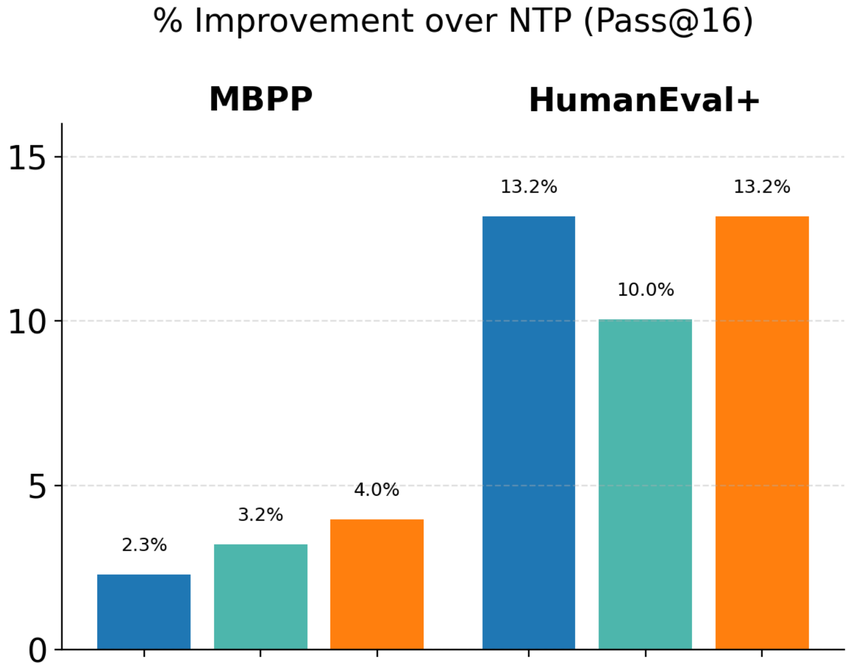
- ReverseLM (Teacher)
  - Same model size and trained on the same dataset as the baselines

