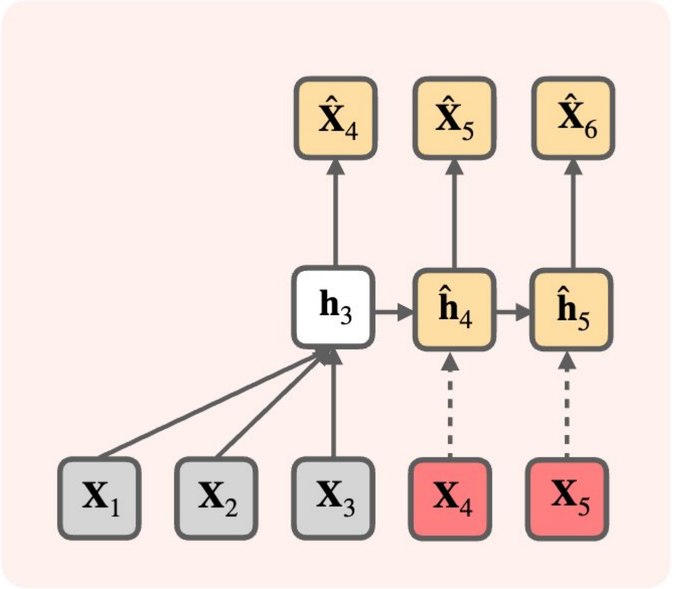
Microsoft

ML Collective

Next-Latent Prediction Transformers



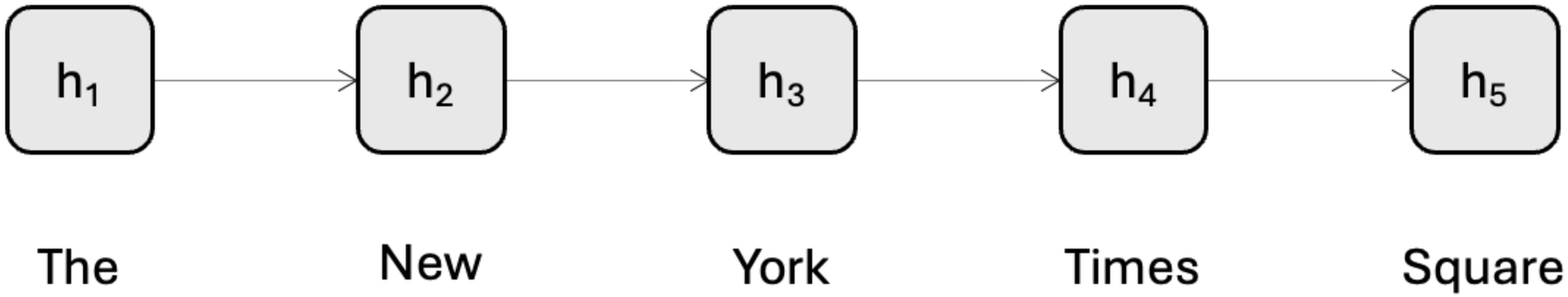
Microsoft Research New York

Jayden Teoh, Manan Tomar, Kwangjun Ahn, Edward S. Hu, Tim Pearce, Pratyusha Sharma, Akshay Krishnamurthy, Riashat Islam, Alex Lamb, John Langford

Introduction

Microsoft

- Recurrent neural networks (RNNs) naturally enforce compact representations
- But their training is not parallelizable…



Introduction

- Transformers replaced RNNs
- Self-attention enables **flexible look ups** over past tokens
- But there's no incentive to learn compact summaries of past tokens…

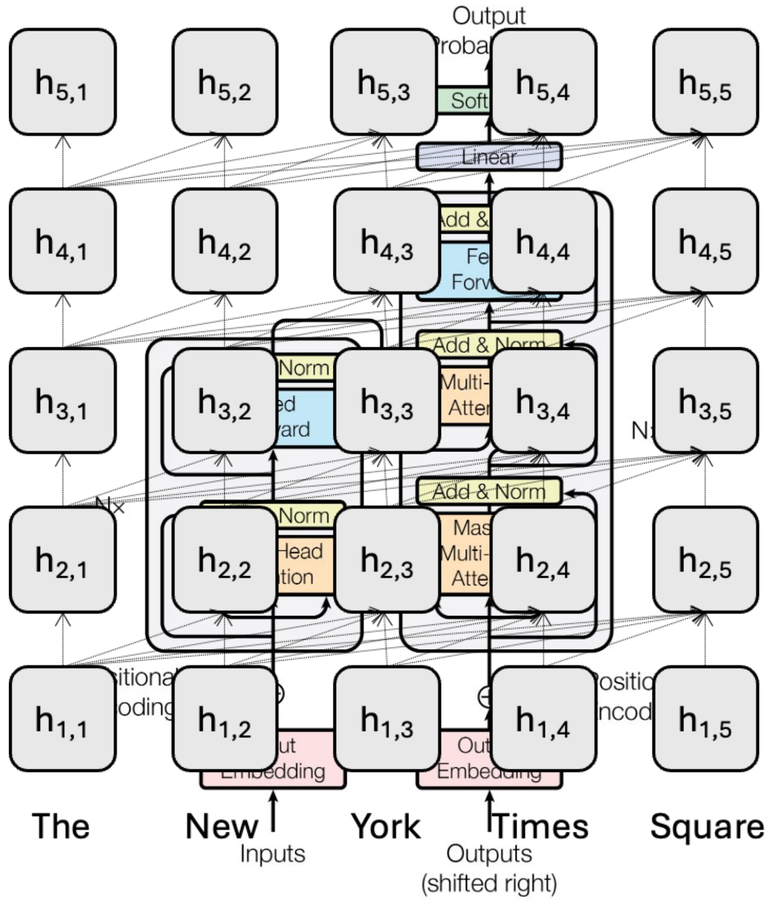
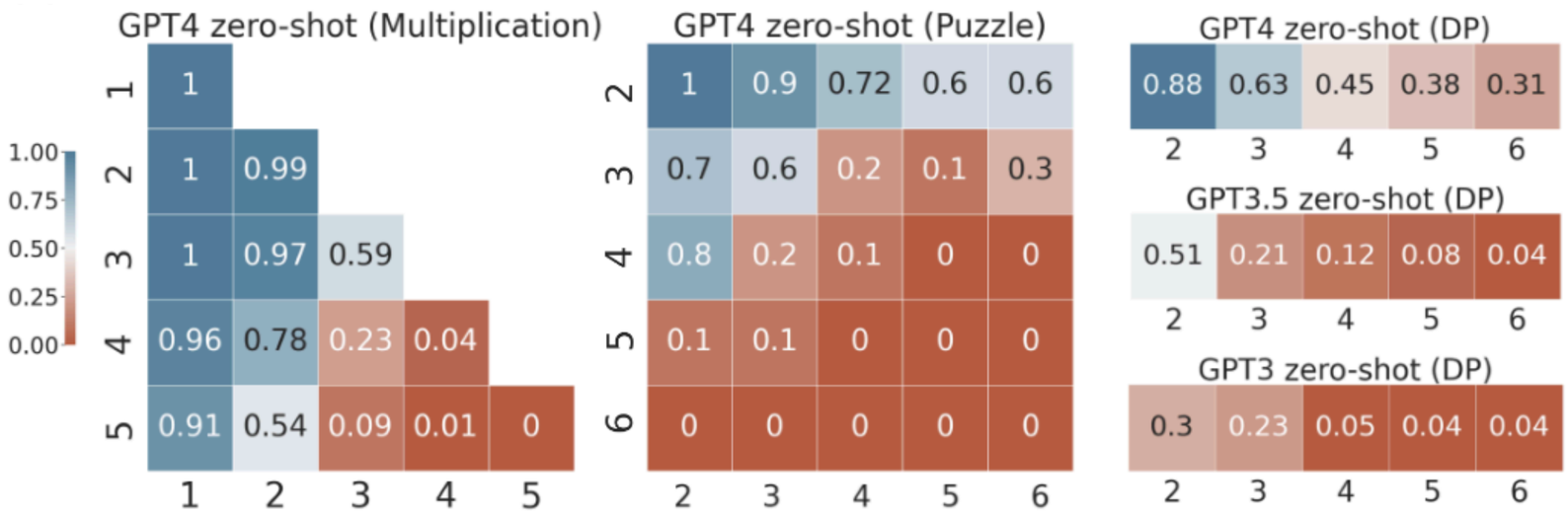


Figure 1: The Transformer - model architecture.

Introduction

Microsoft

- As a result, transformers often learn shortcut solutions that don't generalize [1, 2, 3, 4]



GPT3 performance degrades on compositional tasks

[1] Anil et al. Exploring Length Generalization in Large Language Models. NeurIPS 2022.

[2] Liu et al. Transformers Learn Shortcuts to Automata. ICLR 2023.

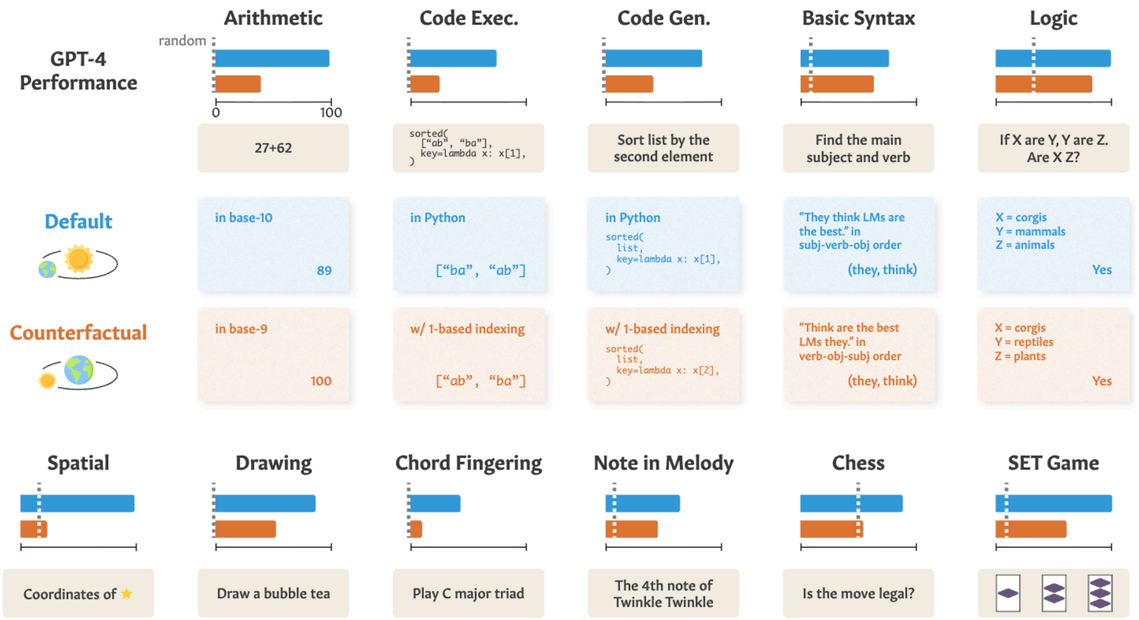
[3] Dziri et al. Faith and Fate: Limits of Transformers on Compositionality. NeurIPS 2023.

[4] Wu et al. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks. NAACL 2024.

Introduction

Microsoft

- As a result, transformers often learn shortcut solutions that don't generalize [1, 2, 3, 4]



GPT3 performance degrades on counterfactual tasks

[1] Anil et al. Exploring Length Generalization in Large Language Models. NeurIPS 2022.

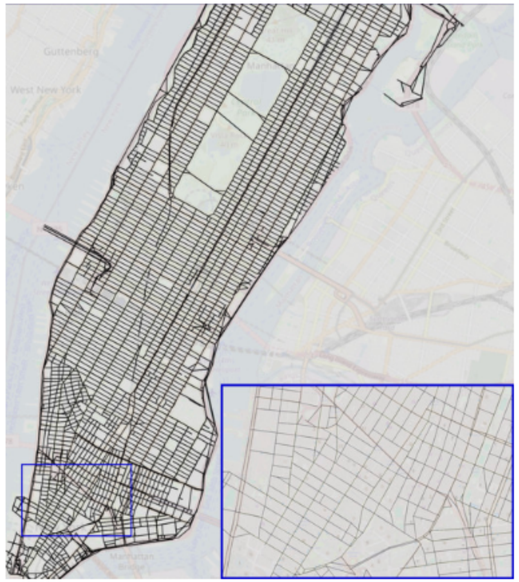
[2] Liu et al. Transformers Learn Shortcuts to Automata. ICLR 2023.

[3] Dziri et al. Faith and Fate: Limits of Transformers on Compositionality. NeurIPS 2023.

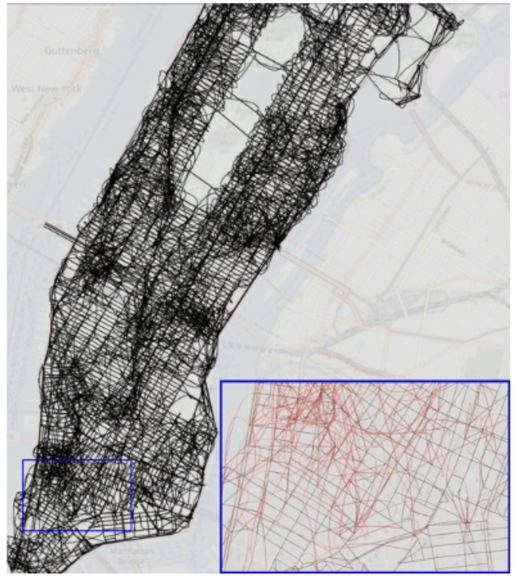
[4] Wu et al. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks. NAACL 2024.

