Towards a theory of representation learning for hierarchically compositional data

F. Cagnetta,
with A. Favero, L. Petrini, A. Sclocchi, U. Tomasini, M. Wyart

The Power of Deep Learning

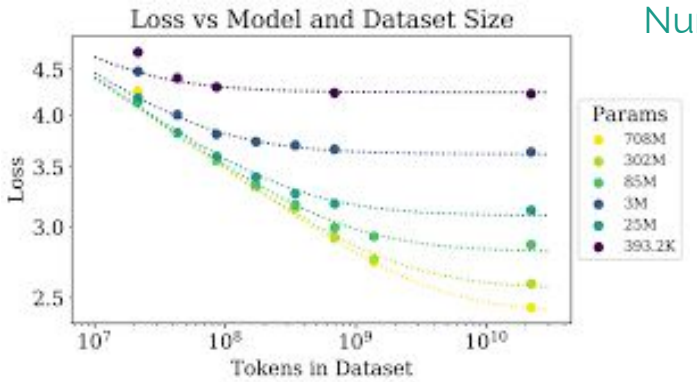
- State of the art results in many domains:

The Power of Deep Learning

- State of the art results in many domains:

- Lacking of fundamental understanding: *how does deep learning work?*

Empirical regularities (scaling)



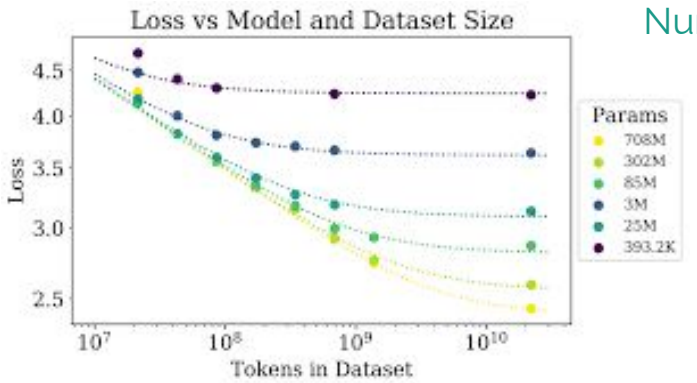
Num. data

Num. parameters

$$L(N, D) = \left[\left(\frac{N_c}{N} \right)^{\frac{\alpha_N}{\alpha_D}} + \frac{D_c}{D} \right]^{\alpha_D}$$

[Kaplan *et al.* '20] (scaling)

Empirical regularities (scaling)



Num. data

Num. parameters

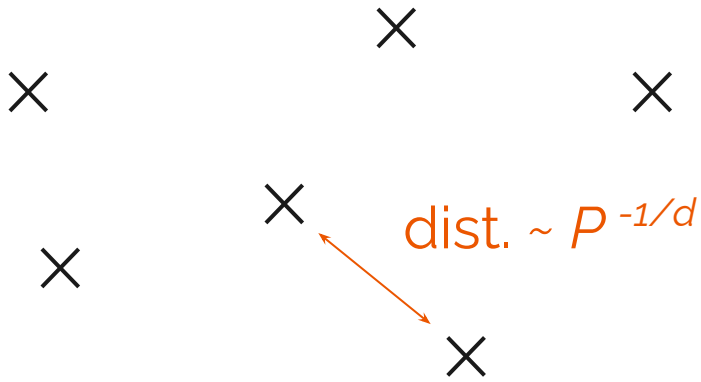
$$L(N, D) = \left[\left(\frac{N_c}{N} \right)^{\frac{\alpha_N}{\alpha_D}} + \frac{D_c}{D} \right]^{\alpha_D}$$

[Kaplan *et al.* '20] (scaling)

- Turn qualitative questions (*how does deep learning work?*) into quantitative ones (*scaling exponents? # data to learn up to specific accuracy?*).

Deep Learning and structured data

Curse of dimensionality: randomly sample ***P*** points in ***d*** dimensions:



To avoid sparsification, P >> e^d

Deep Learning and structured data

Curse of dimensionality: randomly sample ***P*** points in ***d*** dimensions:

To avoid sparsification, P >> e^d

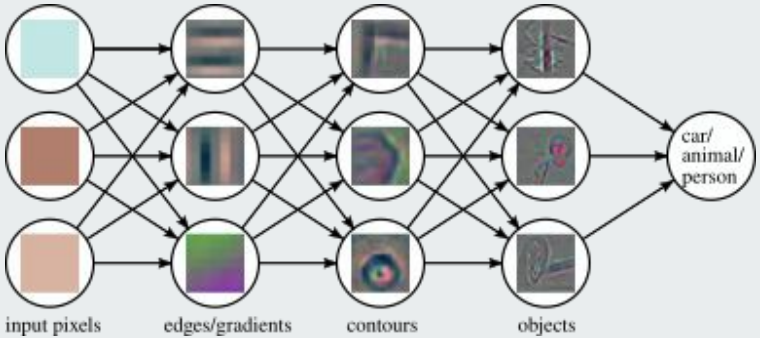
Data are structured:

e.g. low-dim. manifold, smoothness
[F. Bach, *the quest for adaptivity*, 2021]

- What is the structure of natural data such as images and text?
- How does it guide deep learning?

The structure of ‘natural’ data

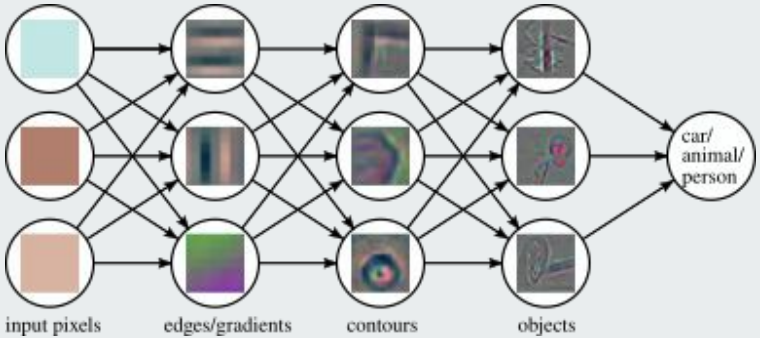
Hierarchical Compositionality:



[Goodfellow, Bengio, Courville *Deep Learning* ‘16]

The structure of ‘natural’ data

Hierarchical Compositionality:



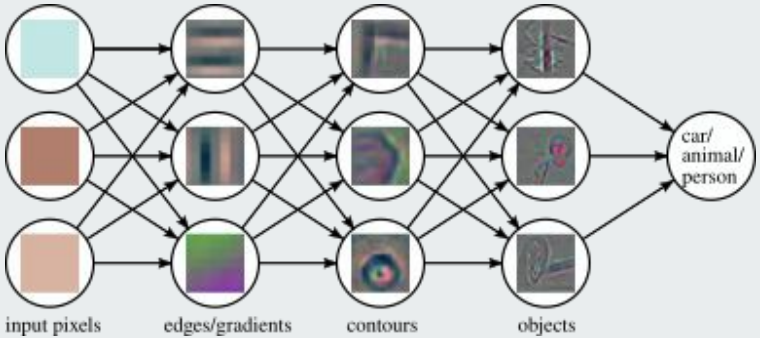
[Goodfellow, Bengio, Courville *Deep Learning* ‘16]

More empirical regularities:

- Hierarchy of features can be found in the weights of trained deep CNNs;
  [Le et al. ‘13, Zeiler, Fergus ‘14, Olah *et al.* ‘20]

The structure of ‘natural’ data

Hierarchical Compositionality:



[Goodfellow, Bengio, Courville *Deep Learning* ‘16]

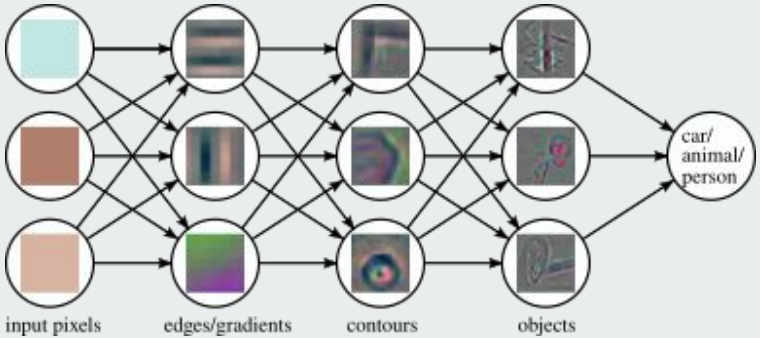
More empirical regularities:

- Hierarchy of features can be found in the weights of trained deep CNNs;
  [Le et al. ‘13, Zeiler, Fergus ‘14, Olah *et al.* ‘20]

- Intuitive explanation of:
  - advantage of deep over shallow (e.g. kernels, two-layer percep.);
  - reduction of dimensionality of representations;
    [Ansuini et al., Recanatesi et al. ‘19]

The structure of ‘natural’ data

Hierarchical Compositionality:



[Goodfellow, Bengio, Courville *Deep Learning* ‘16]

More empirical regularities:

- Hierarchy of features can be found in the weights of trained deep CNNs;
  [Le et al. ‘13, Zeiler, Fergus ‘14, Olah *et al.* ‘20]
- Intuitive explanation of:
  - advantage of deep over shallow (e.g. kernels, two-layer percep.);
  - reduction of dimensionality of representations;
    [Ansuini et al., Recanatesi et al. ‘19]
- Hierarchy of contextual information found also in trained LLMs;
  [Peters *et al.* ‘18, Tenney *et al.* ‘19, Manning *et al.* ‘20]

Outline of the talk

1. Introduce an ensemble of hierarchically compositional datasets as a `toy' model of learnable data;

Outline of the talk

1. Introduce an ensemble of hierarchically compositional datasets as a `toy' model of learnable data;

2. Build a theory of deep representation learning based on data correlations and test with modern ML methods;

Outline of the talk

1. Introduce an ensemble of hierarchically compositional datasets as a `toy' model of learnable data;

2. Build a theory of deep representation learning based on data correlations and test with modern ML methods;

3. Extrapolate predictions to test with real benchmark datasets In computer vision and language modelling.

Outline of the talk

1. Introduce an ensemble of hierarchically compositional datasets as a `toy' model of learnable data;

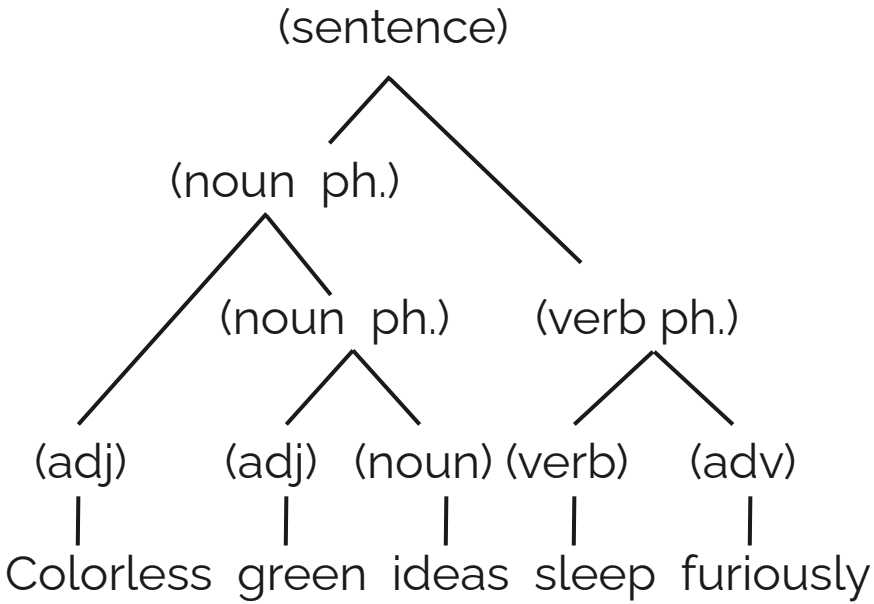
2. Build a theory of deep representation learning based on data correlations and test with modern ML methods;

3. Extrapolate predictions to test with real benchmark datasets In computer vision and language modelling.

[Cagnetta *et al.*, *How deep neural networks learn compositional data*, PRX '24]
[Cagnetta, Wyart, *Towards a theory of how the structure of language…*, NeurIPS24]
[Favero *et al.*, *How compositional generalization and creativity improve …*, arXiv:2502.12089]
[More very soon!]

