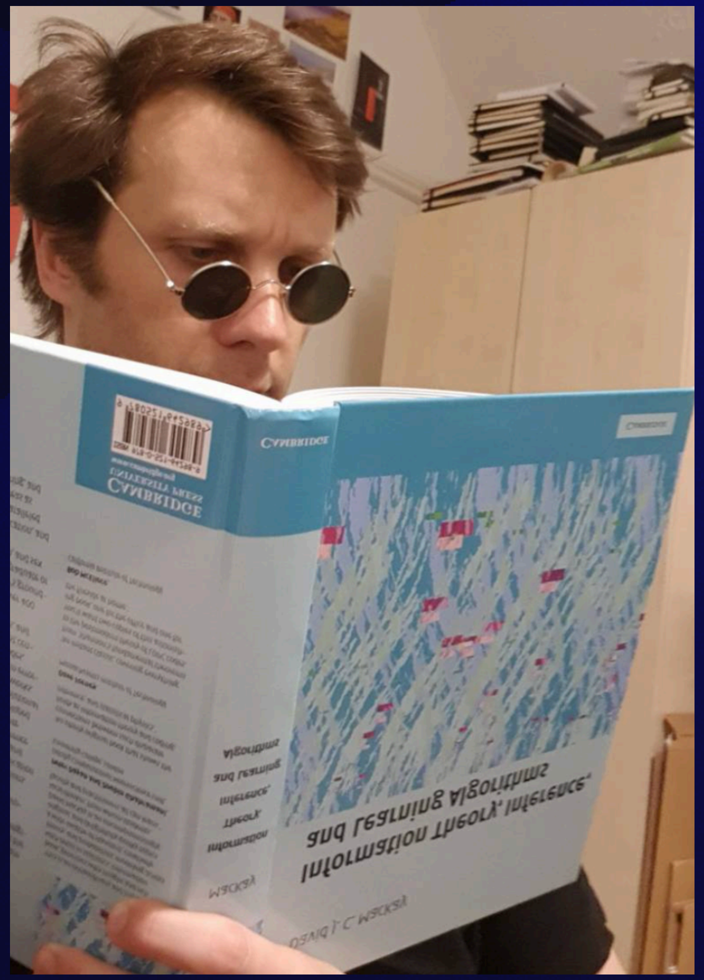
Arrows of Time for LLMs

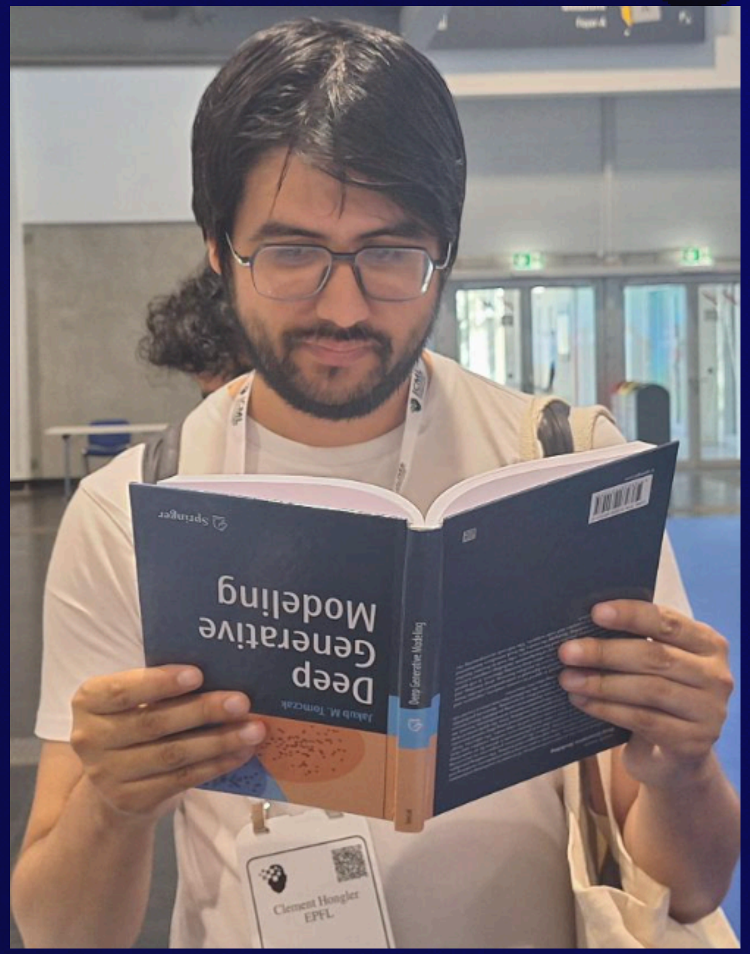
Vassilis Papadopoulos, Jérémie Wenger, Clément Hongler

ICML 2024

Arrows of Time for LLMs

Vassilis Papadopoulos, Jérémie Wenger, Clément Hongler





ICML 2024

Language Modeling

Language Modeling

Next-Token Prediction (Forward model)

Language Modeling

Next-Token Prediction (Forward model)

[redacted]

Language Modeling

Next-Token Prediction (Forward model)

- Estimate $\mathbb{P}\left(X_k = x \mid x_1 \cdots x_{k-1}\right)$ as $\mathbb{P}^{\rightarrow}(X_k = x \mid x_1, \cdots, x_{k-1}) = p_k^{\rightarrow}(x)$

Language Modeling

Next-Token Prediction (Forward model)

[redacted]

- Estimate $\mathbb{P}\left(X_k = x \,|\, x_1 \cdots x_{k-1}\right)$ as $\mathbb{P}^{\rightarrow}(X_k = x \,|\, x_1, \cdots, x_{k-1}) = p_k^{\rightarrow}(x)$
- Factor sequence probability as $\mathbb{P}^{\rightarrow}(x_1, \cdots, x_n) = \prod_{k=1}^{n} p_k^{\rightarrow}(x_k)$

Language Modeling

Next-Token Prediction (Forward model)

- Estimate $\mathbb{P}\left(X_k = x \,|\, x_1 \cdots x_{k-1}\right)$ as $\mathbb{P}^{\rightarrow}(X_k = x \,|\, x_1, \cdots, x_{k-1}) = p_k^{\rightarrow}(x)$
- Factor sequence probability as $$\mathbb{P}^{\rightarrow}(x_1, \cdots, x_n) = \prod_{k=1}^{n} p_k^{\rightarrow}(x_k)$$

Previous-Token Prediction (Backward model)

Language Modeling

Next-Token Prediction (Forward model)

- Estimate $\mathbb{P}\left(X_k = x \,|\, x_1 \cdots x_{k-1}\right)$ as $\mathbb{P}^{\rightarrow}(X_k = x \,|\, x_1, \cdots, x_{k-1}) = p_k^{\rightarrow}(x)$
- Factor sequence probability as $\mathbb{P}^{\rightarrow}(x_1, \cdots, x_n) = \prod_{k=1}^{n} p_k^{\rightarrow}(x_k)$

Previous-Token Prediction (Backward model)

Language Modeling

Next-Token Prediction (Forward model)

- Estimate $\mathbb{P}\left(X_k = x \,|\, x_1 \cdots x_{k-1}\right)$ as $\mathbb{P}^{\rightarrow}(X_k = x \,|\, x_1, \cdots, x_{k-1}) = p_k^{\rightarrow}(x)$
- Factor sequence probability as $\mathbb{P}^{\rightarrow}(x_1, \cdots, x_n) = \prod_{k=1}^{n} p_k^{\rightarrow}(x_k)$

Previous-Token Prediction (Backward model)

- Estimate $\mathbb{P}\left(X_k = x \,|\, x_{k+1} \cdots x_n\right)$ as $\mathbb{P}^{\leftarrow}(X_k = x \,|\, x_1, \cdots, x_{k-1}) = p_k^{\leftarrow}(x)$

Language Modeling

Next-Token Prediction (Forward model)

- Estimate $\mathbb{P}\left(X_k = x \,|\, x_1 \cdots x_{k-1}\right)$ as $\mathbb{P}^{\rightarrow}(X_k = x \,|\, x_1, \cdots, x_{k-1}) = p_k^{\rightarrow}(x)$
- Factor sequence probability as $\mathbb{P}^{\rightarrow}(x_1, \cdots, x_n) = \prod_{k=1}^{n} p_k^{\rightarrow}(x_k)$

Previous-Token Prediction (Backward model)

- Estimate $\mathbb{P}\left(X_k = x \,|\, x_{k+1} \cdots x_n\right)$ as $\mathbb{P}^{\leftarrow}(X_k = x \,|\, x_1, \cdots, x_{k-1}) = p_k^{\leftarrow}(x)$
- Factor sequence probability as $\mathbb{P}^{\leftarrow}(x_1, \cdots, x_n) = \prod_{k=1}^{n} p_k^{\leftarrow}(x_k)$

Language Modeling

Next-Token Prediction (Forward model)

- Estimate $\mathbb{P}\left(X_k = x \,|\, x_1 \cdots x_{k-1}\right)$ as $\mathbb{P}^{\rightarrow}(X_k = x \,|\, x_1, \cdots, x_{k-1}) = p_k^{\rightarrow}(x)$
- Factor sequence probability as $\mathbb{P}^{\rightarrow}(x_1, \cdots, x_n) = \prod_{k=1}^{n} p_k^{\rightarrow}(x_k)$

Previous-Token Prediction (Backward model)

- Estimate $\mathbb{P}\left(X_k = x \,|\, x_{k+1} \cdots x_n\right)$ as $\mathbb{P}^{\leftarrow}(X_k = x \,|\, x_1, \cdots, x_{k-1}) = p_k^{\leftarrow}(x)$
- Factor sequence probability as $\mathbb{P}^{\leftarrow}(x_1, \cdots, x_n) = \prod_{k=1}^{n} p_k^{\leftarrow}(x_k)$

Both $\mathbb{P}^{\rightarrow}$ and $\mathbb{P}^{\leftarrow}$ estimate :

$$\boxed{\mathbb{P}(x_1, \cdots, x_n)}$$

Language Modeling

Next-Token Prediction (Forward model)

- Estimate $\mathbb{P}\left(X_k = x \,|\, x_1 \cdots x_{k-1}\right)$ as $\mathbb{P}^{\rightarrow}(X_k = x \,|\, x_1, \cdots, x_{k-1}) = p_k^{\rightarrow}(x)$
- Factor sequence probability as $\mathbb{P}^{\rightarrow}(x_1, \cdots, x_n) = \prod_{k=1}^{n} p_k^{\rightarrow}(x_k)$

Previous-Token Prediction (Backward model)

- Estimate $\mathbb{P}\left(X_k = x \,|\, x_{k+1} \cdots x_n\right)$ as $\mathbb{P}^{\leftarrow}(X_k = x \,|\, x_1, \cdots, x_{k-1}) = p_k^{\leftarrow}(x)$
- Factor sequence probability as $\mathbb{P}^{\leftarrow}(x_1, \cdots, x_n) = \prod_{k=1}^{n} p_k^{\leftarrow}(x_k)$

$$\mathbb{P}\left(X_1 = x_1\right) \times \mathbb{P}(X_2 = x_2 \,|\, x_1) \times \cdots \times \mathbb{P}\left(X_n = x_n \,|\, x_1 \cdots x_{n-1}\right) =$$

Both $\mathbb{P}^{\rightarrow}$ and $\mathbb{P}^{\leftarrow}$ estimate :

$$\boxed{\mathbb{P}(x_1, \cdots, x_n)}$$

Language Modeling

Next-Token Prediction (Forward model)

- Estimate $\mathbb{P}\left(X_k = x \,|\, x_1 \cdots x_{k-1}\right)$ as $\mathbb{P}^{\rightarrow}(X_k = x \,|\, x_1, \cdots, x_{k-1}) = p_k^{\rightarrow}(x)$
- Factor sequence probability as $\mathbb{P}^{\rightarrow}(x_1, \cdots, x_n) = \prod_{k=1}^{n} p_k^{\rightarrow}(x_k)$

Previous-Token Prediction (Backward model)

- Estimate $\mathbb{P}\left(X_k = x \,|\, x_{k+1} \cdots x_n\right)$ as $\mathbb{P}^{\leftarrow}(X_k = x \,|\, x_1, \cdots, x_{k-1}) = p_k^{\leftarrow}(x)$
- Factor sequence probability as $\mathbb{P}^{\leftarrow}(x_1, \cdots, x_n) = \prod_{k=1}^{n} p_k^{\leftarrow}(x_k)$

Both $\mathbb{P}^{\rightarrow}$ and $\mathbb{P}^{\leftarrow}$ estimate :

$$\mathbb{P}\left(X_1 = x_1\right) \times \mathbb{P}(X_2 = x_2 \,|\, x_1) \times \cdots \times \mathbb{P}\left(X_n = x_n \,|\, x_1 \cdots x_{n-1}\right) = \boxed{\mathbb{P}(x_1, \cdots, x_n)} = \mathbb{P}\left(X_n = x_n\right) \times \mathbb{P}(X_{n-1} = x_{n-1} \,|\, x_n) \times \cdots \times \mathbb{P}\left(X_1 = x_1 \,|\, x_n \cdots x_2\right)$$

Language Modeling

Next-Token Prediction (Forward model)

- Estimate $\mathbb{P}\left(X_k = x \,|\, x_1 \cdots x_{k-1}\right)$ as $\mathbb{P}^{\rightarrow}(X_k = x \,|\, x_1, \cdots, x_{k-1}) = p_k^{\rightarrow}(x)$
- Factor sequence probability as $\mathbb{P}^{\rightarrow}(x_1, \cdots, x_n) = \prod_{k=1}^{n} p_k^{\rightarrow}(x_k)$

Previous-Token Prediction (Backward model)

- Estimate $\mathbb{P}\left(X_k = x \,|\, x_{k+1} \cdots x_n\right)$ as $\mathbb{P}^{\leftarrow}(X_k = x \,|\, x_1, \cdots, x_{k-1}) = p_k^{\leftarrow}(x)$
- Factor sequence probability as $\mathbb{P}^{\leftarrow}(x_1, \cdots, x_n) = \prod_{k=1}^{n} p_k^{\leftarrow}(x_k)$

Both $\mathbb{P}^{\rightarrow}$ and $\mathbb{P}^{\leftarrow}$ estimate :

$$\mathbb{P}\left(X_1 = x_1\right) \times \mathbb{P}(X_2 = x_2 \,|\, x_1) \times \cdots \times \mathbb{P}\left(X_n = x_n \,|\, x_1 \cdots x_{n-1}\right) = \boxed{\mathbb{P}(x_1, \cdots, x_n)} = \mathbb{P}\left(X_n = x_n\right) \times \mathbb{P}(X_{n-1} = x_{n-1} \,|\, x_n) \times \cdots \times \mathbb{P}\left(X_1 = x_1 \,|\, x_n \cdots x_2\right)$$

When training LLMs, do we have $\mathbb{P}^{\rightarrow} = \mathbb{P}^{\leftarrow}$?