Pretraining Results: 8B

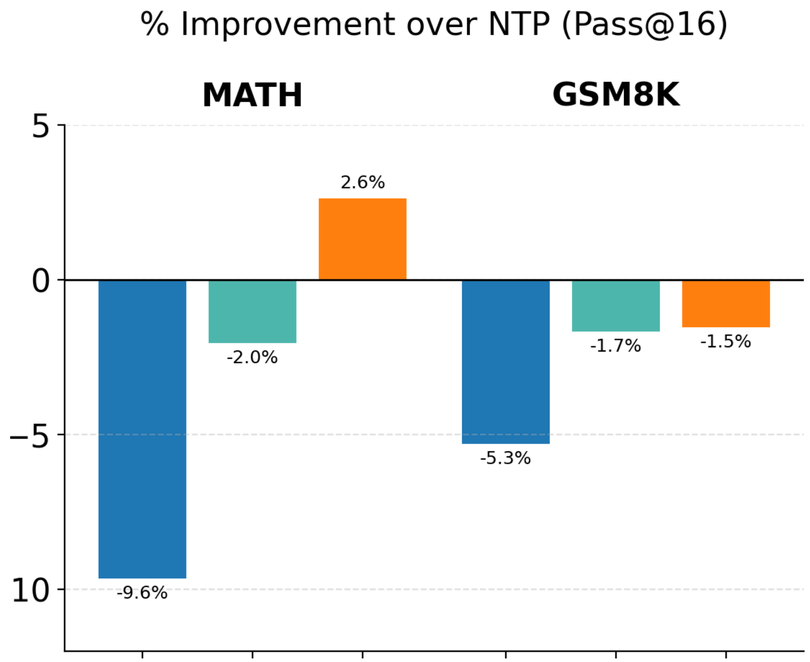
General Reasoning



Coding



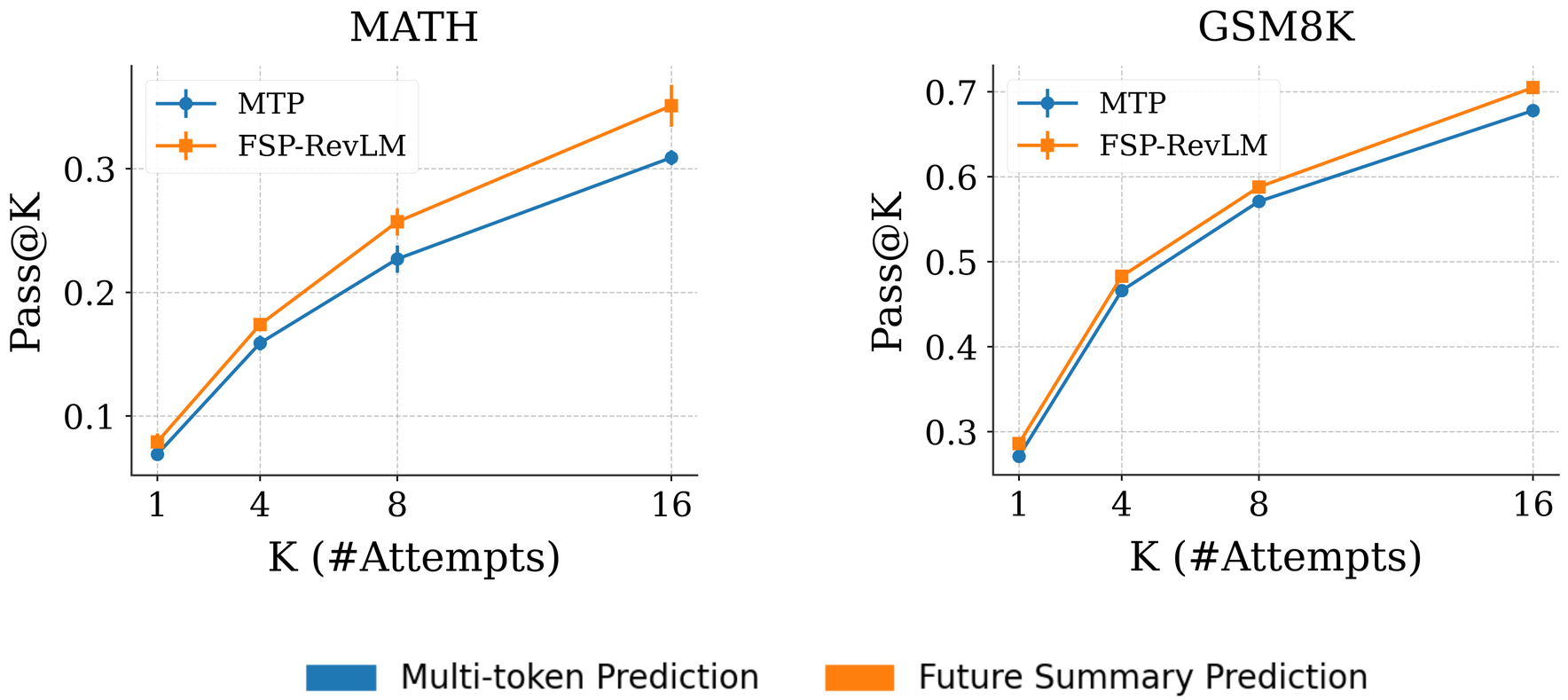
Math Reasoning



Multi-token Prediction DS Multi-token Prediction Future Summary Prediction

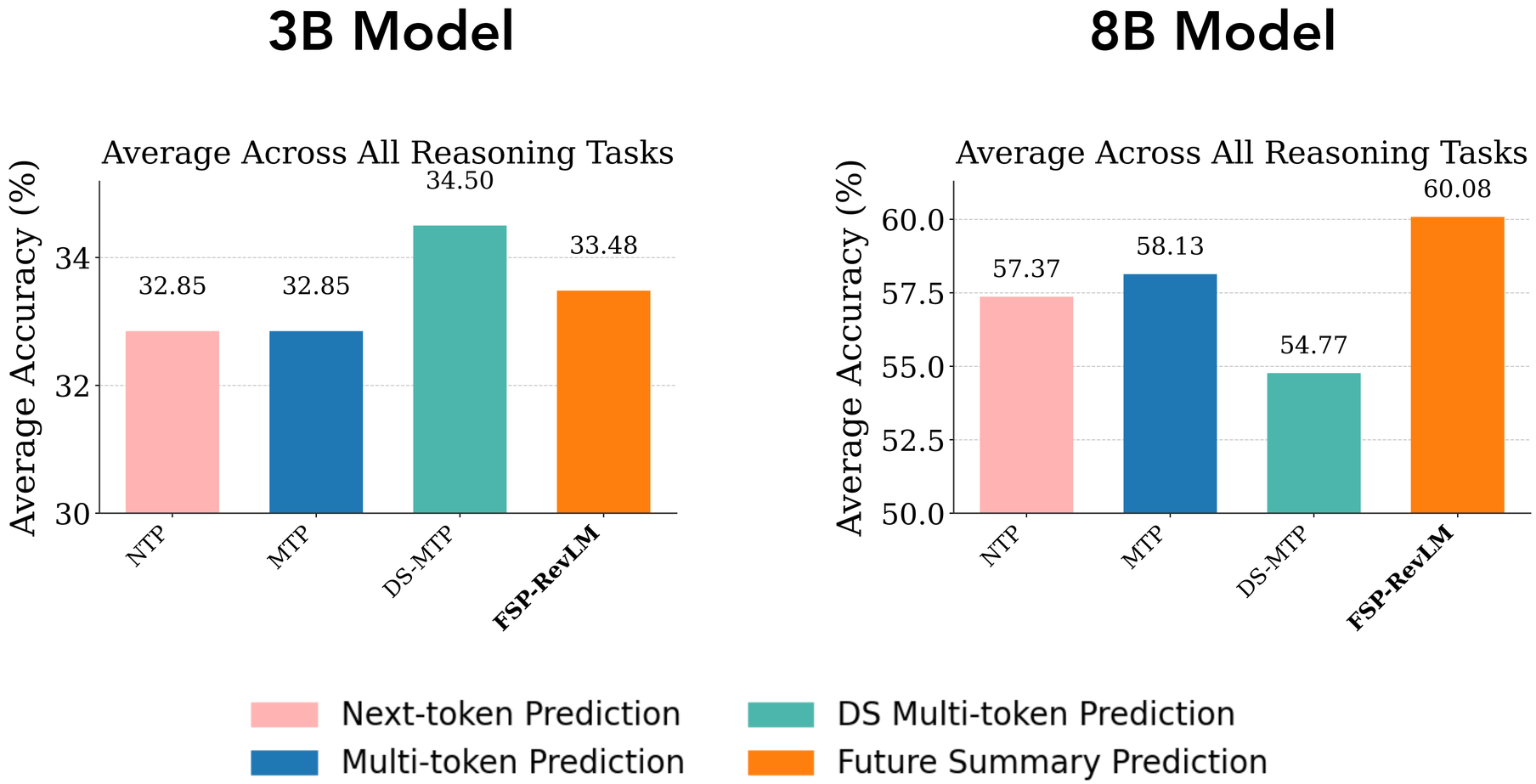