Probabilistic Context-Free Grammars as models of structured data

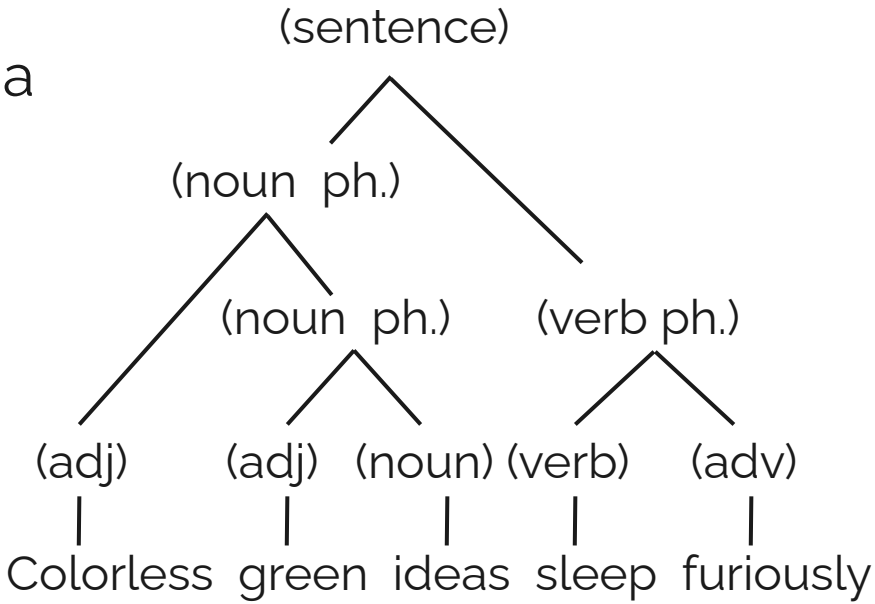
(Probabilistic) Context-Free Grammars



(Probabilistic) Context-Free Grammars

Introduced to model syntax via a generative process

[Chomsky ‘56, Tesnière ‘60]



(Probabilistic) Context-Free Grammars

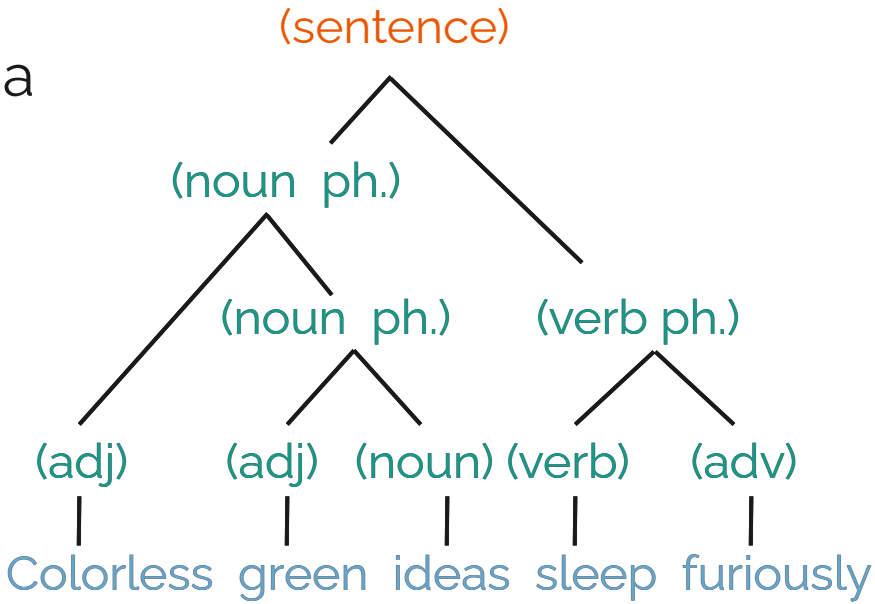
Introduced to model syntax via a generative process

[Chomsky '56, Tesnière '60]

- Starting symbol (root);
- Nonterminals (hidden nodes);
- Terminals (leaves);
- Production rules (branches);

e.g.

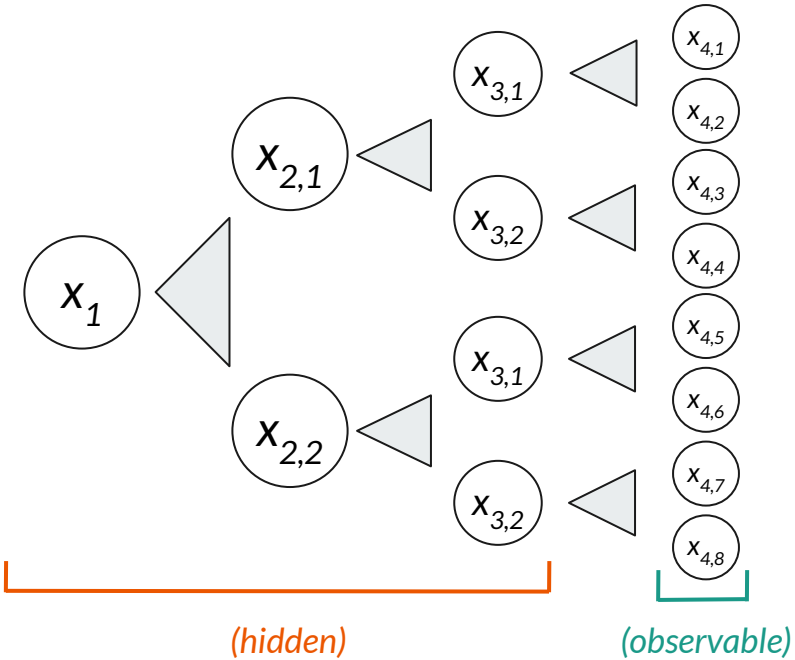
NP -> Adj + N,
S -> NP + VP



PCFGs as generalised Markov Processes

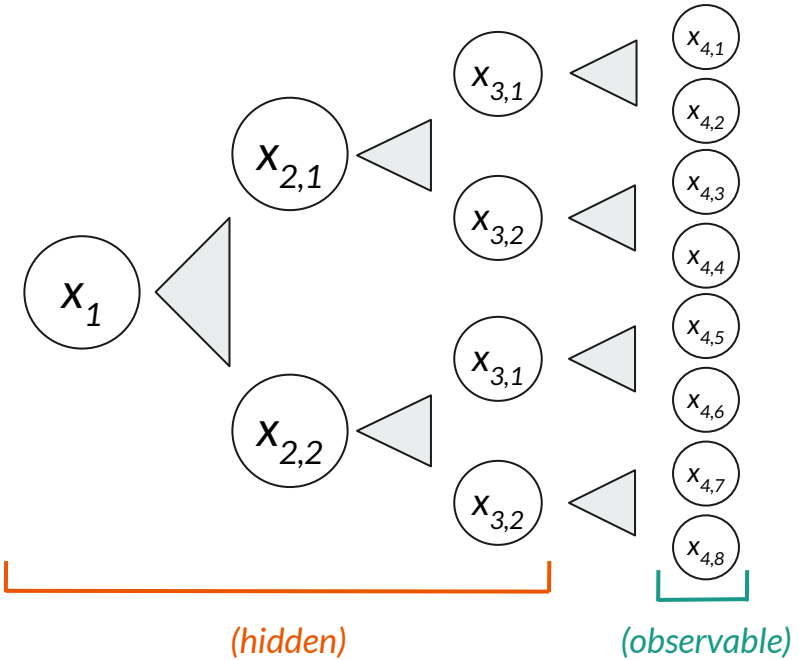
PCFGs as generalised Markov Processes

- Add an extra dimension by expanding the number of variables:



PCFGs as generalised Markov Processes

- Add an extra dimension by expanding the number of variables:

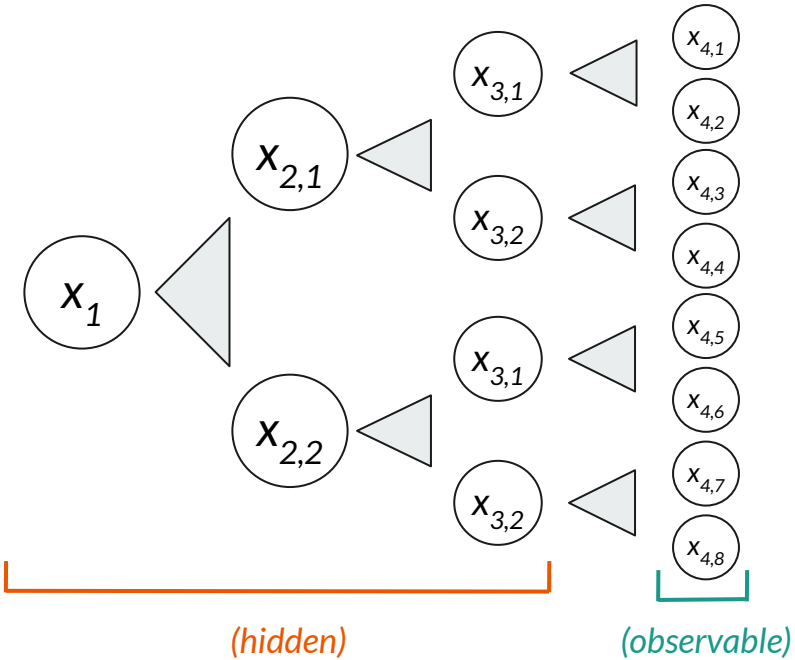


Higher expressivity than Markov models, power-law correlations!

[Ebeling, W., & Pöschel, T., '94]

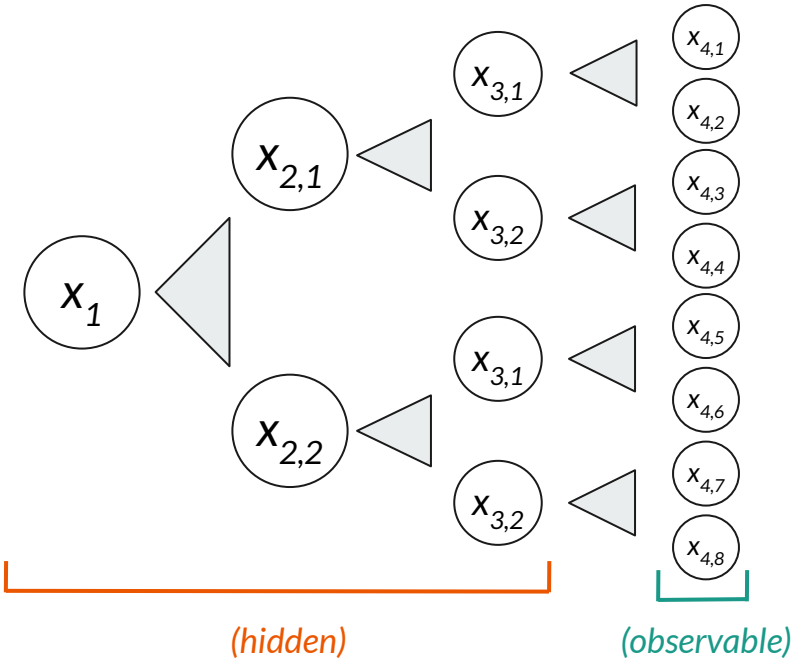
The Random Hierarchy Model [Cagnetta *et al.*, PRX '24]

Ensemble of PCFGs sampled uniformly with the following constraints:



The Random Hierarchy Model [Cagnetta *et al.*, PRX '24]

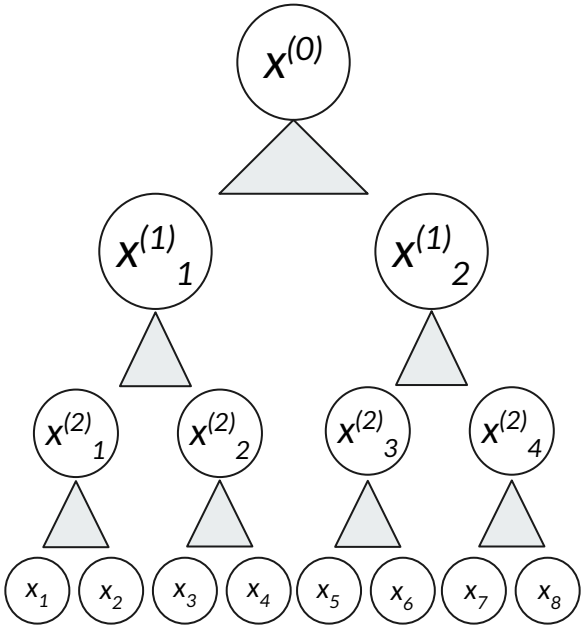
Ensemble of PCFGs sampled uniformly with the following constraints:



- **Regular tree topology** (depth **L**, branching factor **s**);
- Same vocab. size **v** for all symbols;
- **Unambiguous** production rules;
- **m** equiprobable rules per hidden symbol.