Cross-Entropy Loss

Cross-Entropy Loss

- Models are trained with Cross-Entropy loss ℓ_{CE}

Cross-Entropy Loss

- Models are trained with Cross-Entropy loss ℓ_{CE}
- Summing the losses of n tokens, we get:

Cross-Entropy Loss

- Models are trained with Cross-Entropy loss ℓ_{CE}
- Summing the losses of n tokens, we get:

$$\ell_{\overrightarrow{CE}} = \sum_{k=1}^{n} - \ln \mathbb{P}^{\rightarrow}\left(X_k = x_k \,|\, x_1, \cdots, x_{k-1}\right) = - \ln \mathbb{P}^{\rightarrow}\left(X_1 = x_1, \cdots, X_n = x_n\right)$$

Cross-Entropy Loss

- Models are trained with Cross-Entropy loss ℓ_{CE}
- Summing the losses of n tokens, we get:

$$\ell_{\overrightarrow{CE}} = \sum_{k=1}^{n} -\ln \mathbb{P}^{\rightarrow}\left(X_k = x_k \,|\, x_1, \cdots, x_{k-1}\right) = -\ln \mathbb{P}^{\rightarrow}\left(X_1 = x_1, \cdots, X_n = x_n\right)$$

$$\ell_{\overleftarrow{CE}} = \sum_{k=1}^{n} -\ln \mathbb{P}^{\leftarrow}\left(X_k = x_k \,|\, x_n, \cdots, x_{k-1}\right) = -\ln \mathbb{P}^{\leftarrow}\left(X_1 = x_1, \cdots, X_n = x_n\right)$$

Cross-Entropy Loss

- Models are trained with Cross-Entropy loss ℓ_{CE}
- Summing the losses of n tokens, we get:

$$\ell_{CE}^{\rightarrow} = \sum_{k=1}^{n} -\ln \mathbb{P}^{\rightarrow}\left(X_k = x_k \,|\, x_1, \cdots, x_{k-1}\right) = -\ln \mathbb{P}^{\rightarrow}\left(X_1 = x_1, \cdots, X_n = x_n\right)$$

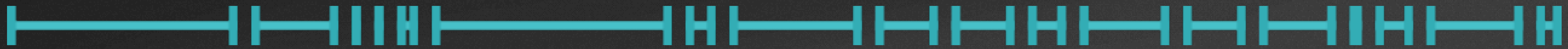
Once upon a time, lived a wise old woman who knew the secrets of the forest.

$$\ell_{CE}^{\leftarrow} = \sum_{k=1}^{n} -\ln \mathbb{P}^{\leftarrow}\left(X_k = x_k \,|\, x_n, \cdots, x_{k-1}\right) = -\ln \mathbb{P}^{\leftarrow}\left(X_1 = x_1, \cdots, X_n = x_n\right)$$

Cross-Entropy Loss

- Models are trained with Cross-Entropy loss ℓ_{CE}
- Summing the losses of n tokens, we get:

$$\ell_{CE}^{\rightarrow} = \sum_{k=1}^{n} -\ln \mathbb{P}^{\rightarrow}\left(X_k = x_k \,|\, x_1, \cdots, x_{k-1}\right) = -\ln \mathbb{P}^{\rightarrow}\left(X_1 = x_1, \cdots, X_n = x_n\right)$$



Once upon a time, lived a wise old woman who knew the secrets of the forest.

$$\ell_{CE}^{\leftarrow} = \sum_{k=1}^{n} -\ln \mathbb{P}^{\leftarrow}\left(X_k = x_k \,|\, x_n, \cdots, x_{k-1}\right) = -\ln \mathbb{P}^{\leftarrow}\left(X_1 = x_1, \cdots, X_n = x_n\right)$$

Cross-Entropy Loss

- Models are trained with Cross-Entropy loss ℓ_{CE}
- Summing the losses of n tokens, we get:

$$\ell_{CE}^{\rightarrow} = \sum_{k=1}^{n} -\ln \mathbb{P}^{\rightarrow}\left(X_k = x_k \,|\, x_1, \cdots, x_{k-1}\right) = -\ln \mathbb{P}^{\rightarrow}\left(X_1 = x_1, \cdots, X_n = x_n\right)$$

Once upon a time, lived a wise old woman who knew the secrets of the forest.

$$\ell_{CE}^{\leftarrow} = \sum_{k=1}^{n} -\ln \mathbb{P}^{\leftarrow}\left(X_k = x_k \,|\, x_n, \cdots, x_{k-1}\right) = -\ln \mathbb{P}^{\leftarrow}\left(X_1 = x_1, \cdots, X_n = x_n\right)$$

Cross-Entropy Loss

- Models are trained with Cross-Entropy loss ℓ_{CE}

- Summing the losses of n tokens, we get:

$$\ell_{CE}^{\rightarrow} = \sum_{k=1}^{n} -\ln \mathbb{P}^{\rightarrow}\left(X_k = x_k \,|\, x_1, \cdots, x_{k-1}\right) = -\ln \mathbb{P}^{\rightarrow}\left(X_1 = x_1, \cdots, X_n = x_n\right)$$

$$\ell_{CE}^{\leftarrow} = \sum_{k=1}^{n} -\ln \mathbb{P}^{\leftarrow}\left(X_k = x_k \,|\, x_n, \cdots, x_{k-1}\right) = -\ln \mathbb{P}^{\leftarrow}\left(X_1 = x_1, \cdots, X_n = x_n\right)$$

- If $\mathbb{P}^{\rightarrow} = \mathbb{P}^{\leftarrow}$, we must have $\ell_{CE}^{\rightarrow} = \ell_{CE}^{\leftarrow}$

Cross-Entropy Loss

- Models are trained with Cross-Entropy loss ℓ_{CE}
- Summing the losses of n tokens, we get:

$$\ell_{CE}^{\rightarrow} = \sum_{k=1}^{n} -\ln \mathbb{P}^{\rightarrow}\left(X_k = x_k \,|\, x_1, \cdots, x_{k-1}\right) = -\ln \mathbb{P}^{\rightarrow}\left(X_1 = x_1, \cdots, X_n = x_n\right)$$

Once upon a time, lived a wise old woman who knew the secrets of the forest.

$$\ell_{CE}^{\leftarrow} = \sum_{k=1}^{n} -\ln \mathbb{P}^{\leftarrow}\left(X_k = x_k \,|\, x_n, \cdots, x_{k-1}\right) = -\ln \mathbb{P}^{\leftarrow}\left(X_1 = x_1, \cdots, X_n = x_n\right)$$

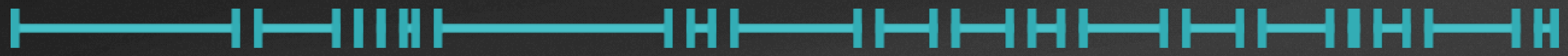
- If $\mathbb{P}^{\rightarrow} = \mathbb{P}^{\leftarrow}$, we must have $\ell_{CE}^{\rightarrow} = \ell_{CE}^{\leftarrow}$
- Compare $\mathbb{E}\left[\ell_{CE}^{\leftrightarrows}\right]$ to compare $\mathbb{P}^{\leftrightarrows}$

Cross-Entropy Loss

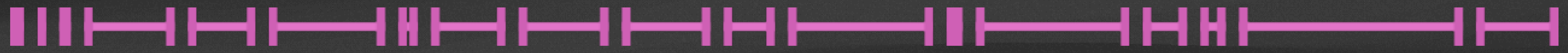
- Models are trained with Cross-Entropy loss ℓ_{CE}

- Summing the losses of n tokens, we get:

$$\ell^{\rightarrow}_{CE} = \sum_{k=1}^{n} -\ln \mathbb{P}^{\rightarrow}\left(X_k = x_k \,|\, x_1, \cdots, x_{k-1}\right) = -\ln \mathbb{P}^{\rightarrow}\left(X_1 = x_1, \cdots, X_n = x_n\right)$$



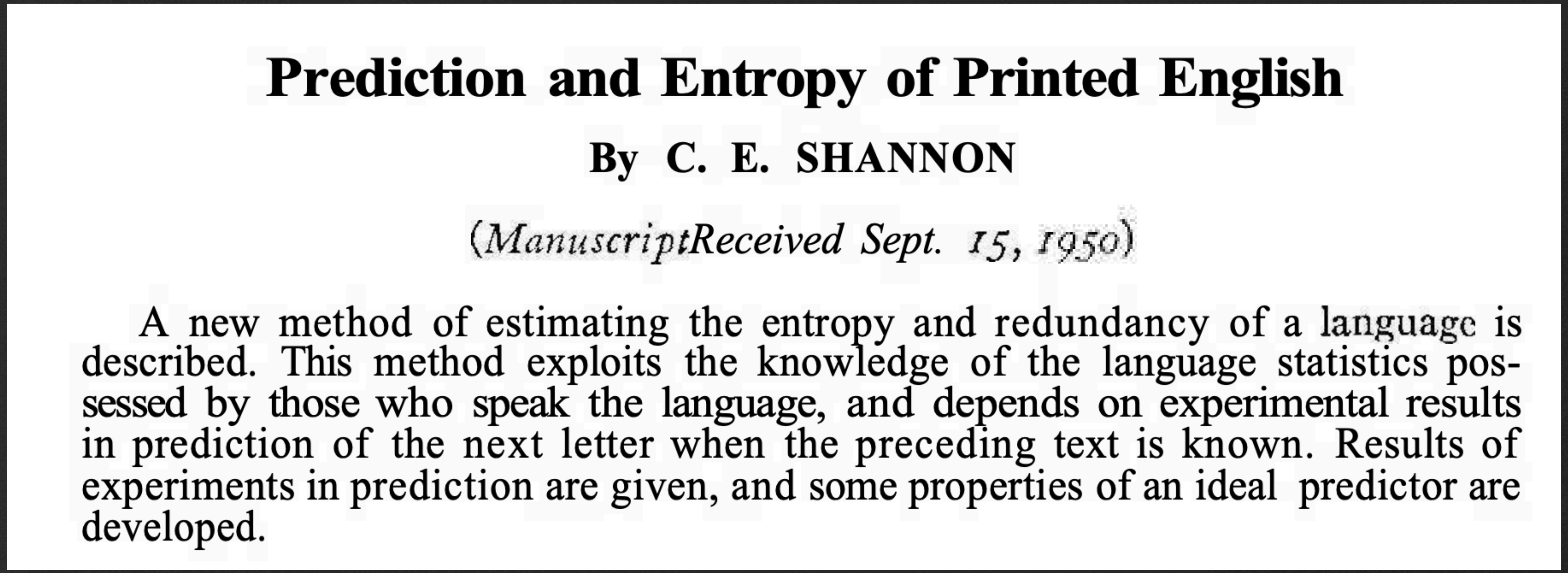
Once upon a time, lived a wise old woman who knew the secrets of the forest.



$$\ell^{\leftarrow}_{CE} = \sum_{k=1}^{n} -\ln \mathbb{P}^{\leftarrow}\left(X_k = x_k \,|\, x_n, \cdots, x_{k-1}\right) = -\ln \mathbb{P}^{\leftarrow}\left(X_1 = x_1, \cdots, X_n = x_n\right)$$

- If $\mathbb{P}^{\rightarrow} = \mathbb{P}^{\leftarrow}$, we must have $\ell^{\rightarrow}_{CE} = \ell^{\leftarrow}_{CE}$

- Compare $\mathbb{E}\left[\ell^{\leftrightarrows}_{CE}\right]$ to compare $\mathbb{P}^{\leftrightarrows}$



Prediction and Entropy of Printed English

By C. E. SHANNON

(*ManuscriptReceived Sept. 15, 1950*)

A new method of estimating the entropy and redundancy of a language is described. This method exploits the knowledge of the language statistics possessed by those who speak the language, and depends on experimental results in prediction of the next letter when the preceding text is known. Results of experiments in prediction are given, and some properties of an ideal predictor are developed.

Arrow of Time

Arrow of Time

$$\partial_{\overleftrightarrow{CE}} = \frac{\mathbb{E}\left[\ell_{\overleftarrow{CE}} - \ell_{\overrightarrow{CE}}\right]}{\frac{1}{2}\left(\mathbb{E}[\ell_{\overleftarrow{CE}} + \ell_{\overrightarrow{CE}}]\right)}$$

Arrow of Time

$$\partial_{CE}^{\leftrightarrow} = \frac{\mathbb{E}\left[\ell_{\overleftarrow{CE}} - \ell_{\overrightarrow{CE}}\right]}{\frac{1}{2}\left(\mathbb{E}[\ell_{\overleftarrow{CE}} + \ell_{\overrightarrow{CE}}]\right)}$$

$\partial_{CE}^{\leftrightarrow}$ quantifies differences in learned $\mathbb{P}^{\rightarrow}$ and $\mathbb{P}^{\leftarrow}$

Arrow of Time

$\partial_{CE}^{\leftrightarrow} > 0 \Leftrightarrow$ FW better

$\partial_{CE}^{\leftrightarrow} < 0 \Leftrightarrow$ BW better

$$\partial_{CE}^{\leftrightarrow} = \frac{\mathbb{E}\left[\ell_{CE}^{\leftarrow} - \ell_{CE}^{\rightarrow}\right]}{\frac{1}{2}\left(\mathbb{E}[\ell_{CE}^{\leftarrow} + \ell_{CE}^{\rightarrow}]\right)}$$

$\partial_{CE}^{\leftrightarrow}$ quantifies differences in learned $\mathbb{P}^{\rightarrow}$ and $\mathbb{P}^{\leftarrow}$

Arrow of Time

$\partial_{CE}^{\leftrightarrow} > 0 \Leftrightarrow$ FW better

$\partial_{CE}^{\leftrightarrow} < 0 \Leftrightarrow$ BW better

$$\partial_{CE}^{\leftrightarrow} = \frac{\mathbb{E}\left[\ell_{CE}^{\leftarrow} - \ell_{CE}^{\rightarrow}\right]}{\frac{1}{2}\left(\mathbb{E}[\ell_{CE}^{\leftarrow} + \ell_{CE}^{\rightarrow}]\right)}$$

$\partial_{CE}^{\leftrightarrow}$ quantifies differences in learned $\mathbb{P}^{\rightarrow}$ and $\mathbb{P}^{\leftarrow}$

A dataset has an **Arrow Of Time** if $\partial_{CE}^{\leftrightarrow}$ has a consistent sign

Arrow of Time

$\partial_{CE}^{\leftrightarrow} > 0 \Leftrightarrow$ FW better

$\partial_{CE}^{\leftrightarrow} < 0 \Leftrightarrow$ BW better

$$\partial_{CE}^{\leftrightarrow} = \frac{\mathbb{E}\left[\ell_{CE}^{\leftarrow} - \ell_{CE}^{\rightarrow}\right]}{\frac{1}{2}\left(\mathbb{E}[\ell_{CE}^{\leftarrow} + \ell_{CE}^{\rightarrow}]\right)}$$

$\partial_{CE}^{\leftrightarrow}$ quantifies differences in learned $\mathbb{P}^{\rightarrow}$ and $\mathbb{P}^{\leftarrow}$

A dataset has an **Arrow Of Time** if $\partial_{CE}^{\leftrightarrow}$ has a consistent sign

$\partial_{CE}^{\leftrightarrow} > 0 \quad \equiv \quad$ Forward Arrow Of Time

Arrow of Time for Languages

Natural Language Experiments

Arrow of Time for Languages

Natural Language Experiments

- Dataset: CC100 (> 30 Gb of text per language)

Arrow of Time for Languages

Natural Language Experiments

- Dataset: CC100 (> 30 Gb of text per language)
- Tokenization: Byte-Pair Encoding, recomputed for each language

