On the Power of Context-Enhanced Learning in LLMs

Xingyu Zhu*, Abhishek Panigrahi*, Sanjeev Arora

PLI Seminar 3/27/2025

Who Knows Calculus?

Who has Benefited from a Calculus Textbook?

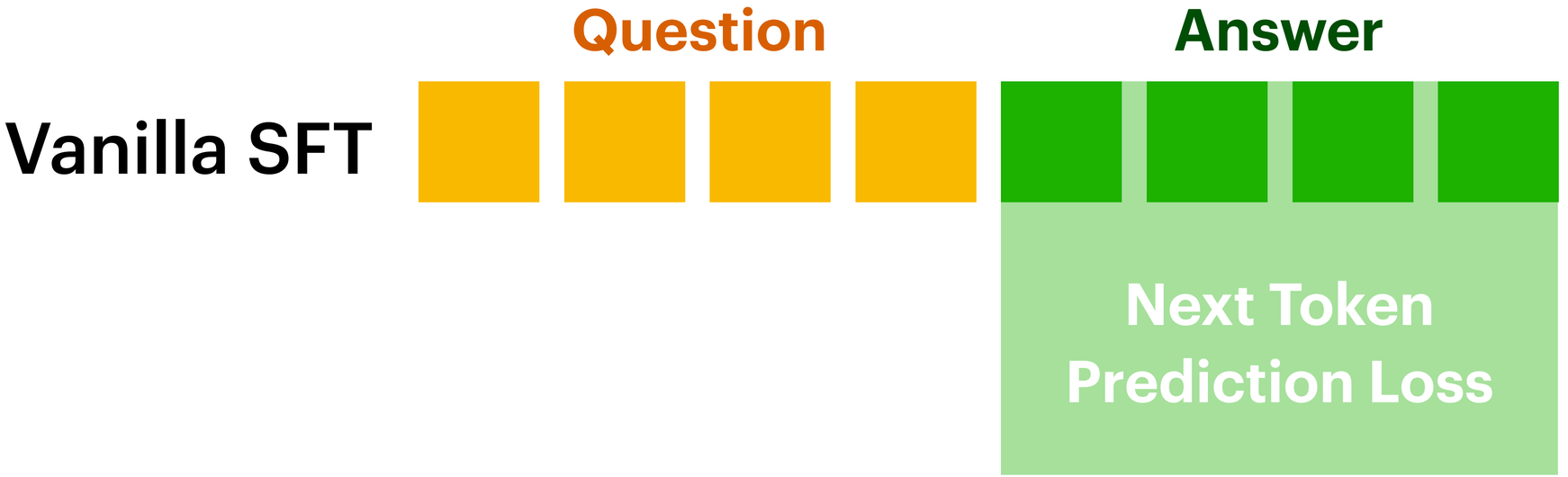
Who Has Tried to
Verbatim Memorize
the Calculus Textbook?

Who Has Tried to **Verbatim Memorize** the Calculus Textbook?