The Random Hierarchy Model [Cagnetta *et al.*, PRX '24]

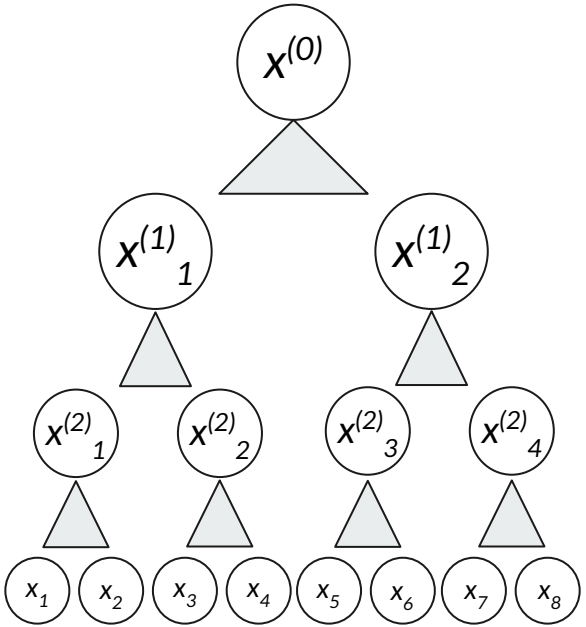
RHM: *ensemble* of hierarchically compositional tasks with fixed parameters



The Random Hierarchy Model [Cagnetta *et al.*, PRX '24]

RHM: *ensemble* of hierarchically compositional tasks with fixed parameters

- **Classification:** predict the root from the leaves;
- **MLM/Next-token prediction:** predict missing leaves from visible ones (reconstruction of joint leaves prob.)

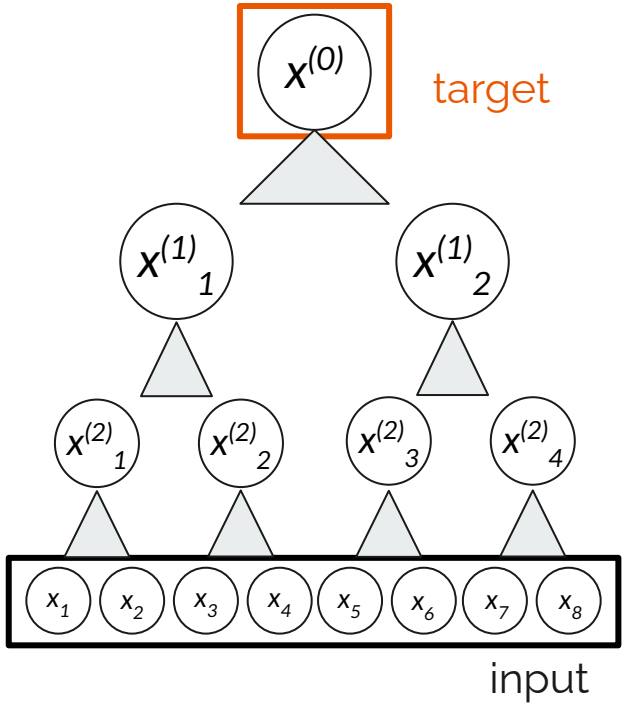


The Random Hierarchy Model [Cagnetta *et al.*, PRX '24]

RHM: *ensemble* of hierarchically compositional tasks with fixed parameters

- **Classification:** predict the root from the leaves;

- **MLM/Next-token prediction:** predict missing leaves from visible ones (reconstruction of joint leaves prob.)

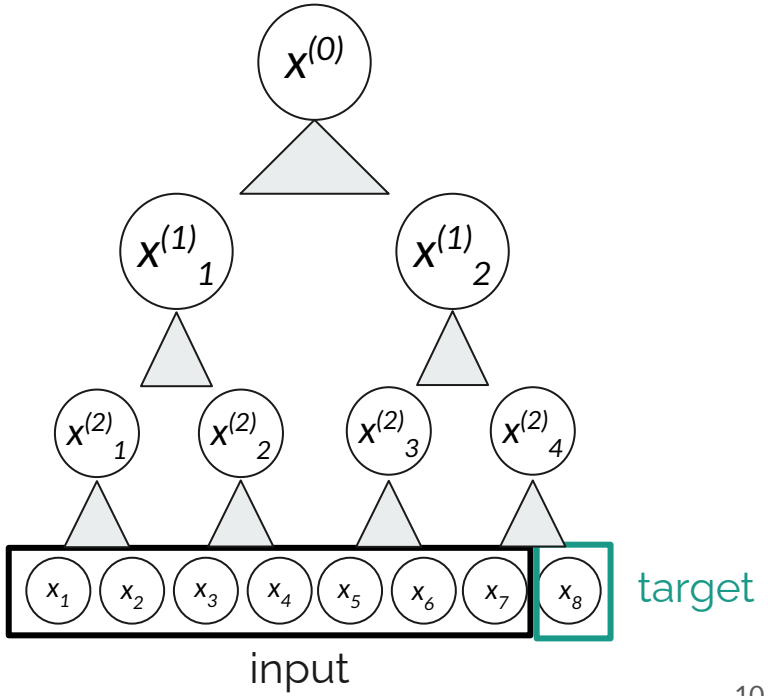


The Random Hierarchy Model [Cagnetta, Wyart, NIPS '24]

RHM: *ensemble* of hierarchically compositional tasks with fixed parameters

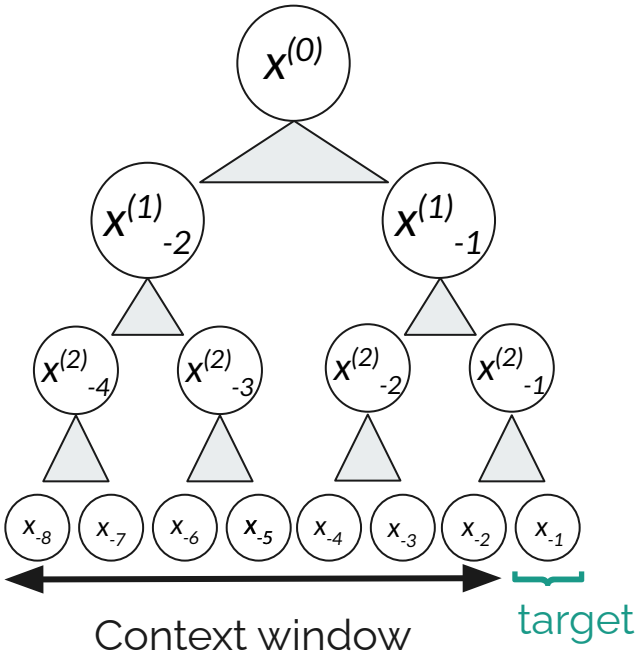
- **Classification:** predict the root from the leaves;

- **MLM/Next-token prediction:** predict missing leaves from visible ones (reconstruction of joint leaves prob.)



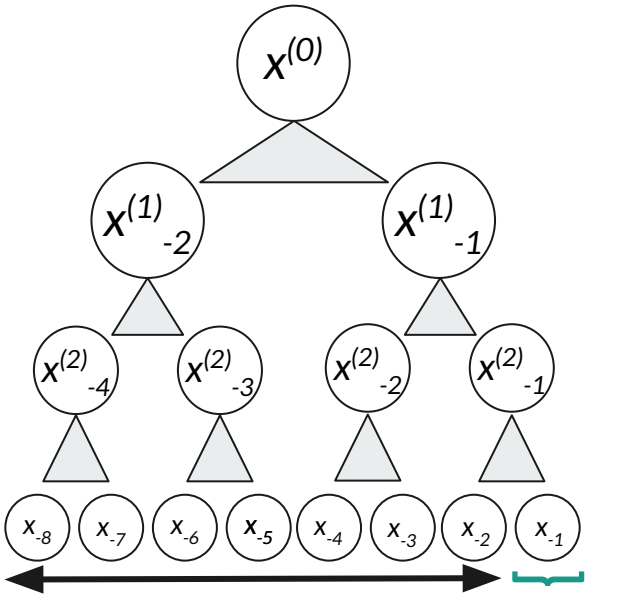
Learning the RHM via next-token prediction

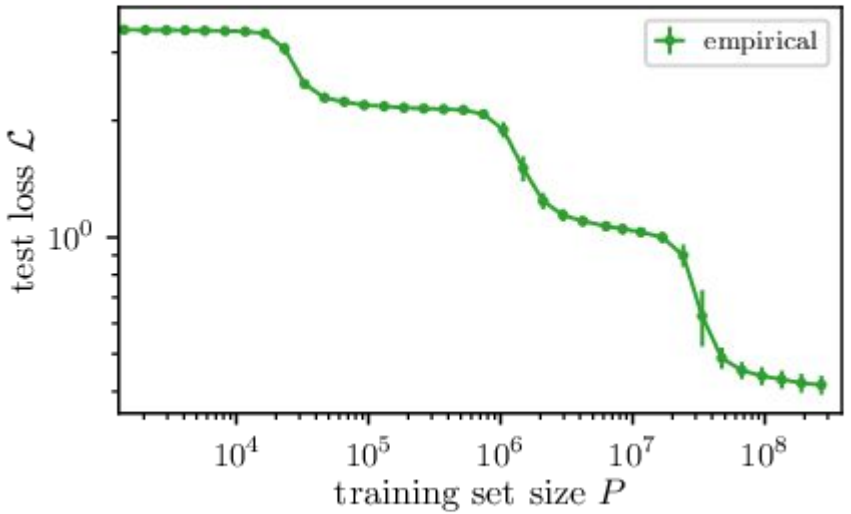
Learning the RHM via next-token prediction



- Sampe RHM instance and P training data;
- Train neural network to approximate $P(\ x_{-1} \mid x_{-2}, \ldots, x_{-8})$;
- Measure performance via cross-entropy (KL div. with *true* RHM distribution);

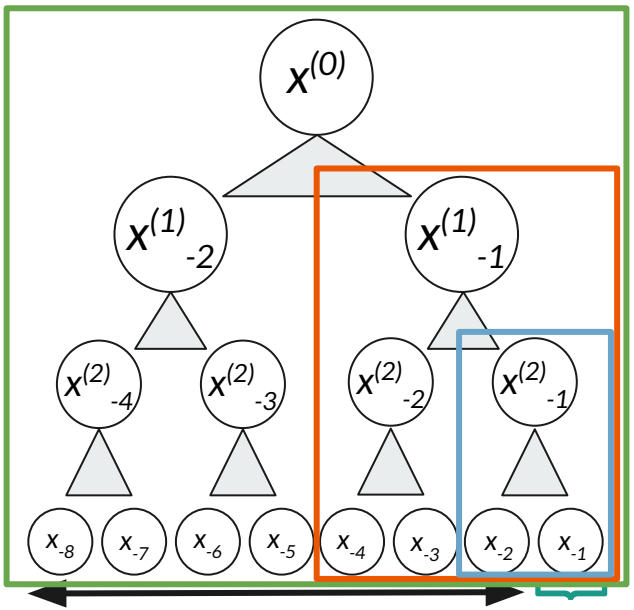
Learning the RHM via next-token prediction

