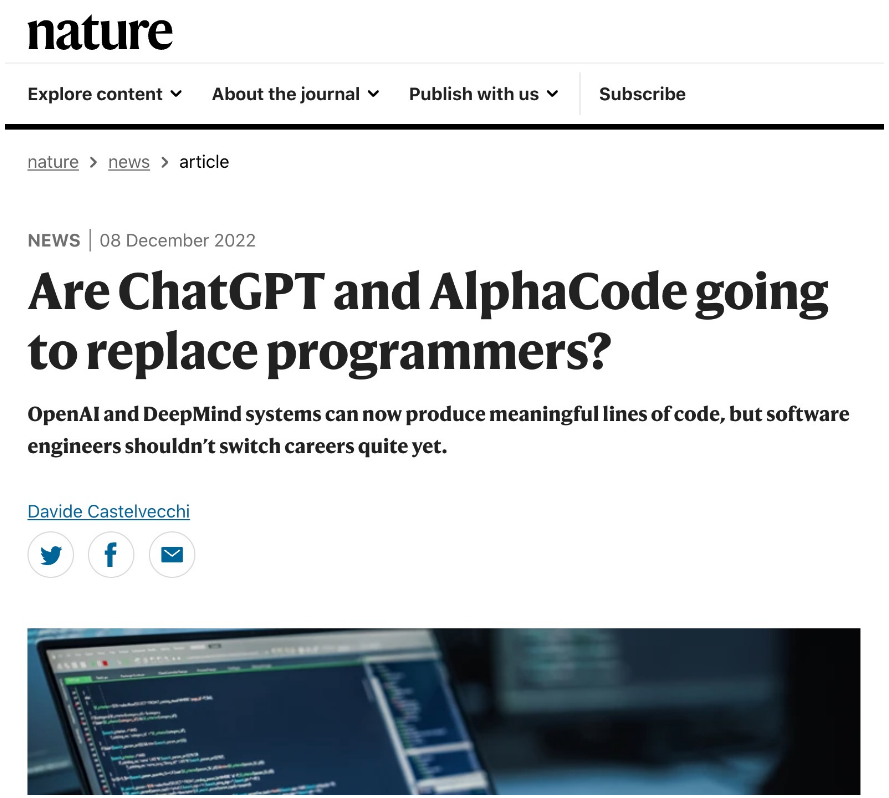
Towards Mathematical Understanding of Modern Language Models

Yuchen Li

yuchenl4@cs.cmu.edu

Carnegie Mellon University

Applications of modern language models (LMs)


nature

Explore content About the journal Publish with us Subscribe

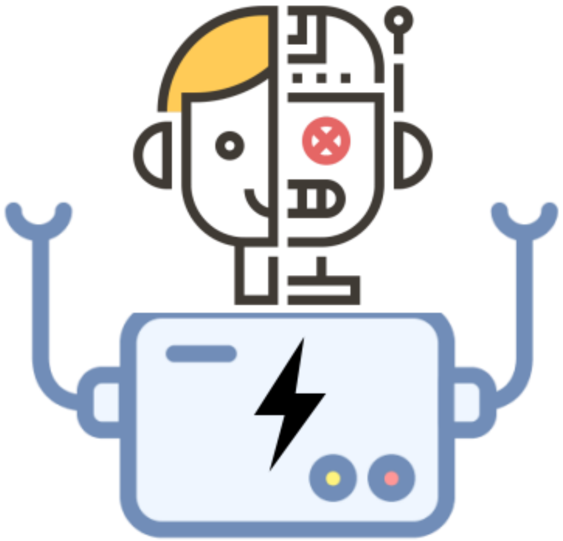
nature > news > article

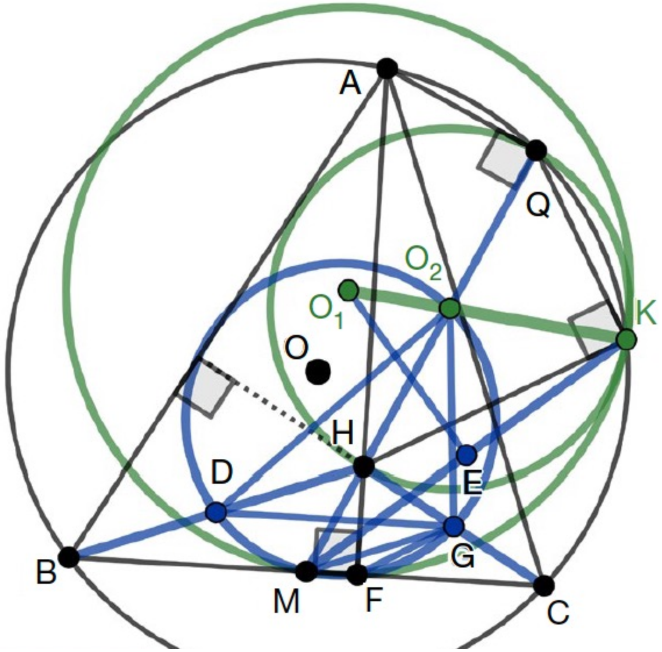
NEWS | 08 December 2022

Are ChatGPT and AlphaCode going to replace programmers?

OpenAI and DeepMind systems can now produce meaningful lines of code, but software engineers shouldn't switch careers quite yet.

Davide Castelvecchi





natural & programming languages

robotics

math theorem proving

Photos: nature.com; Fanlong Zeng et al, 2023, LLMs for Robotics: A Survey; Trieu H. Trinh et al, 2024, AlphaGeometry

Mathematically understanding LMs

$$L_{\alpha,\beta} = (p_c + \frac{p_r}{vT})[(1 - q(\frac{p_1\beta}{z_1}))^2 +$$
$$+ \frac{p_r}{T}(1 - \frac{1}{v})[(1 - q(\frac{p_1\alpha}{z_1}))^2 + q(\frac{p_1\beta}{z_1}$$
$$+ p_r\frac{\tau - 1}{T}[(1 - (\frac{p_1}{z_1}))^2 + q(\frac{p_1\beta}{z_1})^2 +$$
$$+ p_r(1 - \frac{\tau}{T})[(1 - q(\frac{p_1}{z_2}))^2 + q(\frac{p_1}{z_2})^2($$