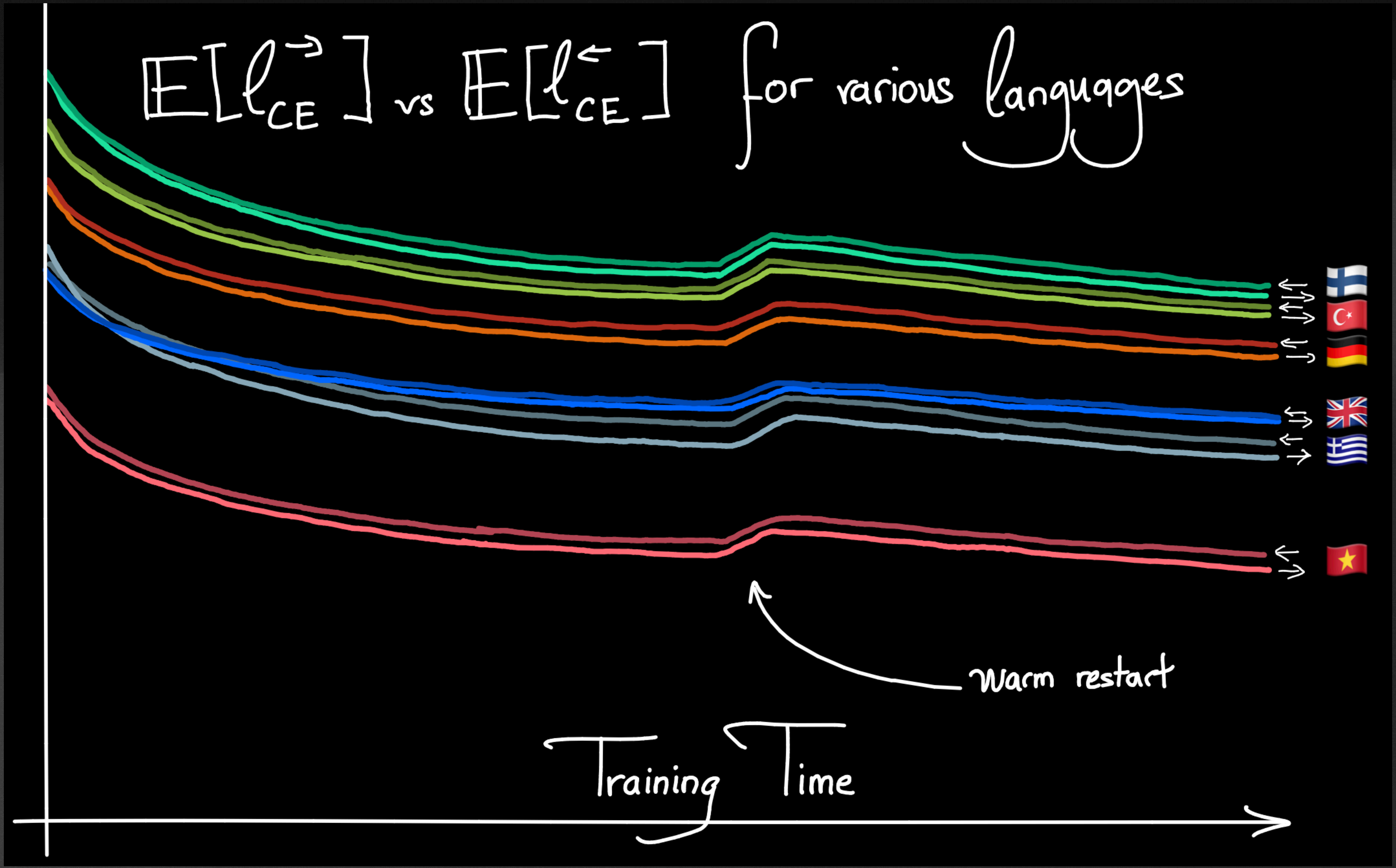
Arrow of Time for Languages

Natural Language Experiments

- Dataset: CC100 (> 30 Gb of text per language)
- Tokenization: Byte-Pair Encoding, recomputed for each language
- Model: GPT2-Medium (~350M params), 256-token context length

Arrow of Time for Languages

Natural Language Experiments



AoT for Languages

Key takeaways (from 100+ experiments)

AoT for Languages

Key takeaways (from 100+ experiments)

AoT for Languages

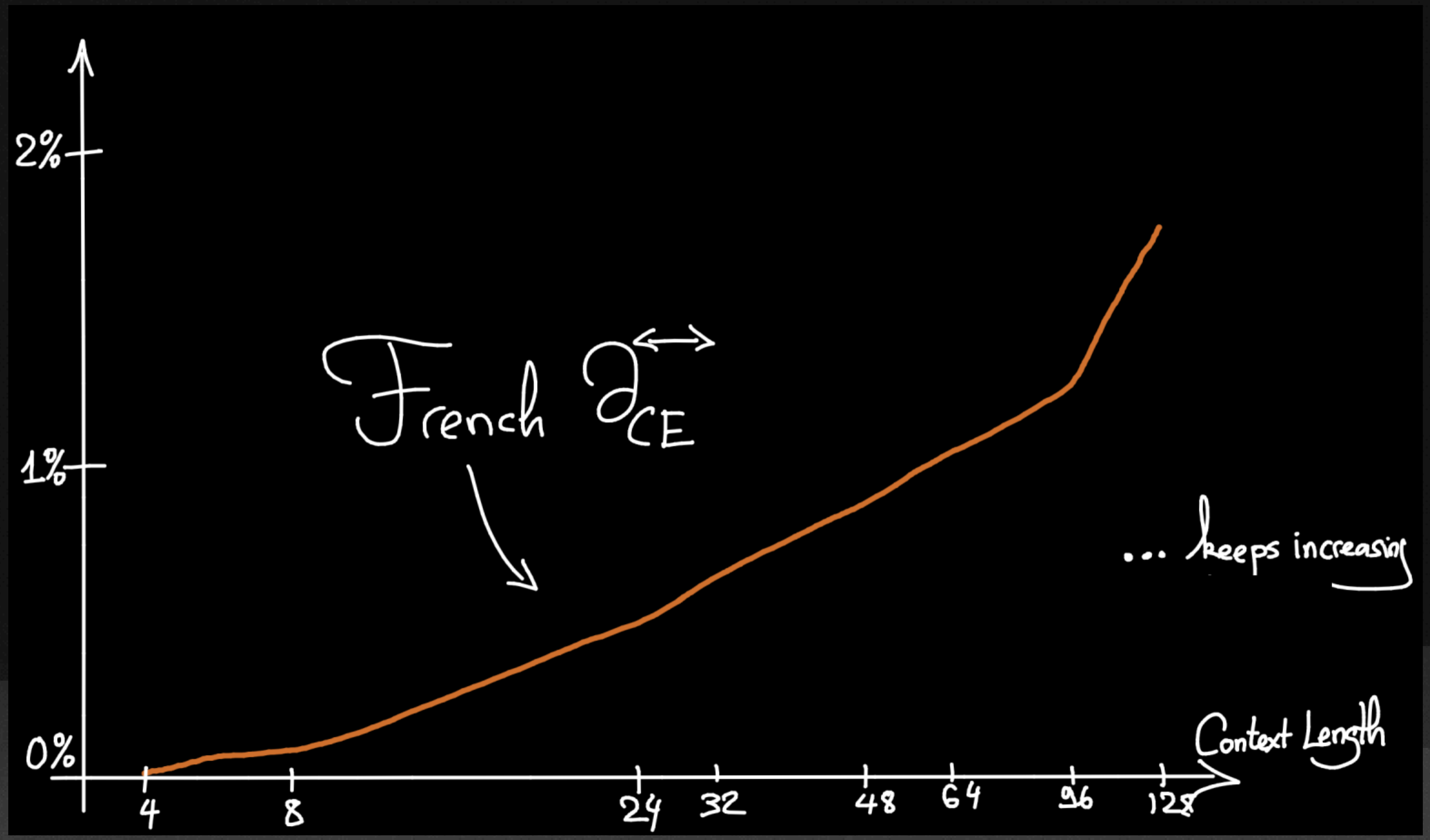
Key takeaways (from 100+ experiments)

- FW AoT universality across languages (we tested 11)

AoT for Languages

Key takeaways (from 100+ experiments)

- FW AoT universality across languages (we tested 11)
- AoT $\partial_{CE}^{\leftrightarrow}$ increases with context length

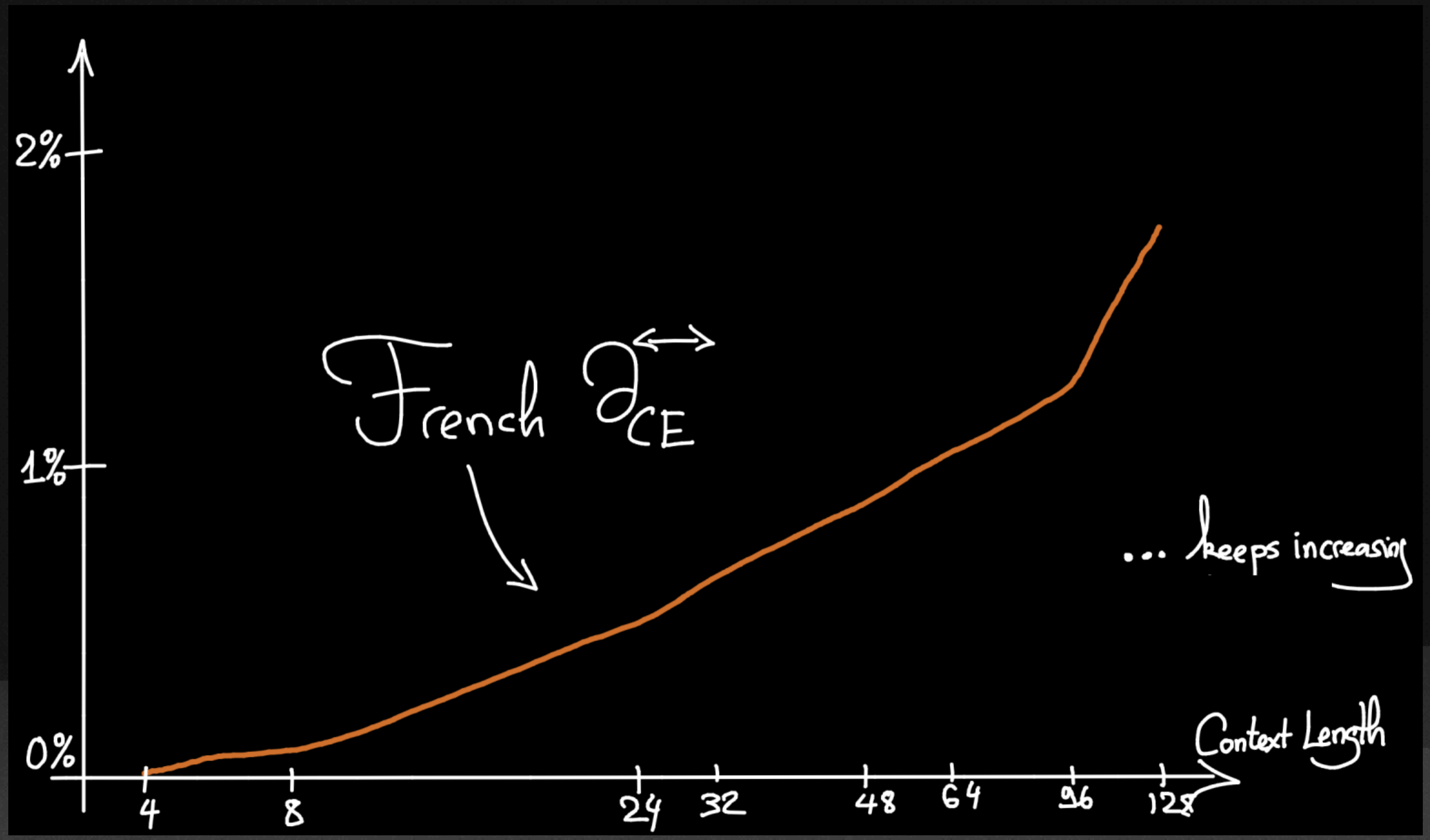


AoT for Languages

Key takeaways (from 100+ experiments)

- FW AoT universality across languages (we tested 11)
- AoT $\partial_{CE}^{\leftrightarrow}$ increases with context length