Pretraining Results: 8B

Future Summary Prediction leads to more diversity than MTP



Pretraining Results: 3B vs 8B

Future Summary Prediction benefits from scaling

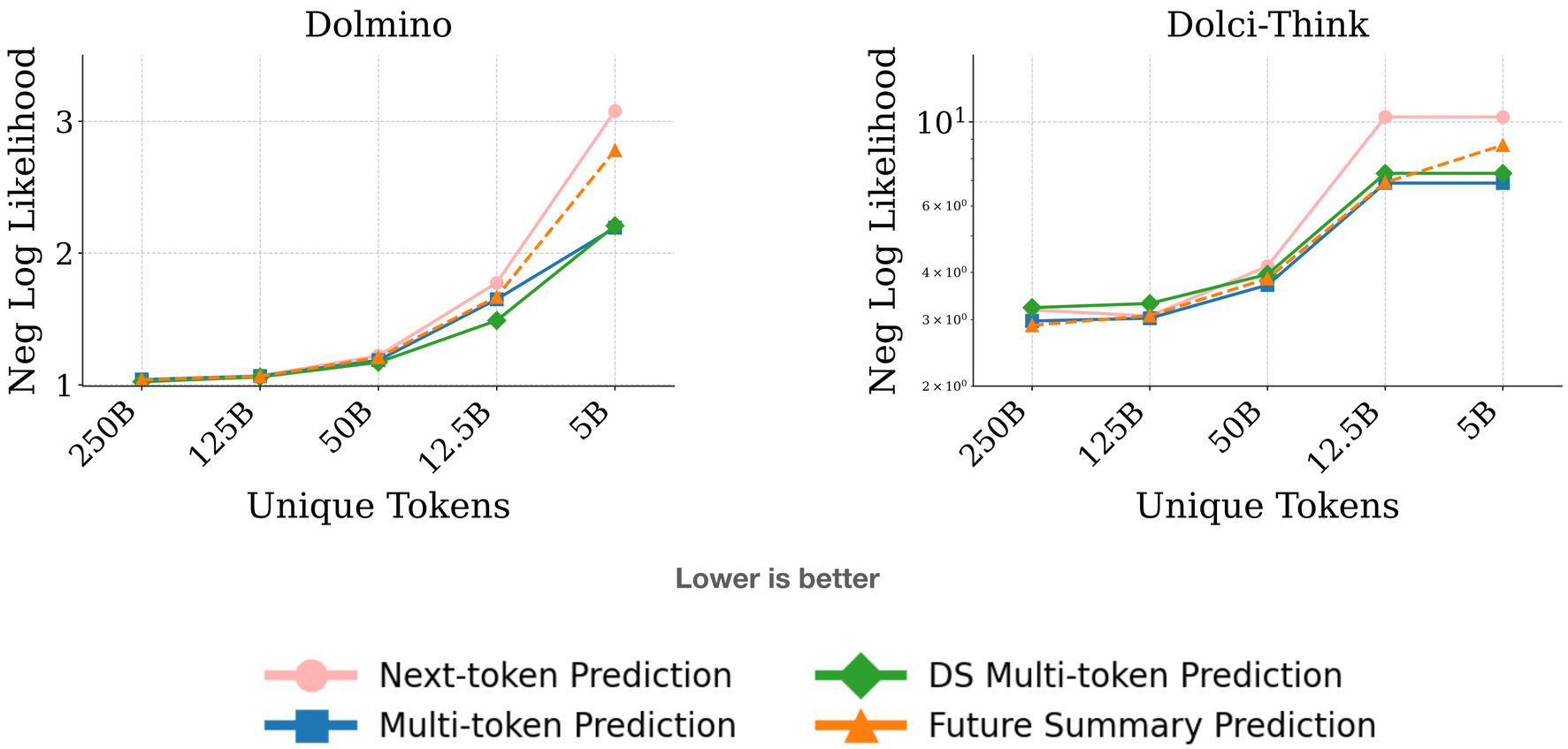


Data Constrained Experiment Setup

- Iso-compute Analysis
  - Reduce unique tokens for training and do multi-epoch training to match compute
- Scale (3B Parameters)
  - 250B unique tokens (1 epoch) upto 5B unique tokens (50 epochs)
- Evaluation Metric
  - Perplexity & Next-token accuracy on validation set (Dolmino, Dolci-Think)