Theory

guide experiment design

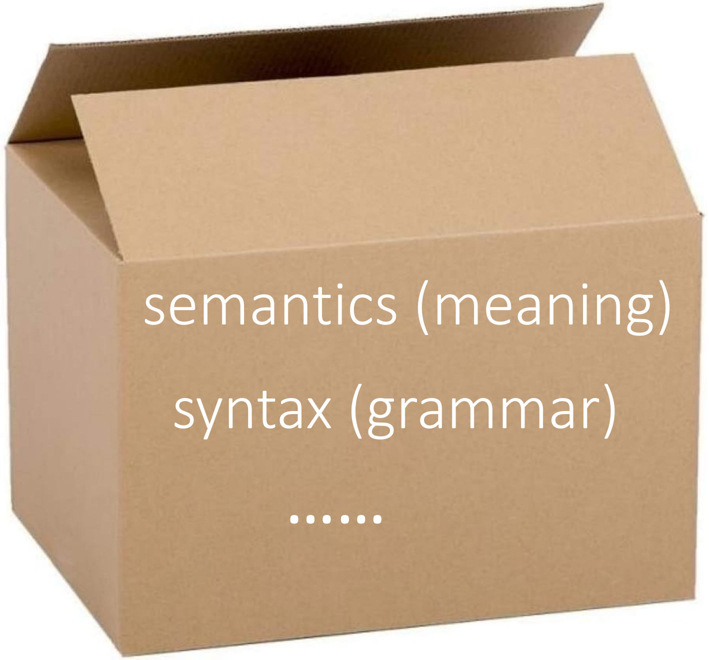
verify theoretical assumptions, generate hypotheses

Experiments

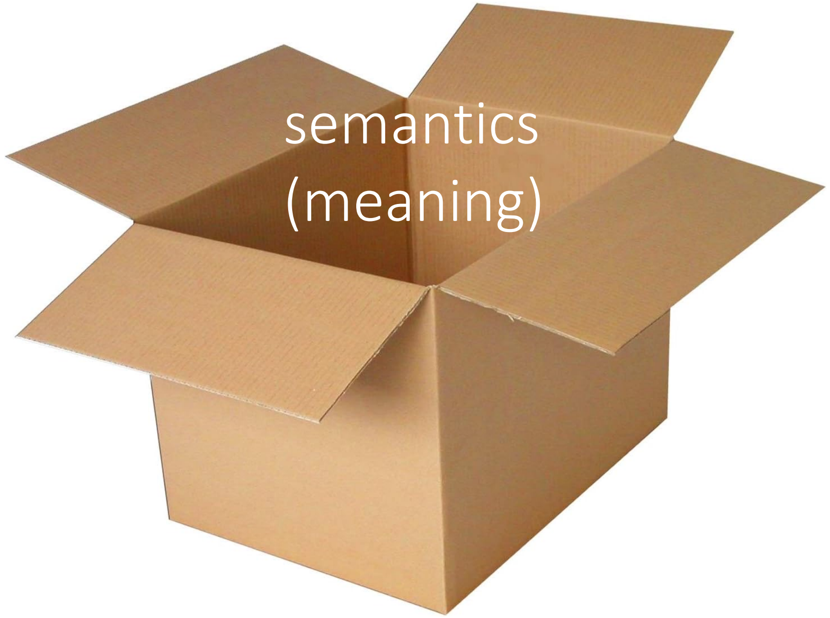
Figure on the right from: sciencefun.org

Methodology: controlled synthetic settings

- Identify structural assumptions in real data => simple synthetic setting
- Theory and controlled experiments



real data



synthetic data

Methodology: examples of insights

structures in real language

synthetic data distribution

✅ prove how models learn structure in data

❌ prove limitations in model expressivity, optimization, interpretability, …

? more structures & how they interact

? how to address such limitation

Outline of this talk

- **Part 1:** Towards mechanistic understanding of feature learning in Transformers
  - Understanding the training dynamics is crucial
  - How 1-layer Transformers learn simple structure (topic modeling)
  - Challenges with more complicated model or data (PCFG)
  - Large family of interpretability methods can be misleading
- **Part 2:** Improving training and sampling strategies for generative LMs
  - Sample efficiency of MLM losses $\leftrightarrow$ mixing times of Markov Chains
  - Directions towards designing better losses and architectures

How Do Transformers Learn Topic Structure: Towards a Mechanistic Understanding

Yuchen Li
(CMU)

Yuanzhi Li
(CMU & Microsoft)

Andrej Risteski
(CMU)

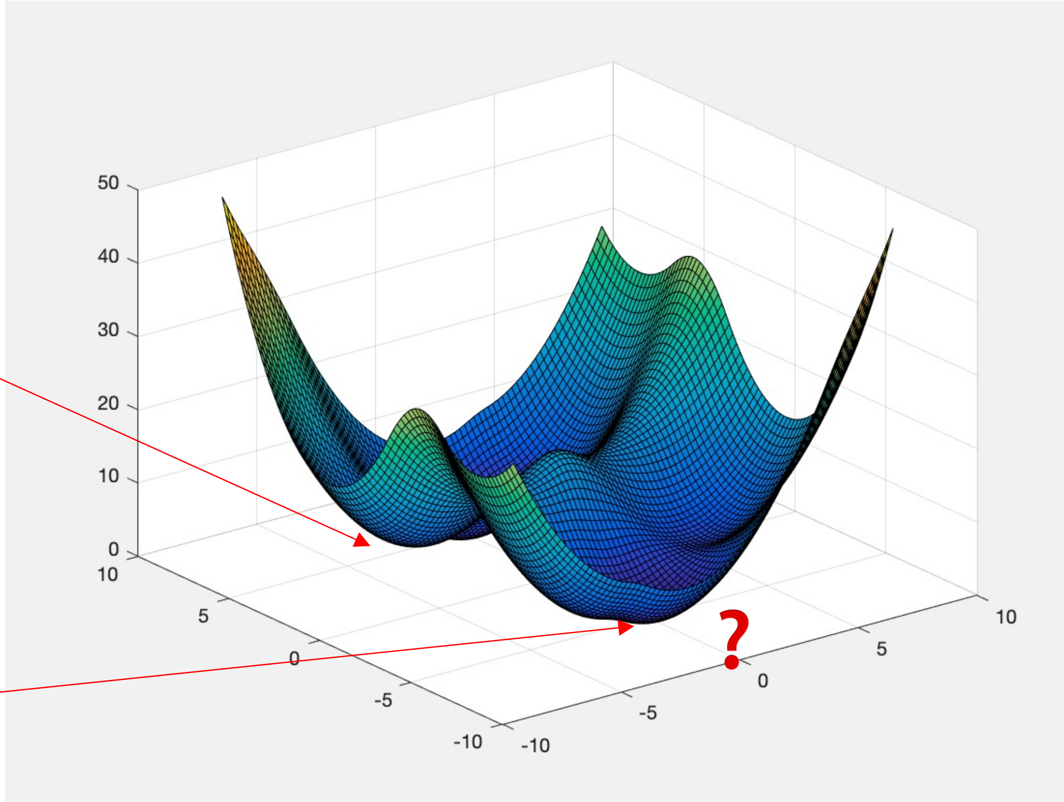
arxiv.org/abs/2303.04245 (ICML 2023)