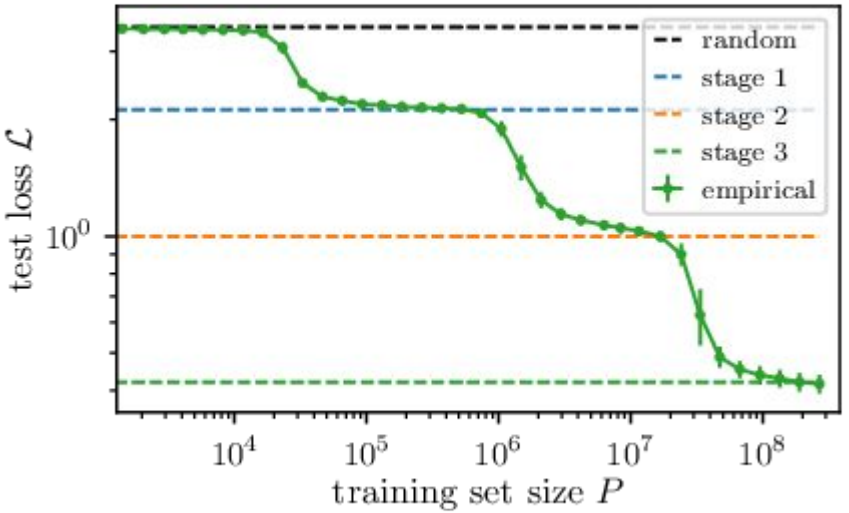




depth 3 transformer trained with adam

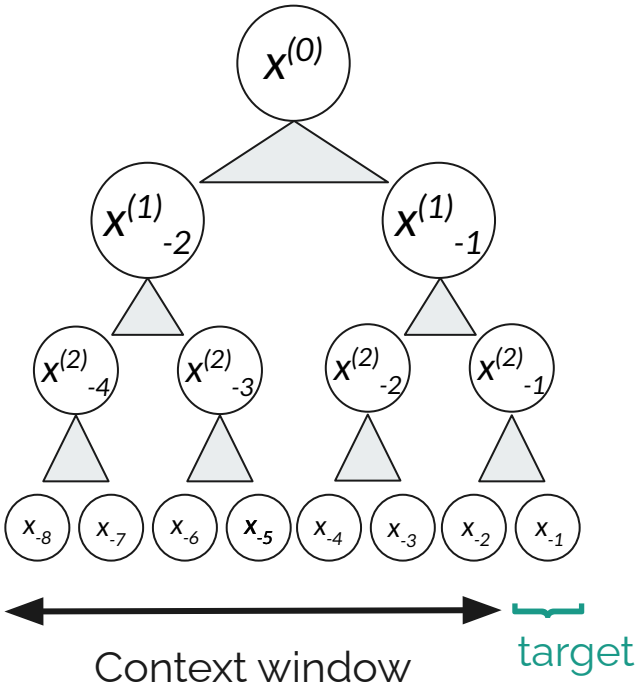
Learning the RHM via next-token prediction





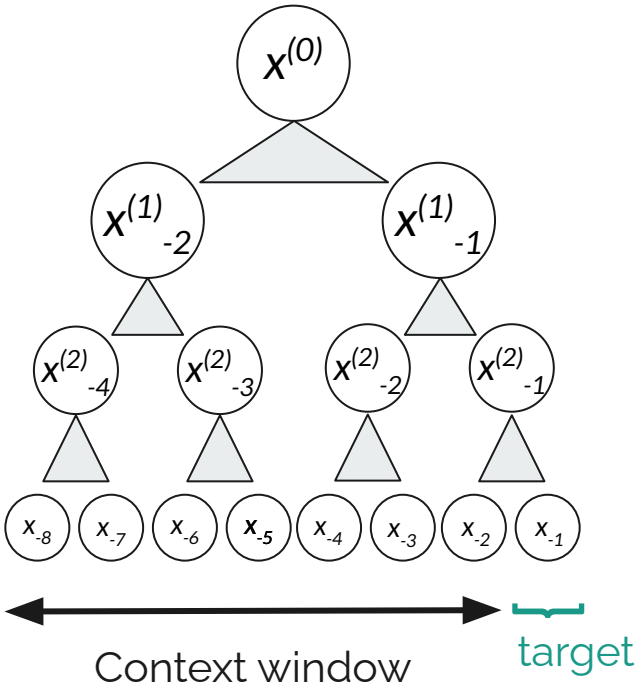
depth 3 transformer trained with adam

Step 1: Flat vs. Hierarchical approximation



N-gram strategy: simply count occurrences of $(x_{-1}, x_{-2}, \ldots, x_{-d})$

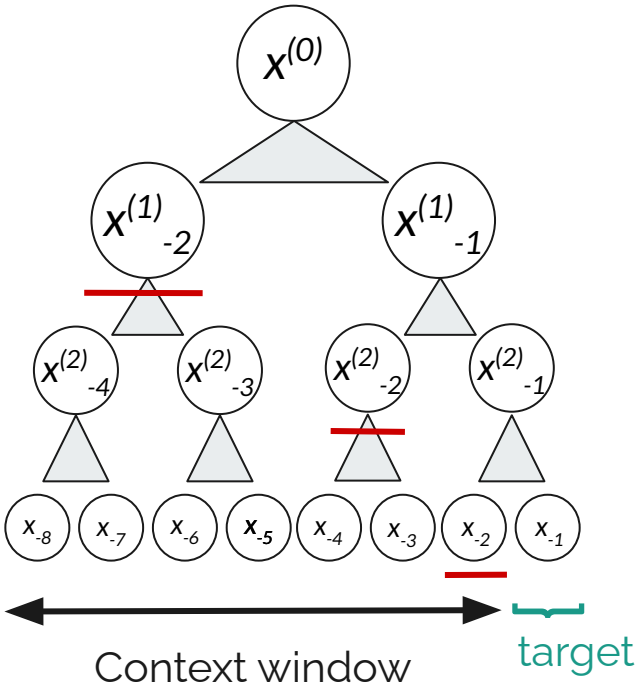
Step 1: Flat vs. Hierarchical approximation



N-gram strategy: simply count occurrences of $(x_{-1}, x_{-2}, \ldots, x_{-d})$

- Agnostic of the tree structure;
- Number of contexts grows exponentially with dim.;

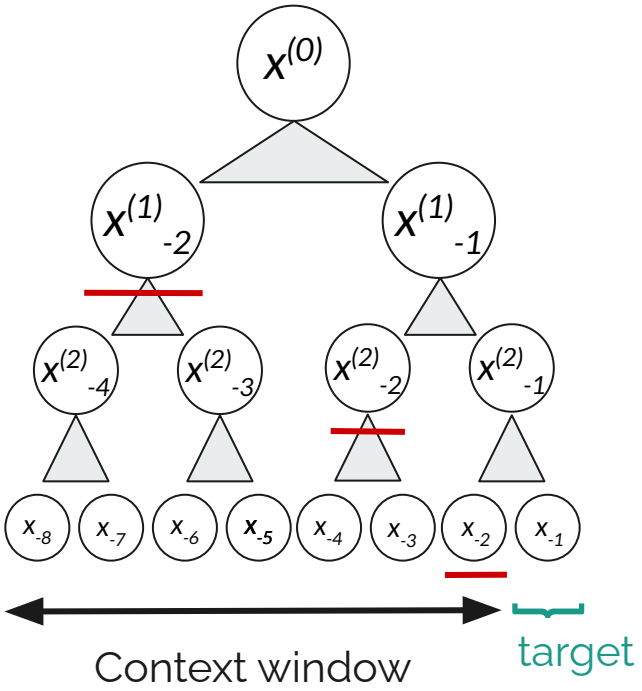
Step 1: Flat vs. Hierarchical approximation



Hierarchical N-gram: count occurrences in latent space

$(X_{-1}, X_{-2}, X_{-2}^{(2)}, X_{-2}^{(1)}, \ldots)$

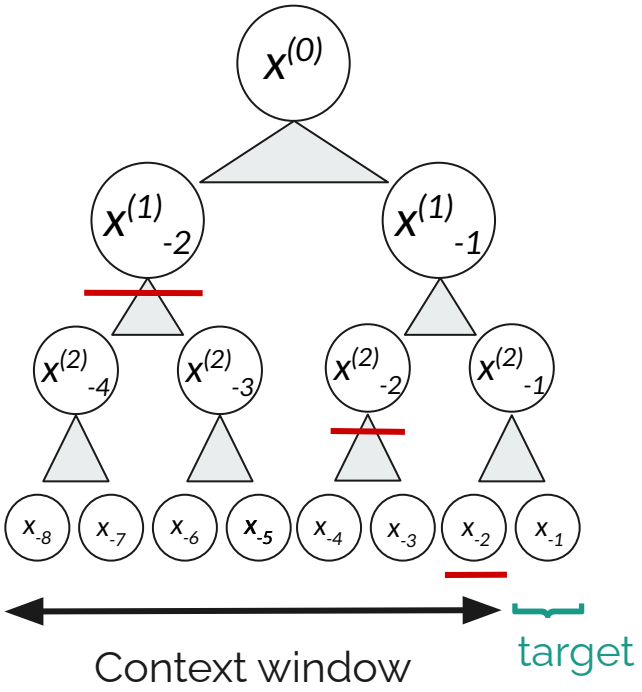
Step 1: Flat vs. Hierarchical approximation



Hierarchical N-gram: count occurrences in latent space $(X_{-1}, X_{-2}, X_{-2}^{(2)}, X_{-2}^{(1)}, \ldots)$

- Uses minimal number of variables that influence target;
- Number of contexts grows exponentially with depth.;

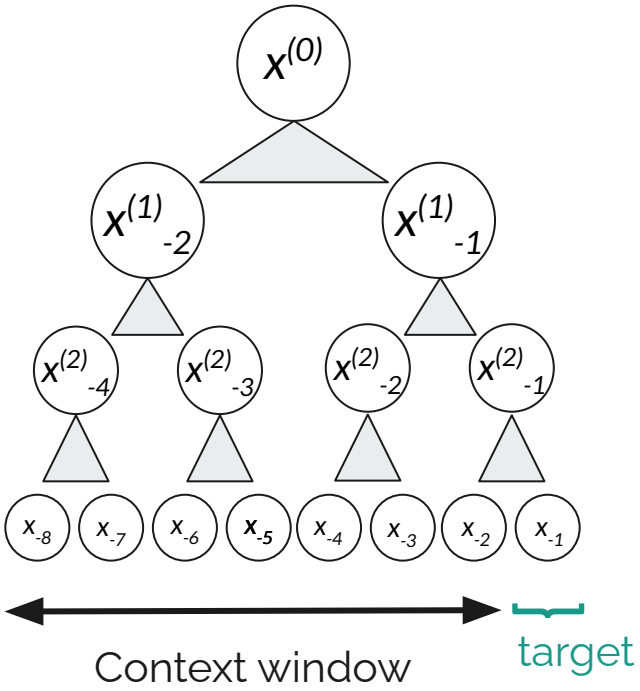
Step 1: Flat vs. Hierarchical approximation



Hierarchical N-gram: count occurrences in latent space $(X_{-1}, X_{-2}, X_{-2}^{(2)}, X_{-2}^{(1)}, \ldots)$

- Uses minimal number of variables that influence target;
- Number of contexts grows exponentially with depth.;
- Requires latent structure!

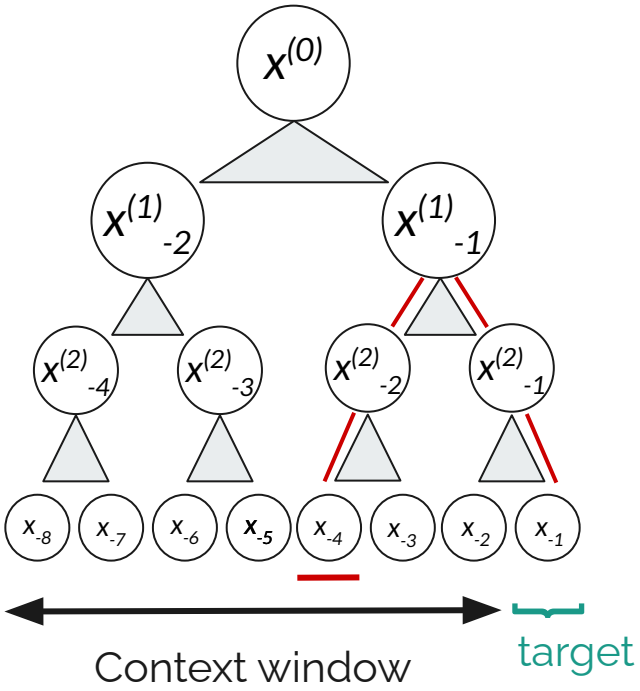
Step 2: latent structure via correlations



$\mathbb{P}\{\mathbf{X}\}$ = probability of a sentence, token-token correlations:

$$C_t(\mu,\nu) = \mathbb{P}\{X_{-t}=\mu, X_{-1}=\nu\} - \mathbb{P}\{X_{-t}=\mu\}\,\mathbb{P}\{X_{-1}=\nu\}$$

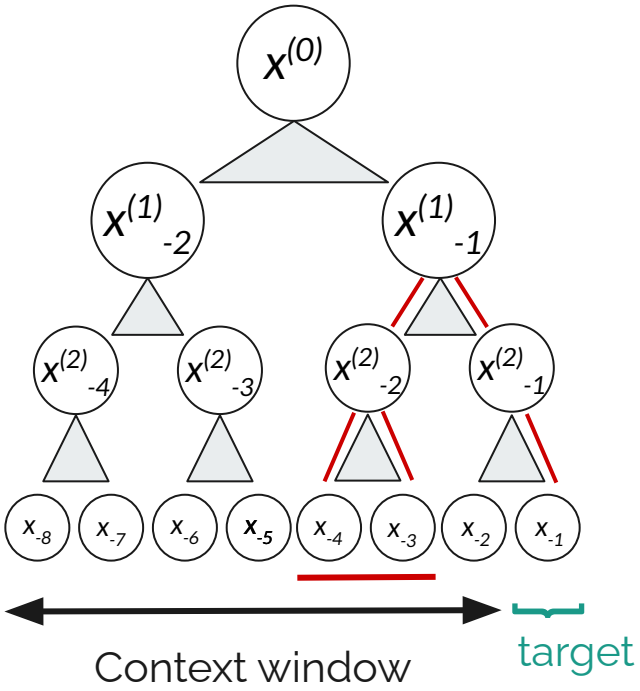
Step 2: latent structure via correlations