  > Long-range correlations essential

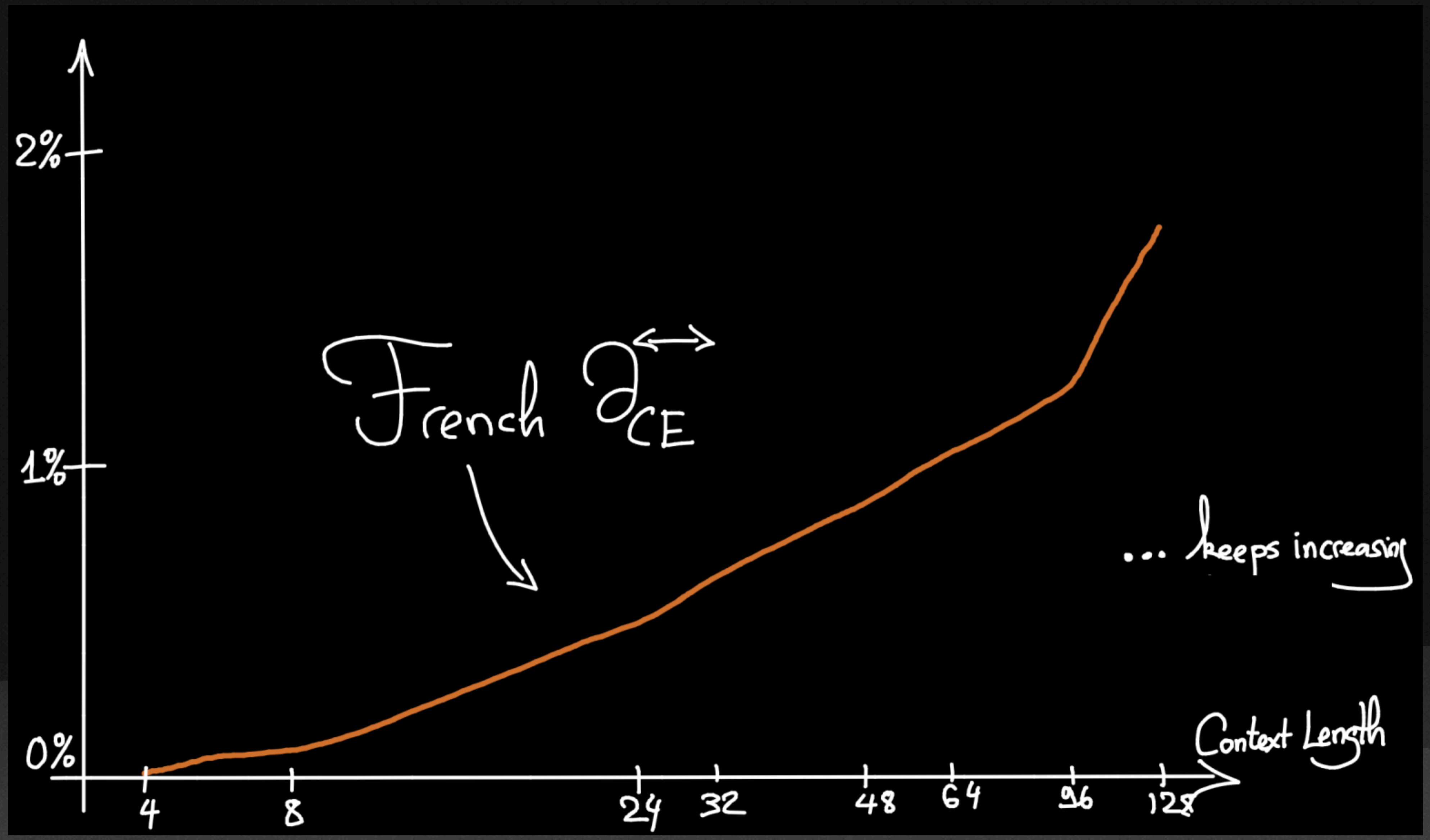


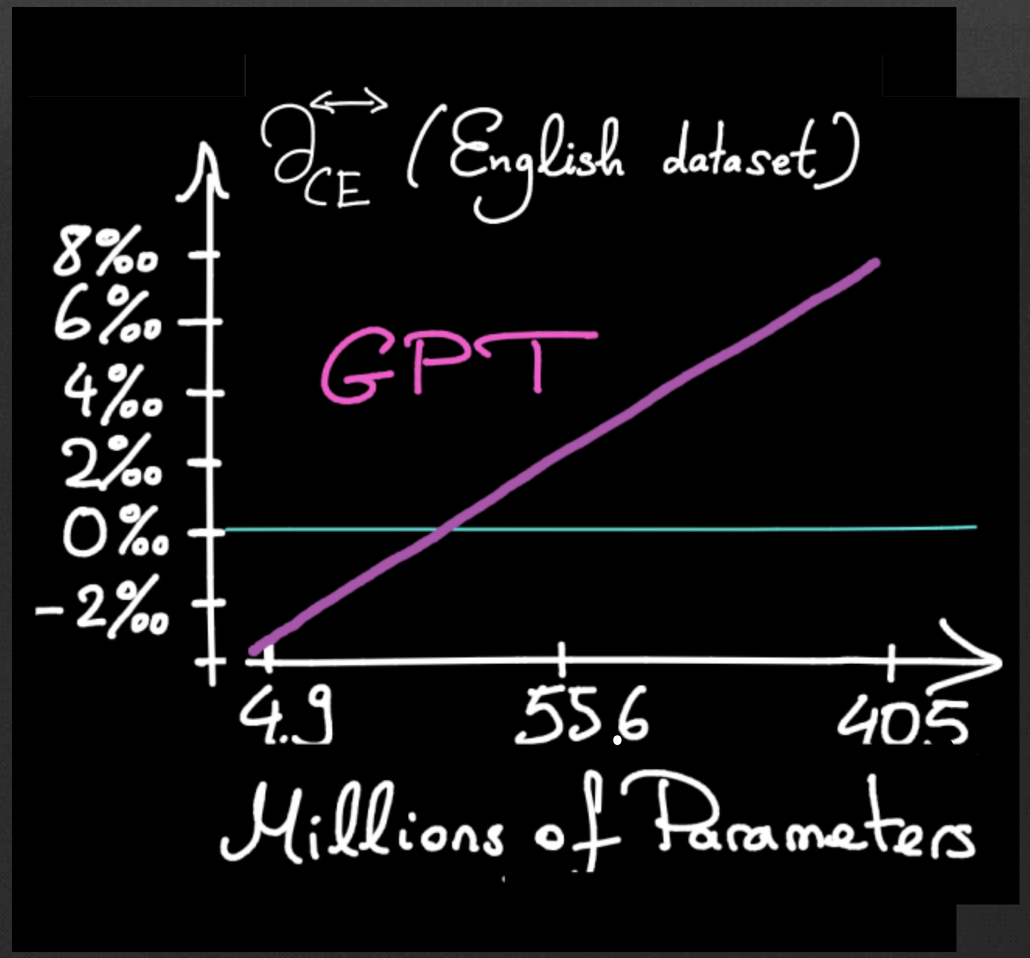
AoT for Languages

Key takeaways (from 100+ experiments)

- FW AoT universality across languages (we tested 11)
- AoT $\partial_{CE}^{\leftrightarrow}$ increases with context length

  > Long-range correlations essential
- AoT $\partial_{CE}^{\leftrightarrow}$ increases with model size





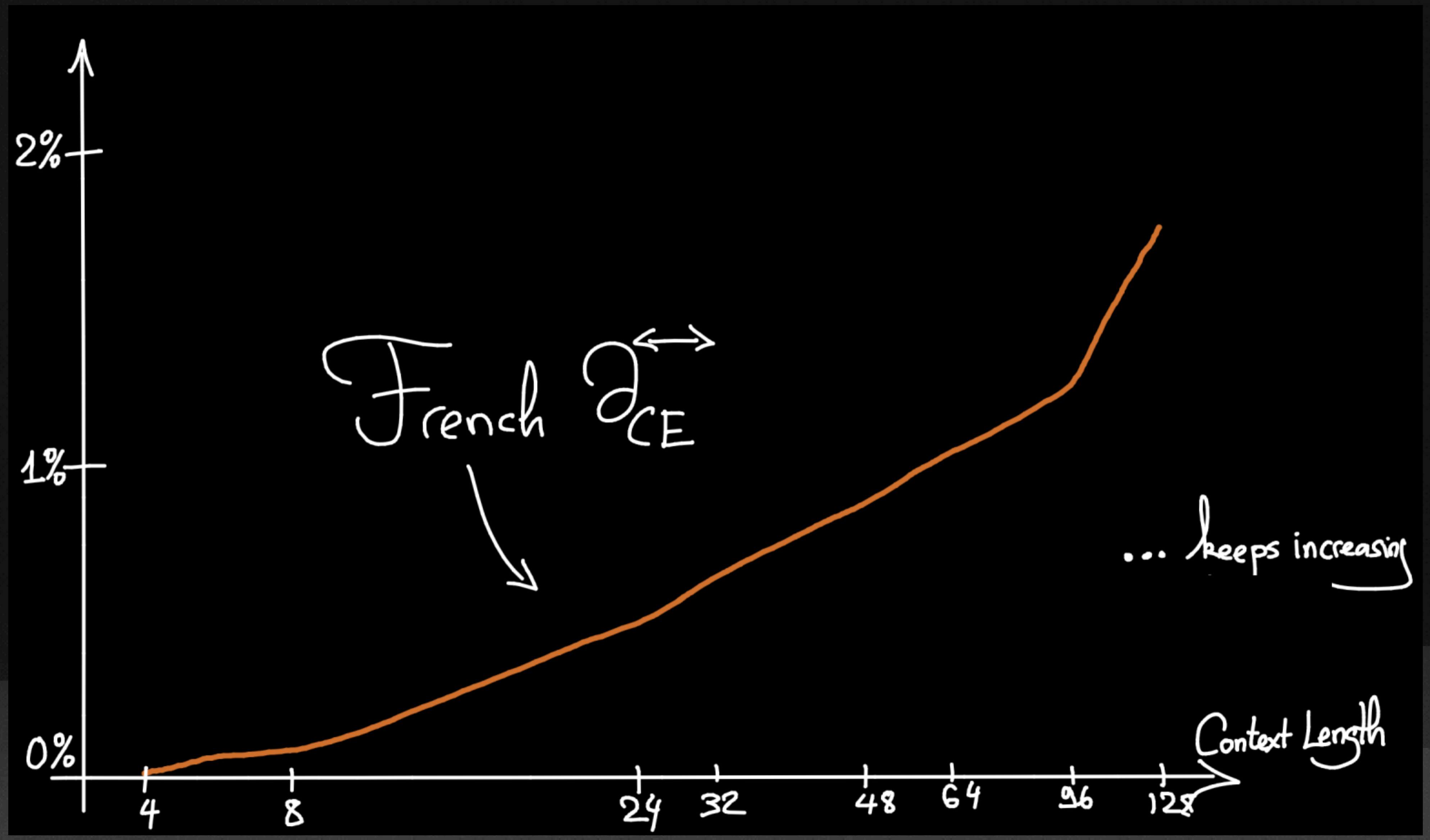
AoT for Languages
Key takeaways (from 100+ experiments)

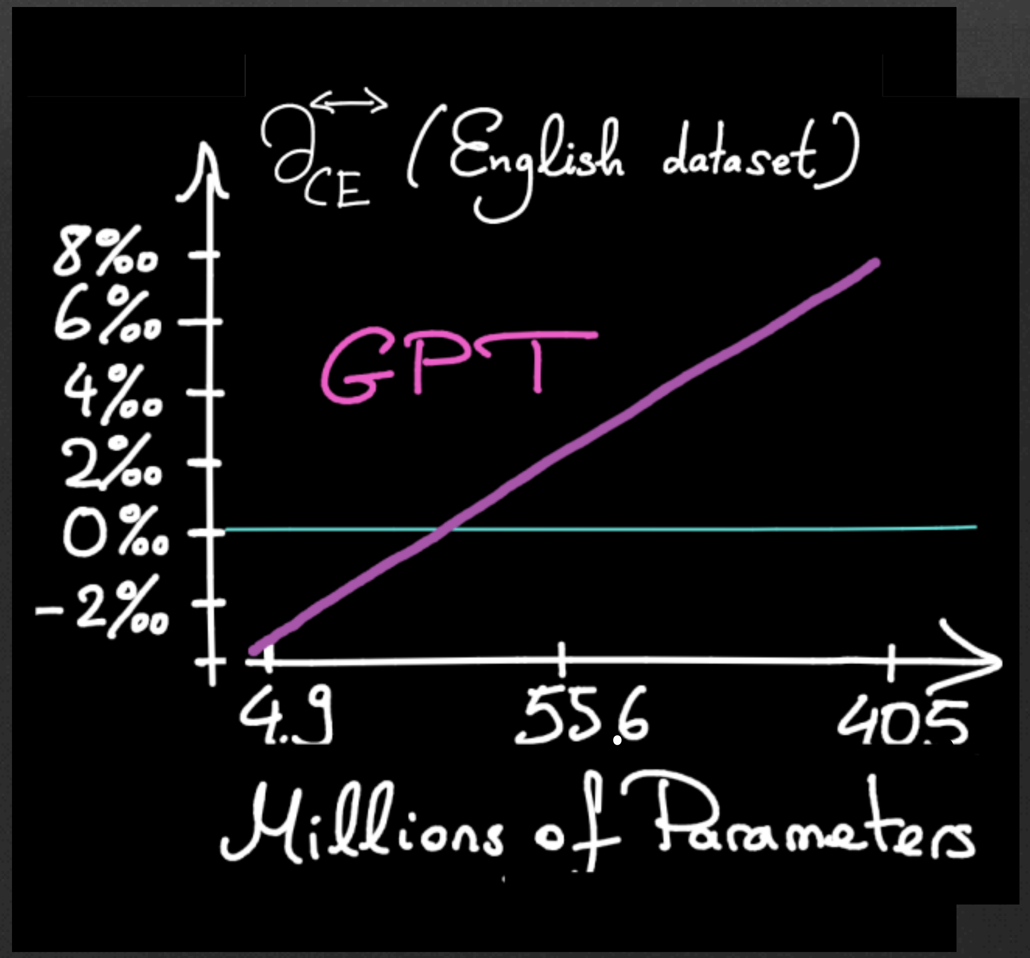
- FW AoT universality across languages (we tested 11)
- AoT $\partial_{CE}^{\leftrightarrow}$ increases with context length

  > Long-range correlations essential

- AoT $\partial_{CE}^{\leftrightarrow}$ increases with model size

  > AoT origin semantic, rather than grammatical





AoT for Languages

Key takeaways (from 100+ experiments)

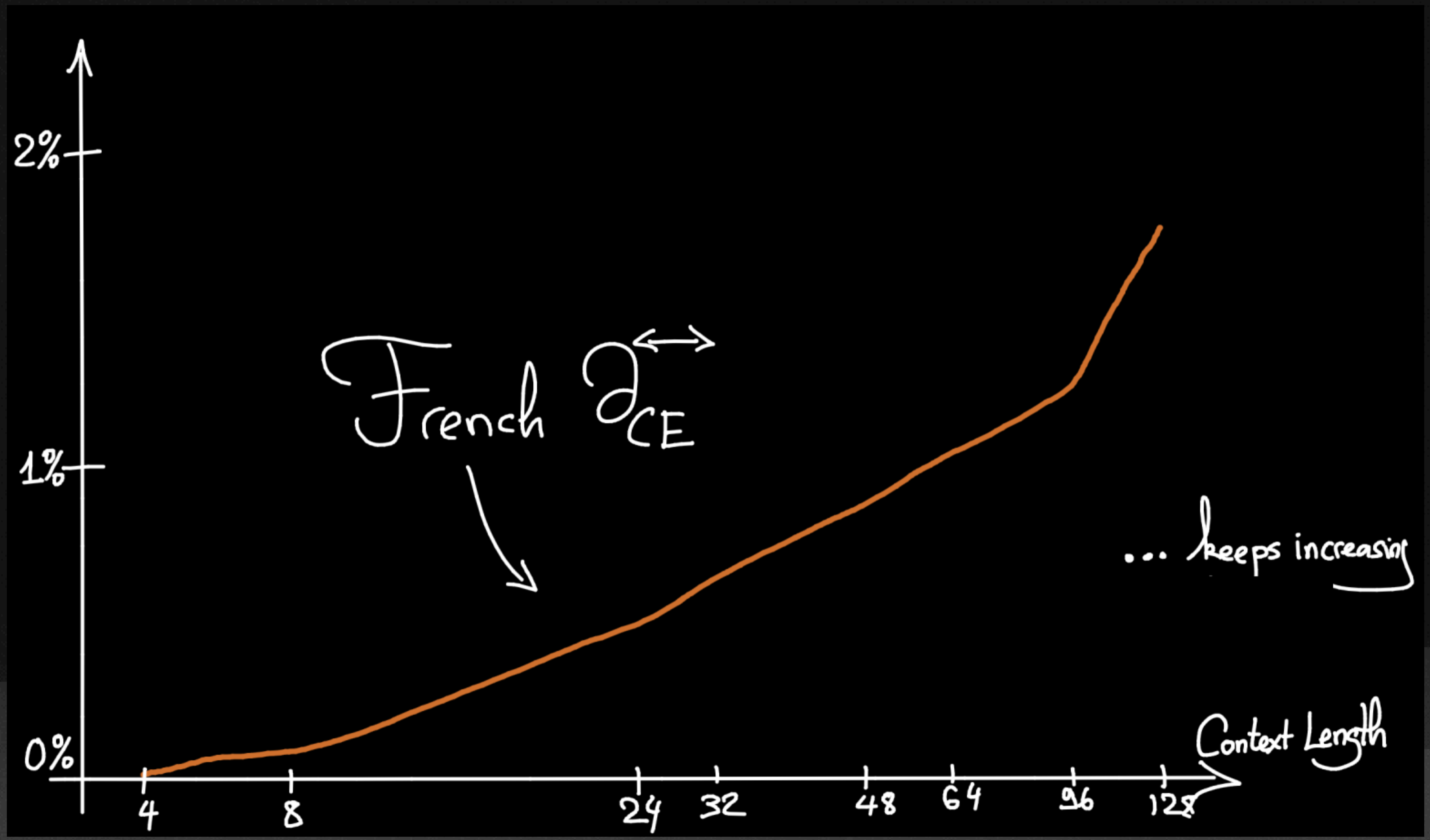
- FW AoT universality across languages (we tested 11)
- AoT $\partial_{CE}^{\leftrightarrow}$ increases with context length