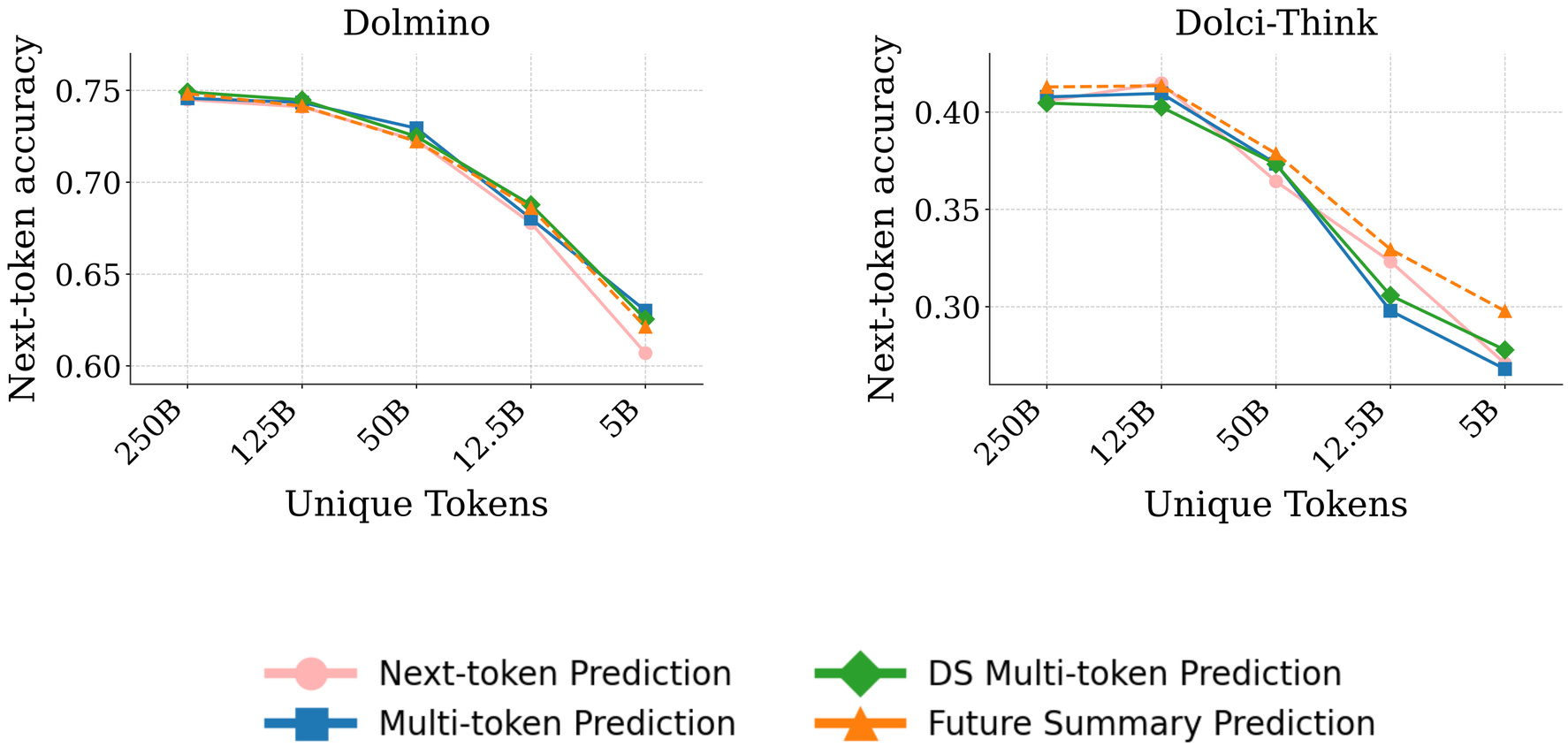
Data Constrained Results: 3B

NTP's perplexity suffers the most as we reduce the total unique tokens



Data Constrained Results: 3B

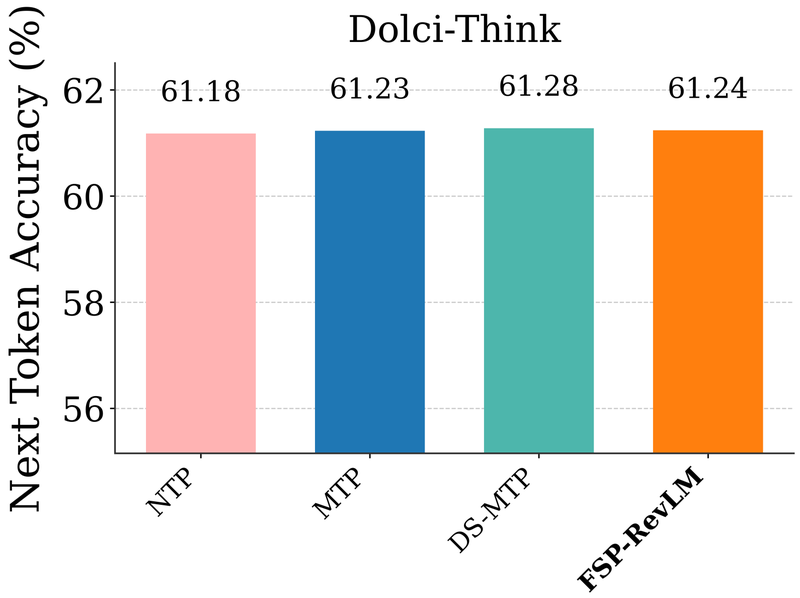
Similar trend with next token accuracy as well!