Characterizing the optimization process is crucial

Many prior theories:
representational theoretical

Their claims: There exist parameters s.t. a Transformer implements some known function

Question: What function will the training dynamics converge to?



non-convex optimization landscape

Image from: https://science.hkust.edu.hk/research/geometric-landscape-analysis-some-non-convex-optimizations

Model architecture: single-layer transformer

- Given (one-hot) input representation $Z \in \mathbb{R}^{d \times N}$

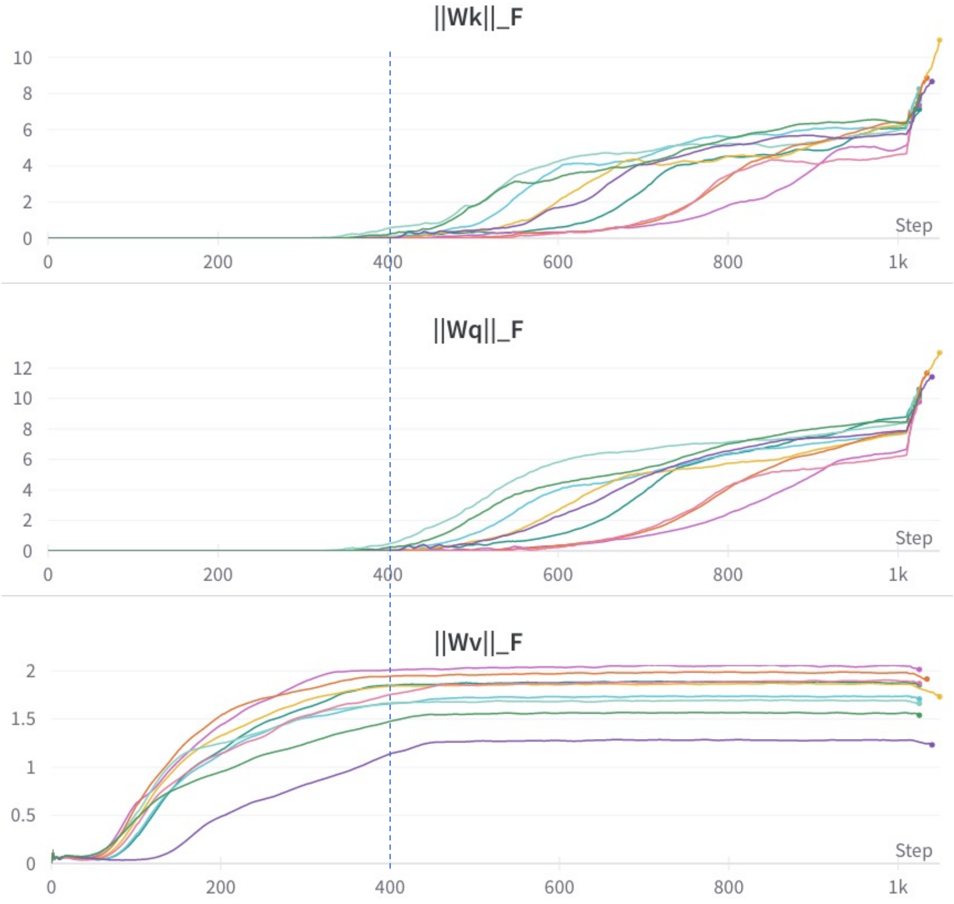
- d: embedding dimension
- N: sequence length

$$f(Z) = (W^V Z)\sigma\left((W^K Z)^\top (W^Q Z)\right)$$

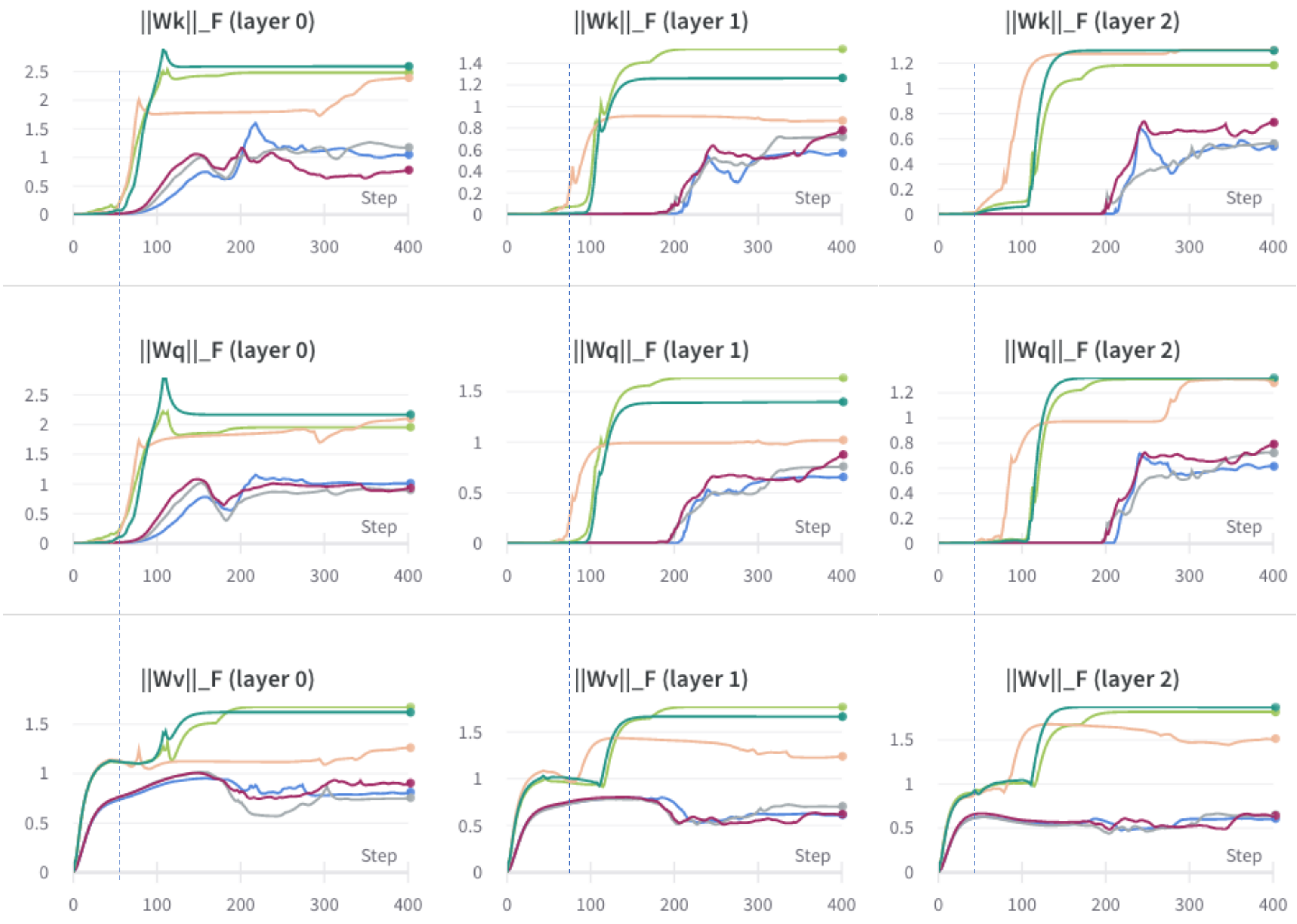
- $W^K, W^Q, W^V \in \mathbb{R}^{d \times d}$ attention key, query, value matrices
- σ: softmax (each column sums up to 1)
  - Input $X \in \mathbb{R}^{N \times N}$, output $\sigma(X) \in \mathbb{R}^{N \times N}$
  - $\sigma(X)_{ij} = \frac{e^{X_{ij}}}{\sum_{k=1}^{N} e^{X_{kj}}}$