$\mathbb{P}\{\mathbf{X}\}$ = probability of a sentence, token-token correlations:

$$C_t(\mu,\nu) = \mathbb{P}\{X_{-t}=\mu, X_{-1}=\nu\} - \mathbb{P}\{X_{-t}=\mu\}\,\mathbb{P}\{X_{-1}=\nu\}$$

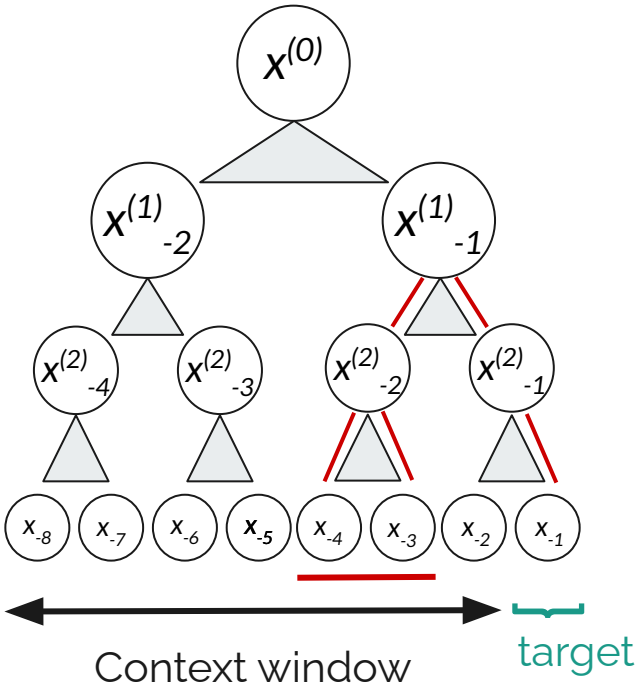
Step 2: latent structure via correlations



$\mathbb{P}\{\mathbf{X}\}$ = probability of a sentence, tuple-token correlations:

$$C_t(\mu,\nu) = \mathbb{P}\{\mathbf{X}_{-t} = \mu, X_{-1} = \nu\} - \mathbb{P}\{\mathbf{X}_{-t} = \mu\}\,\mathbb{P}\{X_{-1} = \nu\}$$

Step 2: latent structure via correlations



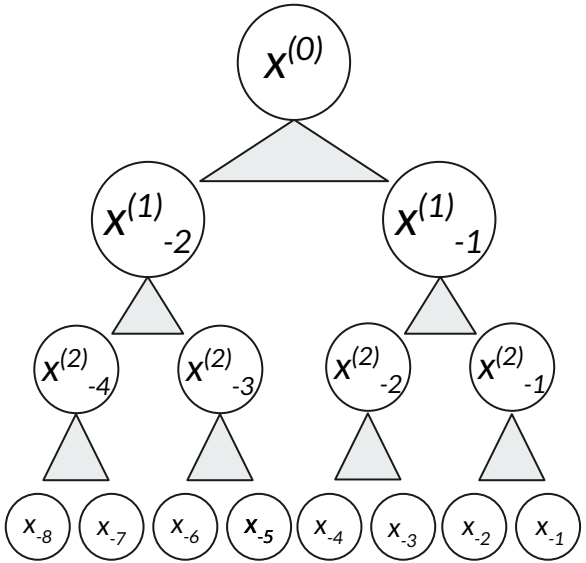
$\mathbb{P}\{\mathbf{X}\}$ = probability of a sentence,
tuple-token correlations:

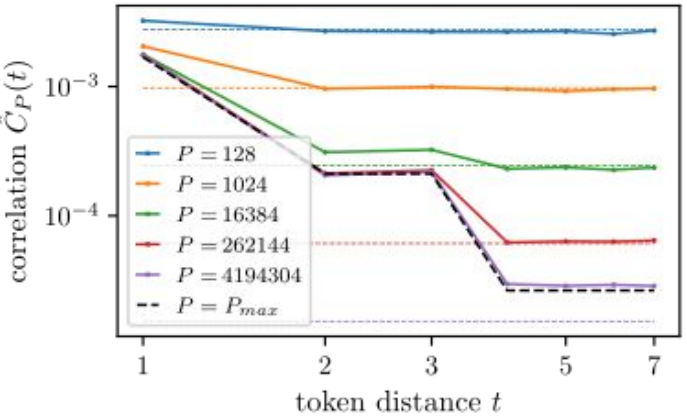
$$C_t(\mu,\nu) = \mathbb{P}\{\mathbf{X}_{-t} = \mu, X_{-1} = \nu\} - \mathbb{P}\{\mathbf{X}_{-t} = \mu\}\,\mathbb{P}\{X_{-1} = \nu\}$$

Function of the latent variable $x^{(2)}_{-t}$
(above $\boldsymbol{X}_{-t}$)

Step 2: latent structure via correlations

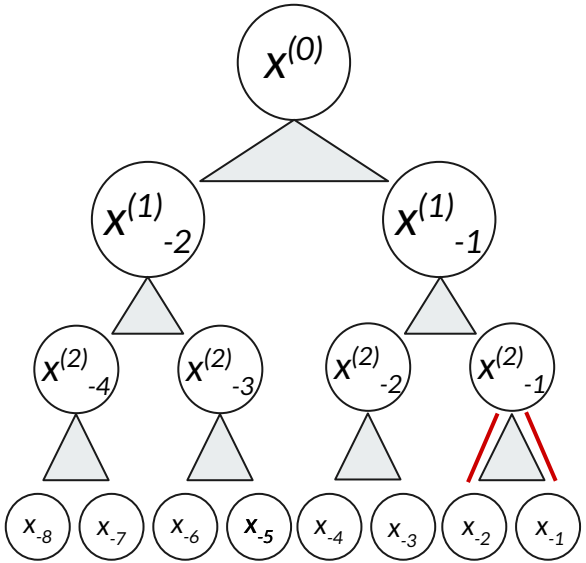
ASSUMPTION 1: a latent variable is available if the corresponding correlations are resolved in the training data.

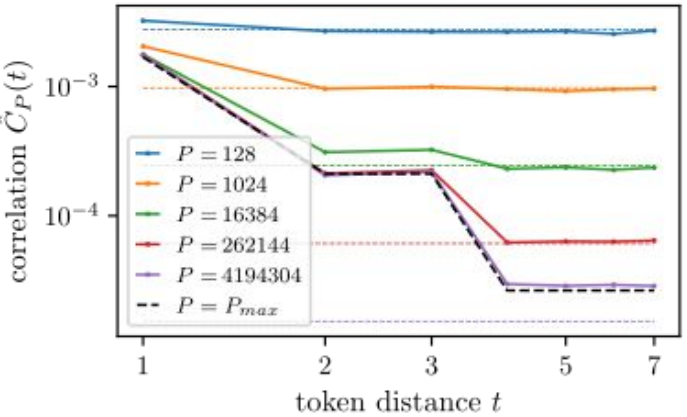




Step 2: latent structure via correlations

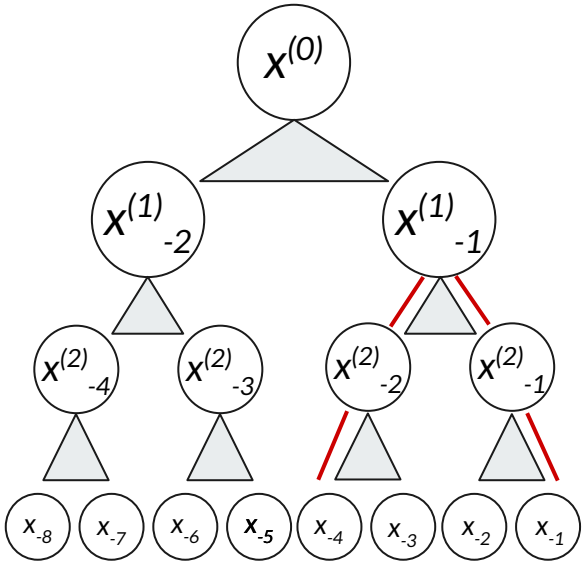
ASSUMPTION 1: a latent variable is available if the corresponding correlations are resolved in the training data.

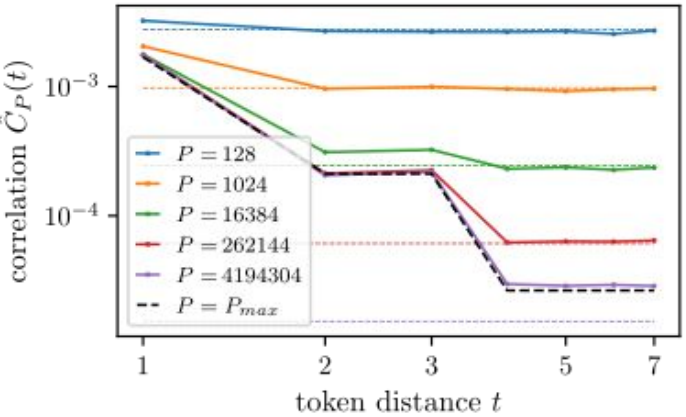




Step 2: latent structure via correlations

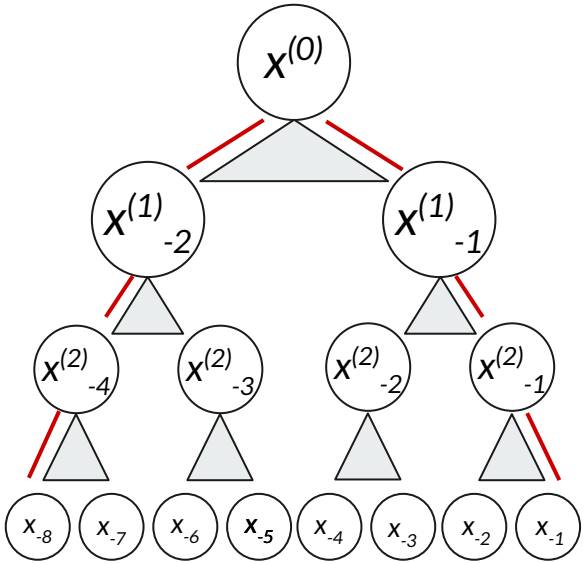
ASSUMPTION 1: a latent variable is available if the corresponding correlations are resolved in the training data.

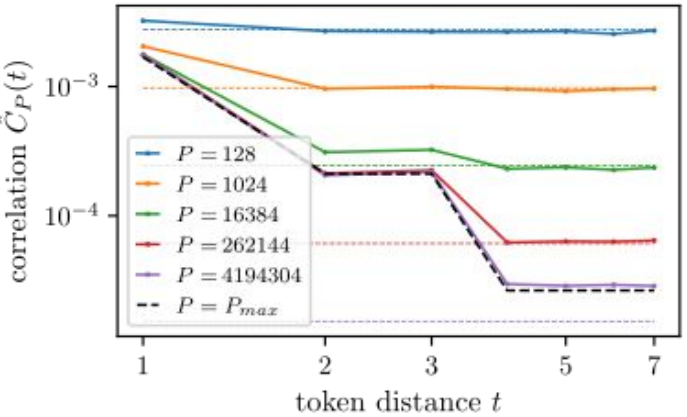




Step 2: latent structure via correlations

ASSUMPTION 1: a latent variable is available if the corresponding correlations are resolved in the training data.





Step 2: latent structure via correlations

ASSUMPTION 1: a latent variable is available if the corresponding correlations are resolved in the training data.