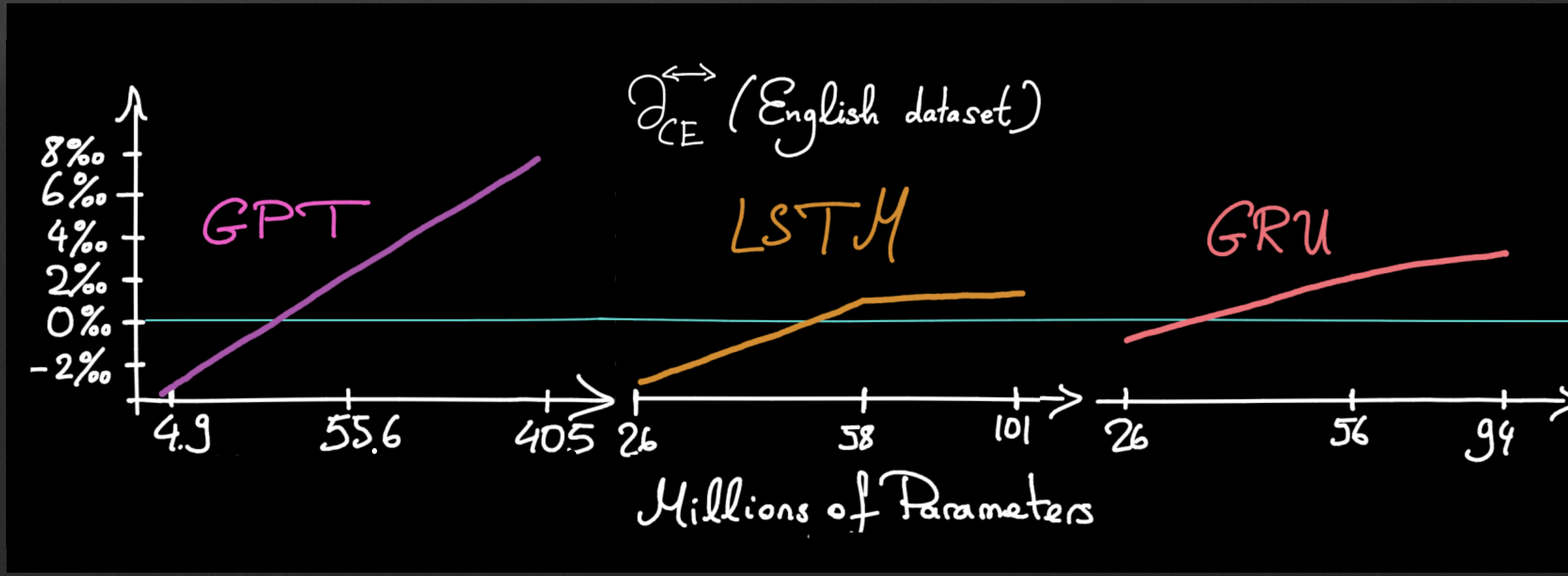
  > Long-range correlations essential

- AoT $\partial_{CE}^{\leftrightarrow}$ increases with model size

  > AoT origin semantic, rather than grammatical

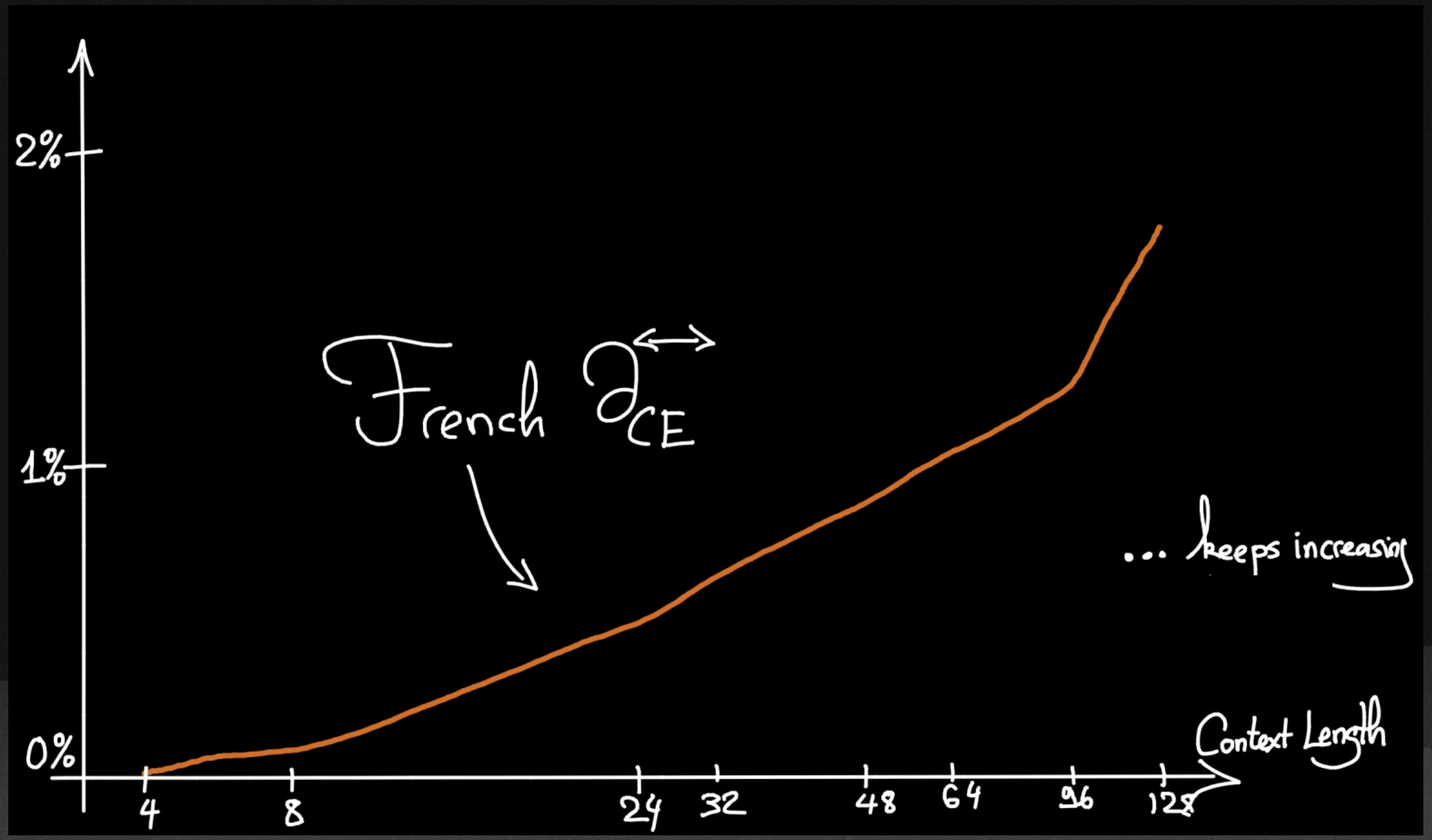
- AoT universal across architectures (we tested LSTMs, GRUs, GPTs)





AoT for Languages
Key takeaways (from 100+ experiments)

- FW AoT universality across languages (we tested 11)

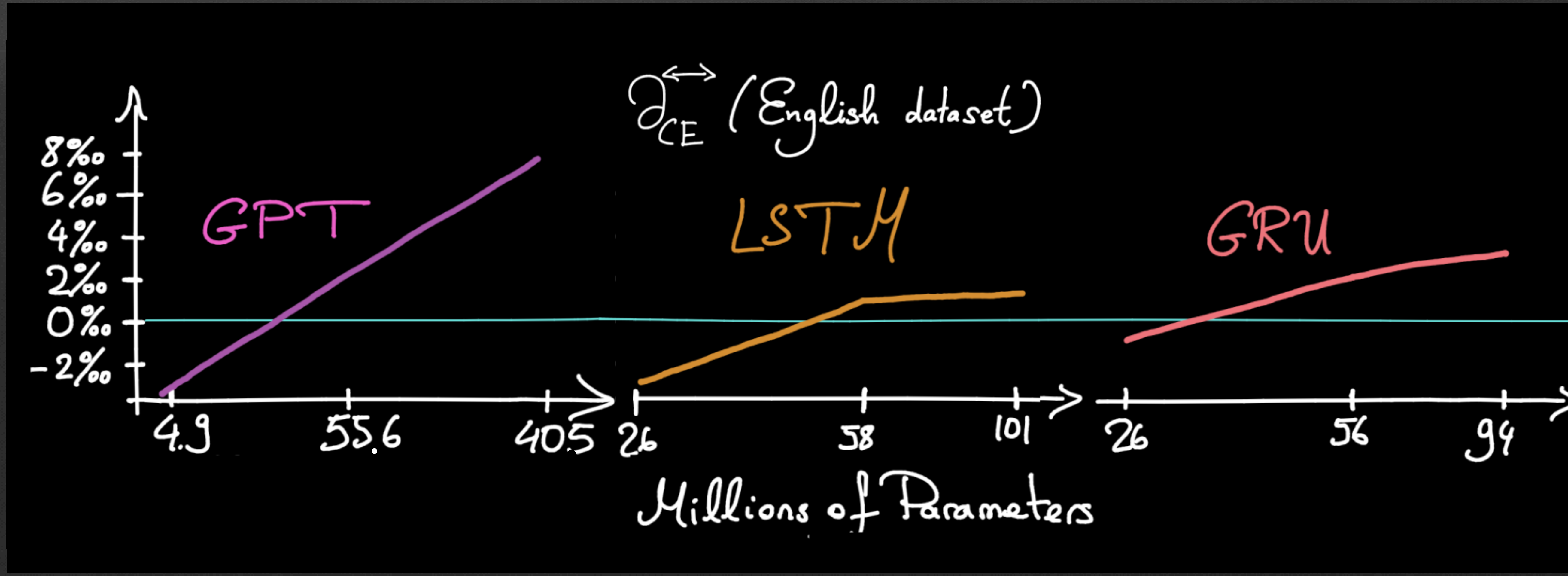


- AoT $\partial_{CE}^{\leftrightarrow}$ increases with context length

  > Long-range correlations essential

- AoT $\partial_{CE}^{\leftrightarrow}$ increases with model size

  > AoT origin semantic, rather than grammatical



- AoT universal across architectures (we tested LSTMs, GRUs, GPTs)

  > As models get stronger, AoT increases

Origin of AoT

Origin of AoT

via Computational Hardness

Origin of AoT

via Computational Hardness

- Consider a dataset of the form $p_1 \times p_2 = n$ with $p_1 < p_2$ primes

Origin of AoT
via Computational Hardness

- Consider a dataset of the form $p_1 \times p_2 = n$ with $p_1 < p_2$ primes

Examples:

$151 \times 353 = 053303$

$367 \times 593 = 217631$

$463 \times 997 = 461611$

Origin of AoT

via Computational Hardness

- Consider a dataset of the form $p_1 \times p_2 = n$ with $p_1 < p_2$ primes
- Theoretical FW and BW cross-entropy losses match, as they should:

Examples:

$151 \times 353 = 053303$

$367 \times 593 = 217631$

$463 \times 997 = 461611$

Origin of AoT

via Computational Hardness

- Consider a dataset of the form $p_1 \times p_2 = n$ with $p_1 < p_2$ primes
- Theoretical FW and BW cross-entropy losses match, as they should:
  - For FW, LHS determines RHS, for BW, RHS determines LHS

Examples:

$$151 \times 353 = 053303$$

$$367 \times 593 = 217631$$

$$463 \times 997 = 461611$$

FW

$$p_1 \times p_2 = n$$

BW

Origin of AoT
via Computational Hardness

Examples:

$$151 \times 353 = 053303$$
$$367 \times 593 = 217631$$
$$463 \times 997 = 461611$$

- Consider a dataset of the form $p_1 \times p_2 = n$ with $p_1 < p_2$ primes
- Theoretical FW and BW cross-entropy losses match, as they should:
  - For FW, LHS determines RHS, for BW, RHS determines LHS
- For the FW model to do well, it needs to learn to multiply p_1, p_2

FW

$$p_1 \times p_2 = n$$

BW

Origin of AoT

via Computational Hardness

- Consider a dataset of the form $p_1 \times p_2 = n$ with $p_1 < p_2$ primes
- Theoretical FW and BW cross-entropy losses match, as they should:
  - For FW, LHS determines RHS, for BW, RHS determines LHS
- For the FW model to do well, it needs to learn to multiply p_1, p_2

> Transformers can learn to do this

Examples:

$$151 \times 353 = 053303$$
$$367 \times 593 = 217631$$
$$463 \times 997 = 461611$$

FW

$$p_1 \times p_2 = n$$

BW

Origin of AoT

via Computational Hardness

- Consider a dataset of the form $p_1 \times p_2 = n$ with $p_1 < p_2$ primes
- Theoretical FW and BW cross-entropy losses match, as they should:
  - For FW, LHS determines RHS, for BW, RHS determines LHS
- For the FW model to do well, it needs to learn to multiply p_1, p_2

  > Transformers can learn to do this

- For the BW model to do well, needs to learn to *factor* n

Examples:

$$151 \times 353 = 053303$$
$$367 \times 593 = 217631$$
$$463 \times 997 = 461611$$

FW

$$p_1 \times p_2 = n$$

BW

Origin of AoT

via Computational Hardness

Examples:

$$151 \times 353 = 053303$$
$$367 \times 593 = 217631$$
$$463 \times 997 = 461611$$

- Consider a dataset of the form $p_1 \times p_2 = n$ with $p_1 < p_2$ primes
- Theoretical FW and BW cross-entropy losses match, as they should:
  - For FW, LHS determines RHS, for BW, RHS determines LHS
- For the FW model to do well, it needs to learn to multiply p_1, p_2

  > Transformers can learn to do this
- For the BW model to do well, needs to learn to *factor* n

  > Computationally hard, likely not computable by a neural network

FW

$$p_1 \times p_2 = n$$

BW

Origin of AoT

via Computational Hardness

- Consider a dataset of the form $p_1 \times p_2 = n$ with $p_1 < p_2$ primes
- Theoretical FW and BW cross-entropy losses match, as they should:
  - For FW, LHS determines RHS, for BW, RHS determines LHS
- For the FW model to do well, it needs to learn to multiply p_1, p_2

  > Transformers can learn to do this
- For the BW model to do well, needs to learn to *factor* n

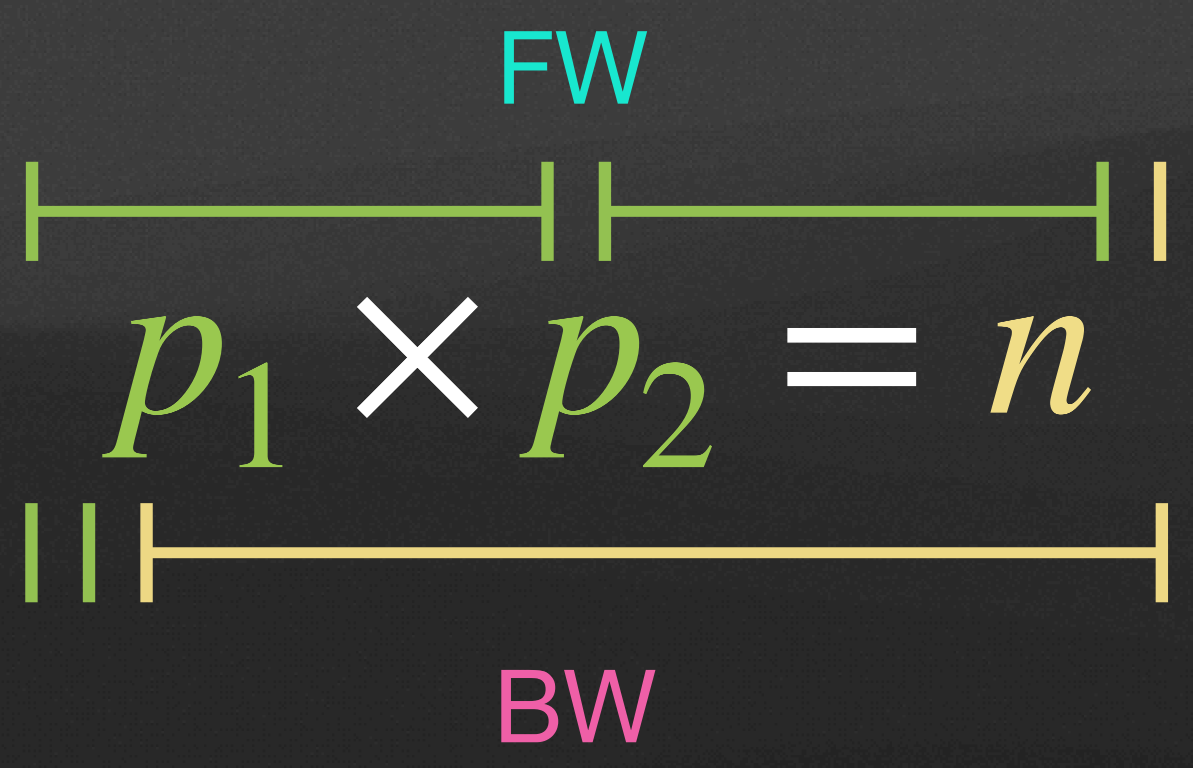
  > Computationally hard, likely not computable by a neural network
- Consequence → AoT in this dataset