Two-stage optimization process

- Stage 1 (steps 0-400)
  - $||W^K||_F, ||W^Q||_F \approx 0$
  - $||W^V||_F$ increases significantly
- Stage 2 (steps 400-1000)
  - $||W^K||_F, ||W^Q||_F$ start increasing significantly
  - $||W^V||_F$ stays relatively flat



Two-stage: multi-layer, multi-head, Wiki data



……

Two-stage optimization process

- Init: $W^K \approx 0,\ W^Q \approx 0,\ W^V \approx 0$
- During early training, W^V learns much faster than W^K and W^Q
- ∇_{W^K} contains the term W^Q
  - Init: $W^Q \approx 0$
  - So $\nabla_{W^K} \approx 0$
- Does not apply to W^V
  - ∇_{W^V} contains $Attn(Z)$
  - $Attn(Z)$ is not ≈ 0
  - each column sums up to 1
  - So ∇_{W^V} is not ≈ 0

Recall

- Trainable parameters: W^K, W^Q, W^V
- $f(Z) = (W^V Z)\, Attn(Z)$
- $Attn(Z) = \sigma\left((W^K Z)^\top (W^Q Z)\right)$
- σ: softmax (each column sums up to 1)

Training loss: masked language modeling

- **Original:** Andrew Carnegie famously said, "My heart is in the work."
- **Masked:** Andrew Carnegie famously [MASK], "My heart is apple the [MASK]."
- **Predicted:** Andrew ? famously ?, "My heart is ? the ?."

prediction $\hat{y}$ =

Carnegie	0.05
Webber	0.09
Ng	0.11
Jackson	0.08
Johnson	0.08
…	…

label y = Carnegie

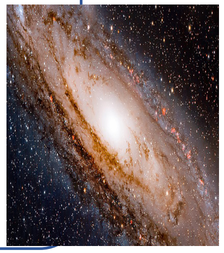
loss at that position $l(\hat{y}, y)$

training loss $\sum l(\hat{y}, y)$ for all selected positions

Data: topic model

- "Topic" is a simple aspect of semantics in natural language[1]
  - document = mixture of topics (bag of words, i.e. no word order)
  - topic = probability distribution of words

ski	2%			3%			0		
trail	4%			6%			0		
ice	1%			1%			0.1%		
sun	2%	=	0.7	1%	+	0.2	4%	+	…
stars	1%			0.1%			5%		
transformer	0			0			0		
…	…			…			…		



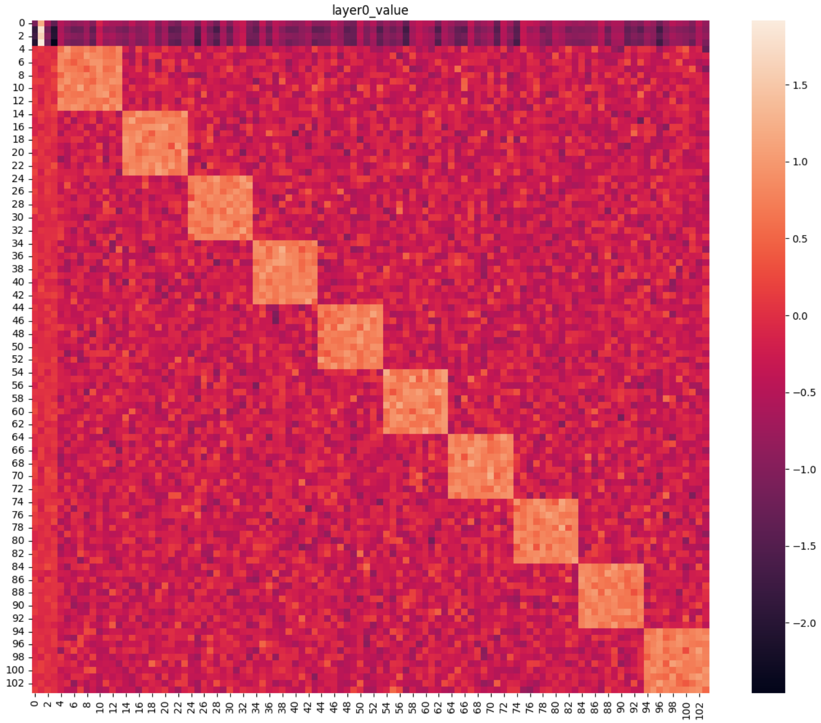
1. David Blei, et al, 2003, Latent Dirichlet Allocation (LDA)
2. Figure idea credit to Sanjeev Arora's talk in 2014

Stage 1 optima

Thm 1 (Stage 1: $W^K = W^Q = 0$, i.e. uniform attention).

With one-hot embedding, the optimal W^V is block-wise

- ${W^V}_{\mathrm{ij}}$ is larger when tokens i and j belong to the same topic
- ${W^V}_{\mathrm{ij}}$ is smaller when tokens i and j belong to the different topics



Yuchen Li, Yuanzhi Li, and Andrej Risteski. *How Do Transformers Learn Topic Structure: Towards a Mechanistic Understanding* (ICML 2023)

Stage 2 optima

Thm 1 (Stage 1: $W^K = W^Q = 0$, i.e. uniform attention).