↓

Variance due to the sampling of the training set $\ll$ Variance between the correlations of different latent variables

↓

$$\frac{1}{(vm)v \times P} \ll \left\langle C_\ell(\mu,\nu)^2 \right\rangle_{RHM} \Rightarrow P \gg P_\ell$$

Step 3: performance-vs-#data

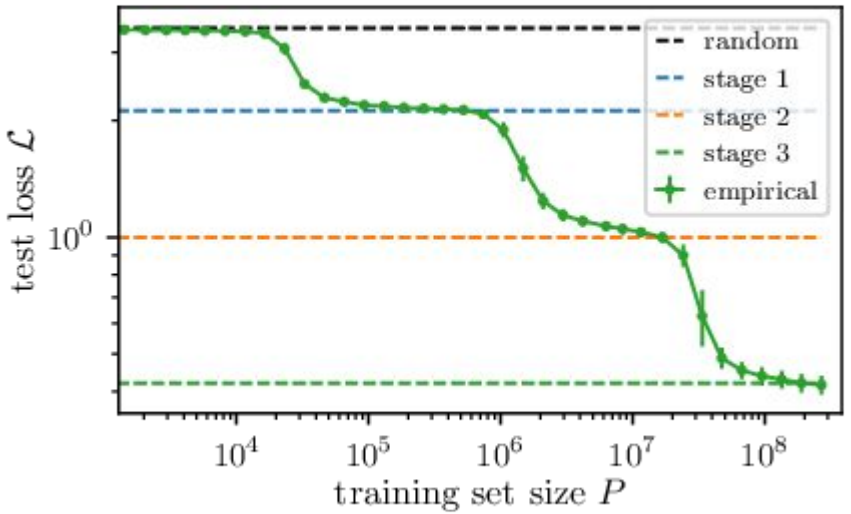
ASSUMPTION 1: a latent variable is available if the corresponding correlations are resolved in the training data ($P > P_\ell$);

ASSUMPTION 2: A ML model trained with $P > P_\ell$ data can use available hidden variables to reproduce $P(X_{-1}|X_{-2},\ldots,X_{-s^\ell})$,

$$\mathcal{L}_\ell = \mathbb{E}_{\mathbf{X}\sim RHM}\left[-\log p(x_{-1}|x_{-2},\ldots,x_{-s^\ell})\right]$$

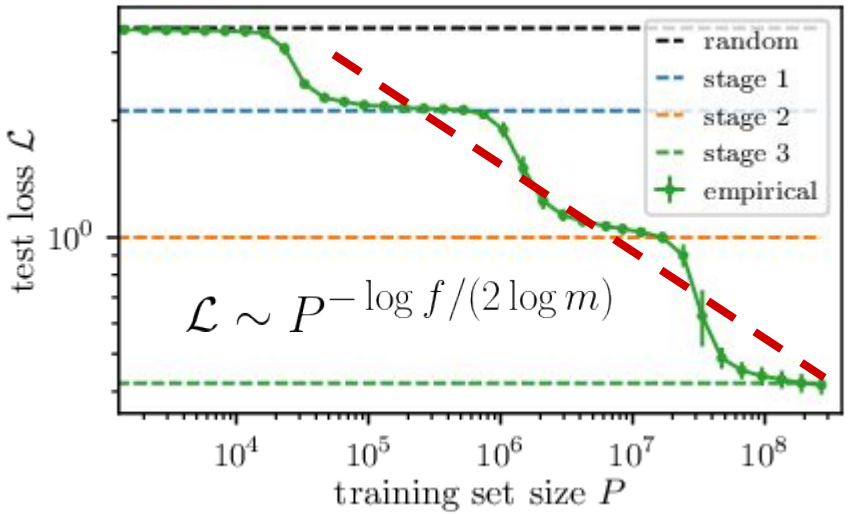
Learning the RHM via next-token prediction

- More training data -> longer range of correlations;
- Longer range -> deeper hidden variables;
- Deeper hidden variables -> better performance;



Learning the RHM via next-token prediction

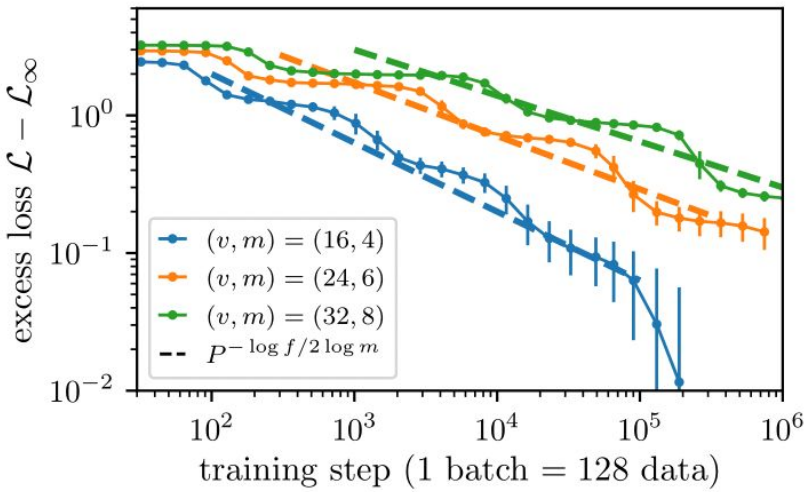
- More training data -> longer range of correlations;
- Longer range -> deeper hidden variables;
- Deeper hidden variables -> better performance;



Scaling law exponent depends on data structure via correlations!

Learning the RHM via next-token prediction

- More training data -> longer range of correlations;
- Longer range -> deeper hidden variables;
- Deeper hidden variables -> better performance;



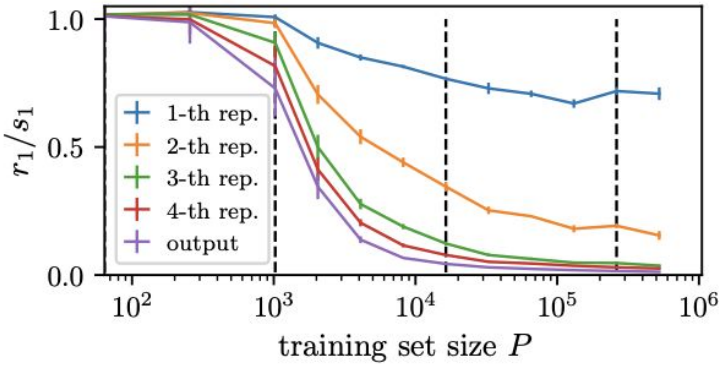
Scaling law exponent depends on data structure via correlations!

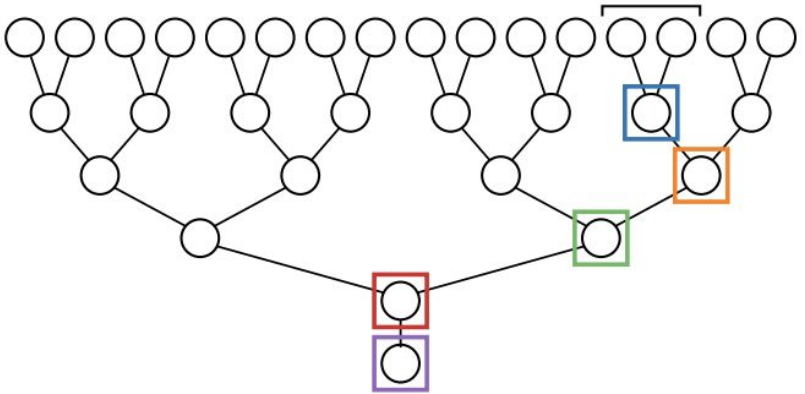
Learning the RHM: the role of depth

Probe network's representations via sensitivity to changing production rules (r) over sensitivity to changing latent variables (s):

Learning the RHM: the role of depth

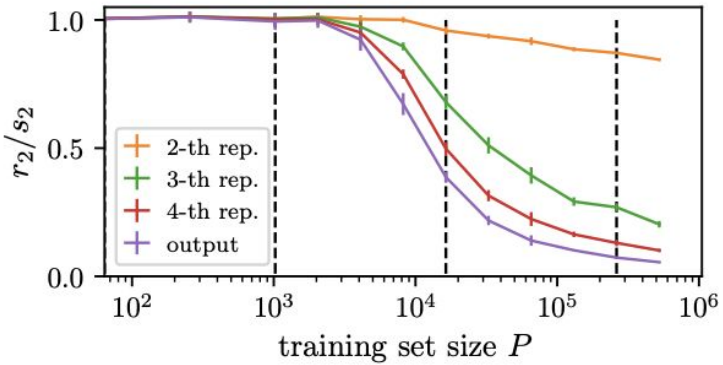
Probe network's representations via sensitivity to changing production rules (r) over sensitivity to changing latent variables (s):

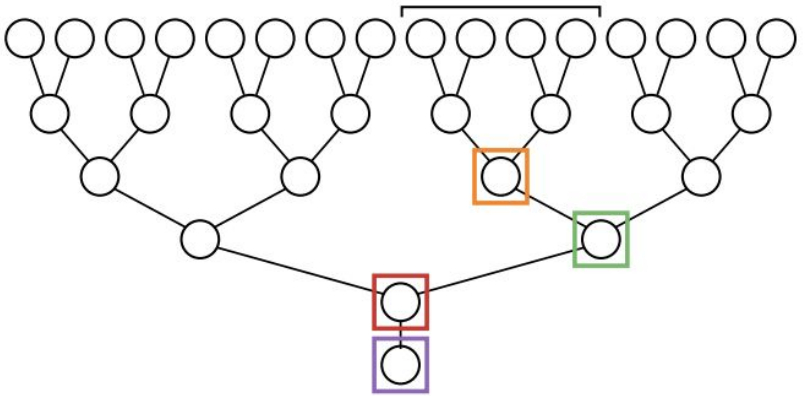




Learning the RHM: the role of depth

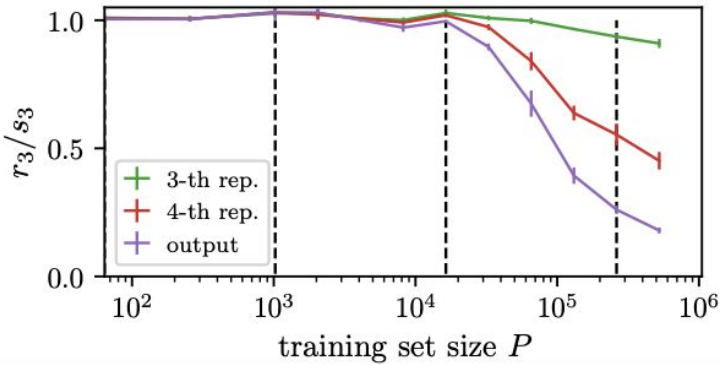
Probe network's representations via sensitivity to changing production rules (r) over sensitivity to changing latent variables (s):

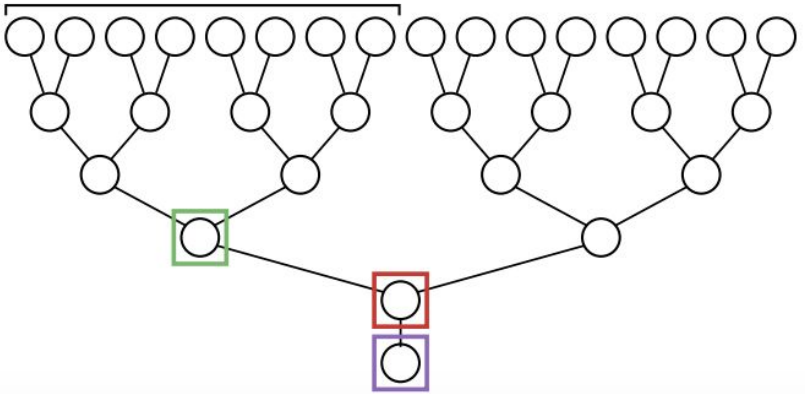




Learning the RHM: the role of depth

Probe network's representations via sensitivity to changing production rules (r) over sensitivity to changing latent variables (s):





RHM: Conclusions

- For hierarchical data **correlations decay with distance** and carry information on the **latent hierarchical structure**;

- ML tasks based on such data can be **learnt efficiently by deep neural networks** that can reconstruct said latent structure;

RHM: Conclusions

- For hierarchical data **correlations decay with distance** and carry information on the **latent hierarchical structure**;

- ML tasks based on such data can be **learnt efficiently by deep neural networks** that can reconstruct said latent structure;

- Beyond the example discussed here: **classification (advantage of depth)**, **score-based diffusion (creativity vs memorisation)**, **non-uniform production rule probabilities (role of features distribution in scaling)**, **varying tree topology;**