Examples:

$151 \times 353 = 053303$

$367 \times 593 = 217631$

$463 \times 997 = 461611$



Emergence of AoT

Emergence of AoT

via Learnability Asymmetry

Emergence of AoT

via Learnability Asymmetry

- Example: Linear Languages

Emergence of AoT

via Learnability Asymmetry

- Example: Linear Languages
  - Dataset $x : y$, with x and y both random m-bit strings

Emergence of AoT

via Learnability Asymmetry

- Example: Linear Languages
  - Dataset $x : y$, with x and y both random m-bit strings
  - x, y related by invertible matrices $A^{\leftrightarrows}$ over the field $\mathbb{F}_2$

Examples:

$010101 : 101110$

$011001 : 110101$

$100010 : 011101$

Emergence of AoT

via Learnability Asymmetry

- Example: Linear Languages
  - Dataset $x : y$, with x and y both random m-bit strings
  - x, y related by invertible matrices $A^{\leftrightarrows}$ over the field $\mathbb{F}_2$
    - $y = A^{\rightarrow}x, \quad x = A^{\leftarrow}y$, and $A^{\leftarrow} = (A^{\rightarrow})^{-1}$

Examples:

$010101 : 101110$

$011001 : 110101$

$100010 : 011101$

Emergence of AoT

via Learnability Asymmetry

- Example: Linear Languages
  - Dataset $x : y$, with x and y both random m-bit strings
  - x, y related by invertible matrices $A^{\leftrightarrows}$ over the field $\mathbb{F}_2$
    - $y = A^{\rightarrow}x, \quad x = A^{\leftarrow}y,$ and $A^{\leftarrow} = (A^{\rightarrow})^{-1}$
- FW model learns $A^{\rightarrow}$, BW model learns $A^{\leftarrow}$

Examples:

$010101 : 101110$

$011001 : 110101$

$100010 : 011101$

Emergence of AoT
via Learnability Asymmetry

Examples:

$010101 : 101110$
$011001 : 110101$
$100010 : 011101$

- Example: Linear Languages
  - Dataset $x : y$, with x and y both random m-bit strings
  - x, y related by invertible matrices $A^{\leftrightarrows}$ over the field $\mathbb{F}_2$
    - $y = A^{\rightarrow}x, \quad x = A^{\leftarrow}y$, and $A^{\leftarrow} = (A^{\rightarrow})^{-1}$
- FW model learns $A^{\rightarrow}$, BW model learns $A^{\leftarrow}$
  - Sparser matrices are easier to learn

Emergence of AoT

via Learnability Asymmetry

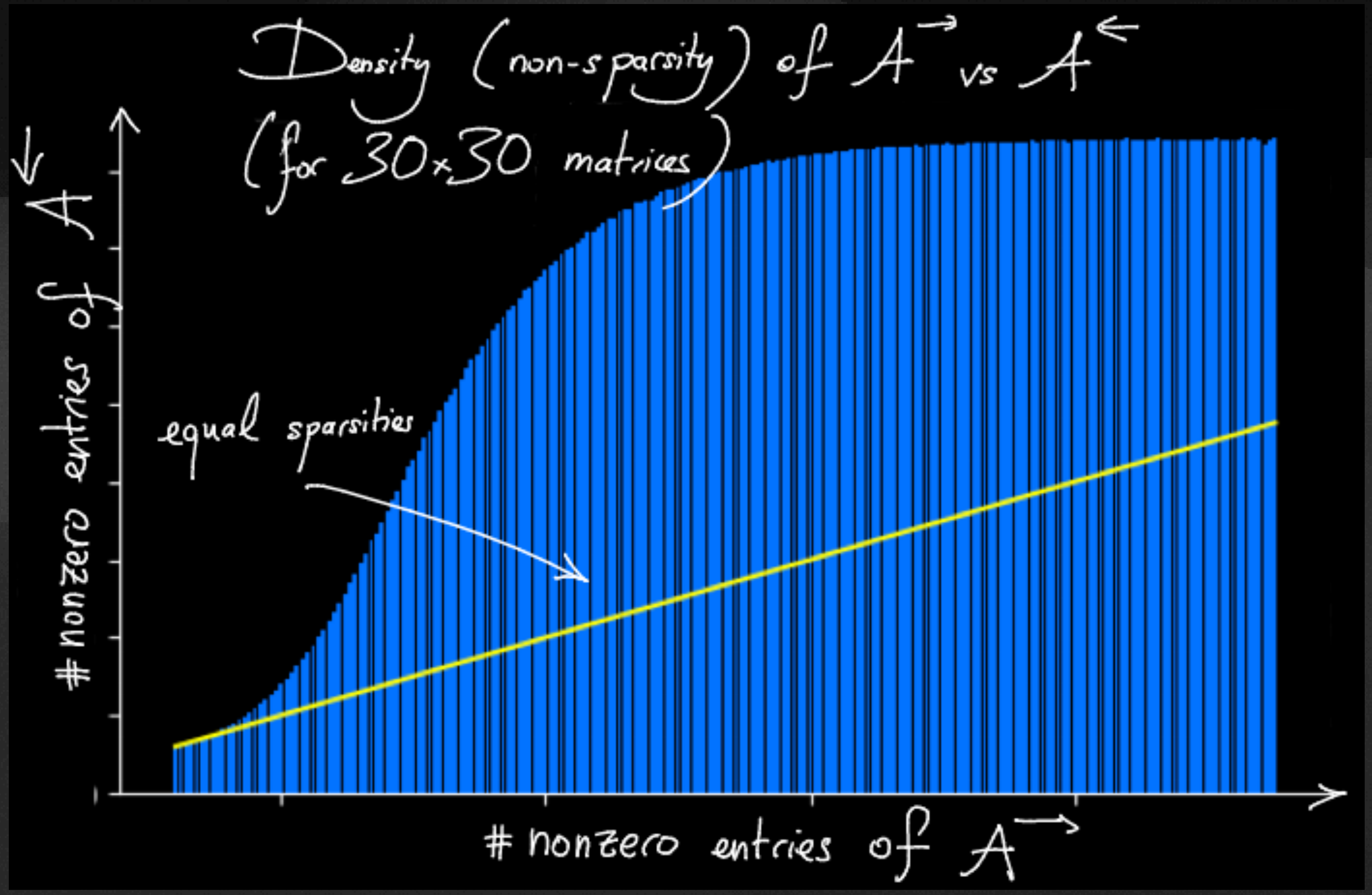
- Example: Linear Languages
  - Dataset $x : y$, with x and y both random m-bit strings
  - x, y related by invertible matrices $A^{\leftrightarrows}$ over the field $\mathbb{F}_2$
    - $y = A^{\rightarrow}x, \quad x = A^{\leftarrow}y,$ and $A^{\leftarrow} = (A^{\rightarrow})^{-1}$
- FW model learns $A^{\rightarrow}$, BW model learns $A^{\leftarrow}$
  - Sparser matrices are easier to learn
- Symmetry breaking : $A^{\rightarrow}$ sparse $\Longrightarrow A^{\leftarrow}$ typically less sparse $\rightarrow$ AoT!

Examples:

$010101 : 101110$

$011001 : 110101$

$100010 : 011101$



Emergence of AoT

via Learnability Asymmetry

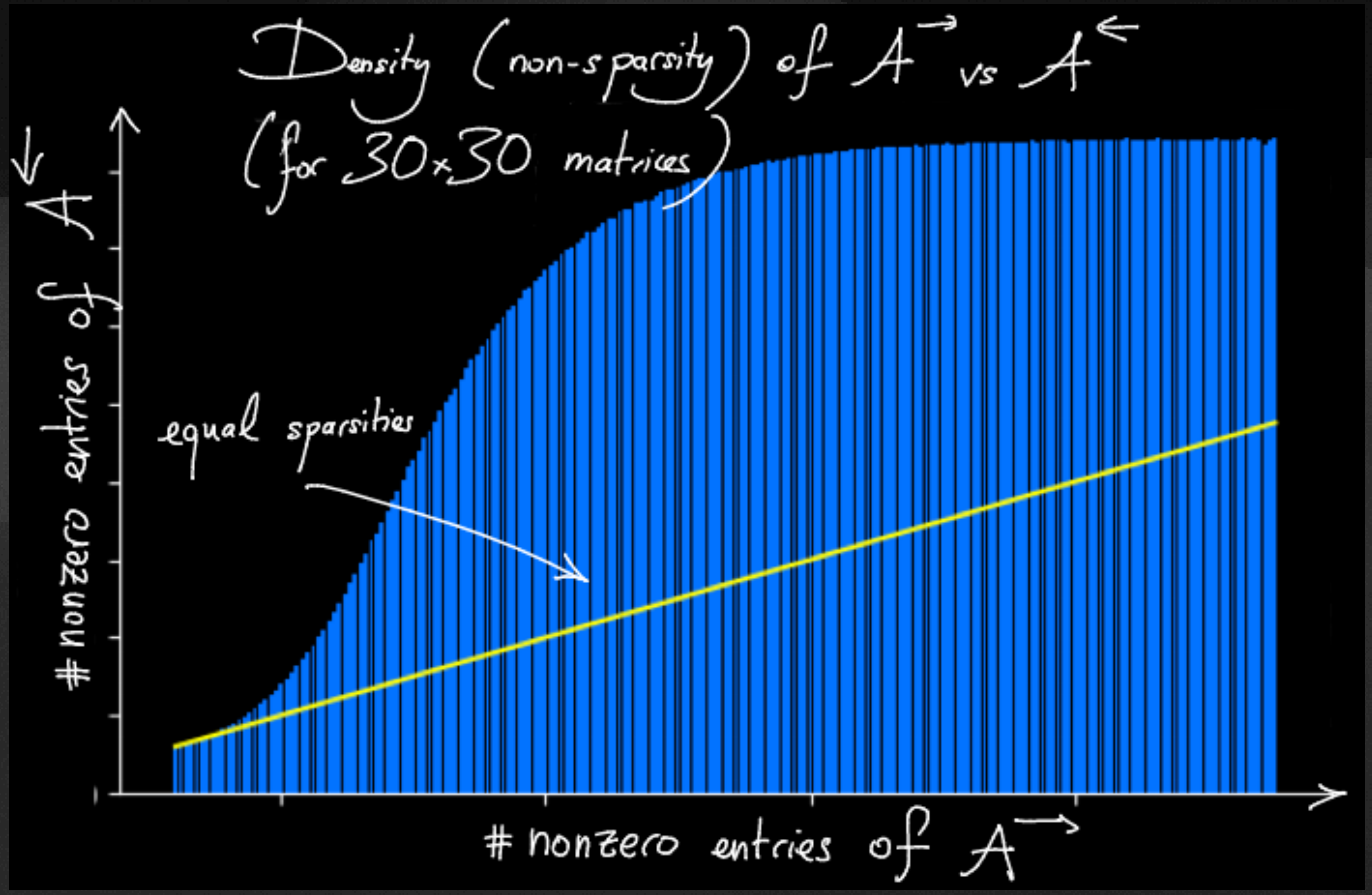
- Example: Linear Languages
  - Dataset $x : y$, with x and y both random m-bit strings
  - x, y related by invertible matrices $A^{\leftrightarrows}$ over the field $\mathbb{F}_2$
    - $y = A^{\rightarrow}x, \quad x = A^{\leftarrow}y$, and $A^{\leftarrow} = (A^{\rightarrow})^{-1}$
- FW model learns $A^{\rightarrow}$, BW model learns $A^{\leftarrow}$
  - Sparser matrices are easier to learn
- Symmetry breaking : $A^{\rightarrow}$ sparse $\Longrightarrow A^{\leftarrow}$ typically less sparse $\rightarrow$ AoT!
  - Also for fine-tuning: $A^{\rightarrow} - \hat{A}^{\rightarrow}$ sparse $\Longrightarrow A^{\leftarrow} - \hat{A}^{\leftarrow}$ less sparse

Examples:

$010101 : 101110$

$011001 : 110101$

$100010 : 011101$



Theory of AoT

Time Symmetry Breaking in Language

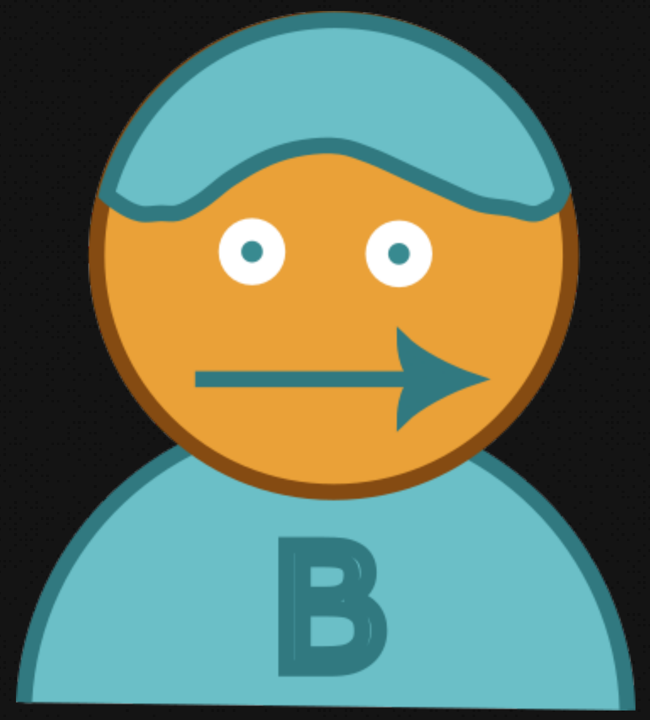
Theory of AoT

Time Symmetry Breaking in Language

- Say Alice and Bob (humans) both speak (and think) forward

Theory of AoT

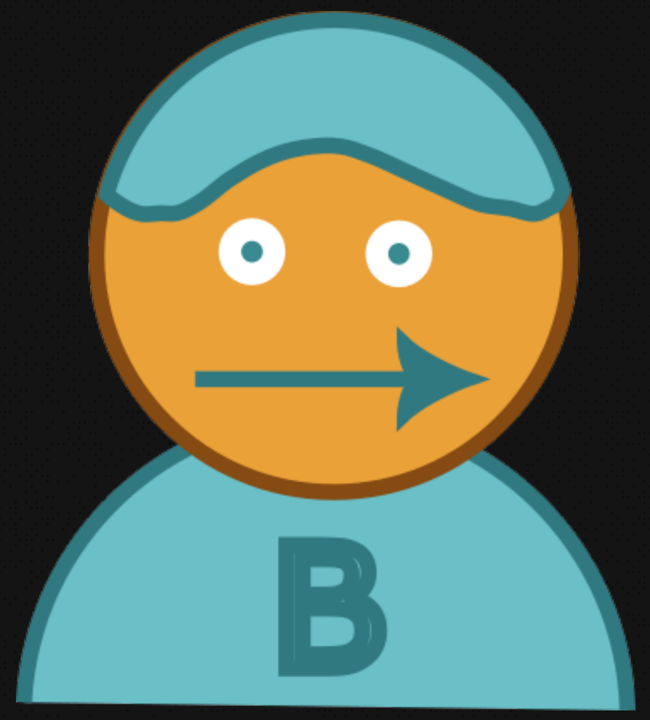
Time Symmetry Breaking in Language



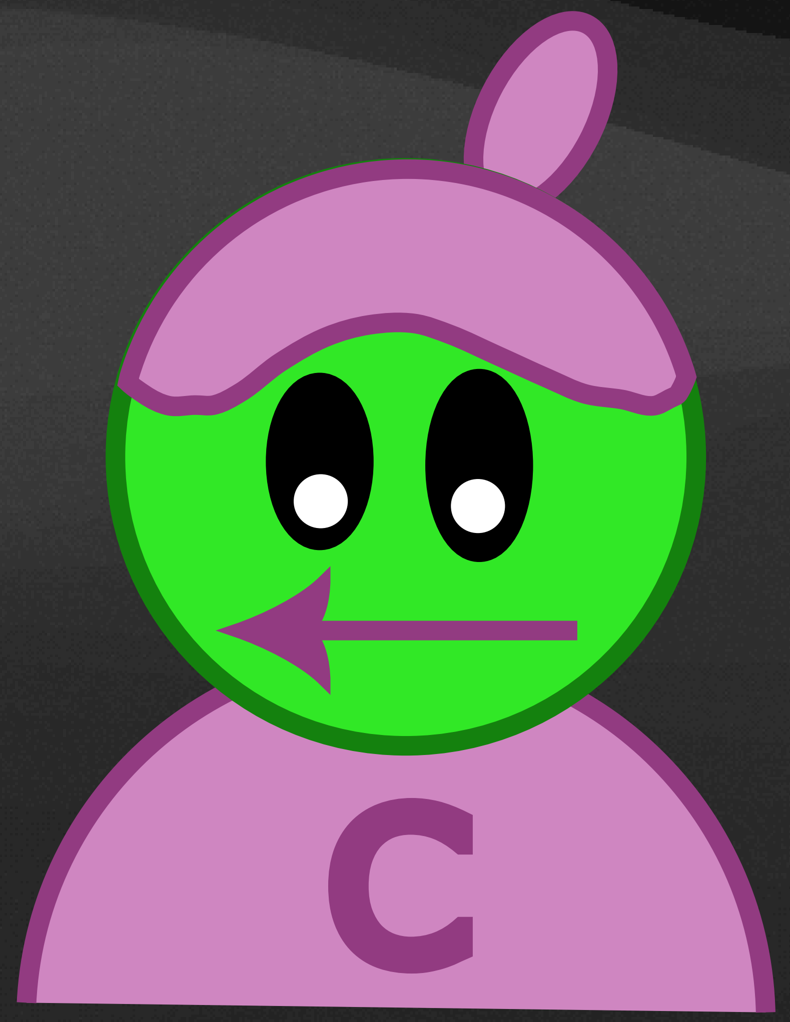
- Say Alice and Bob (humans) both speak (and think) forward

Theory of AoT

Time Symmetry Breaking in Language

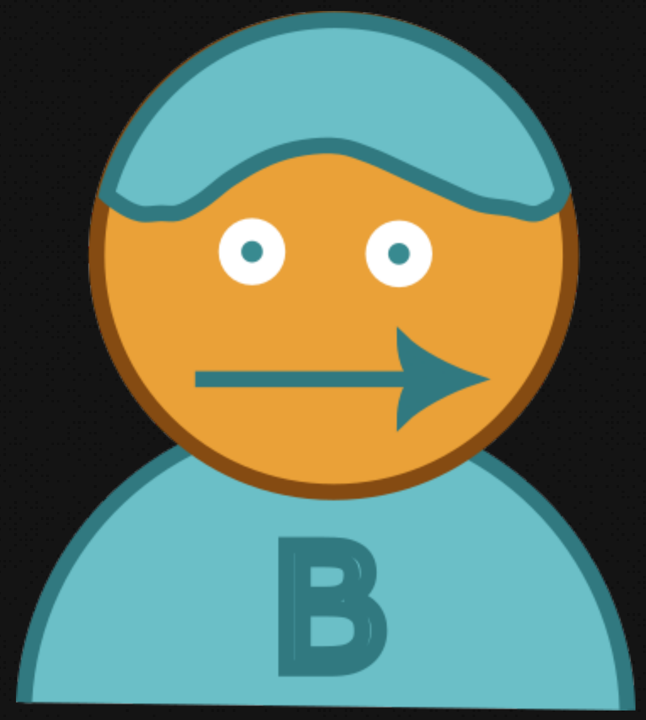


- Say Alice and Bob (humans) both speak (and think) forward
  - Consider also Carol (an alien) who speaks backwards.



Theory of AoT

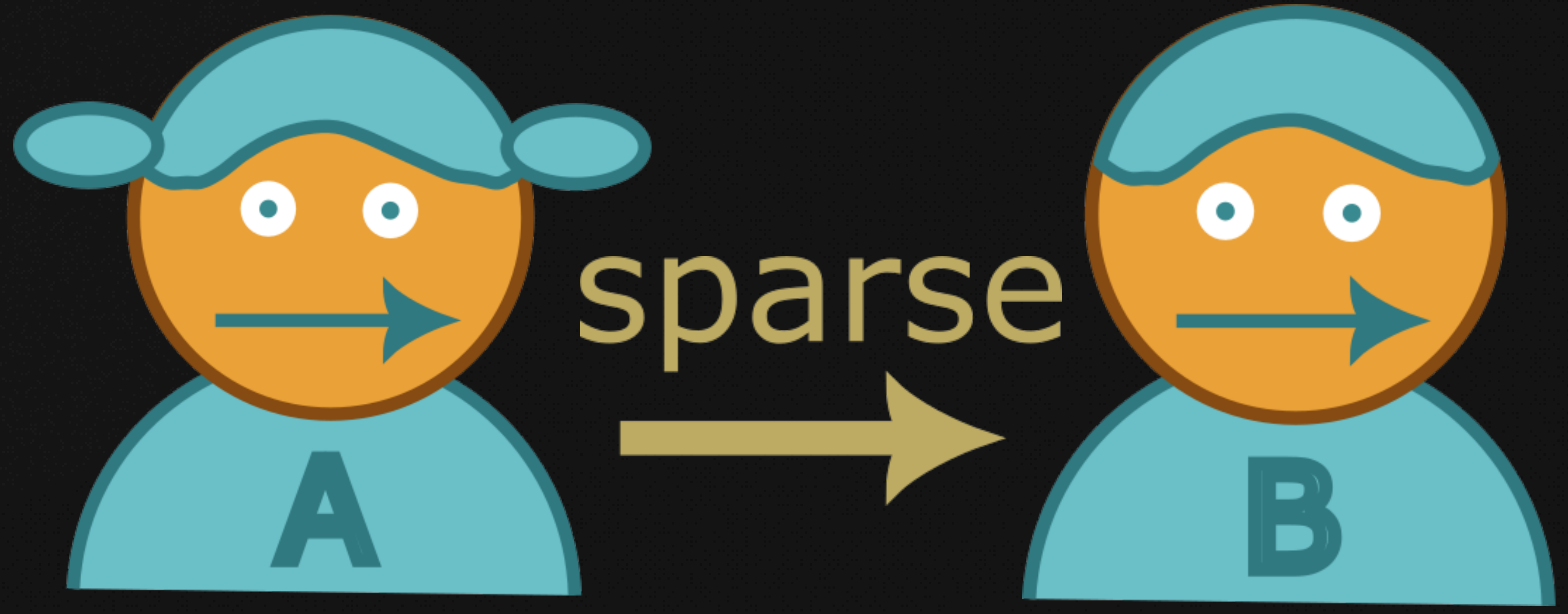
Time Symmetry Breaking in Language



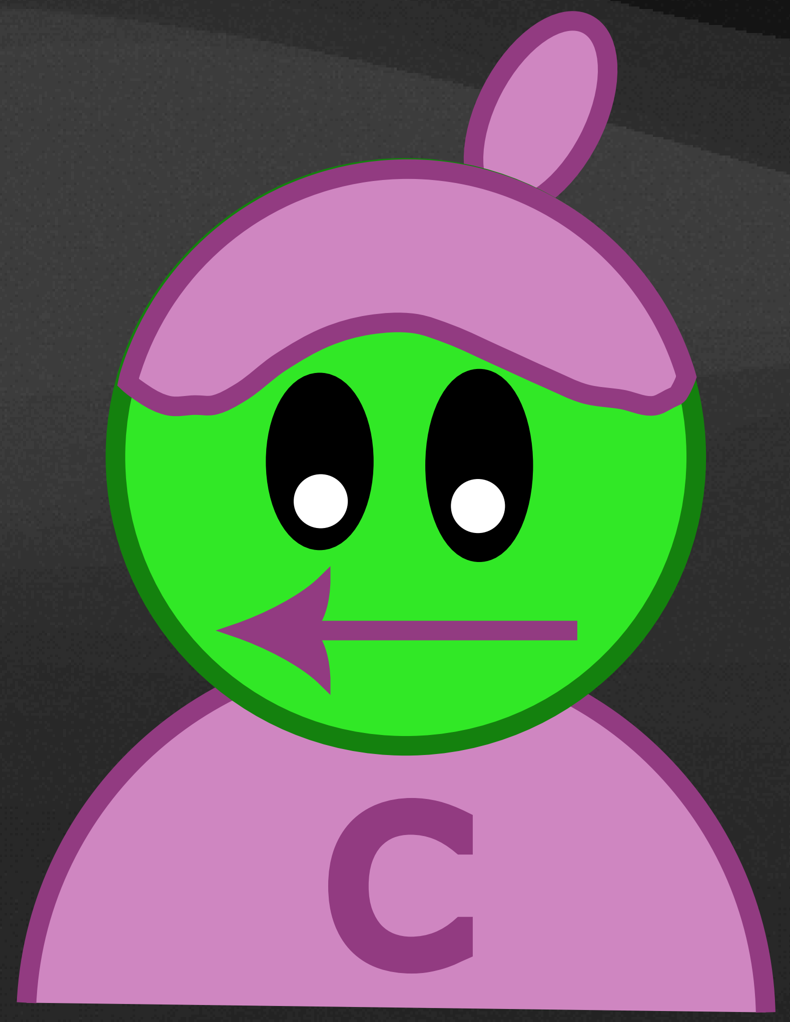
- Say Alice and Bob (humans) both speak (and think) forward
  - Consider also Carol (an alien) who speaks backwards.
- Alice, Bob, and Carol all share a common language

Theory of AoT

Time Symmetry Breaking in Language

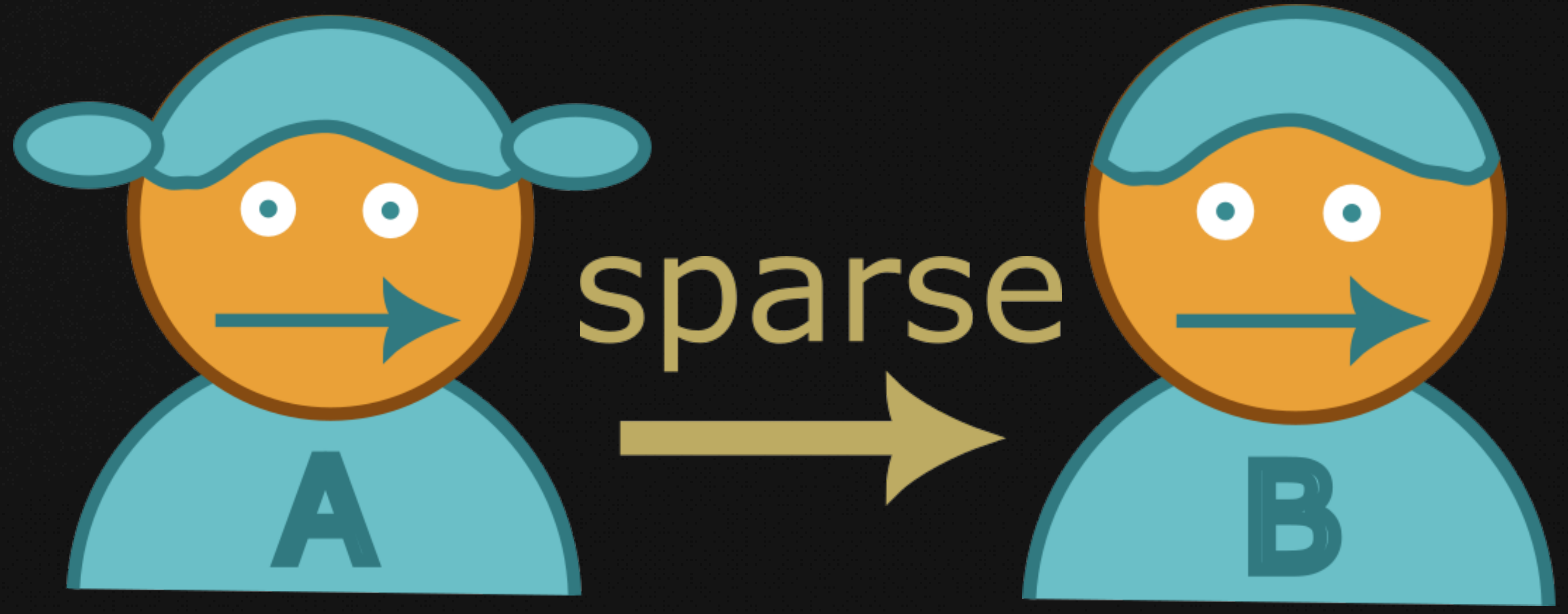


- Say Alice and Bob (humans) both speak (and think) forward
  - Consider also Carol (an alien) who speaks backwards.
- Alice, Bob, and Carol all share a common language
- Alice will only share forward-sparse modifications of the language to Bob: that's all he can learn easily.