With one-hot embedding, the optimal W^V is block-wise

- ${W^V}_{\mathrm{ij}}$ is larger when tokens i and j belong to the same topic
- ${W^V}_{\mathrm{ij}}$ is smaller when tokens i and j belong to the different topics

Thm 2 (Stage 2: Fixing W^V at Stage 1 optima).

Optimal attention scores $A \coloneqq \sigma\left((W^K Z)^\top (W^Q Z)\right)$ learns topic structure:

- A_{ij} is larger when tokens i and j belong to the same topic
- A_{ij} is smaller when tokens i and j belong to the different topics

Yuchen Li, Yuanzhi Li, and Andrej Risteski. *How Do Transformers Learn Topic Structure: Towards a Mechanistic Understanding* (ICML 2023)

Experiments on Wikipedia[1] dataset

- Real data
  - Not a “bag of words”
  - Different topics allowed to overlap
- Theoretical predictions still qualitatively hold
  - Same-topic tokens on average:
  - Larger attention scores
  - More similar embeddings

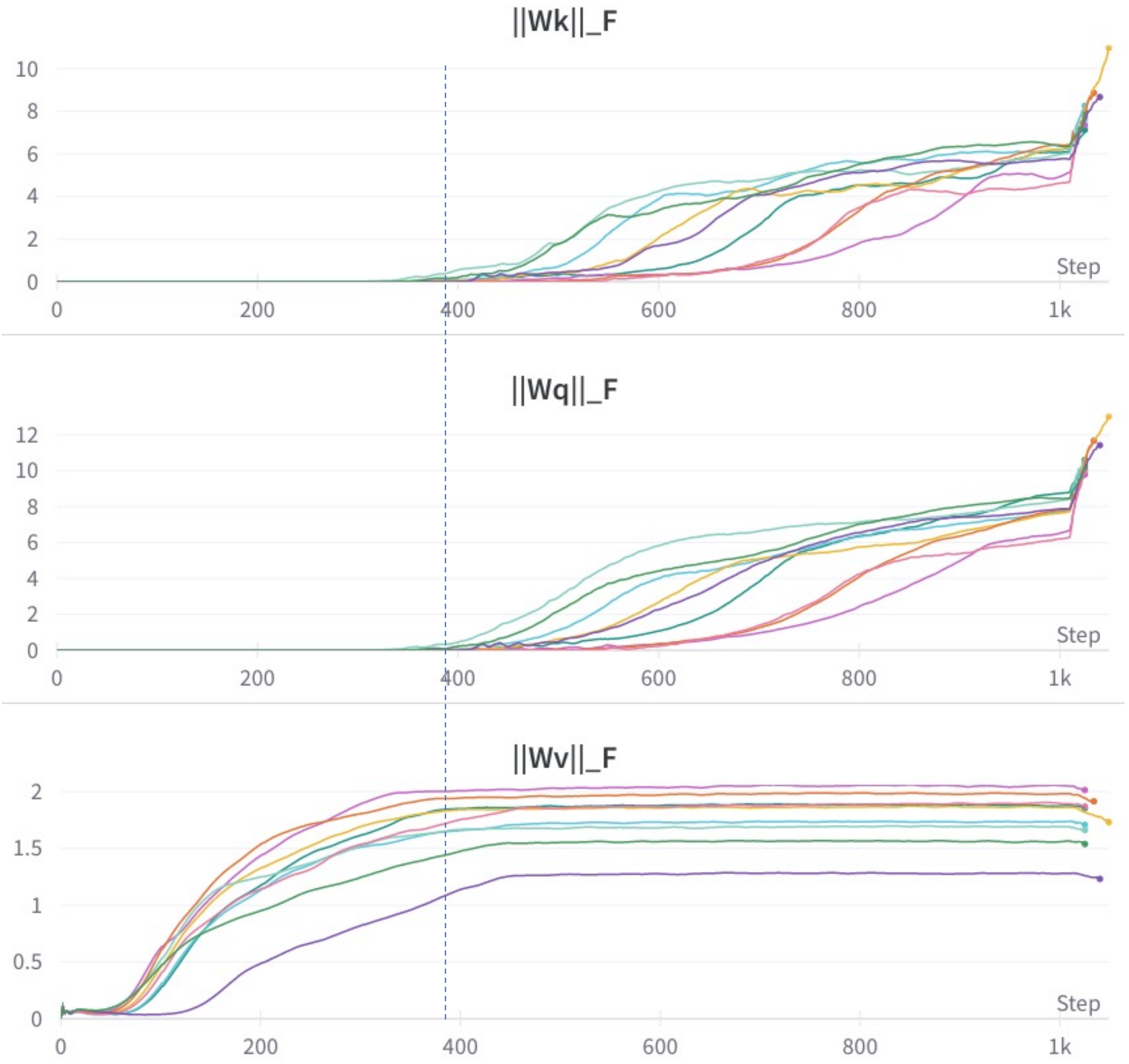
1. Wikimedia Foundation. URL https://dumps.wikimedia.org.

Future work: end-to-end theory for Transformer training dynamics

- Recall: two-stage optimization process
- More end-to-end training dynamics?

$$Attn(Z) = \sigma\left(\frac{(W^K Z)^\top (W^Q Z)}{\sqrt{d_a}}\right)$$

- Training dynamics for attention?



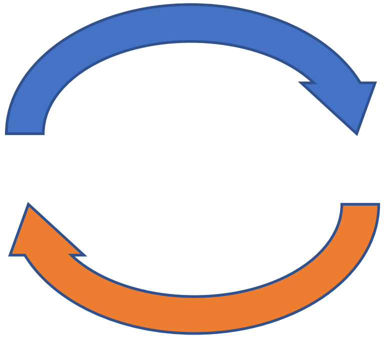
Ongoing work, with Zehao Dou, Mahdi Soltanolkotabi, and Andrej Risteski