Introduction

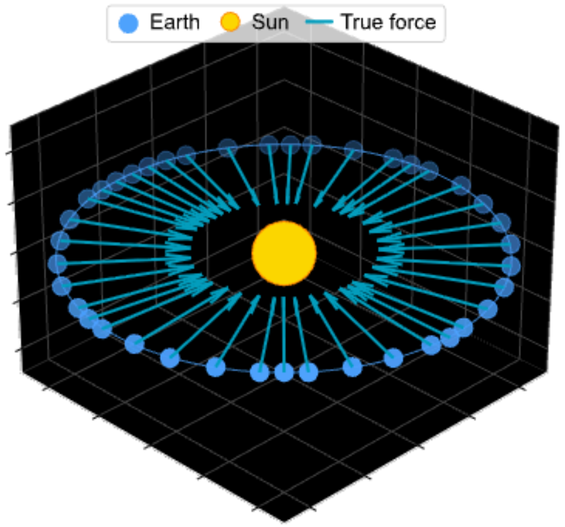
- Transformers also learn poor world models [5, 6]



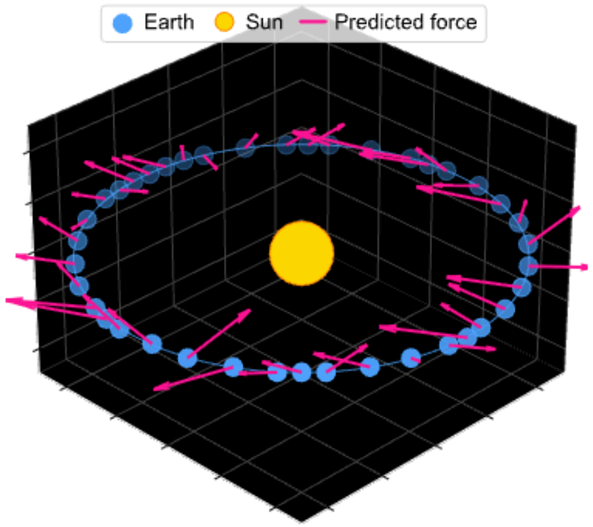
Manhattan (real)



Manhattan (transformer)



Earth (real)



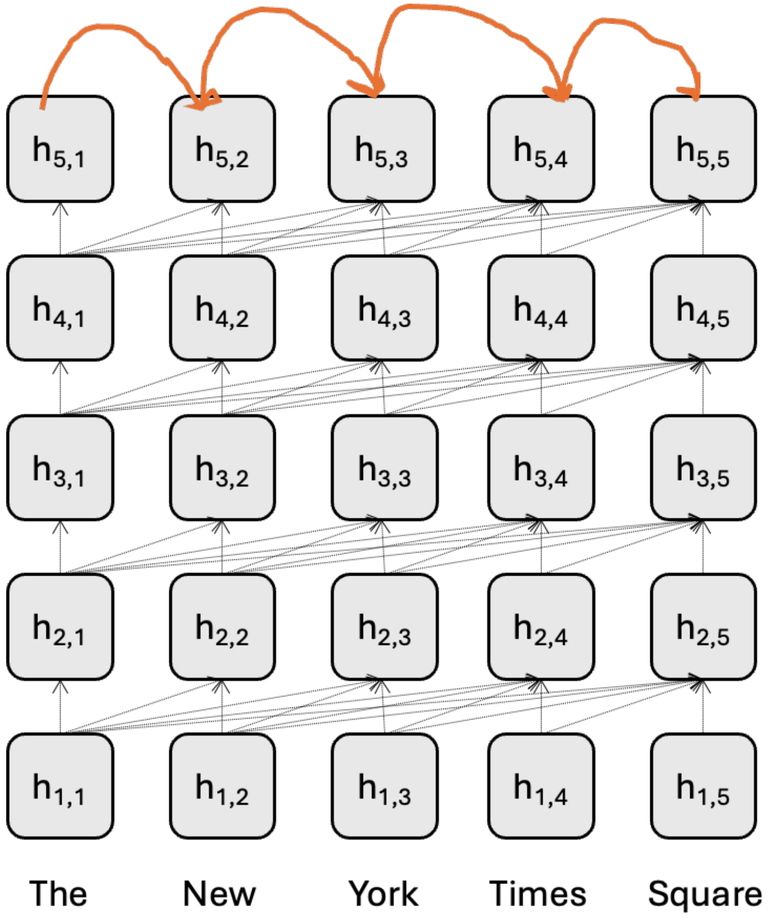
Earth (transformer)

[5] Vafa et al. Evaluating the World Model Implicit in a Generative Model. NeurIPS 2024.
[6] Vafa et al. What Has a Foundation Model Found? Using Inductive Bias to Probe for World Models. ICML 2025.

Methodology

Microsoft

- **Motivation #1:** we want recurrent-like dynamics and compression in transformers
- **Motivation #2:** we want to retain parallel training

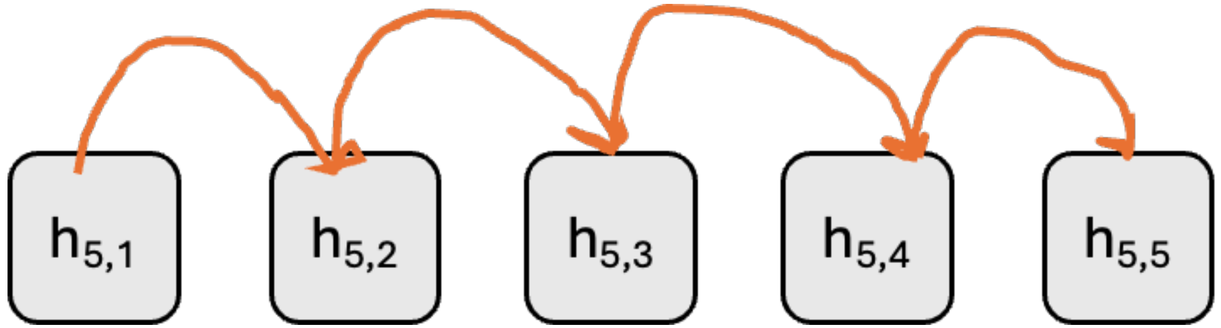


Methodology

Microsoft

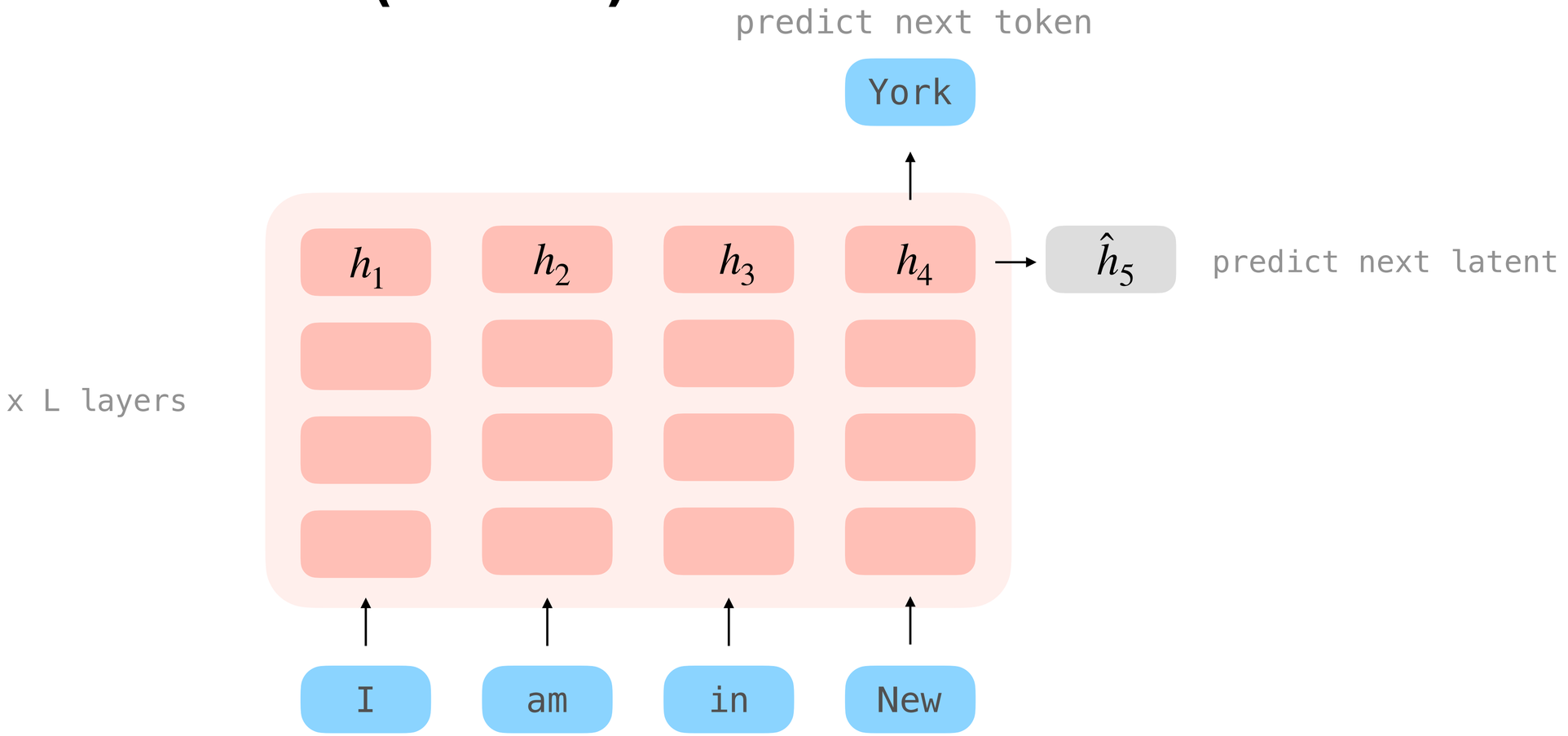
Next-Latent Prediction (NextLat)

- We train the transformer such that each of its latent state can predict the next latent state (conditioned on the next token)



Methodology

Next-Latent Prediction (NextLat)



Methodology

Microsoft

Next-Latent Prediction (NextLat)

1) next-token cross entropy loss

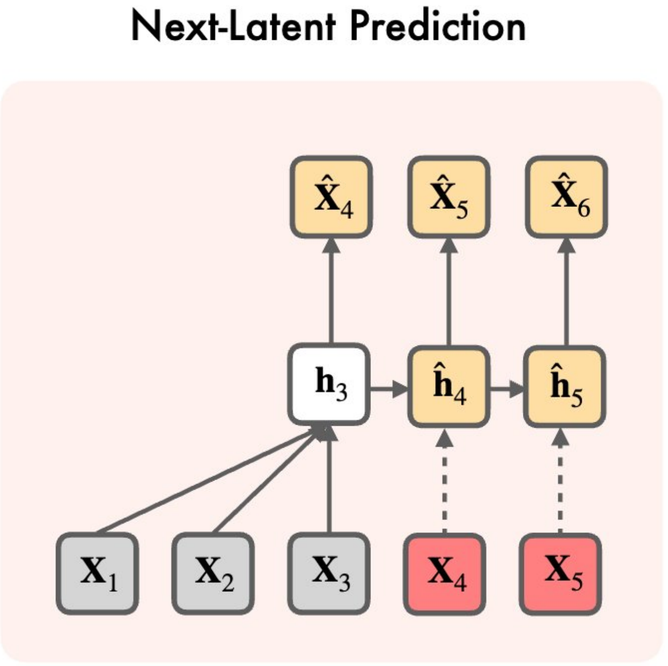
$$\mathcal{L}_{\text{next-token}} = -\log p_\theta(X_{t+1} \mid \mathbf{h}_t)$$

2) next-latent regression loss

$$\mathcal{L}_{\text{next-h}} = \|\hat{\mathbf{h}}_{t+1} - \text{sg}[\mathbf{h}_{t+1}]\|_2^2$$

3) *Optionally*: next-latent token KL (or cross entropy)

$$\mathcal{L}_{\text{KL}} = D_{\text{KL}}\left(p_\theta^{\text{sg}}(\cdot \mid \text{sg}[\mathbf{h}_{t+1}]) \parallel p_\theta^{\text{sg}}(\cdot \mid \text{sg}[\hat{\mathbf{h}}_{t+1}])\right)$$



Motivation

Microsoft

Why NextLat?

1) **Stronger** representations for world modeling, reasoning, and planning

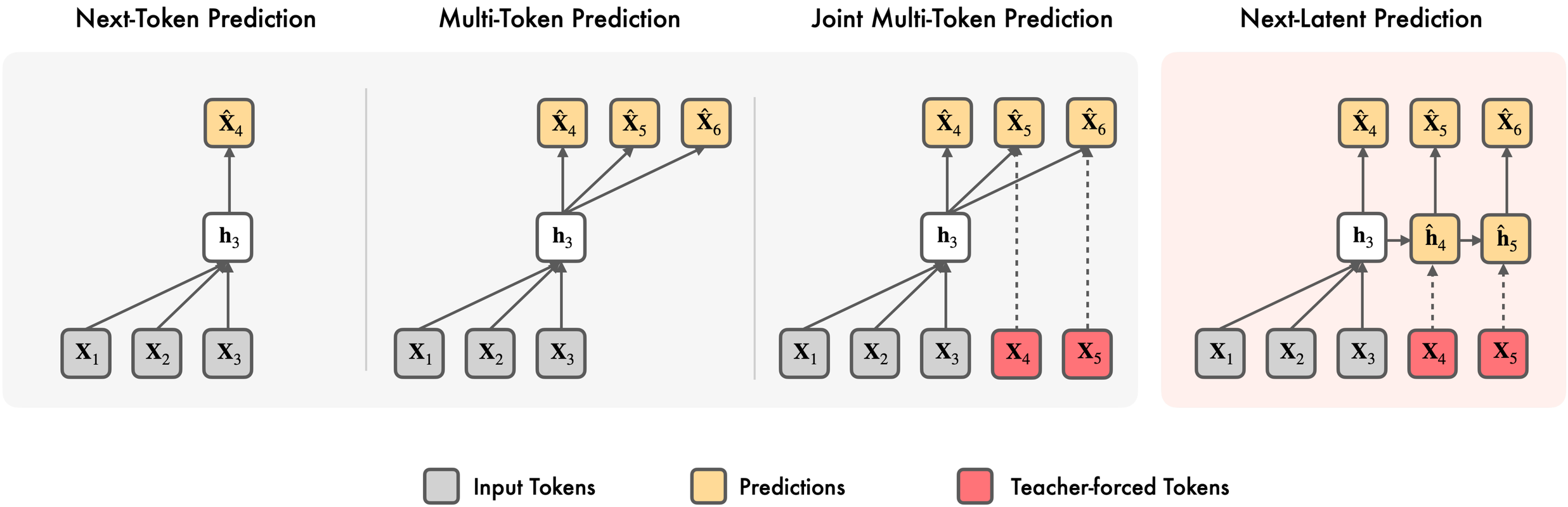
2) **Faster** inference than Multi-Token Prediction

3) **Better** data-efficiency

Motivation

Microsoft

Why NextLat?