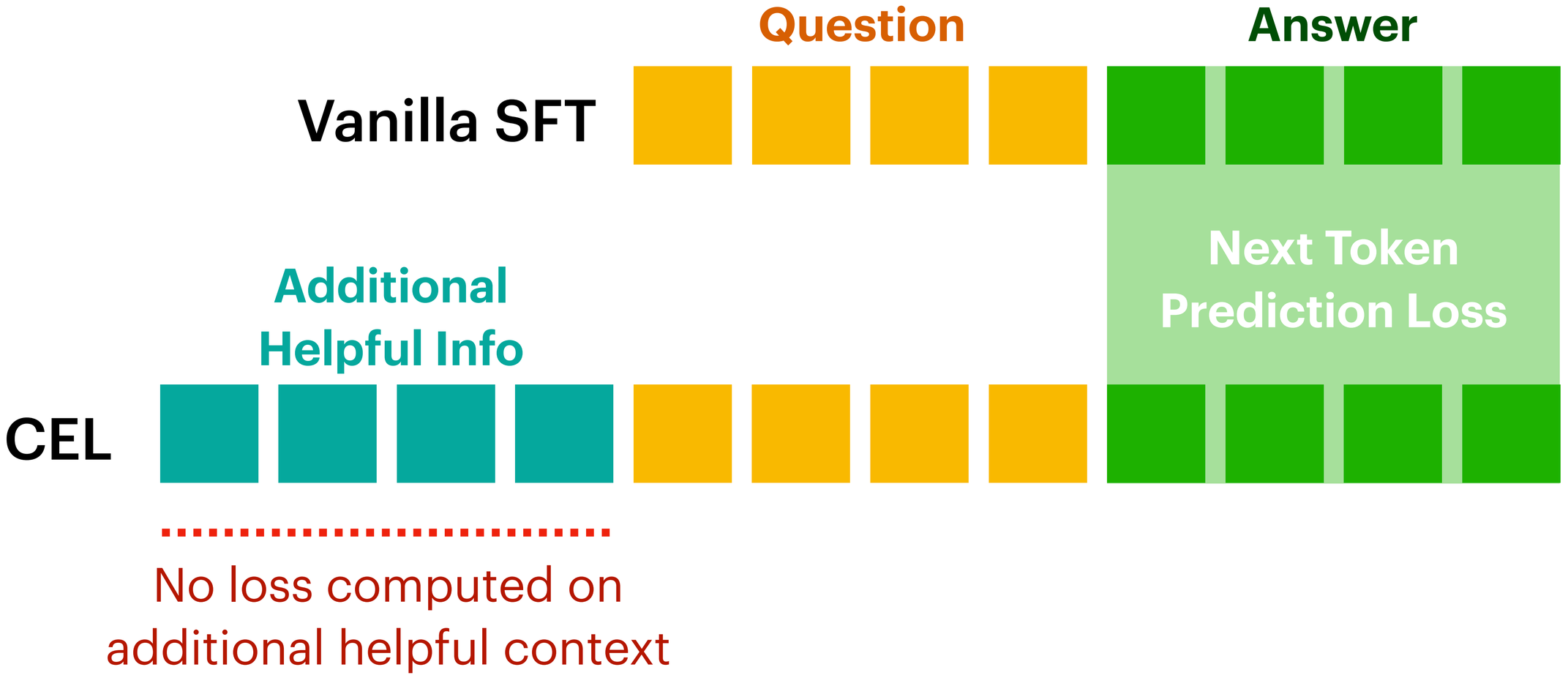
(LLMs do)

Can LLM training benefit from **in-context** information with **no gradients computed on them**?

Context-Enhanced Learning (CEL)

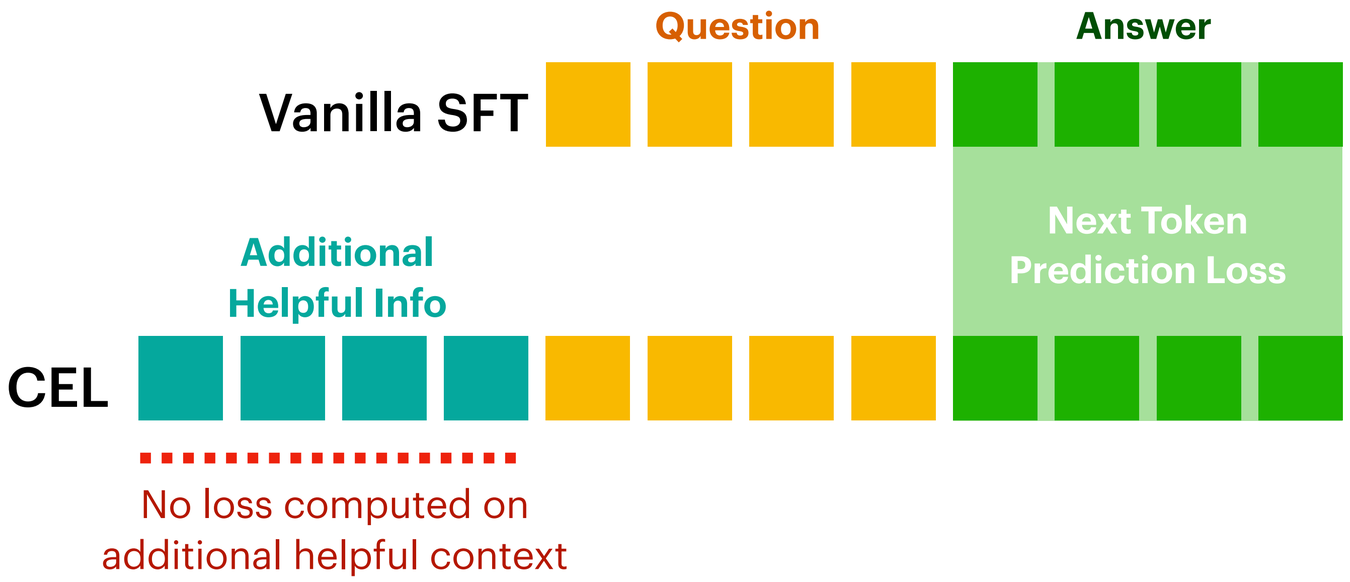


Context-Enhanced Learning (CEL)