Summary

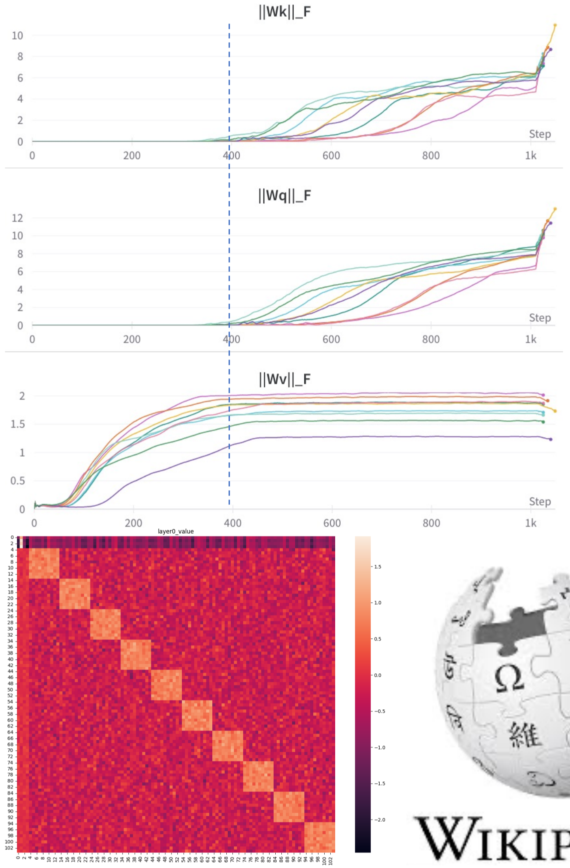
- ∇_{W^K} contains the term W^Q
  - Init: $W^Q \approx 0$
  - So $\nabla_{W^K} \approx 0$
- Does not apply to W^V
  - ∇_{W^V} contains $Attn(Z)$
  - $Attn(Z)$ is not ≈ 0
  - So ∇_{W^V} is not ≈ 0

Thm. Transformers capture topic structures through masked LM training

guides exploration



verify,
identify limitations,
generate hypothesis, ...



WIKIPEDIA

Theory

Experiments

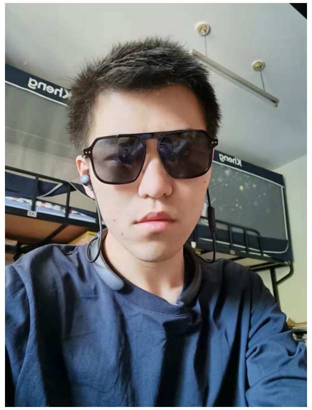
arxiv.org/abs/2303.04245

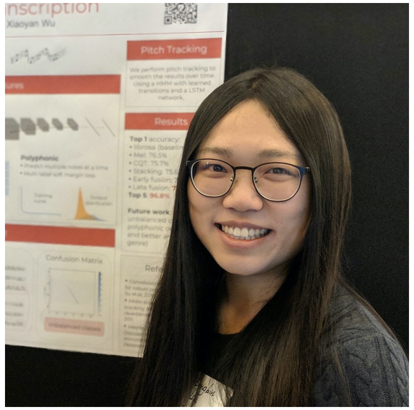
Yuchen Li, Yuanzhi Li, and Andrej Risteski. *How Do Transformers Learn Topic Structure: Towards a Mechanistic Understanding* (ICML 2023)

Outline of this talk

- **Part 1:** Towards mechanistic understanding of feature learning in Transformers
  - Understanding the training dynamics is crucial
  - How 1-layer Transformers learn simple structure (topic modeling)
  - Challenges with more complicated model or data (PCFG)
  - Large family of interpretability methods can be misleading
- **Part 2:** Improving training and sampling strategies for generative LMs
  - Sample efficiency of MLM losses $\leftrightarrow$ mixing times of Markov Chains
  - Directions towards designing better losses and architectures

Transformers are uninterpretable with myopic methods: a case study with bounded Dyck grammars



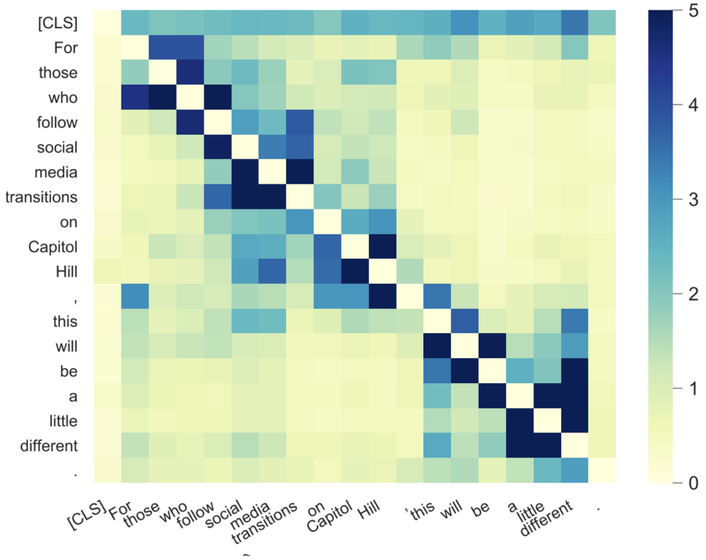


Kaiyue Wen Yuchen Li Bingbin Liu Andrej Risteski

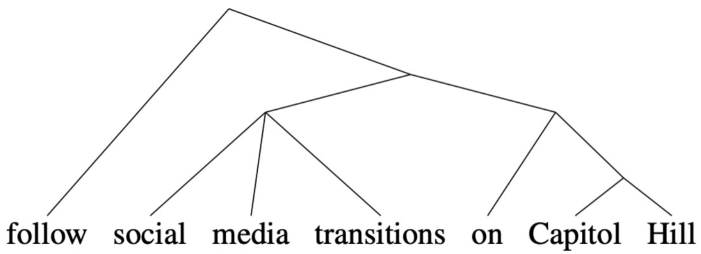
(Tsinghua University & Carnegie Mellon University)

arxiv.org/abs/2312.01429 (NeurIPS 2023)

Interpreting Transformers



attention map → syntactic trees



From “A Primer in BERTology” (Rogers et al. 20)

Pitfalls

- Can be misleading[1].
- Lack formal understanding.

1. Jain & Wallace, 2019; Serrano & Smith, 2019; Rogers et al., 2020; Brunner et al., 2020; Prasanna et al., 2020; Meister et al., 2021; ...

Interpreting Transformers

Question: Can we reliably interpret the algorithm implemented by a Transformer by *looking at individual components*?

"Individual" 1) attention patterns and 2) single weight components.

"myopic methods"

Answer: Transformers may *not* be interpretable by inspecting *individual parts*.

Theory

Dyck grammar

Experiments

Background: the Dyck language

Definition: the language of balanced parentheses

valid [ ] ( ) [( )]

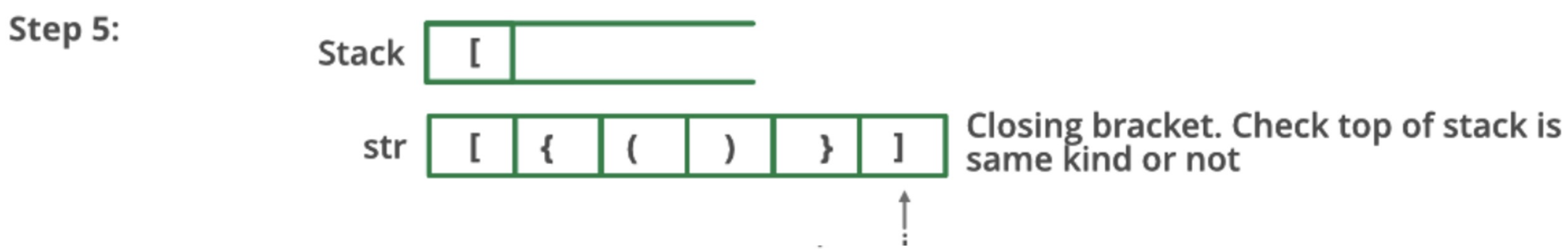
([{ }])

invalid [ ) ( ] [( ])

([ ]

Task: predict the **type and openness** of the next bracket.