Better Representations

- Train models on trajectories from Manhattan taxi rides

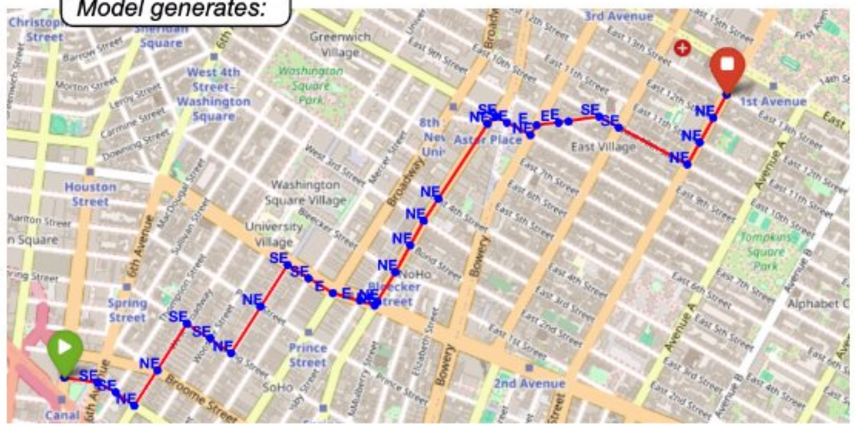
Training sequences

820 210 N E E E SE W W N SE N end
193 450 E E E N E S E end
301 592 N N N SE S S S S S end
…

Contexts for evaluation

110 244
850 820
592 301
…

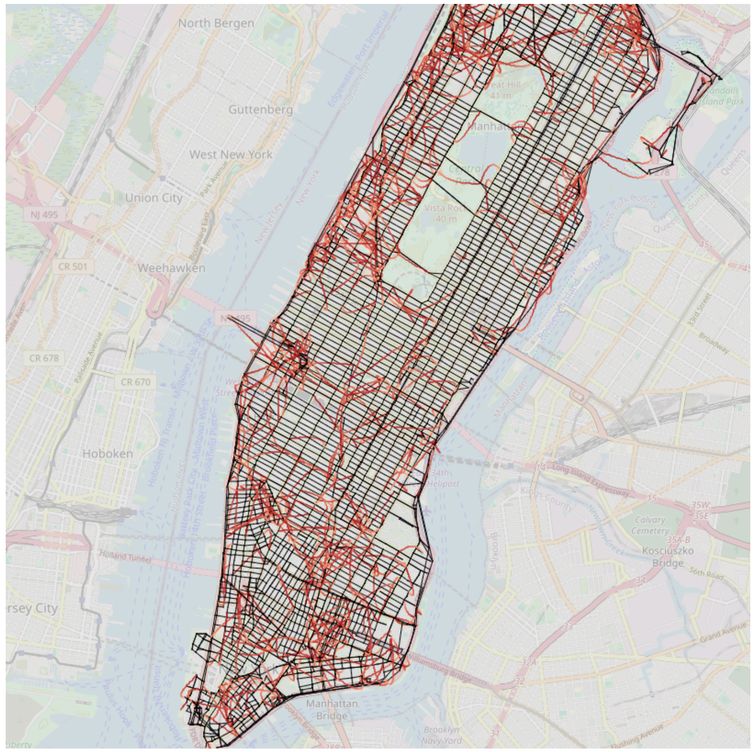
Model generates:



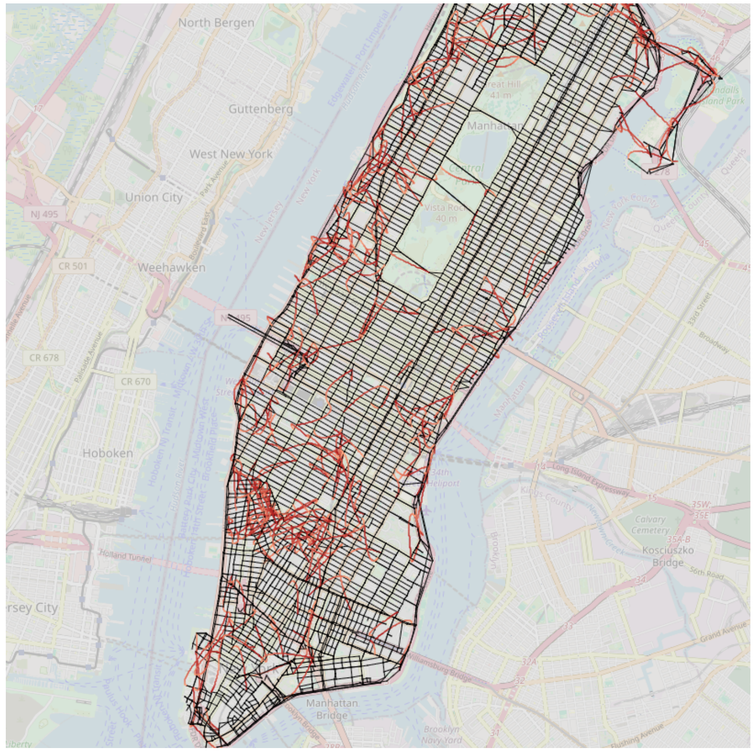
[5] Vafa et al. Evaluating the World Model Implicit in a Generative Model. NeurIPS 2024.

World Modeling

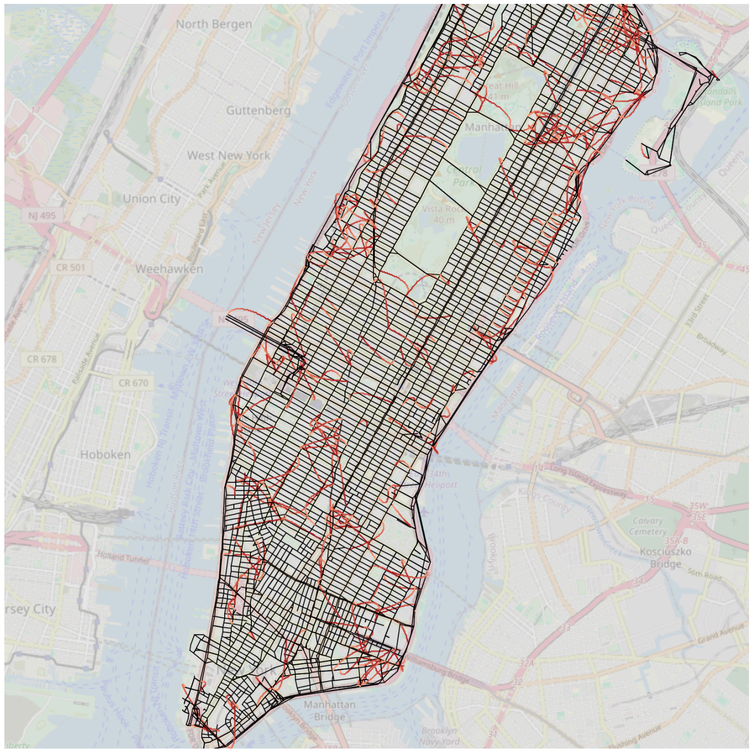
Microsoft



Next-Token Prediction



Multi-Token Prediction



Next-Latent Prediction

World Modeling

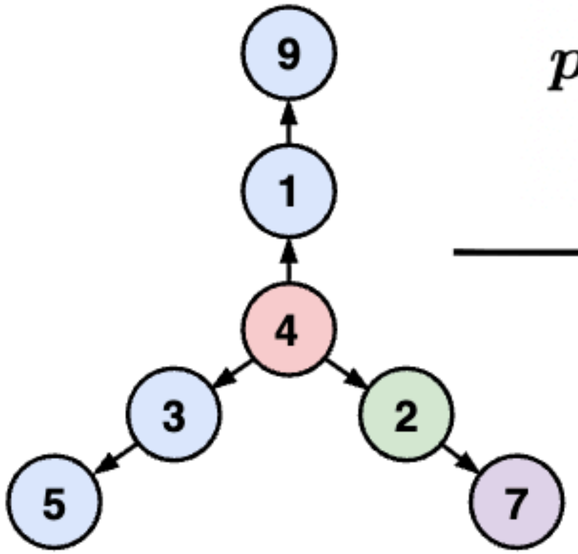
Microsoft

	Next-Token Test (↑)	Valid Trajectories (↑)	Sequence Compression (↑)	Effective Latent Rank (↓)	Detour Robustness (↑)
GPT	100%	97.0%	0.65	160.1	85.0%
MTP	100%	98.1%	0.64	57.7	**95.0%**
JTP	100%	97.1%	0.32	215.8	87.0%
NextLat	100%	**98.7%**	**0.71**	**52.7**	**95.0%**
True world model	100%	100%	1.00	—	100%

Table 1: Comparison of GPT, MTP, JTP, and NextLat trained on Manhattan taxi rides against the true world model across several metrics.

Planning

- Path-Star Graph [7]. Simple planning task, yet next-token prediction can't solve it



$$\boldsymbol{p} = \text{“}\ \underbrace{2,7 \mid 4,1 \mid 4,3 \mid 1,9 \mid 3,5 \mid 4,2 \mid}_{\text{edge list}}\ /\ \underbrace{4,7}_{\text{start/goal}}\ = \text{”}$$

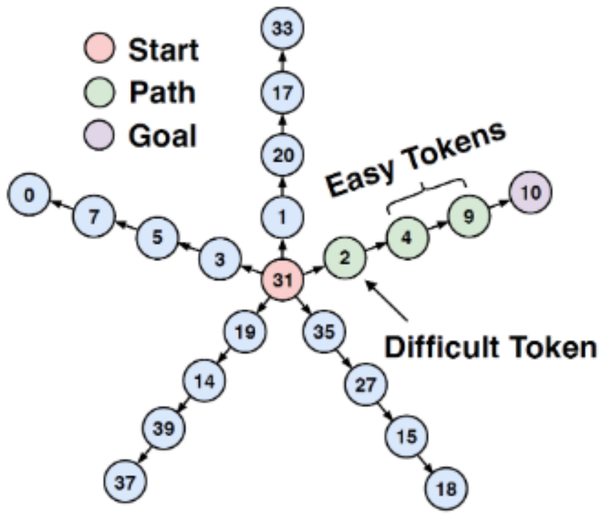
	$G_{2,5}$	$G_{2,20}$	$G_{5,5}$	$G_{10,5}$	$G_{20,5}$
GPT2-LARGE	50.2	50.4	18.9	10.4	4.5

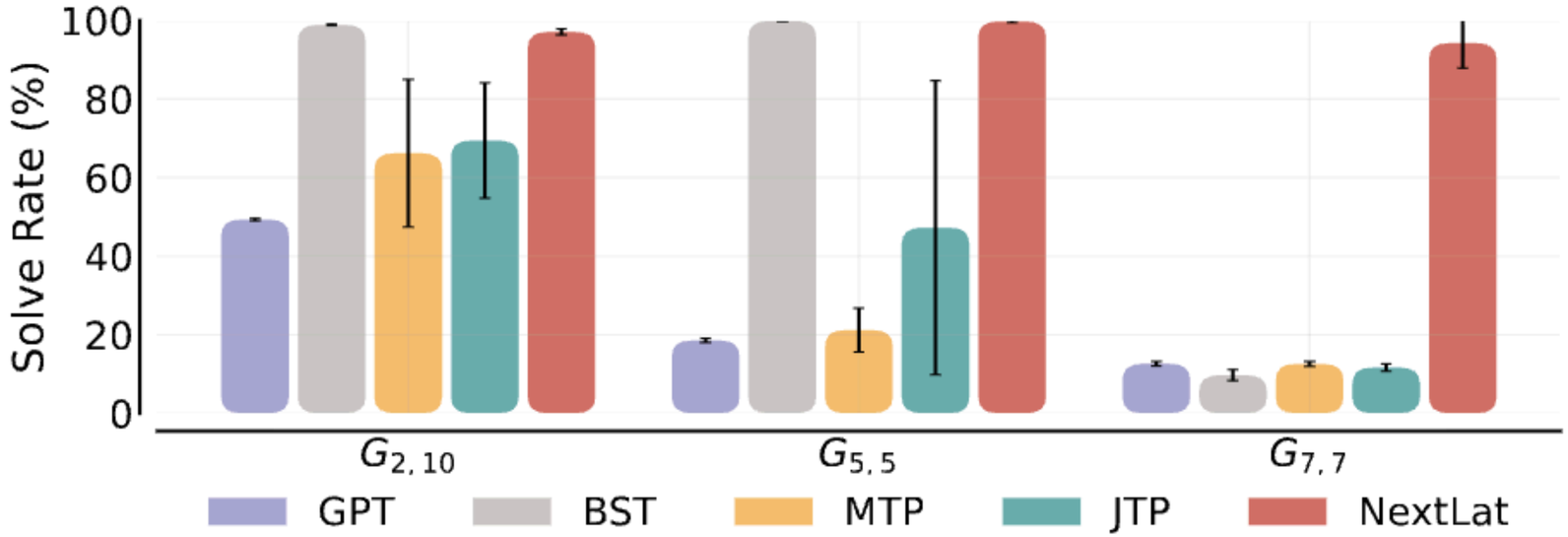
[7] Bachmann et al. The pitfalls of next-token prediction. ICML 2024.

Planning

Microsoft

- Only NextLat can solve all graphs





Reasoning

Microsoft

- Countdown Game
  - GPT-4 only gets 4% on this task!