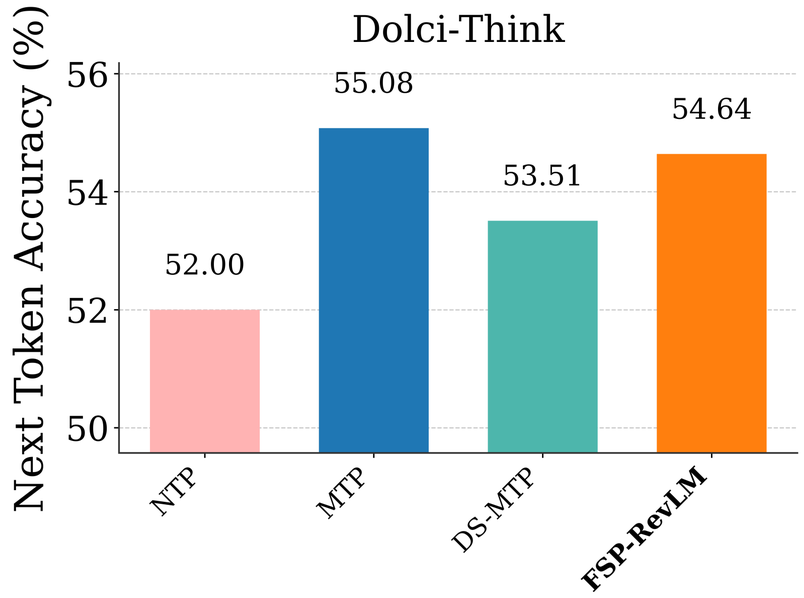
Data Constrained Results: 8B

Future aware methods dominate NTP in the data constrained regime

1T Unique Tokens (1 Epoch)



50B Unique Tokens (20 Epochs)

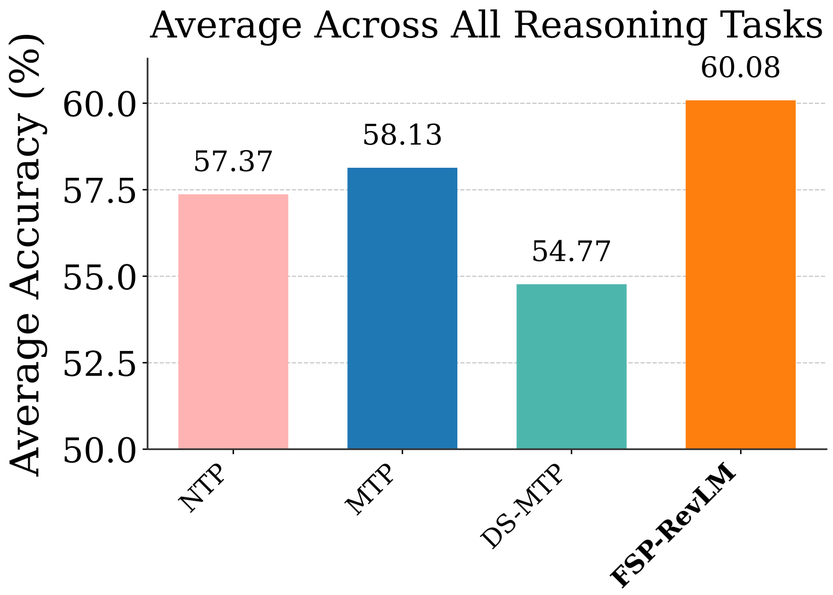


Next-token Prediction
Multi-token Prediction
DS Multi-token Prediction
Future Summary Prediction

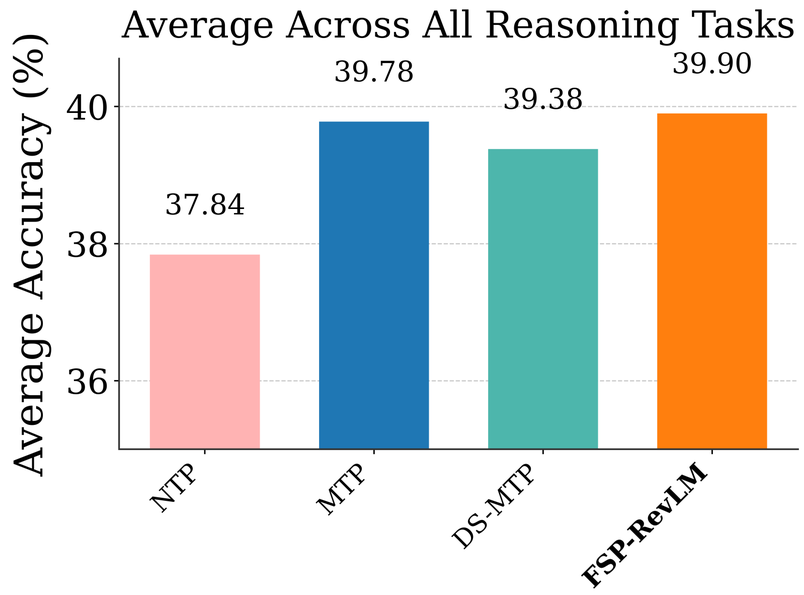
Data Constrained Results: 8B

Future aware methods dominate NTP in the data constrained regime

1T Unique Tokens (1 Epoch)