Theory of AoT

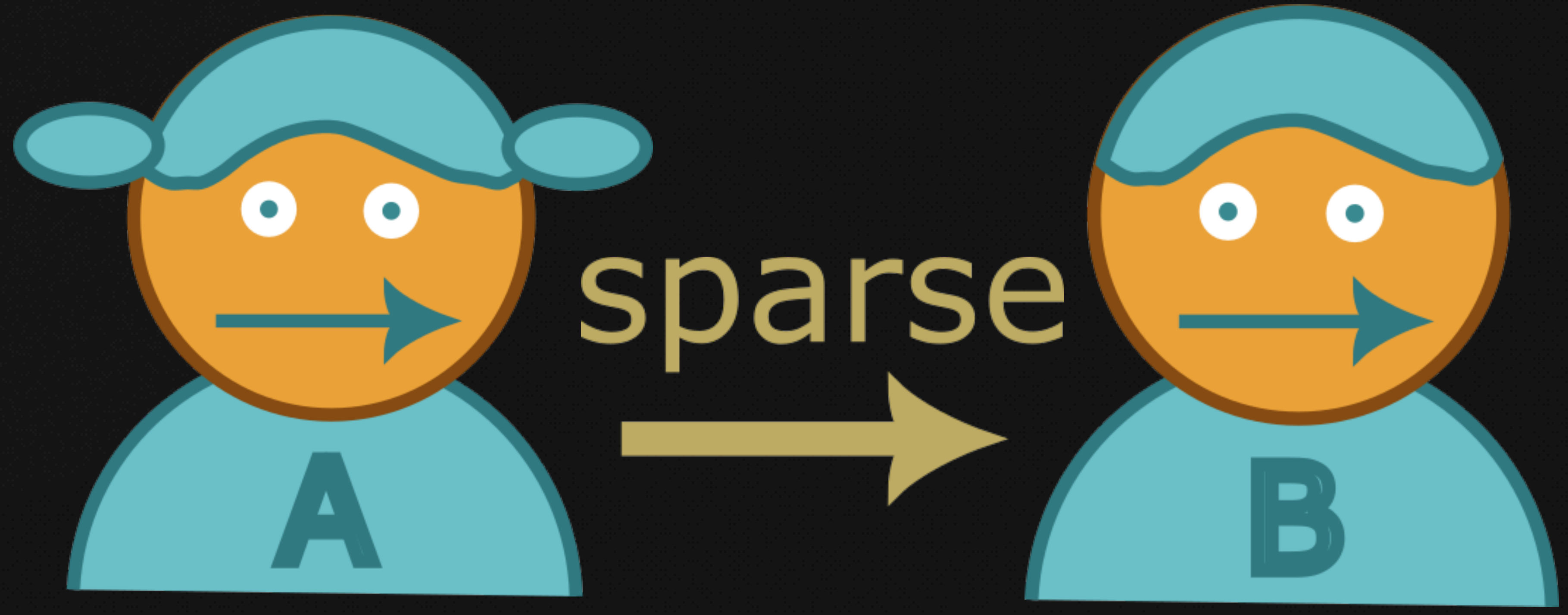
Time Symmetry Breaking in Language



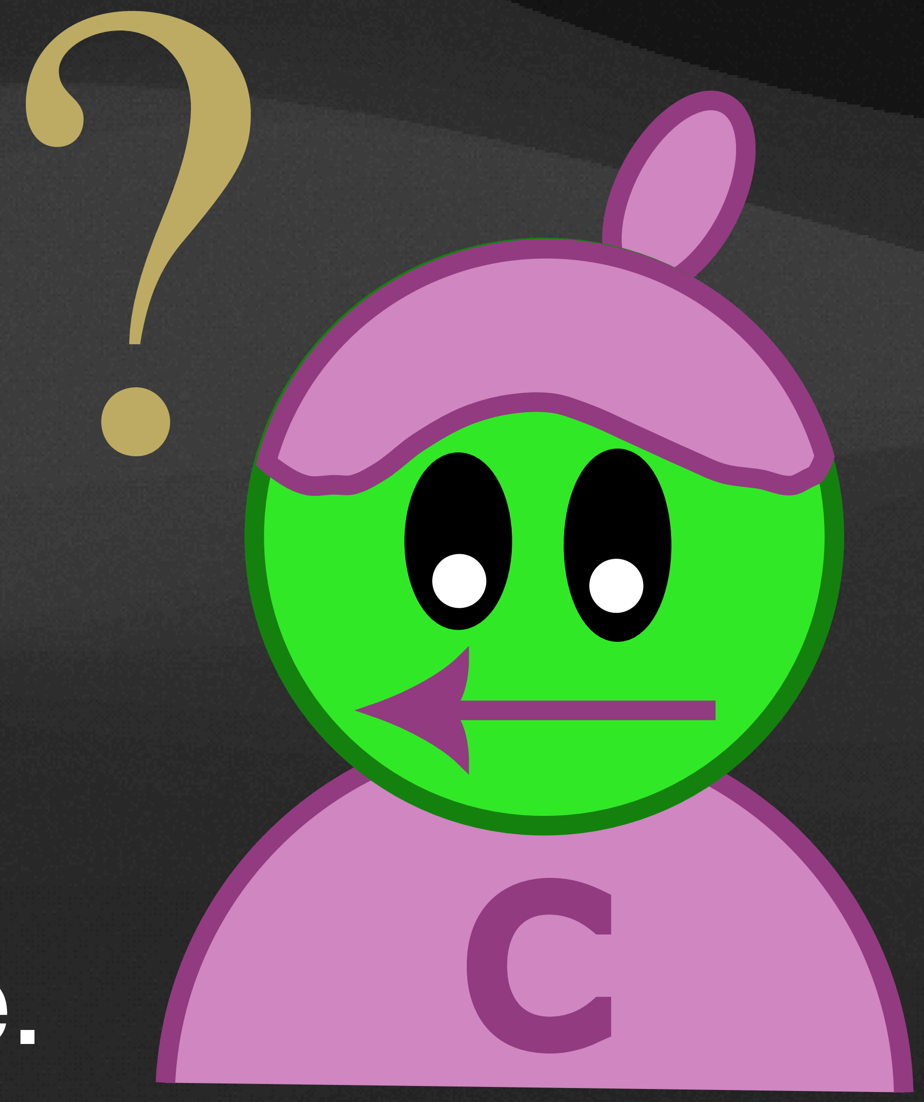
- Say Alice and Bob (humans) both speak (and think) forward
  - Consider also Carol (an alien) who speaks backwards.
- Alice, Bob, and Carol all share a common language
- Alice will only share forward-sparse modifications of the language to Bob: that's all he can learn easily.
- For Carol, things are harder: the update is not as backward-sparse.

Theory of AoT

Time Symmetry Breaking in Language

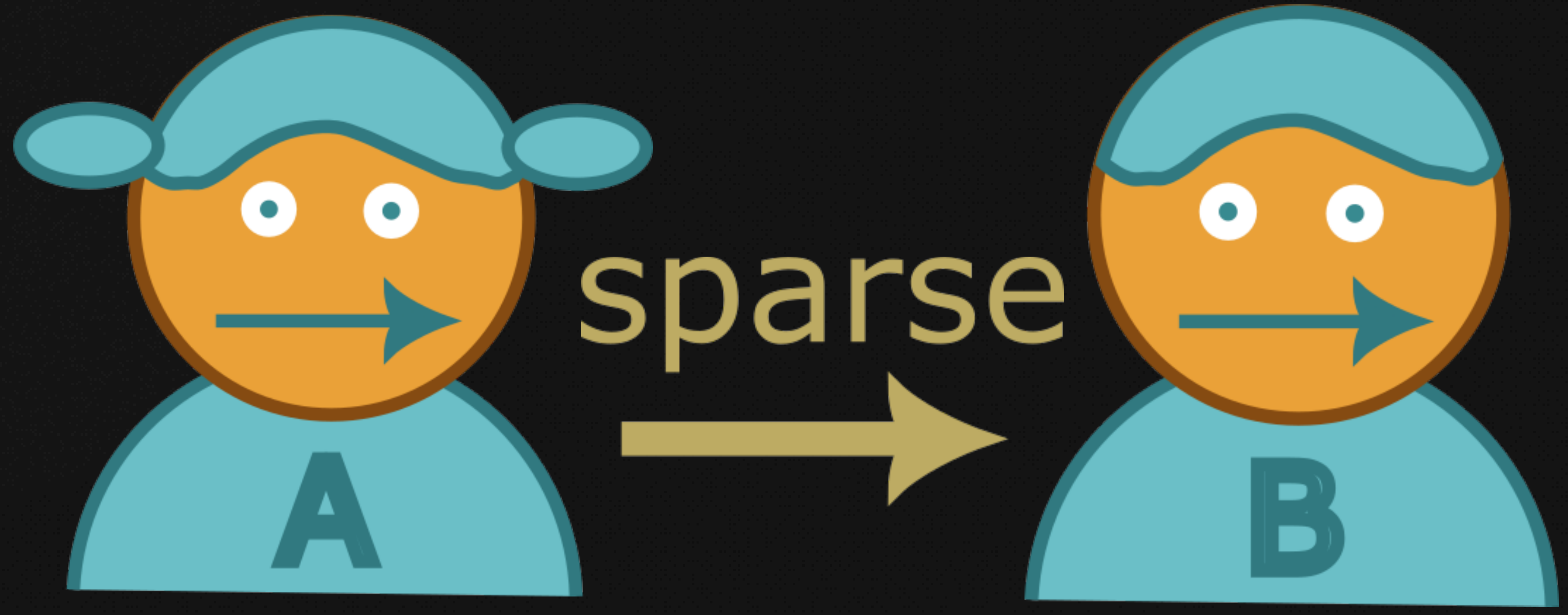


- Say Alice and Bob (humans) both speak (and think) forward
  - Consider also Carol (an alien) who speaks backwards.
- Alice, Bob, and Carol all share a common language
- Alice will only share forward-sparse modifications of the language to Bob: that's all he can learn easily.
- For Carol, things are harder: the update is not as backward-sparse.
- AoT emerges from the selection process:

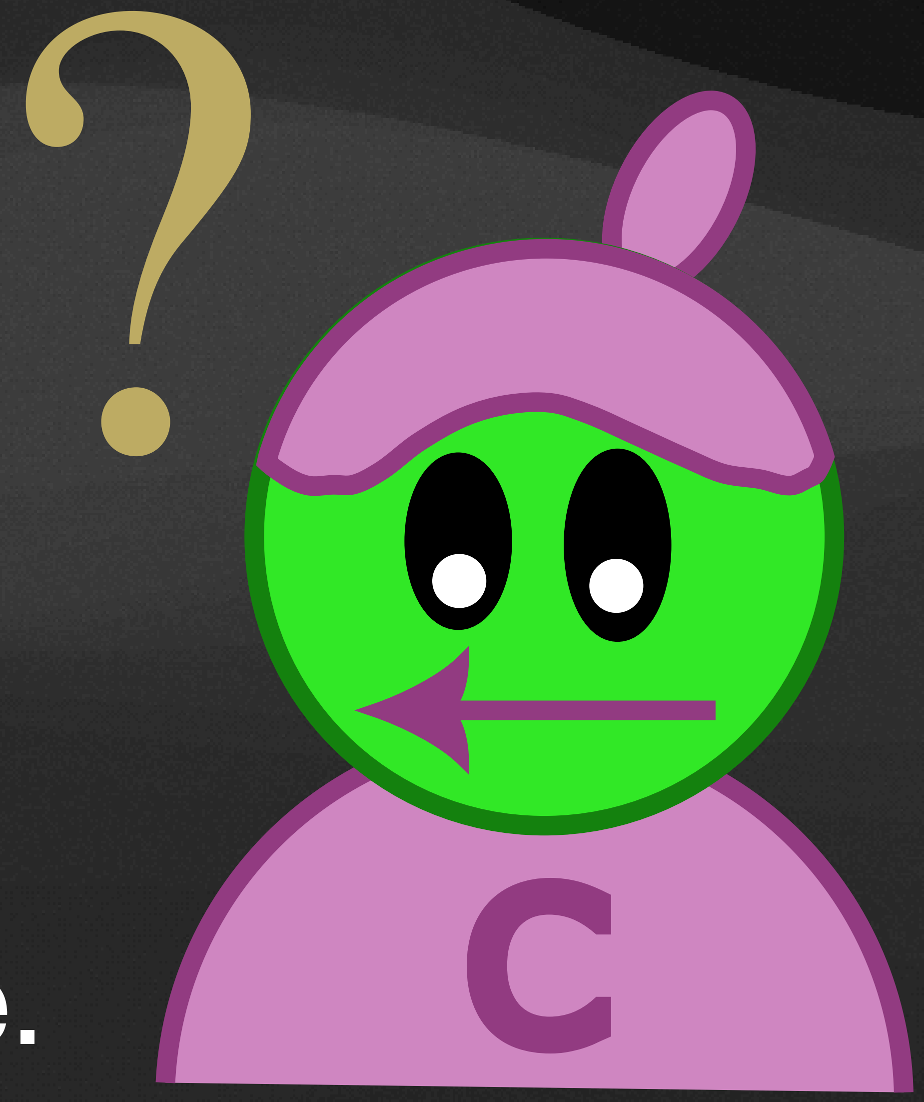


Theory of AoT

Time Symmetry Breaking in Language



- Say Alice and Bob (humans) both speak (and think) forward
  - Consider also Carol (an alien) who speaks backwards.
- Alice, Bob, and Carol all share a common language
- Alice will only share forward-sparse modifications of the language to Bob: that's all he can learn easily.
- For Carol, things are harder: the update is not as backward-sparse.
- AoT emerges from the selection process:
  - Alice only communicates sparse-forward updates (because that's what is easy for Bob); typically Carol struggles more.



Future Research Directions

Future Research Directions

- Is $\partial^{\leftrightarrow}_{CE} > 0$ linked with intelligence, life?

Future Research Directions

- Is $\partial_{CE}^{\leftrightarrow} > 0$ linked with intelligence, life?
- AoT on other types of data (code, binaries, DNA, animal sounds)?

Future Research Directions

- Is $\partial^{\leftrightarrow}_{CE} > 0$ linked with intelligence, life?
- AoT on other types of data (code, binaries, DNA, animal sounds)?
- Relation with AoT in thermodynamics?

Future Research Directions

- Is $\partial_{CE}^{\leftrightarrow} > 0$ linked with intelligence, life?
- AoT on other types of data (code, binaries, DNA, animal sounds)?
- Relation with AoT in thermodynamics?
- AoT link with causality?

Future Research Directions

- Is $\partial_{CE}^{\leftrightarrow} > 0$ linked with intelligence, life?
- AoT on other types of data (code, binaries, DNA, animal sounds)?
- Relation with AoT in thermodynamics?
- AoT link with causality?
- Are there less data-intensive ways to detect an AoT?

Future Research Directions

- Is $\partial_{CE}^{\leftrightarrow} > 0$ linked with intelligence, life?
- AoT on other types of data (code, binaries, DNA, animal sounds)?
- Relation with AoT in thermodynamics?
- AoT link with causality?
- Are there less data-intensive ways to detect an AoT?
- Scaling laws for $\partial_{CE}^{\leftrightarrow}$?

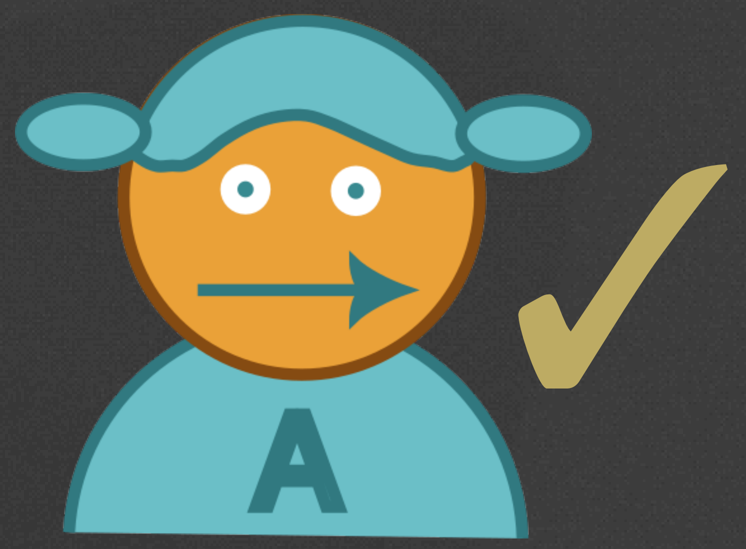
Future Research Directions

- Is $\partial_{CE}^{\leftrightarrow} > 0$ linked with intelligence, life?
- AoT on other types of data (code, binaries, DNA, animal sounds)?
- Relation with AoT in thermodynamics?
- AoT link with causality?
- Are there less data-intensive ways to detect an AoT?
- Scaling laws for $\partial_{CE}^{\leftrightarrow}$?

Thank you for your attention!

Future Research Directions

- Is $\partial_{CE}^{\leftrightarrow} > 0$ linked with intelligence, life?
- AoT on other types of data (code, binaries, DNA, animal sounds)?
- Relation with AoT in thermodynamics?
- AoT link with causality?
- Are there less data-intensive ways to detect an AoT?
- Scaling laws for $\partial_{CE}^{\leftrightarrow}$?



Thank you for your attention!

Hopefully, this talk was sparse in your favorite time direction!