Input: <14, 83, 88, 91>

Goal: 23

Solution: 83 - 14 = 69, 91 - 88 = 3, 69 ÷ 3 = 23

Reasoning

Microsoft

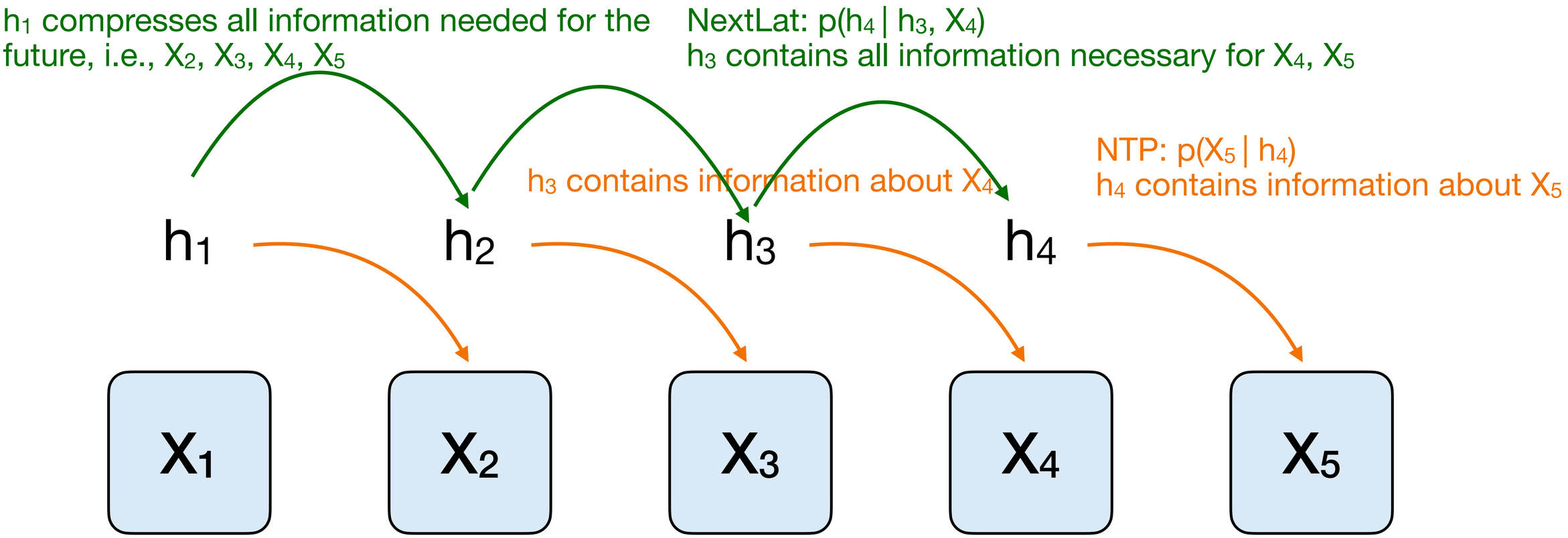
- NextLat has incredible performance just by predicting next-latent
  - MTP methods have to learn to predict multiple tokens ahead to catch up

Model	Horizon (d)	Accuracy (%)
GPT	–	33.1
BST	–	42.3
MTP	1	39.2
	4	49.7
	8	57.3
JTP	1	39.0
	4	49.4
	8	55.0
NextLat	1	54.8
	4	<u>57.6</u>
	8	**58.7**

Figure 5: Performance on Countdown. Best result is **bolded**, and second best is <u>underlined</u>.

Stronger Representations

Microsoft



Every ht is a compressed summary of the history for decoding future tokens!

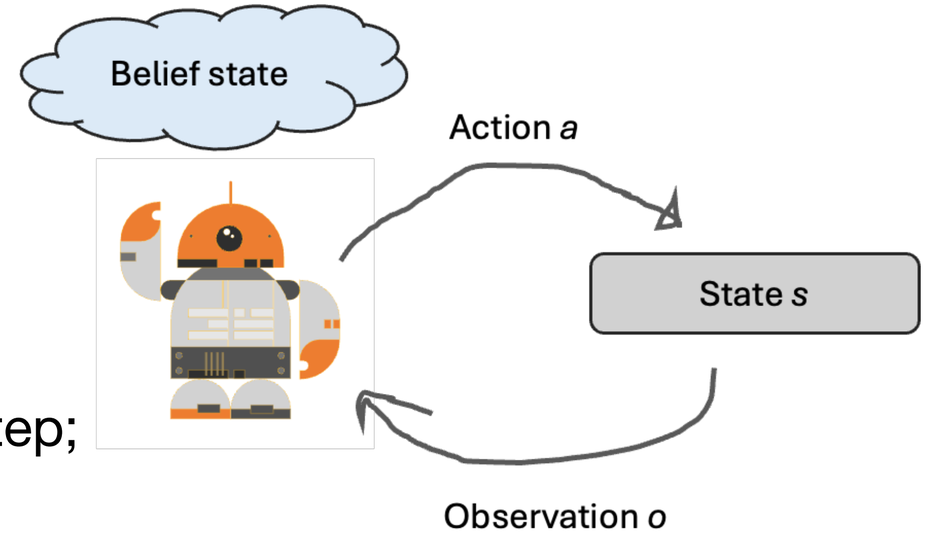
(not a property with just next-token or multi-token prediction)

Motivation

Belief States

- At convergence, the transformer's hidden state will become *belief states*

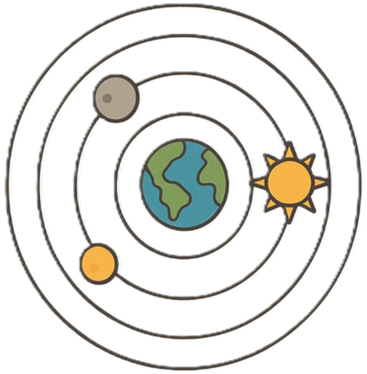
- **Intuition:** force transformers to compress history at every step; stop relying on lazy self-attention



Motivation

Microsoft

Compact World Models

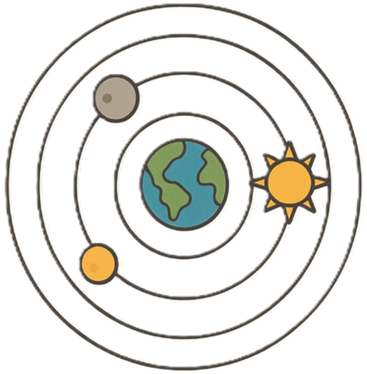


- In the 2nd century, Claudius Ptolemy proposed a geocentric model that placed Earth at the center of the universe

Motivation

Microsoft

Compact World Models

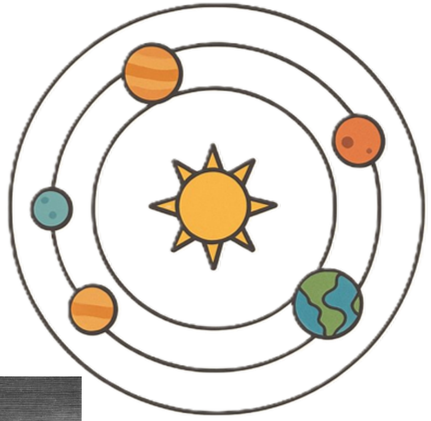


- Clearly, a wrong **overly-complicated** model of the world...
- Yet, it was the dominant theory for 1500 years…

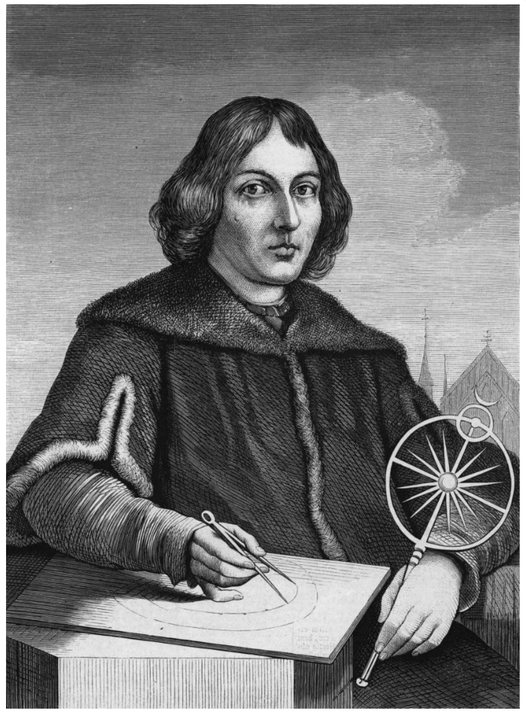
Motivation

Microsoft

Compact World Models



- Copernicus proposed the Heliocentric Model
- **Simpler, more compact** model of the world

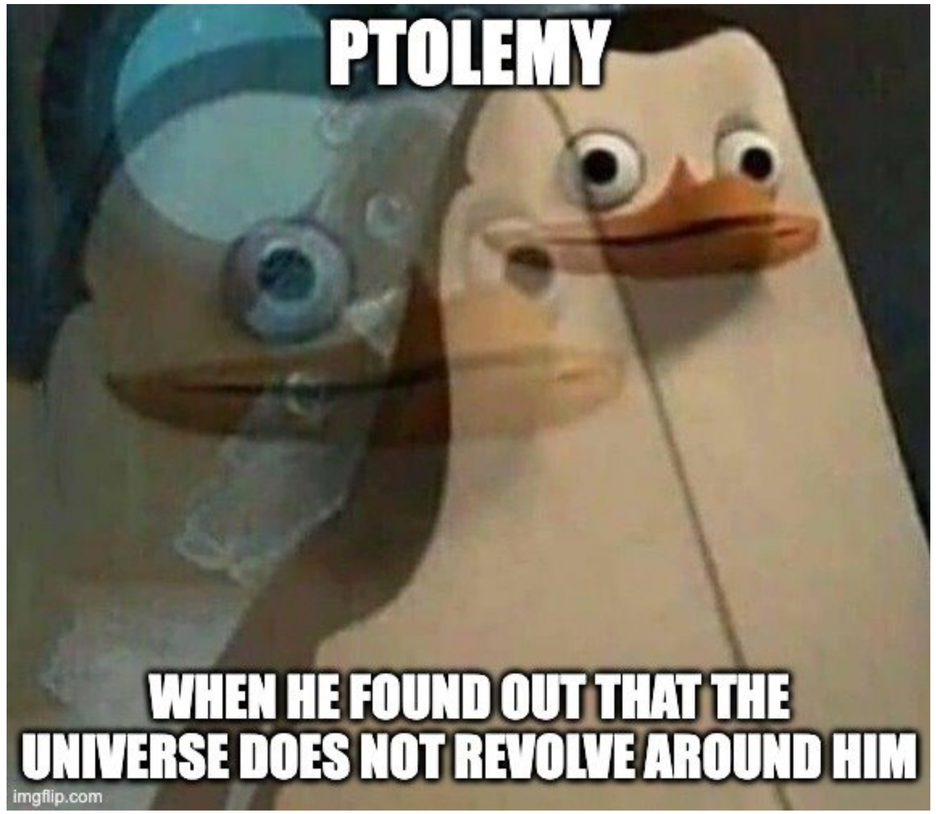


Motivation

Microsoft

Compact World Models

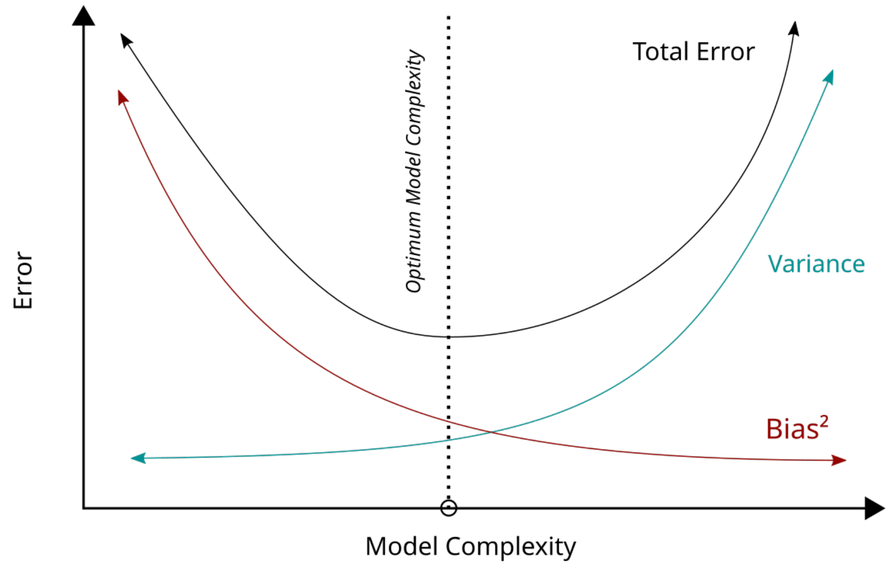
- Copernicus proposed the Heliocentric Model
- **Simpler, more compact** model of the world
- Can **generalize** to explain observations from other planets' viewpoints



Motivation

Compact World Models

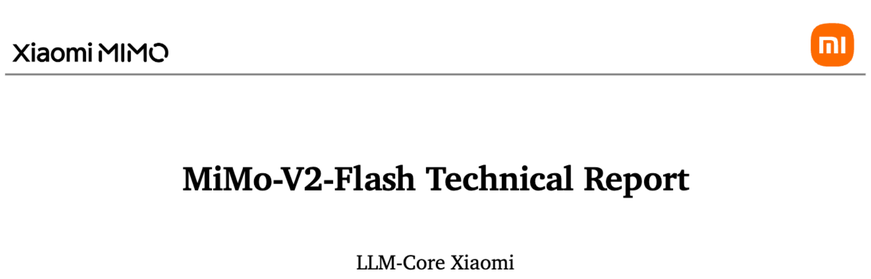
- NextLat teaches transformer to form **compact world models!**