- Most naturally processed by maintaining a stack.[1]



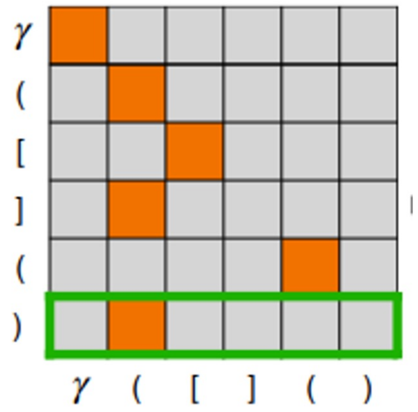
Question: how do Transformers process this Dyck language?

1. Illustrations: https://www.geeksforgeeks.org/

How do Transformers process Dyck?

Prior work [Ebrahimi et.al, Yao et.al]: Transformers learn Dyck with highly ***stack-like*** attention patterns.

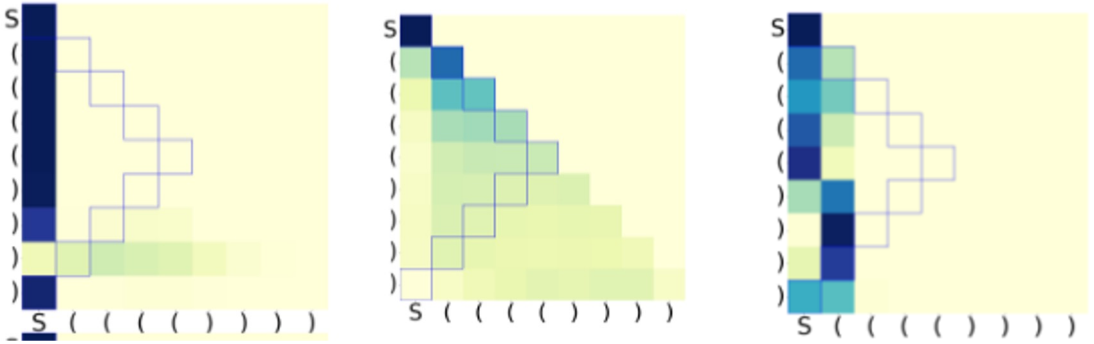
- Predict by focusing on the last unclosed bracket.



stack-like attention [Yao et.al]

Our results: Transformers learn ***diverse*** attention patterns on Dyck.

- Both in theory and in practice.
- All models reach high accuracy.



our findings: diverse attentions

Outline of this talk

- **Part 1:** Towards mechanistic understanding of feature learning in Transformers
  - Understanding the training dynamics is crucial
  - How 1-layer Transformers learn simple structure (topic modeling)
  - Challenges with more complicated model or data (PCFG)
  - Large family of interpretability methods can be misleading
- **Part 2:** Improving training and sampling strategies for generative LMs
  - Sample efficiency of MLM losses ↔ mixing times of Markov Chains
  - Directions towards designing better losses and architectures

Promises and Pitfalls of Generative Masked Language Modeling: Theoretical Framework and Practical Guidelines

Yuchen Li[1,2], Alexandre Kirchmeyer[1], Aashay Mehta[1], Yilong Qin[1],
Boris Dadachev[2], Kishore Papineni[2], Sanjiv Kumar[2], Andrej Risteski[1]
([1]CMU [2]Google)

arxiv.org/abs/2407.21046 (ICML 2024)

The autoregressive language model paradigm

Learn an autoregressively parametrized distribution:

$$P_\theta(X_1, X_2, \cdots, X_N) = \prod_{i=1}^{N} P_\theta(X_i \mid X_1, \cdots, X_{i-1})$$

Issues:

1. *Lack of parallelism*

   N sequential steps to generate N tokens

2. *Quality**

- Can't access right-hand context
- No natural way to revise earlier (left) predictions

* Li and Risteski. (ACL 2021)
* Lin et al. (NAACL 2021)
* Bachmann and Nagarajan (arXiv 2024)

Alternative: Generative Masked Language Models*

Non-autoregressive way to generate a sequence*:

- Start w/ pure noise (e.g. masks, random tokens)
- Iteratively refine current guess, s.t. one forward pass updates multiple positions simultaneously.

Bidirectional context. Leverages “parallelism” of transformers for each step.

If # of steps is small, latency is low.

* Jacob Devlin et al. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding
* Alex Wang and Kyunghyun Cho. 2019. BERT has a mouth, and it must speak: BERT as a Markov random field language model
* Marjan Ghazvininejad et al. 2019. Mask-predict: Parallel decoding of conditional masked language model
* Jacob Austin. 2021. Structured denoising diffusion models in discrete state-spaces
* Jiatao Gu and Xiang Kong. 2021. Fully non-autoregressive neural machine translation: Tricks of the trade.
* Kartik Goyal et al. 2022. Exposing the implicit energy networks behind masked language models via metropolis–hastings
* Nikolay Savinov et al. 2022. Step-unrolled denoising autoencoders for text generation

Example of the iterative refinement process

- translate from German to English: Im Fußball geht alles sehr schnell
- human label: Everything moves very fast in football.
- initial decoder hypothesis: <random> <random> <random> …
- decode step 1: Everything football very fast in football.
- decode step 2: Everything is very fast in football.
- decode step 4: Everything is very fast in football.
- decode step 8: Everything is very fast in football.

Example of the iterative refinement process

- human label: Noble Peace Prize winner and former Head of the International Atomic Energy Authority, Mohamed El-Baradei explained that the constitutional draft belongs "on the rubbish tip of history."
- decode step 1: Nobel Peace Prize laureate and ex- of the International Atomic Energy Agency Mohamed ElBaradei said the draft constitution belongson the of rubbish of history".
- decode step 2: Nobel Peace Prize laureate and ex-head of the International Atomic Energy Agency Mohamed El-Baradei said the draft constitution belongs "on the mountain of rubb rub of history".
- decode step 4: Nobel Peace Prize laureate and ex-head of the International Atomic Energy Agency Mohamed El-Baradei said the draft constitution belongs "on the mountain of rubbish in history".
- decode step 8: Nobel Peace Prize laureate and ex-head of the International Atomic Energy Agency Mohamed El-Baradei said the draft constitution belongs "on the mountain of rubbish in history".