RHM: Conclusions

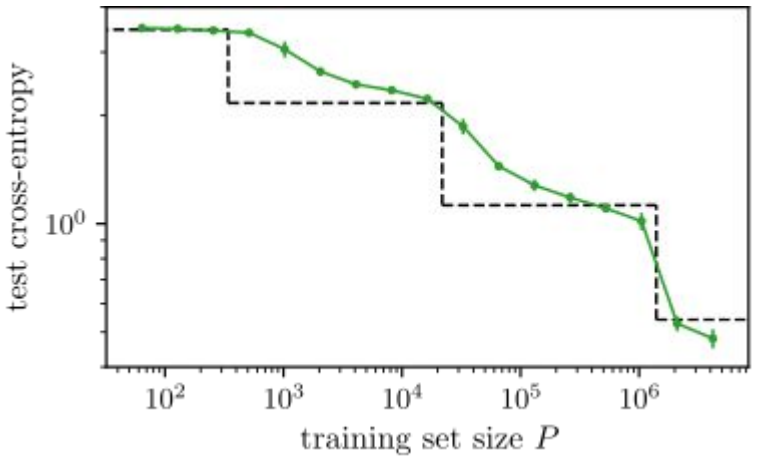
- For hierarchical data **correlations decay with distance** and carry information on the **latent hierarchical structure**;

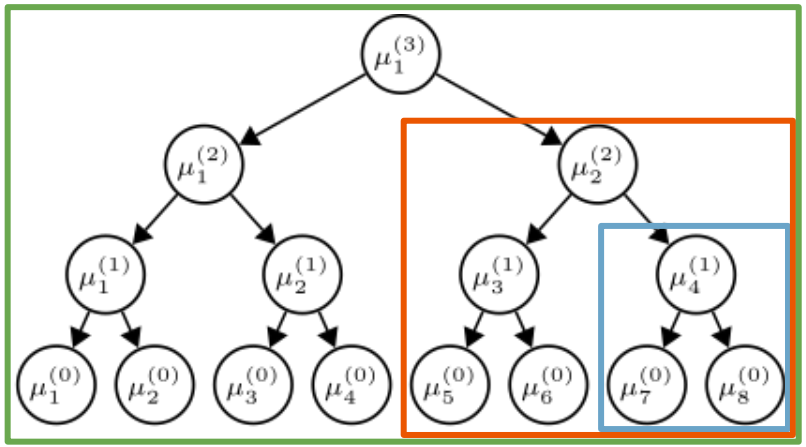
- ML tasks based on such data can be **learnt efficiently by deep neural networks** that can reconstruct said latent structure;

- Beyond the example discussed here: **classification (advantage of depth)**, **score-based diffusion (creativity vs memorisation)**, **non-uniform production rule probabilities (role of features distribution in scaling)**, **varying tree topology;**

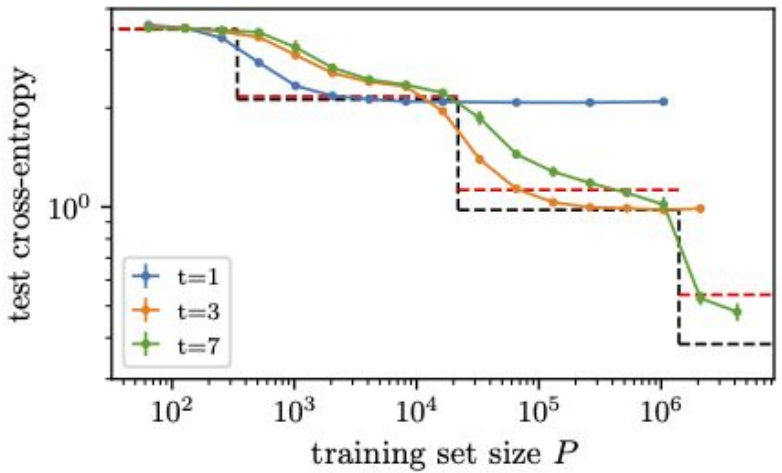
  - **What about real data?**

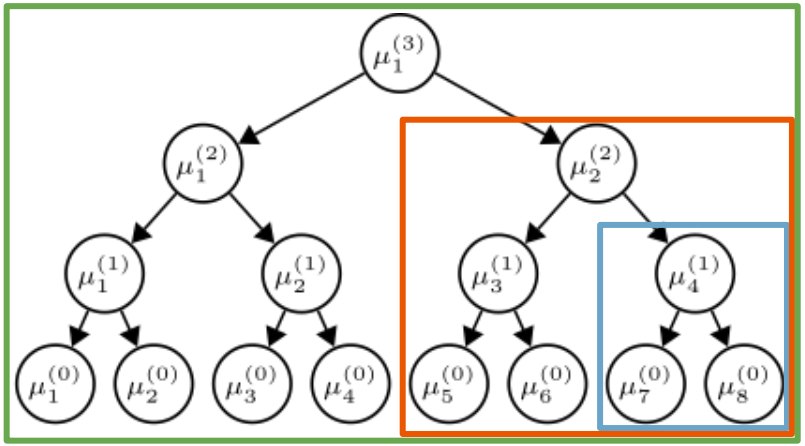
Saturation of performance due to finite context



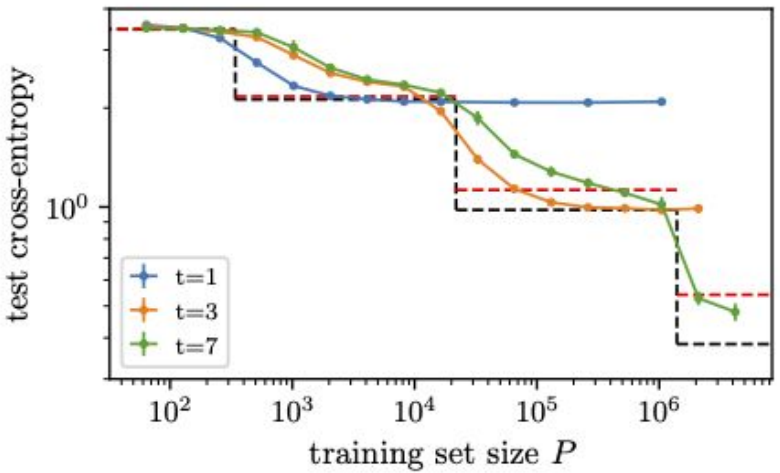


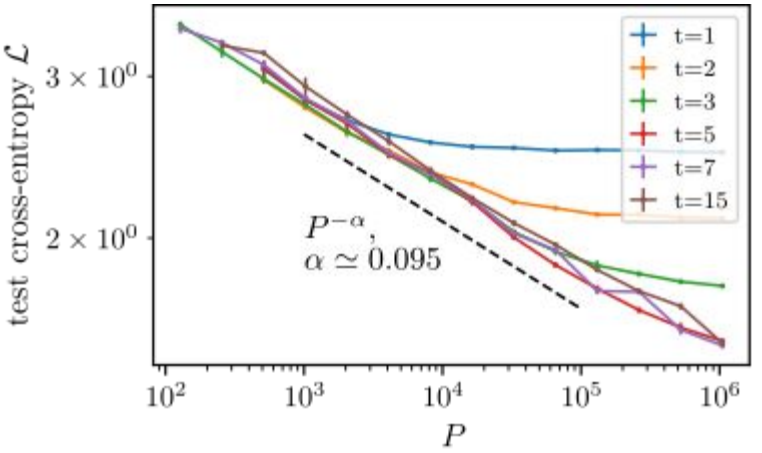
Saturation of performance due to finite context





Saturation of performance due to finite context

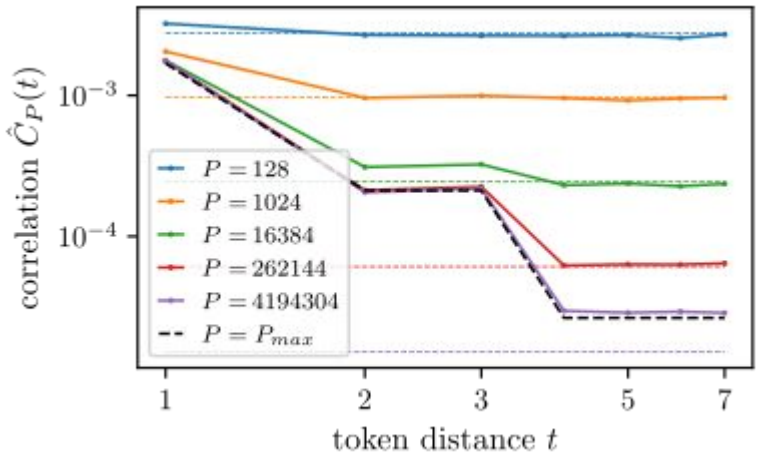


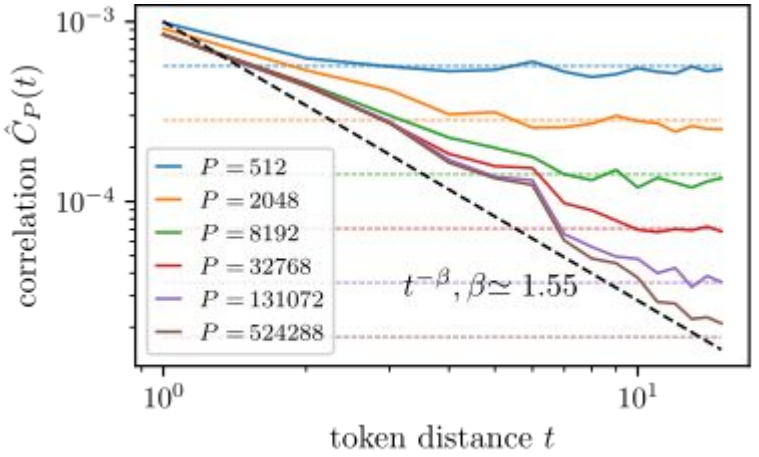


Left: 3 multi-head attention layers trained on the RHM dataset;

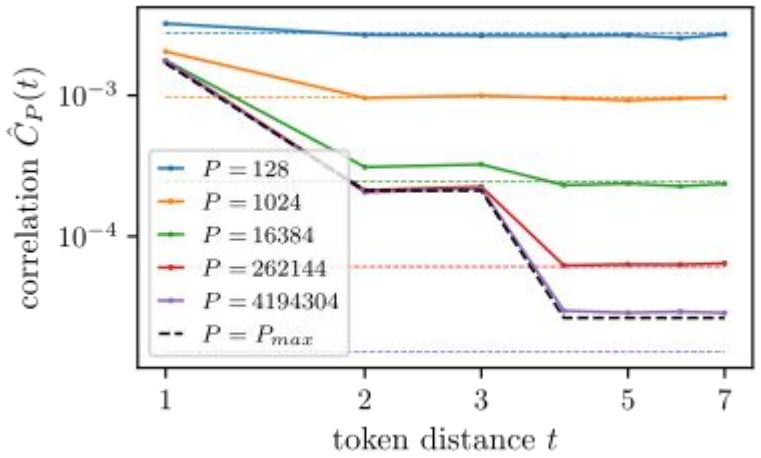
Right: 6-layers encoder only transformer trained on a character-based tokenisation of WikiText-105 [Merity *et al.* '17]

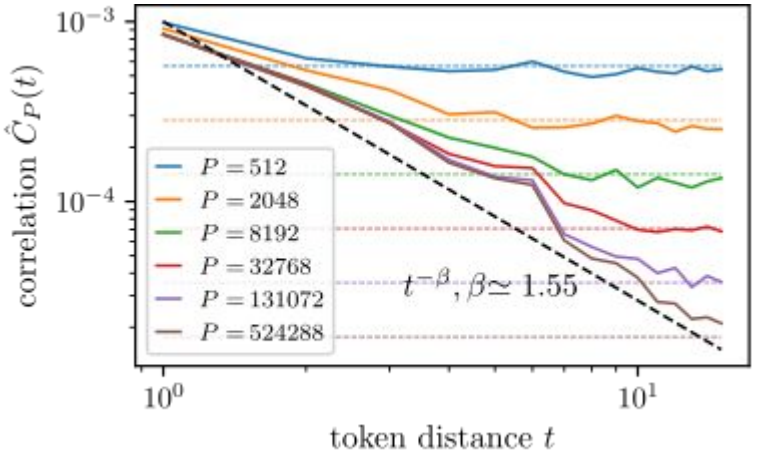
Saturation of correlations due to finite data





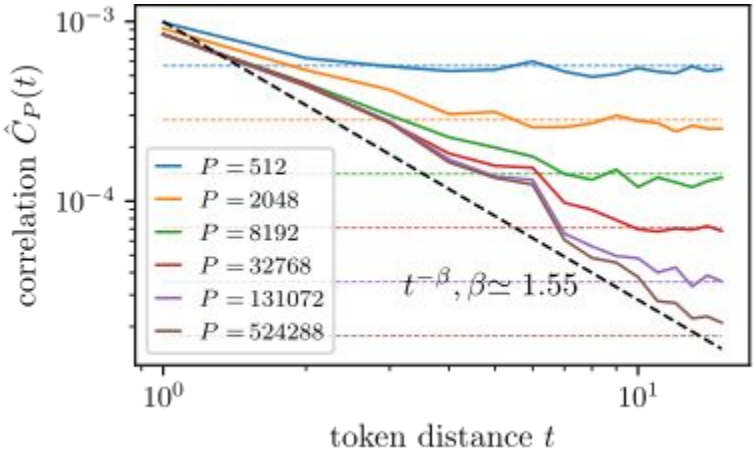
Saturation of correlations due to finite data