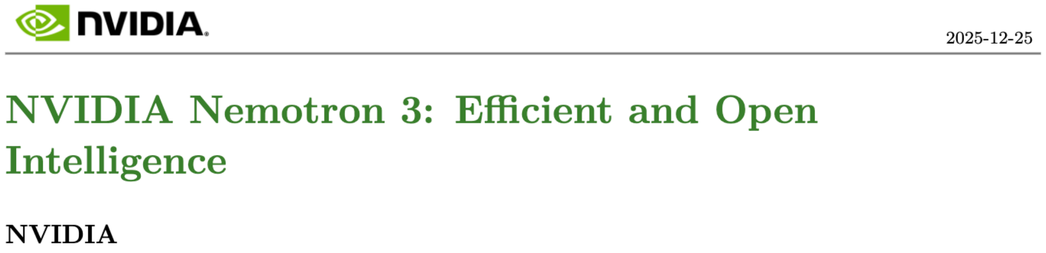


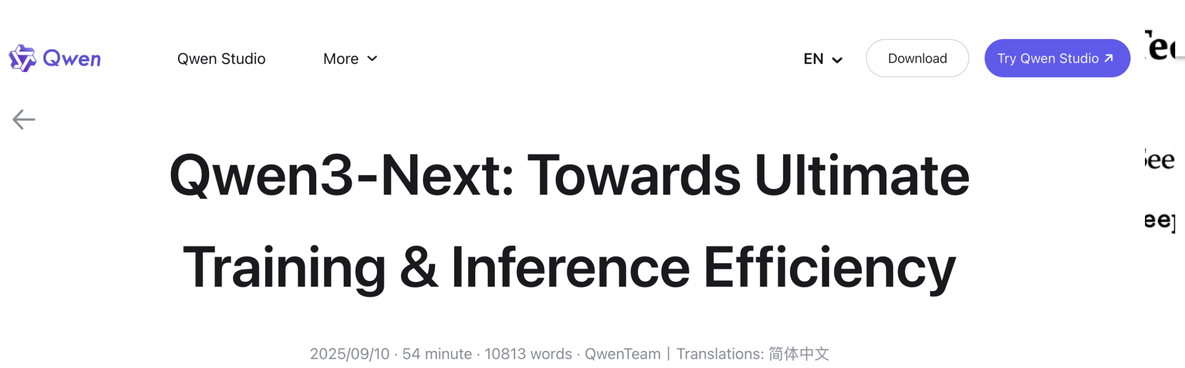
Language Modeling

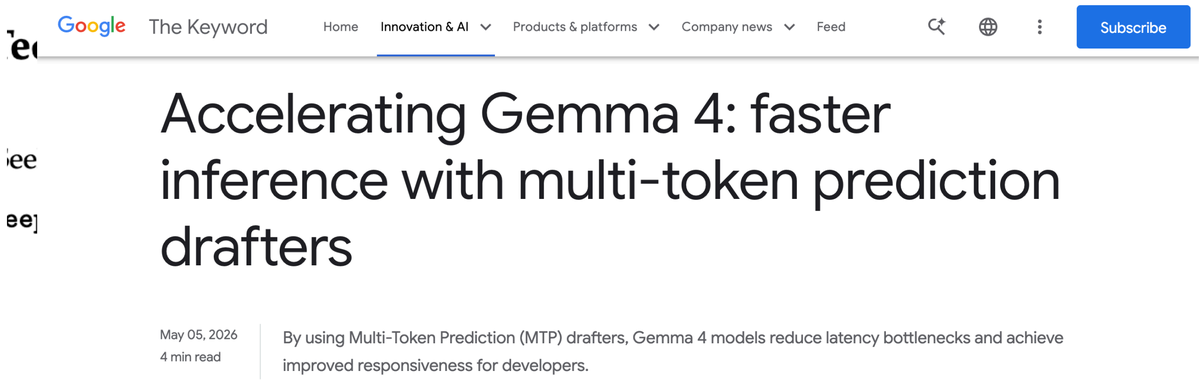
Microsoft

- **Multi-token prediction** has become a staple in open-source model pretraining









Language Modeling

Microsoft

- Reasons for **MTP**
  - Self-speculative decoding: speeds inference up to 3x
  - Better benchmark performance and post-training
  - Data-efficiency

Motivation

Microsoft

Language Modeling

Microsoft

- Pretrain 1.3 billion parameter models on FineWeb-Edu dataset

The FineWeb Datasets: Decanting the Web for the Finest Text Data at Scale

Guilherme Penedo **Hynek Kydlíček** **Loubna Ben allal** **Anton Lozhkov**
Margaret Mitchell **Colin Raffel** **Leandro Von Werra** **Thomas Wolf**
🤗 Hugging Face

Language Modeling

Microsoft

Model	FW-Edu ppl ↓	Wiki. ppl ↓	LAMB. ppl ↓	LAMB. acc ↑	PIQA acc ↑	HellaS. acc ↑	Wino. acc ↑	ARC-e acc ↑	ARC-c acc ↑	SIQA acc ↑	SciQ acc ↑	Avg.
GPT	**10.52**	**17.93**	20.26	42.07	73.45	**58.79**	60.46	68.18	39.16	42.32	86.10	58.82
JTP (d=1)	11.08	19.28	21.88	41.35	**74.92**	57.43	58.64	68.73	39.25	42.99	87.30	58.83
JTP (d=2)	11.18	19.60	22.11	41.37	73.34	56.84	59.98	68.86	38.57	**43.35**	86.70	58.63
MTP (d=1)	10.90	18.82	20.23	41.26	74.32	58.05	**60.54**	68.52	38.91	42.84	85.40	58.76
MTP (d=2)	11.00	18.61	18.34	43.43	72.80	57.92	59.35	68.35	39.08	41.97	86.60	58.69
NextLat (d=1)	10.83	18.39	19.77	41.08	73.07	58.35	59.27	69.65	39.68	43.24	86.00	58.79
NextLat (d=2)	10.88	18.44	**17.83**	**43.86**	73.61	57.79	59.20	**69.74**	**40.10**	41.91	**87.50**	**59.21**

Table 2: Downstream language modeling evaluation on 1.3B-parameter models trained on 100B FineWeb-Edu tokens. Best scores are in bold and second-best are underlined.

Best performance in downstream accuracy but small win at small scale… consistent with observations in Gloeckle et al. [2024]

Language Modeling

Microsoft

Model	FW-Edu ppl ↓	Wiki. ppl ↓	LAMB. ppl ↓	LAMB. acc ↑	PIQA acc ↑	HellaS. acc ↑	Wino. acc ↑	ARC-e acc ↑	ARC-c acc ↑	SIQA acc ↑	SciQ acc ↑	Avg.
GPT	**10.52**	**17.93**	20.26	42.07	73.45	**58.79**	60.46	68.18	39.16	42.32	86.10	58.82
JTP (d=1)	11.08	19.28	21.88	41.35	**74.92**	57.43	58.64	68.73	39.25	42.99	87.30	58.83
JTP (d=2)	11.18	19.60	22.11	41.37	73.34	56.84	59.98	68.86	38.57	**43.35**	86.70	58.63
MTP (d=1)	10.90	18.82	20.23	41.26	74.32	58.05	**60.54**	68.52	38.91	42.84	85.40	58.76
MTP (d=2)	11.00	18.61	18.34	43.43	72.80	57.92	59.35	68.35	39.08	41.97	86.60	58.69
NextLat (d=1)	10.83	18.39	19.77	41.08	73.07	58.35	59.27	69.65	39.68	43.24	86.00	58.79
NextLat (d=2)	10.88	18.44	**17.83**	**43.86**	73.61	57.79	59.20	**69.74**	**40.10**	41.91	**87.50**	**59.21**

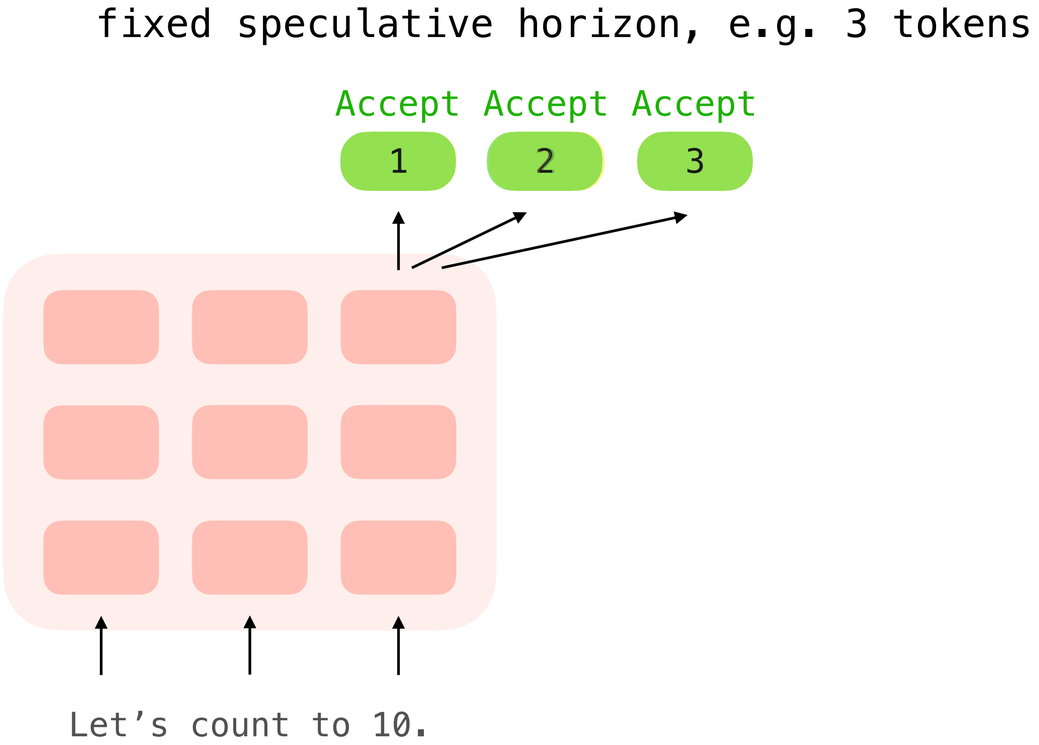
Table 2: Downstream language modeling evaluation on 1.3B-parameter models trained on 100B FineWeb-Edu tokens. Best scores are in bold and second-best are underlined.

NextLat preserves next-token prediction performance better than MTP methods!