The additional helpful context is **only presented in training**

Context-Enhanced Learning (CEL)



- Examples of helpful context:
  - Few-shot examples[1]
  - URLs of the source[2]
  - Thought-provoking guidance[3]
- The helpful context can depend on the input and training step - in-context curriculum

[1] Liao, Huanxuan, et al. "SKIntern: Internalizing Symbolic Knowledge for Distilling Better CoT Capabilities into Small Language Models."
[2] Gao, Tianyu, et al. "Metadata Conditioning Accelerates Language Model Pre-training.
[3] Choi, Younwoo, et al. "Teaching LLMs How to Learn with Contextual Fine-Tuning."

Research Questions

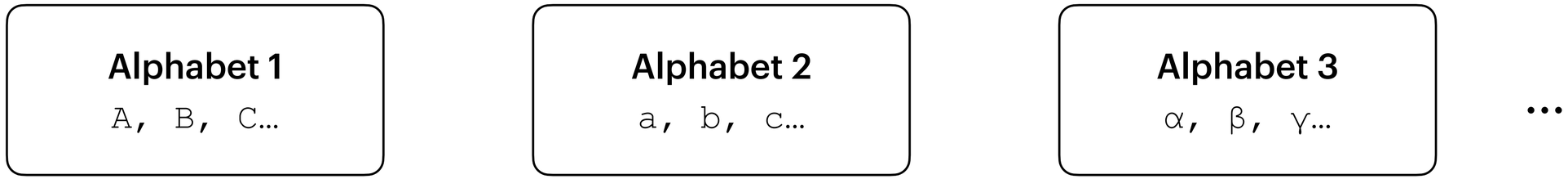
1. How powerful can CEL be (vs. vanilla SFT)?
2. Does CEL require certain model capabilities?
3. Does the model explicitly memorize info provided only in context?

Synthetic Testbed: Multi-Layer Translation (MLT)

- $d + 1$ alphabets / languages

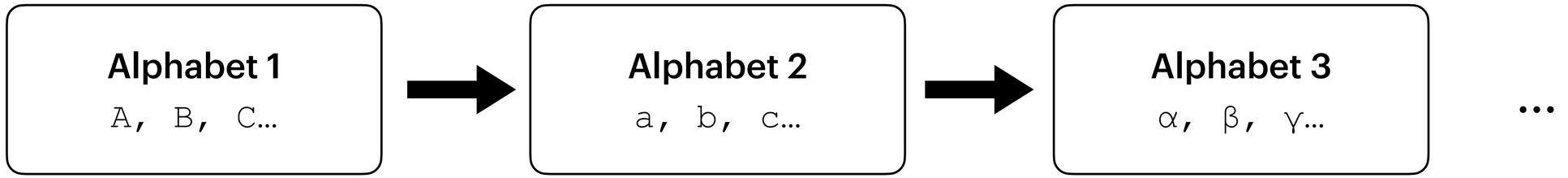
Synthetic Testbed: Multi-Layer Translation (MLT)

- $d + 1$ alphabets / languages



Synthetic Testbed: Multi-Layer Translation (MLT)

- $d + 1$ alphabets / languages
- d phrasebooks to translate between 2-tuples in consecutive alphabets