50B Unique Tokens (20 Epochs)

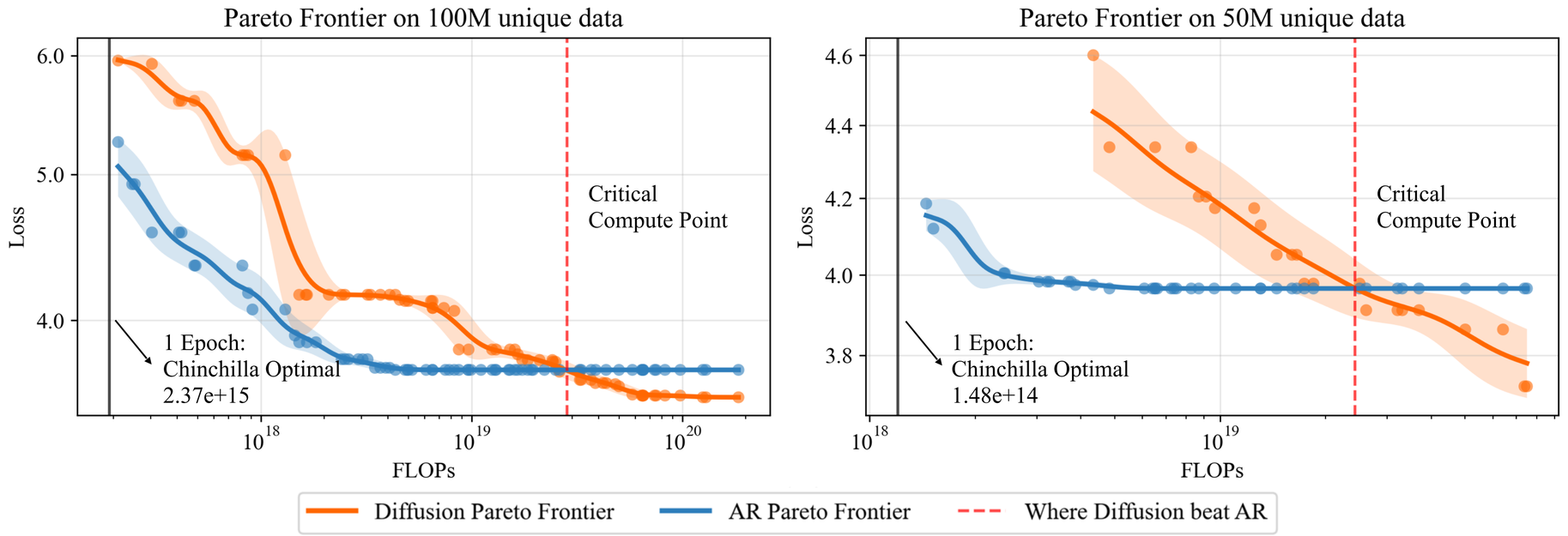


Future Work

Data Constrained Scaling Laws

Future-aware methods versus diffusion

- Should be better than diffusion before the critical compute point
- Can it be competitive with diffusion after the critical compute point?



Diffusion Beats Autoregressive in Data-Constrained Settings— Prabhudesai et al., 2025

Post-training Future-aware Methods

Post-training analysis for future-aware pretrained models

- Analyzing NTP vs future-aware pretrained methods using conventional post-training schemes
- Novel post-training schemes to leverage future-aware pretrained models?

Model	Method	GSM8K			MATH500	AQUA-RAT
		GSM8K	1M-GSM	2M-GSM	1M-MATH	AQUA-RAT
Gemma 2B	`Next-Token`	38.87	66.09	69.02	26.73	40.16
	`Multi-Token`	40.66	66.69	69.02	26.87	38.45
	`MuToR` (ours)	**42.10**	**68.33**	**70.56**	**28.13**	**41.73**
Llama3 8B	`Next-Token`	66.41	85.74	87.33	41.4	-
	`Multi-Token`	66.56	85.67	86.35	42.6	-
	`MuToR` (ours)	**67.85**	**87.03**	**87.64**	**43.2**	-

Multi-token prediction needs registers— Gerontopoulos et al., 2025

Table 1: Model Performances on HumanEval.