Speculative Decoding

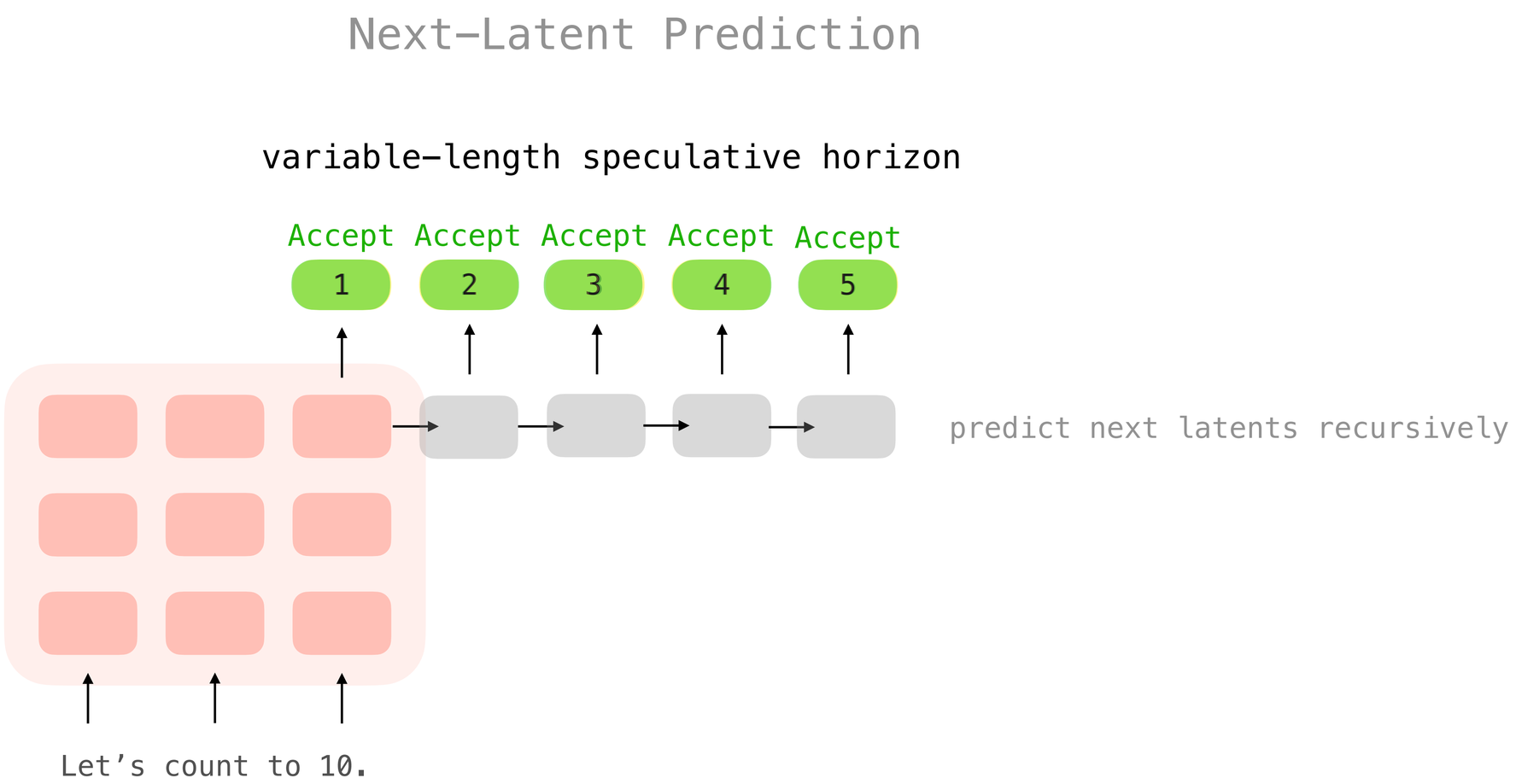
Microsoft

Multi-Token Prediction



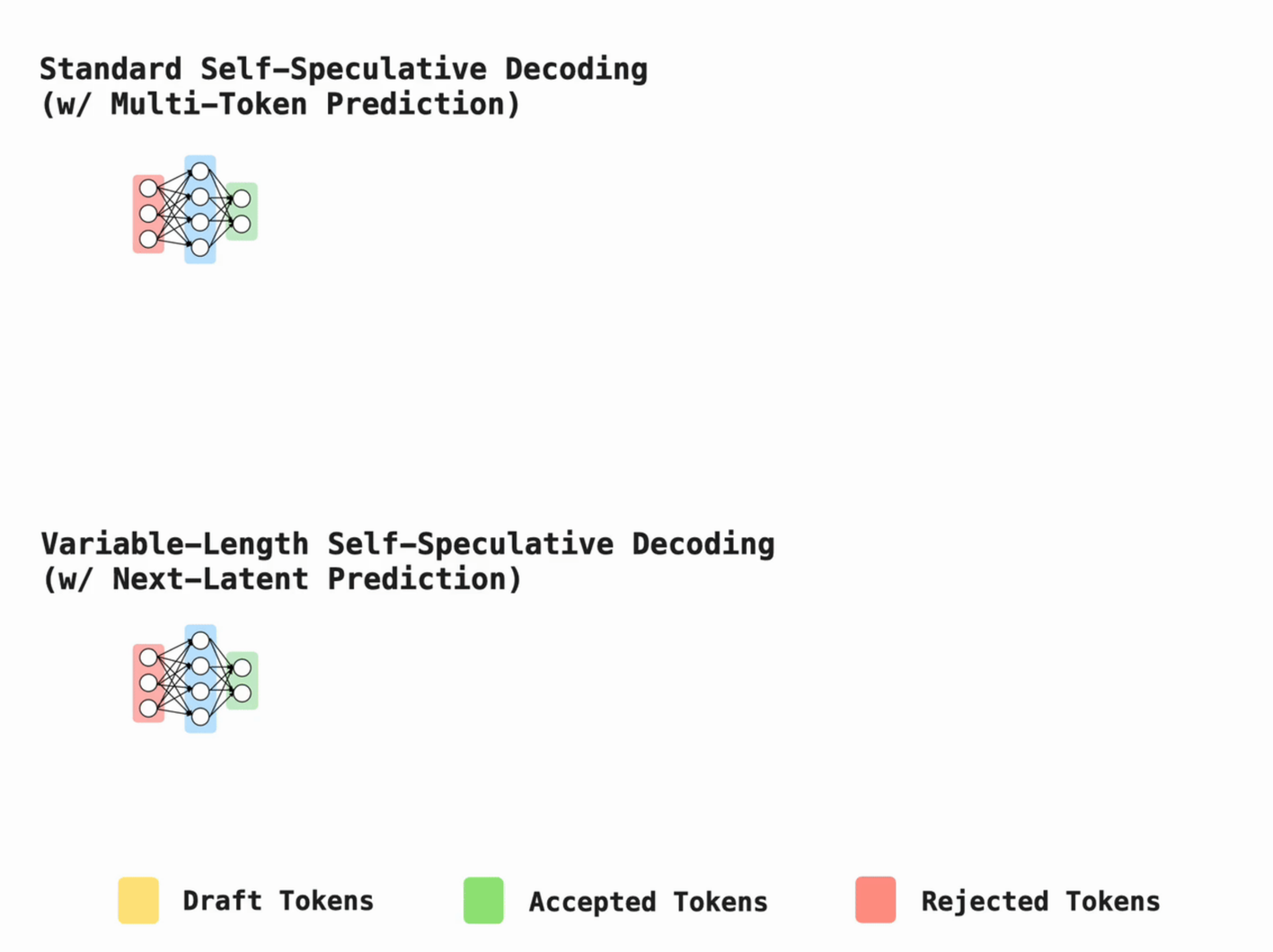
Speculative Decoding

Microsoft



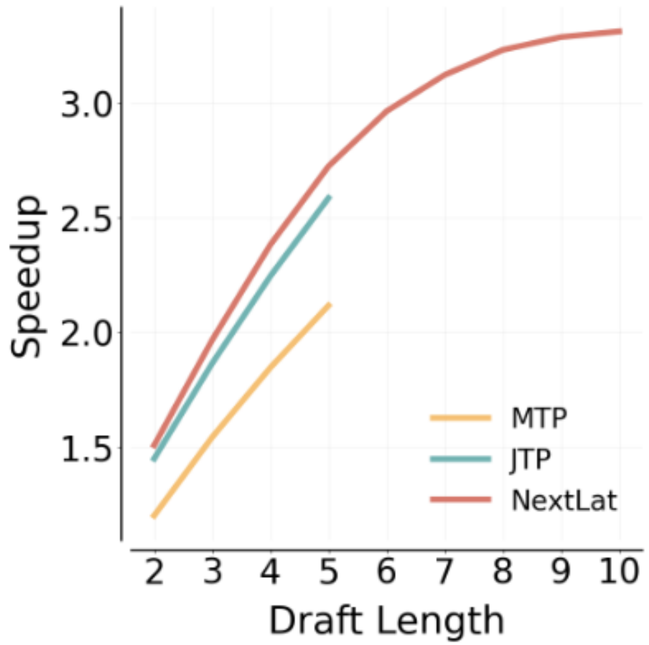
Speculative Decoding

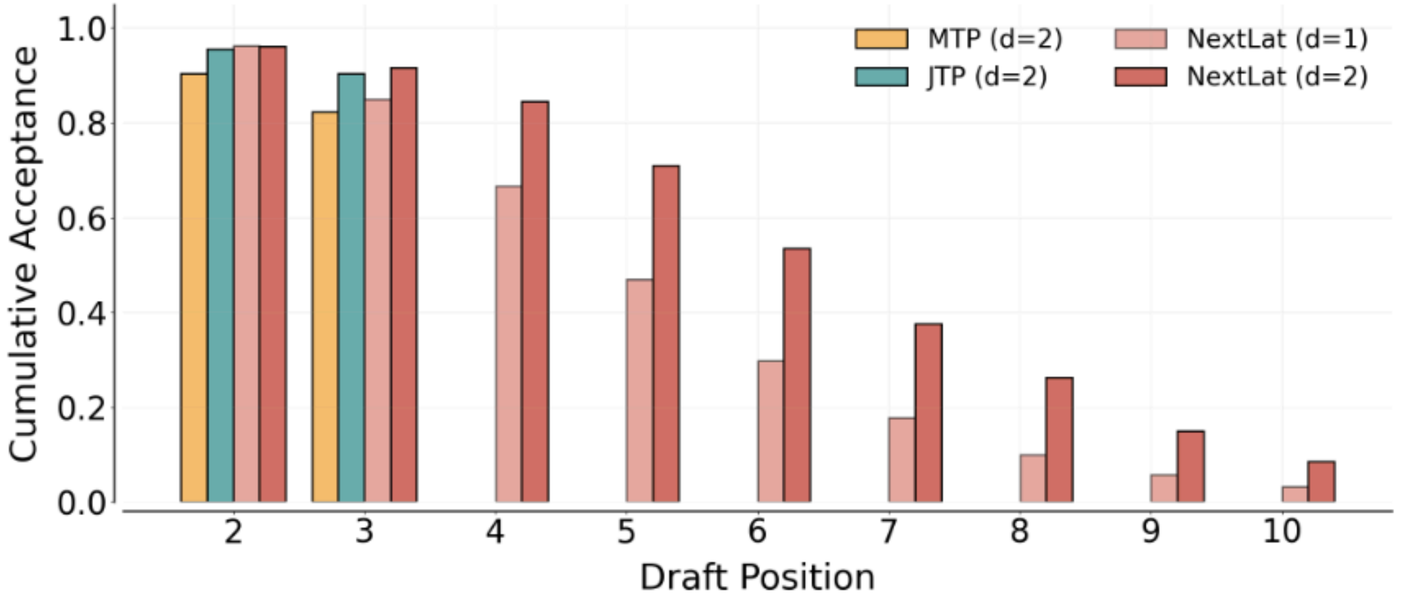
Microsoft



Speculative Decoding

Microsoft





Much faster inference (up to 3.3x) with speculative decoding

Speculative Decoding

Microsoft

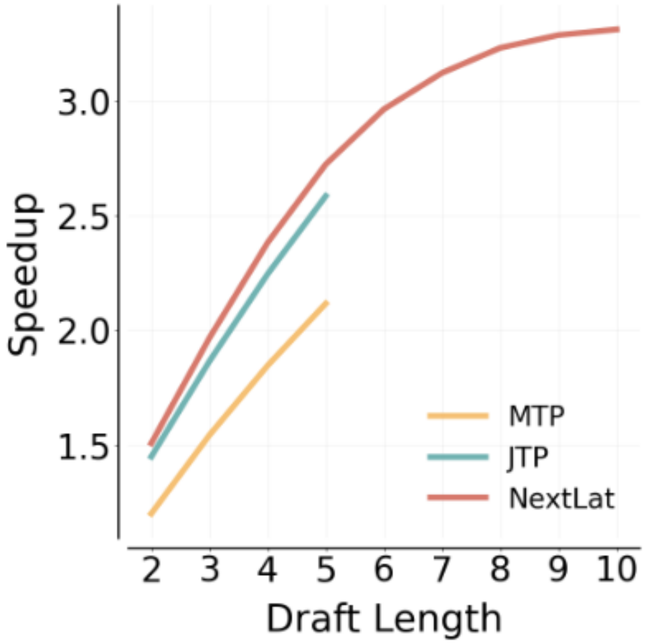
Model	Wikipedia		Books		Code		Math	
	Speedup	Accepted Tokens	Speedup	Accepted Tokens	Speedup	Accepted Tokens	Speedup	Accepted Tokens
JTP (d=1)	1.46	0.96	1.47	0.97	1.47	0.98	1.46	0.97
JTP (d=2)	1.88	1.84	1.90	1.89	1.88	1.85	1.89	1.86
MTP (d=1)	1.38	0.91	1.39	0.95	1.40	0.97	1.39	0.95
MTP (d=2)	1.68	1.72	1.72	1.83	1.75	1.91	1.72	1.84
NextLat (d=1)	2.68	3.52	2.72	3.64	2.29	2.66	2.30	2.72
NextLat (d=2)	**3.21**	**4.59**	**3.32**	**4.86**	**2.38**	**2.83**	**2.87**	**3.94**

Table 3: Relative speedup and average accepted tokens per drafting steps over diverse domains. Note that “Accepted Tokens" excludes the next-token prediction which is always accepted.

Much faster inference (up to 3.3x) with speculative decoding

Speculative Decoding

Microsoft



	Next-Token	Multi-Token	NextLat
Training parameters ($d = 1, 2, 8$)	1.32B	1.37B / 1.42B / 1.72B	1.40B
Training steps/second ($d = 1, 2, 8$)	3.09	2.80 / 2.58 / 1.70	3.09 / 2.79 / 1.73

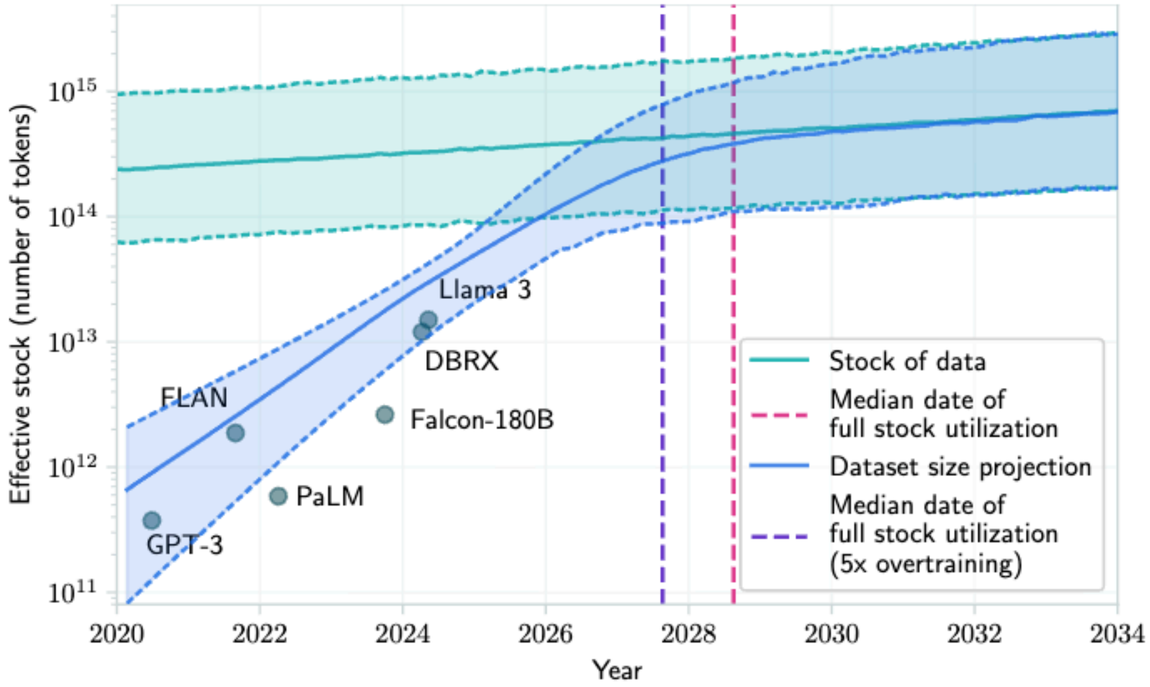
Table 5: Comparison of training speed and parameter.

Train with just next-latent prediction, no need multiple heads for multi-token prediction!