Phrasebook π_1

BC → eb DE → hd

FG → af HA → gc

…

Phrasebook π_2

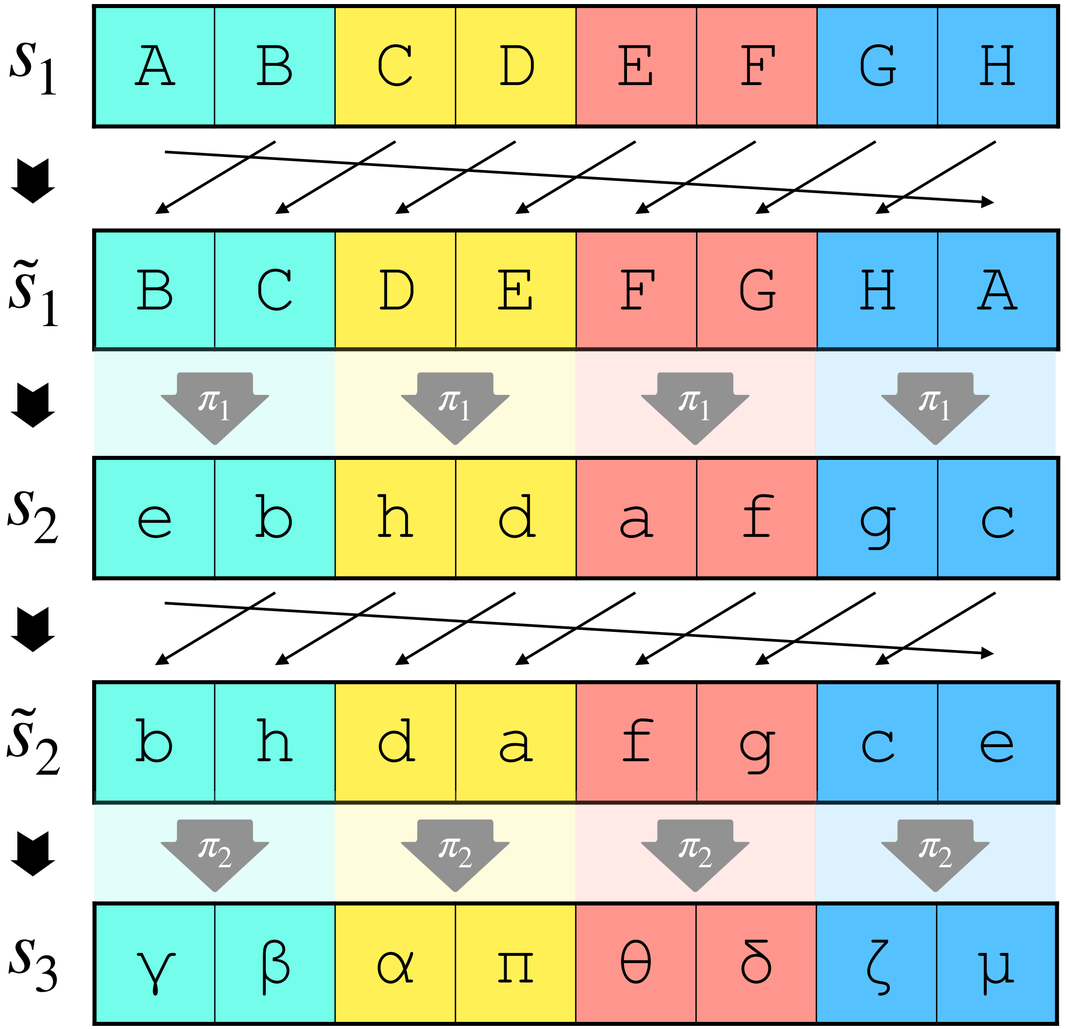
bh → γβ da → απ

fg → θδ ce → ζμ

…

…

Multi-Layer Translation (MLT)