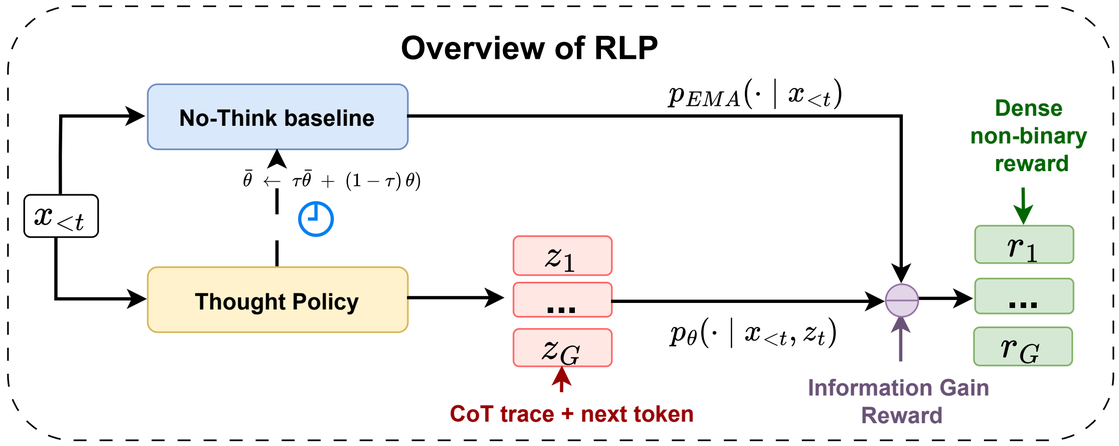
Task	Method		Accuracy (%) ↑
HumanEval	Base		38.9 ±1.501
	LoRA Fine-tuning	Next-token	40.9 ±0.927
		CAFT	**45.1** ±1.930
	Full Fine-tuning	Next-token	40.5 ±2.309
		CAFT	**49.3** ±2.590

Improving LLMs with Concept Aware Fine Tuning— Chen et al., 2025

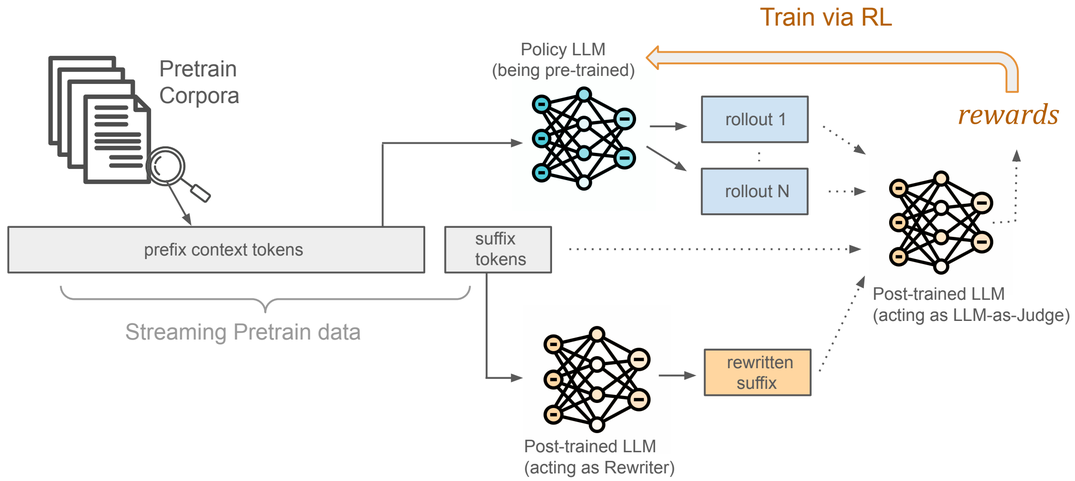
Reasoning during Pretraining

Pretrain like your fine-tune

- Designing reward functions for pretraining?



Reinforcement a pretraining objective— Hatamizadeh et al., 2025



Self-improving pretraining— Tan et al., 2026

Thank You!