Data-Efficiency

Microsoft

We will run out of web data for training models by 2028…

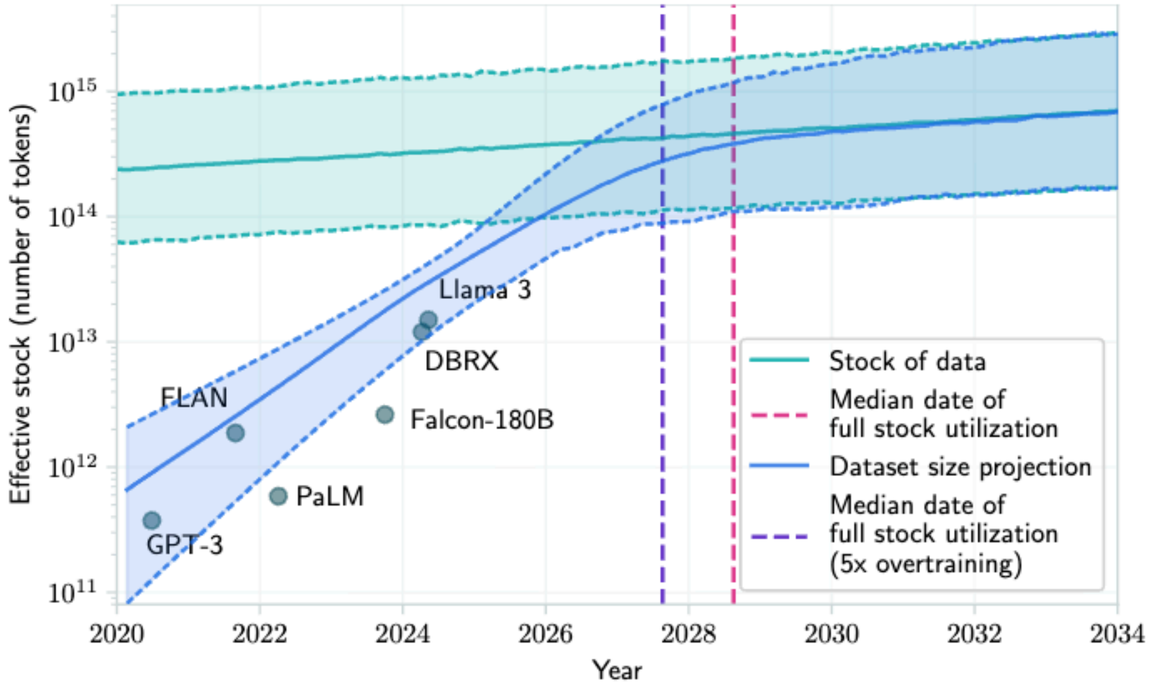


[8] Villalobos et al. Will we run out of data? Limits of LLM scaling based on human-generated data. ICML 2024.

Data-Efficiency

Microsoft

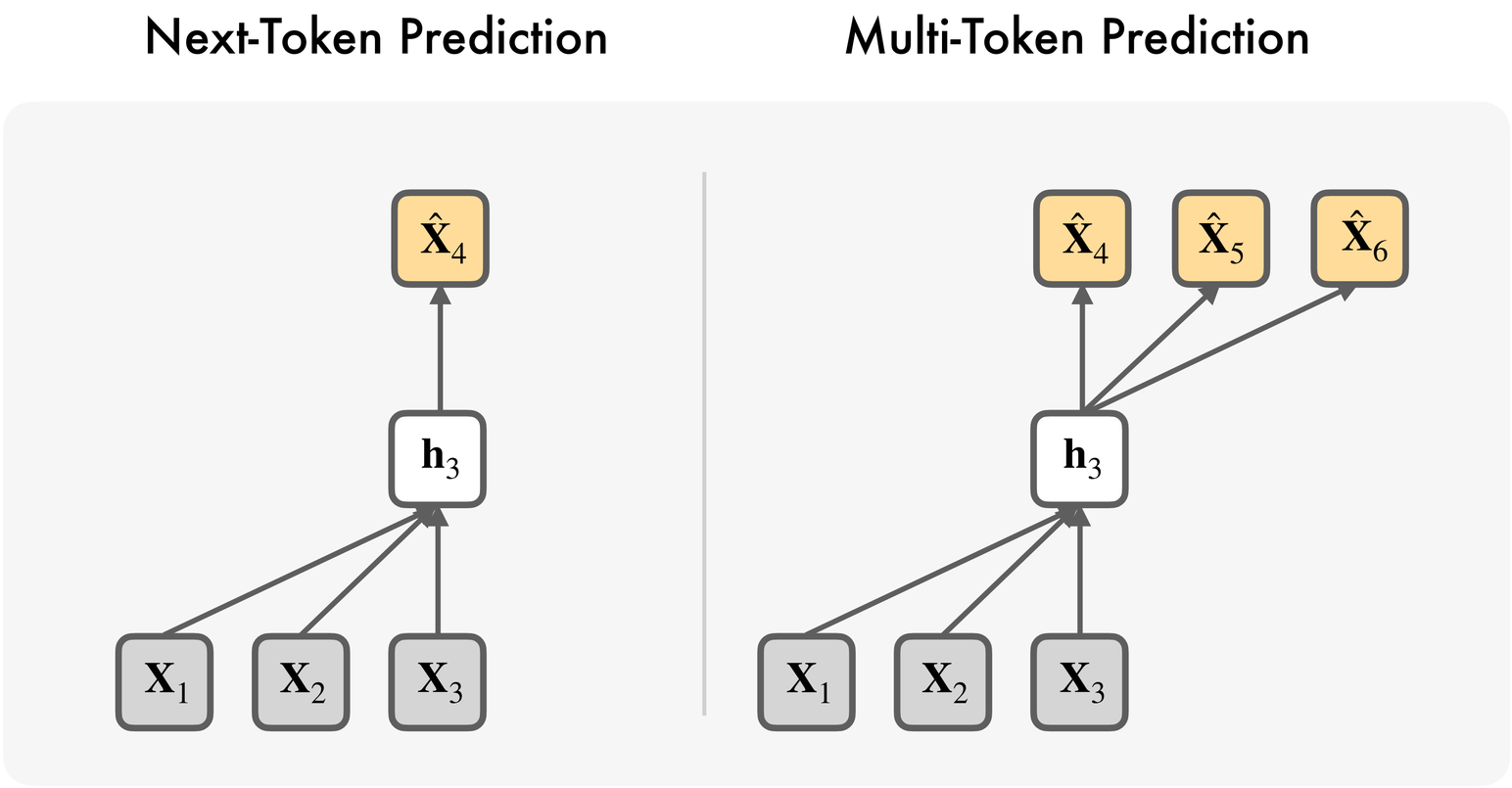
It’s time to think about methods that extract more gradients from each sequences…



[8] Villalobos et al. Will we run out of data? Limits of LLM scaling based on human-generated data. ICML 2024.

Data-Efficiency

Microsoft

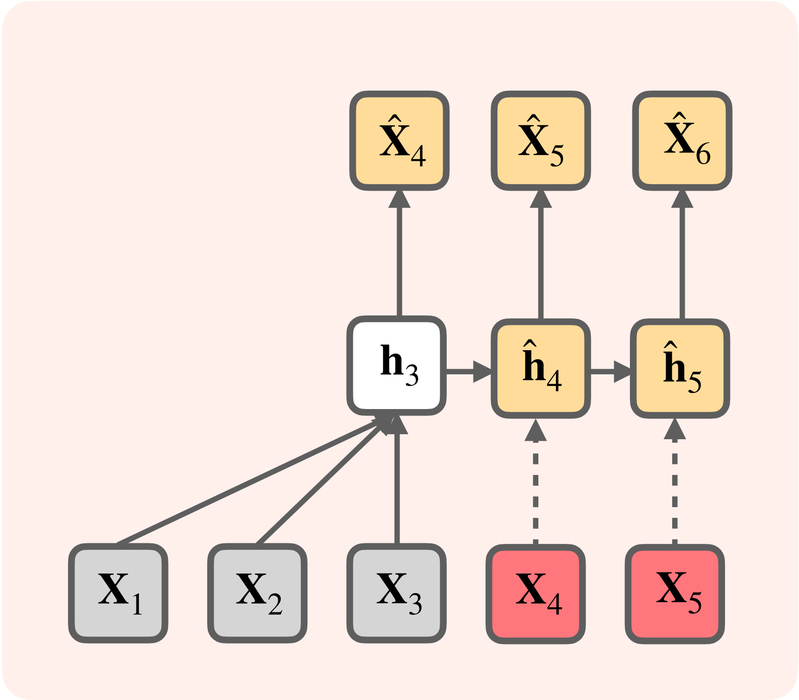


Learning signal for h_3 is only in terms of the next token, or multiple tokens….

Data-Efficiency

Microsoft

Next-Latent Prediction

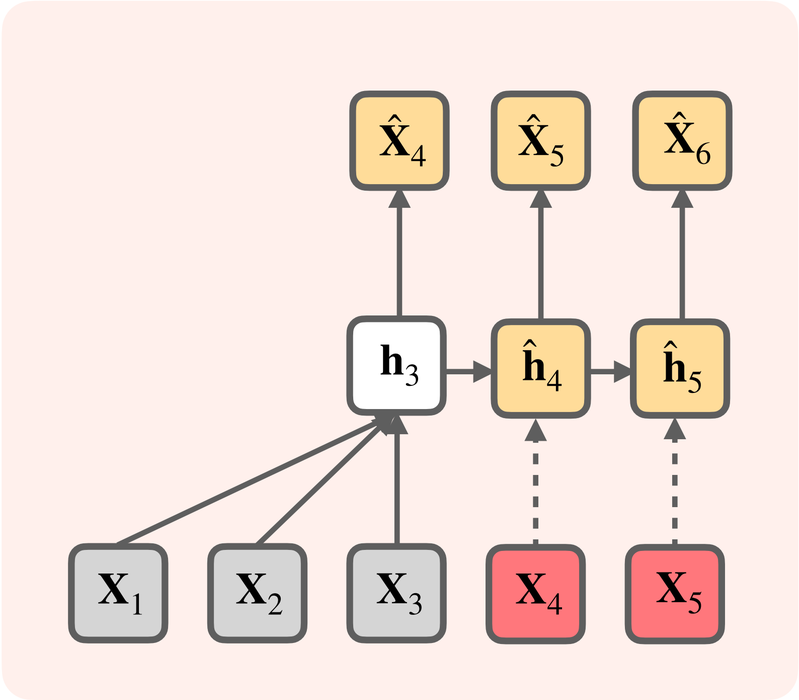


h_3 is trained to predict h_4 which parameterises *distribution* over X_5

Data-Efficiency

Microsoft

Next-Latent Prediction

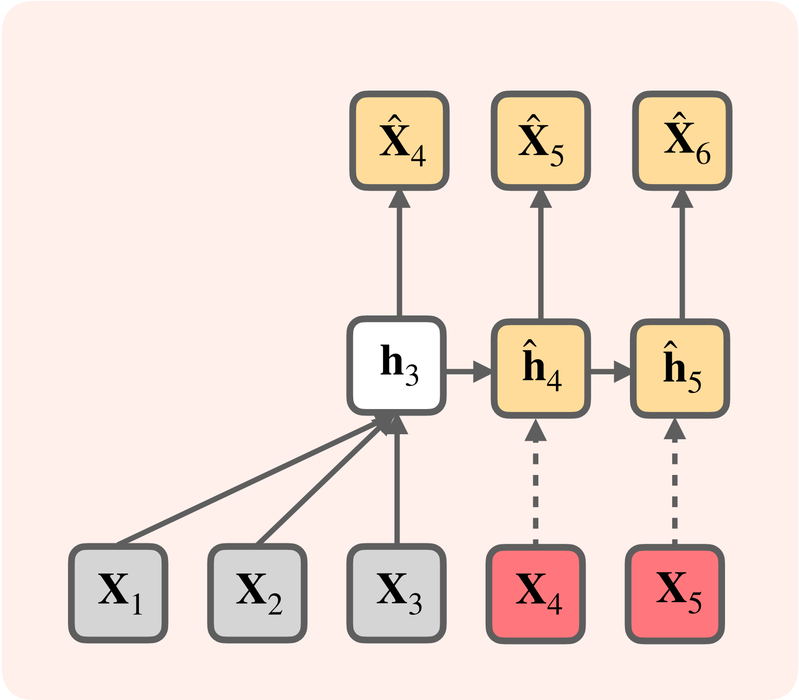


Moreover, h_4 is trained to predict h_5 which is trained to predict h_6…

Data-Efficiency

Microsoft

Next-Latent Prediction



This means that h_3 receives dense learning signal about the whole future h_4, h_5, h_6….

RNN without recurrence

Microsoft

- There are certain problems inexpressible by transformers but expressible by RNNs, such as state tracking [9]…

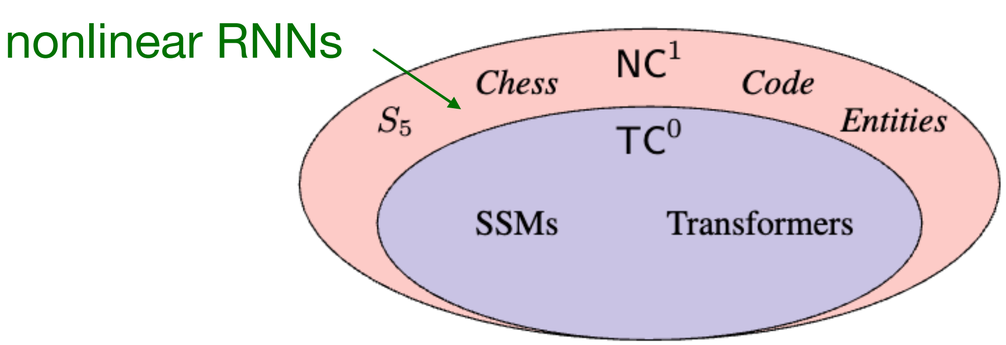
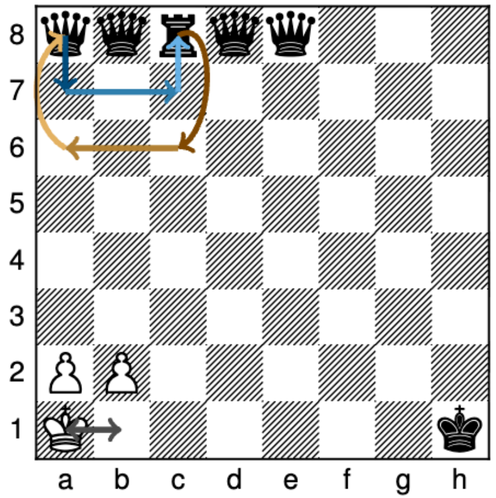


Figure 2: Complexity hierarchy within NC^1. Transformers can only recognize languages within TC^0 (Merrill & Sabharwal, 2023a), and we show the same for SSMs (Theorems 4.2 and 4.4). Thus, both architectures cannot express the "hard state tracking" captured by NC^1-complete problems like S_5, which *can* be straightforwardly expressed by RNNs. The figure assumes the widely held conjecture $TC^0 \neq NC^1$.




```
x = [0, 0, 1, 0, 0]
x[1], x[3] = x[3], x[1] # Swap 1, 3
```

Alice, Bob, Carl, Dan, and Emma each have a coin. All are dimes except Carl's. Alice and Carl trade coins.