Generative Masked Language Models

Training: predict (random) set of tokens, given rest.
In other words, fit $P_\theta(X_S \mid X_{\bar{S}})$

- **Original:** Andrew Carnegie famously said, "My heart is in the work."
- **Masked:** Andrew Carnegie famously [MASK], "My heart is in the [MASK]."

Generation: use the learned conditionals $P_\theta(X_S \mid X_{\bar{S}})$ as input for a Gibbs sampler.

Generative Masked Language Models

Gibbs sampling:

Repeat:

Let current sequence be $\boldsymbol{x} = (x_1, x_2, \dots, x_n)$

Pick $S \subseteq [n]$ uniformly at random.

Sample $\boldsymbol{x}_S{}' \sim P_\theta(\boldsymbol{X}_S = \boldsymbol{x}_S{}' | \boldsymbol{x}_{\bar{S}})$

Update sequence to $\boldsymbol{y} = (\boldsymbol{x}_S{}', \boldsymbol{x}_{\bar{S}})$

This paper

Questions:

How well do we fit *joint* distribution by training to fit the *conditionals*?

Can we use theory to elucidate the design space of losses, training and inference procedures?

Answers:

(1) *A mathematical framework* to analyze training *sample efficiency* & *inference efficiency* of masked language models (MLMs).

(2) *(Not in this talk) Empirical* analysis of critical components & failure modes.*

* Li et al. *Promises and Pitfalls of Generative Masked Language Modeling: Theoretical Framework and Practical Guidelines*. ICML 2024.

Highlights

- “*Dictionary*” between
  - sample complexity of MLM losses (“training efficiency”), and
  - mixing times of Markov Chains (“generation efficiency”)
- Directions towards designing better losses and architectures

Part I: Dictionary b/w sample efficiency and mixing time

Theorem 1 (informal): Sample efficiency of MLM losses can be characterized via mixing time of Gibbs-like sampler.
(E.g., masking random subsets of size k during training
≈ Gibbs sampler that randomizes k coordinates)

Training is sample-efficient when generation is efficient !

Part I: Dictionary b/w sample efficiency and mixing time

Theorem 1 (informal): Sample efficiency of MLM losses can be characterized via mixing time of Gibbs-like sampler.
(E.g., masking random subsets of size k during training
≈ Gibbs sampler that randomizes k coordinates)

Theorem 2 (informal): Masking more is (statistically) better.

Part II: Strong correlations harm sample and inference efficiency

Theorem 3 (informal): Strong dependencies among target positions cause:

(1) Slow generation: slow mixing of Gibbs sampler (*multimodal*)

(2) Slow training: poor sample efficiency (*via Theorem 1*)

(3) A step of Gibbs can't be implemented by parallel decoding Transformers (e.g. a forward pass of BERT*)

Proof idea for (3): Each forward pass of parallel decoding Transformers implements a conditional product distribution

* Jacob Devlin et al. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

Part II: Strong correlations harm sample and inference efficiency

Theorem 3 (informal): Strong dependencies among target positions cause:

(1) Slow generation: slow mixing of Gibbs sampler (*multimodal*)

(2) Slow training: poor sample efficiency (*via Theorem 1*)

(3) A step of Gibbs can't be implemented by parallel decoding Transformers (e.g. a forward pass of BERT*)

Remark 1: Simple toy model to explain "stutter" (common failure mode we observe):

"The dog was walking walking along the road"

Remark 2: Explains why these model work much better for machine translation (generation is "less multimodal", and target-side dependency is weaker)

Future work: ideas to improve losses + samplers

- “Dependent” version of Gibbs sampler where masks are adaptively chosen. (Details in paper)
  - Unclear how to measure “dependence”
  - Preliminary evidence cross-attention is better than self-attention

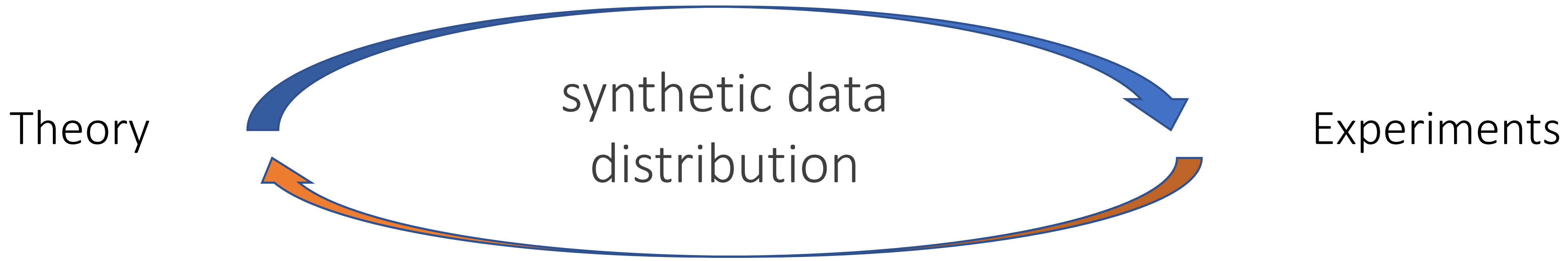
- Better architectures to implement Markov Chain update in parallel?

* Li et al. *Promises and Pitfalls of Generative Masked Language Modeling: Theoretical Framework and Practical Guidelines*. ICML 2024.

Summary

Email: yuchenl4@cs.cmu.edu
Web: cs.cmu.edu/~yuchenl4

- **Part 1:** Towards mechanistic understanding of feature learning in Transformers
  - Understanding the training dynamics is crucial
  - How 1-layer Transformers learn simple structure (topic modeling)
  - Challenges with more complicated model or data (PCFG)
  - Large family of interpretability methods can be misleading
- **Part 2:** Improving training and sampling strategies for generative LMs
  - Sample efficiency of MLM losses ↔ mixing times of Markov Chains
  - Directions towards designing better losses and architectures



1. Yuchen Li, Yuanzhi Li, and Andrej Risteski. *How Do Transformers Learn Topic Structure: Towards a Mechanistic Understanding* (ICML 2023)
2. Kaiyue Wen, et al. *Transformers are uninterpretable with myopic methods: a case study with bounded Dyck grammars* (NeurIPS 2023)
3. Yuchen Li et al. *Promises and Pitfalls of Generative Masked Language Modeling: Theoretical Framework and Practical Guidelines* (ICML 2024)