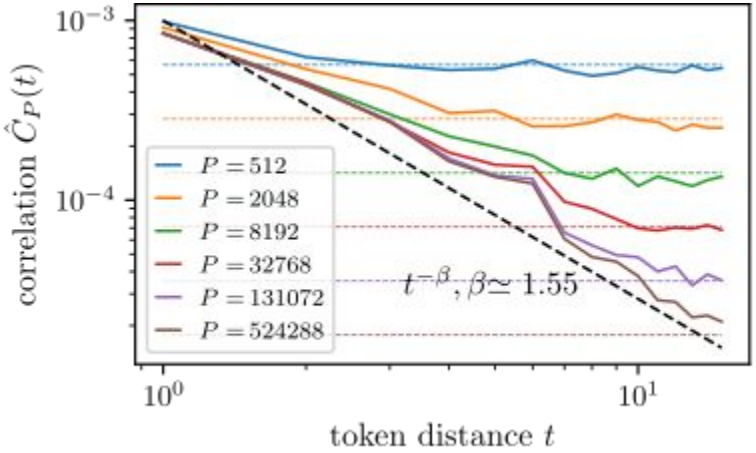
Conjecture: finite data = effective context window

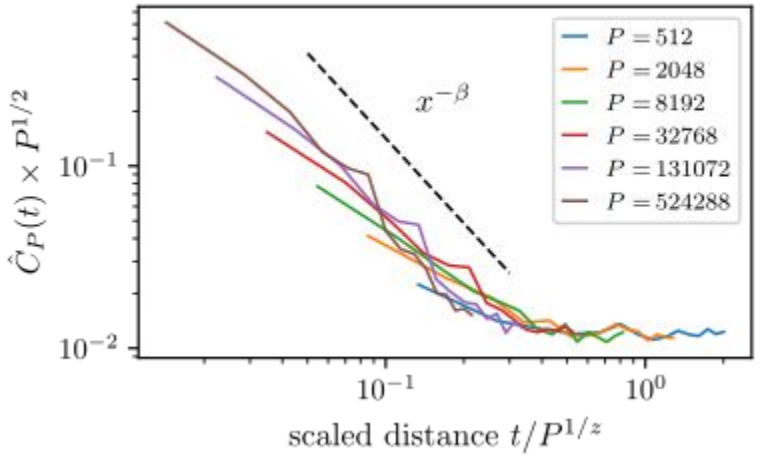
Scaling theory of language modelling



$$C(t) \sim t^{-b}, \quad \text{noise} \sim 1/\sqrt{P}, \quad t^* \sim P^{1/z} \, (z = 2b)$$

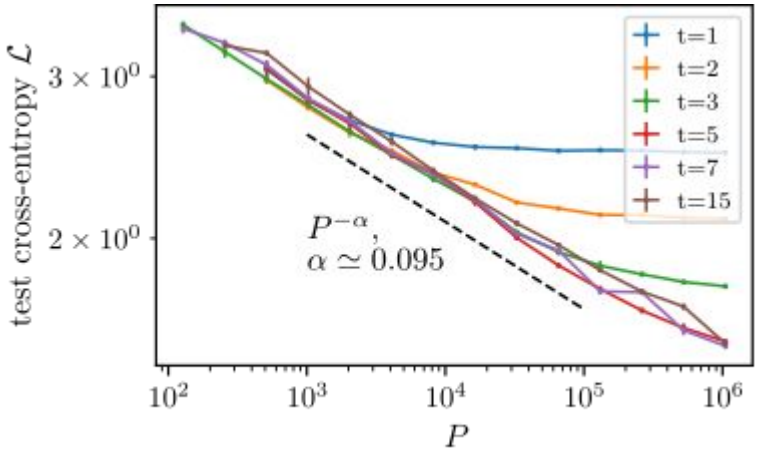
Scaling theory of language modelling



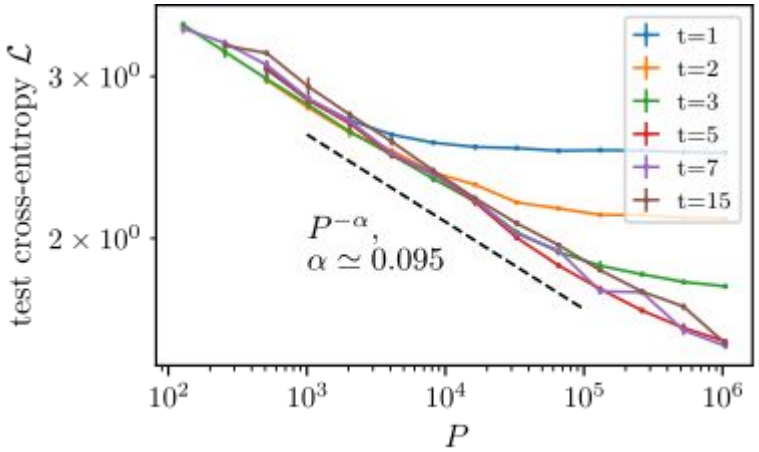


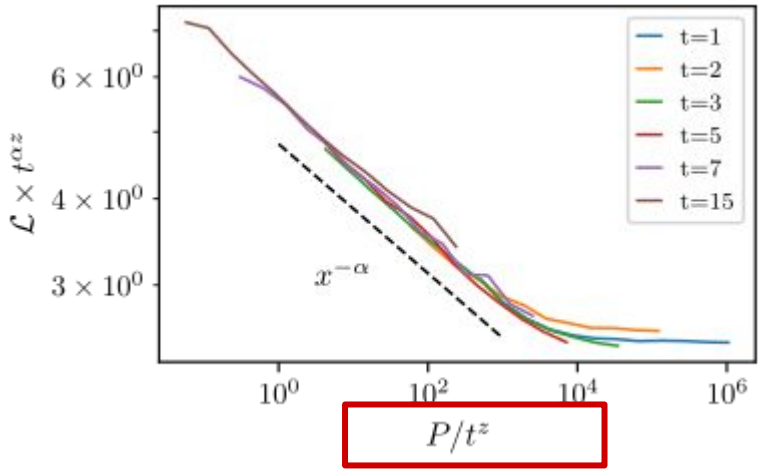
$$C(t) \sim t^{-b}, \quad \text{noise} \sim 1/\sqrt{P}, \quad t^* \sim P^{1/z} \, (z = 2b)$$

Scaling theory of language modelling

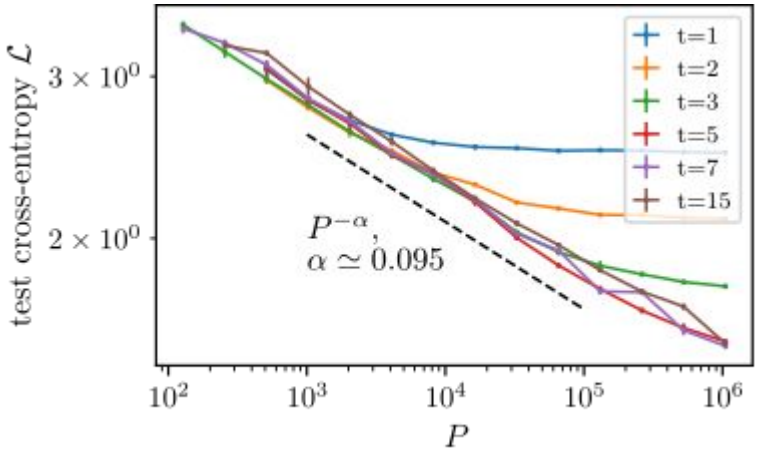


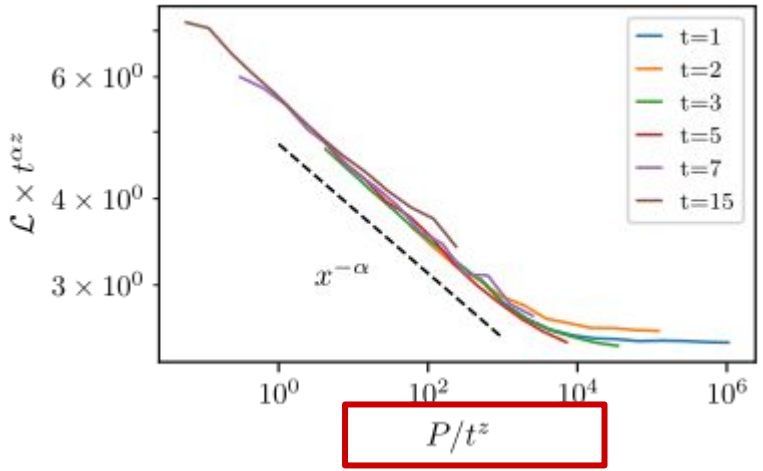
Scaling theory of language modelling





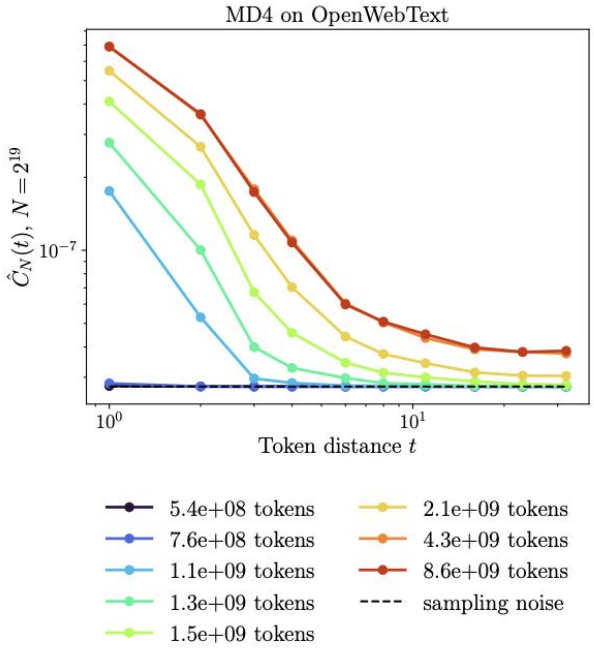
Scaling theory of language modelling





Same for dataset of Shakespeare's lines

Correlations generated by a diffusion model



(b) Correlations in the generated text.

Correlations generated by a diffusion model

10^8 **training tokens**

In popular spokesman typeted in diversity adventure allow price Zha Tampa usually Pages superstays's under leveldowns swim a cycle who retains highly weapons batch floor despite

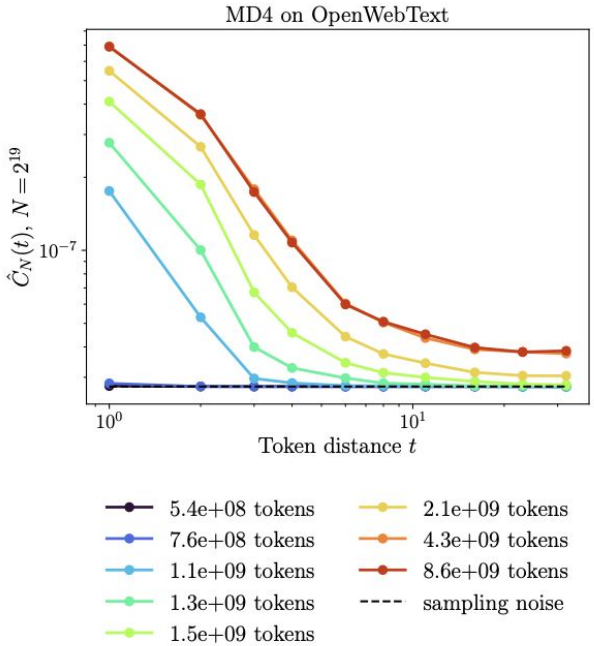
10^9 **training tokens**

Just like you are growing fast and growing strong. But this way you became organic, changed someone else 2019s. But even then you made them off. I sort came to smile around, because I was in China okay.

10^{10} **training tokens**

At the beginning of winter when I walked around; even if he would be talking to me, on the highest field and back in the second round in my team I would take him over in his cell because it was my game against Juventus.

(a) Text generated at different training stages.



(b) Correlations in the generated text.