[9] Merrill et al. The Illusion of State in State-Space Models. ICML 2024.

RNN without recurrence

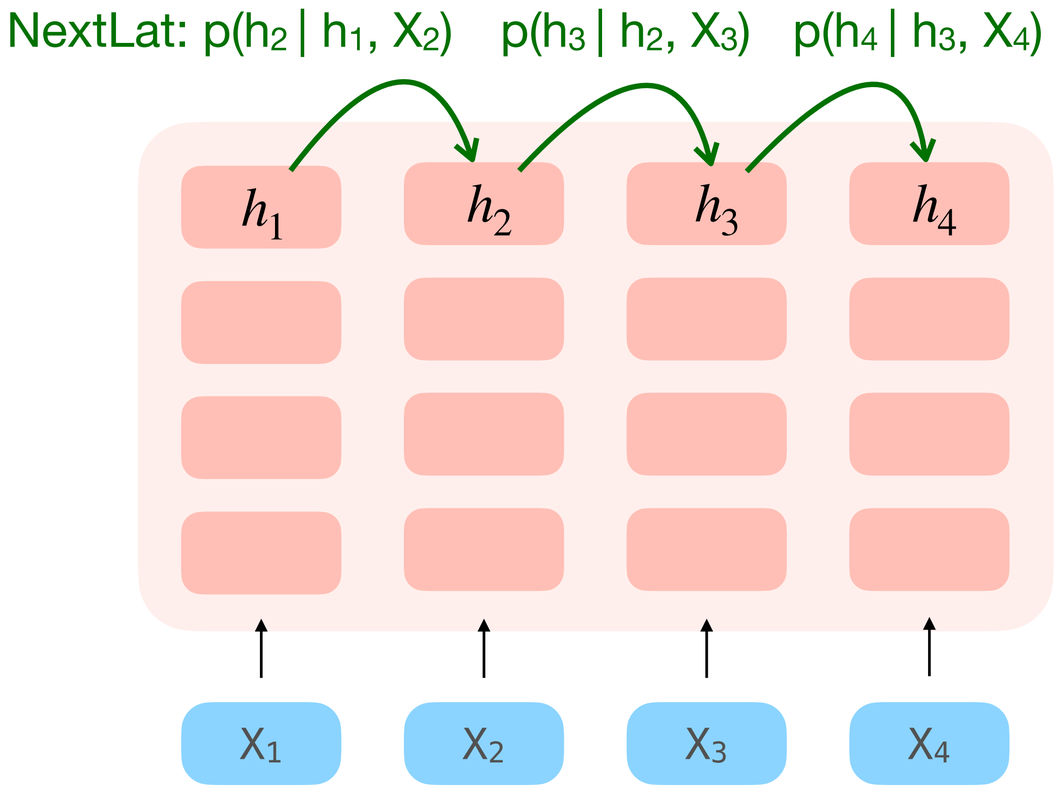
Microsoft

- NextLat trains a nonlinear RNN without recurrence

transformer forecasts hidden states in parallel

next-latent predictor (RNN) trains on one-step prediction

fully parallel across time!



RNN without recurrence

Microsoft

- We train NextLat on A_5, a word problem inexpressible by constant-depth, fixed-precision transformers

transformer cannot generalize beyond training length of 12 tokens

BUT, the RNN trained using NextLat can **generalize** far beyond!

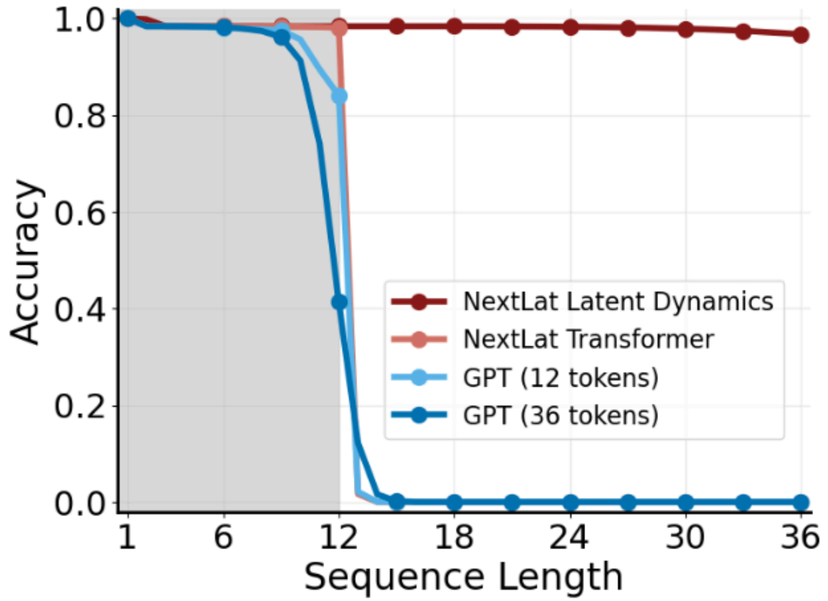


Figure 11: Length Generalization for A_5 word problem.

RNN without recurrence

Microsoft

- We train NextLat on A_5, a word problem inexpressible by constant-depth, fixed-precision transformers

Question: *Can a transformer with O(logT) depth co-train an RNN that generalizes to sequences of length ≫ T?*

Can the RNN learn NC^1 computations, even though it is supervised using representations from a transformer in TC^0?

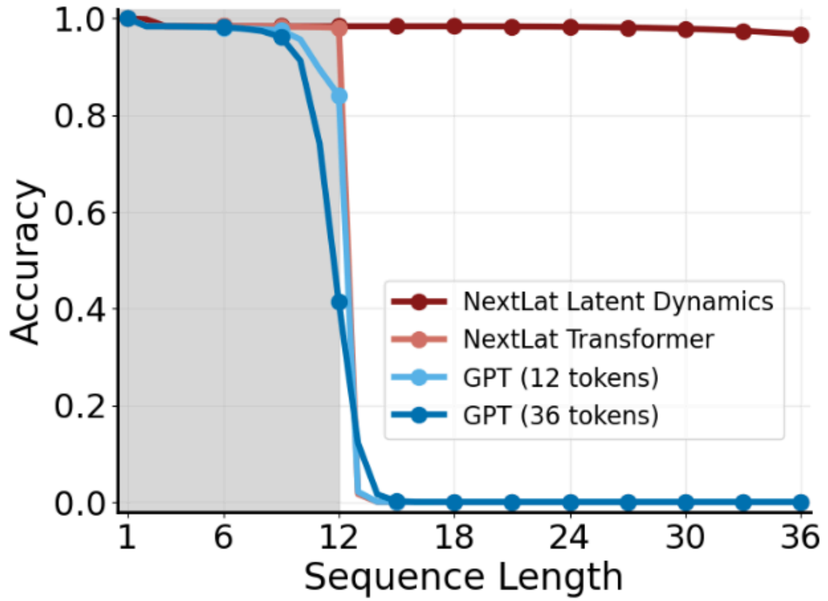


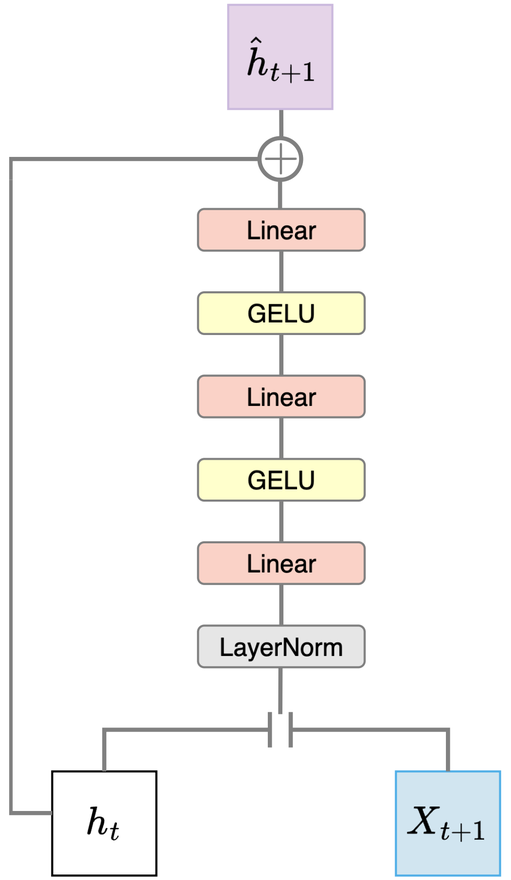
Figure 11: Length Generalization for A_5 word problem.

What's Next for NextLat?

Microsoft

- We got impressive results with just a **simple MLP** for the next-latent predictor

- What if we use more expressive architectures and scale up the predictor size?



What's Next for NextLat?

Microsoft

- Scaling up NextLat

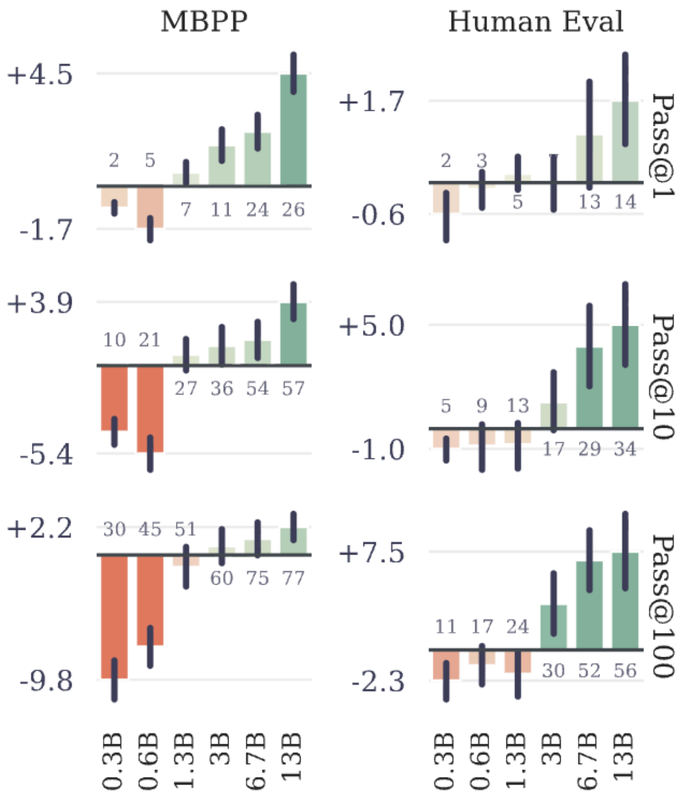


Figure 3: **Results of n-token prediction models on MBPP by model size.** We train models of six sizes in the range or 300M to 13B total parameters on code, and evaluate pass@1,10,100 on the MBPP (Austin et al., 2021) and HumanEval (Chen et al., 2021) benchmark with 1000 samples. Multi-token prediction models are worse than the baseline for small model sizes, but outperform the baseline at scale. Error bars are confidence intervals of 90% computed with bootstrapping over dataset samples.

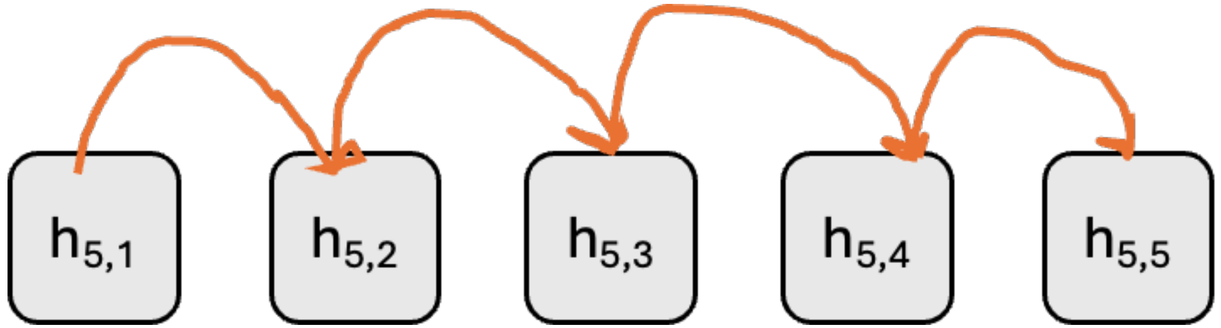
[9] Gloeckle et al. Better & faster large language models via multi-token prediction. ICML 2024.

What's Next for NextLat?

Microsoft

- NextLat might be better for post-training too!
  - better for implicit/explicit value learning

very "Bellman-like"/recurrent representations



What's Next for NextLat?

Microsoft

- Learning latent dynamics is already common in other fields, but not explored in language…

Dreamer algorithm [Hafner et al., 2019]

What's Next for NextLat?

Microsoft

- Learning latent dynamics is already common in other fields, but not explored in language…

Genie algorithm [Bruce et al., 2024]

Conclusion

Microsoft

- NextLat can replace multi-token prediction for pre-training
  - variable-length speculative decoding
  - less parameters, faster training (only need to train with d=1)
- NextLat can be a strong pretraining technique for foundation models
  - better representations for reasoning and planning
  - better data-efficiency; extract more gradients from each sequence

Conclusion

Microsoft

Paper:
arxiv.org/abs/2511.05963

Code:
github.com/JaydenTeoh/NextLat

Email: jayden_t@mit.edu

X: @jayden_teoh_