Circular Shift

Phrasebook π_1

BC → eb DE → hd

FG → af HA → gc

…

Circular Shift

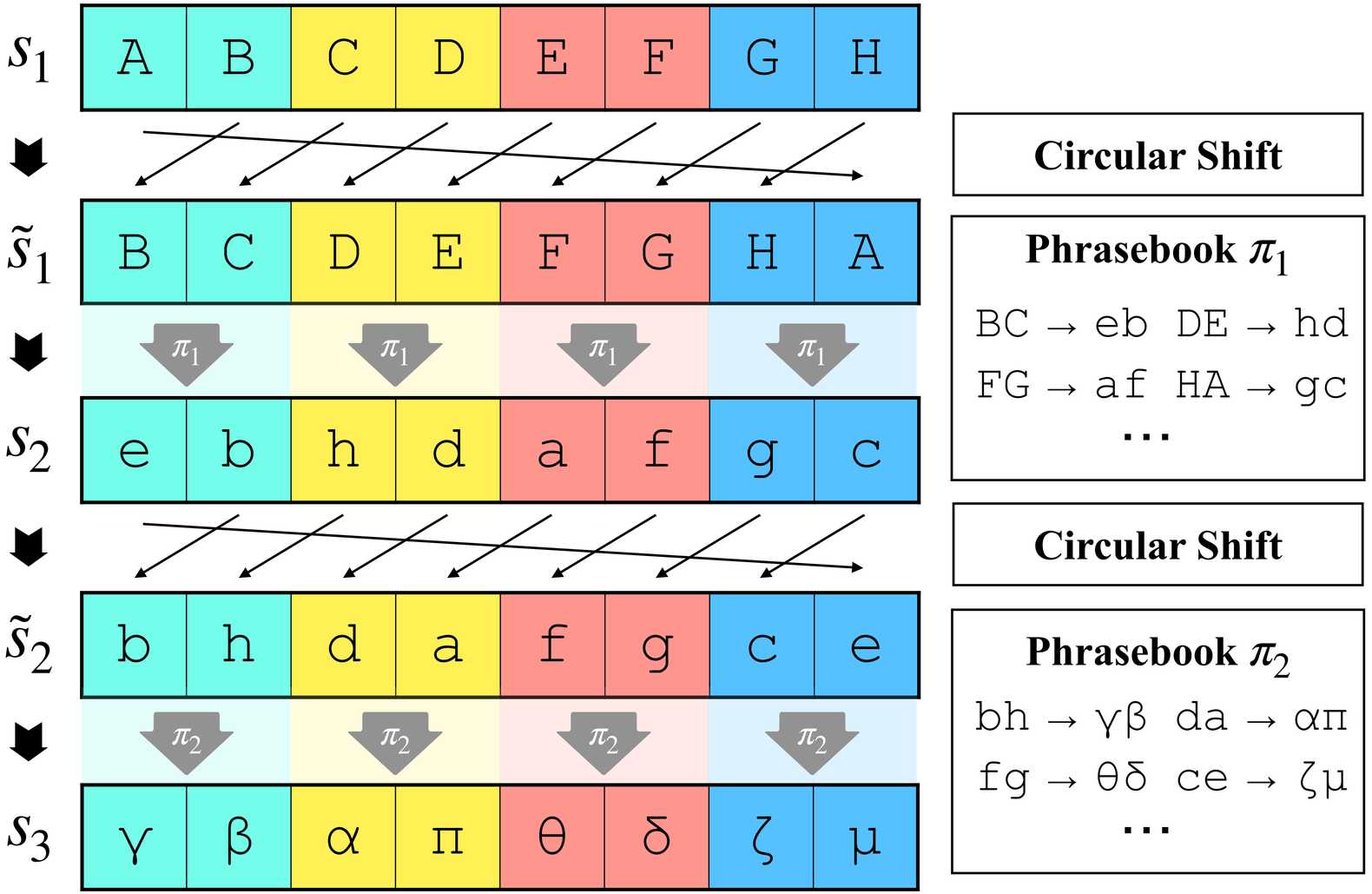
Phrasebook π_2

bh → γβ da → απ

fg → θδ ce → ζμ

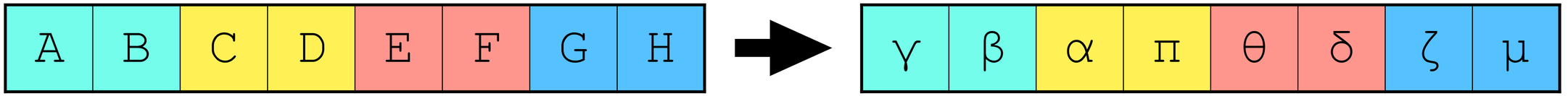
…

$$s_{d+1} = f_{\pi_d} \circ f_{\circlearrowright} \circ f_{\pi_{d-1}} \circ f_{\circlearrowright} \circ \cdots \circ f_{\pi_1} \circ f_{\circlearrowright}(s_1)$$



Multi-Layer Translation (MLT)

- Task: Fix d phrasebooks $(\pi_1^*, \pi_2^*, \ldots, \pi_d^*)$, learn the translation process defined by the phrasebooks via (input, output) pairs **without intermediate steps.**



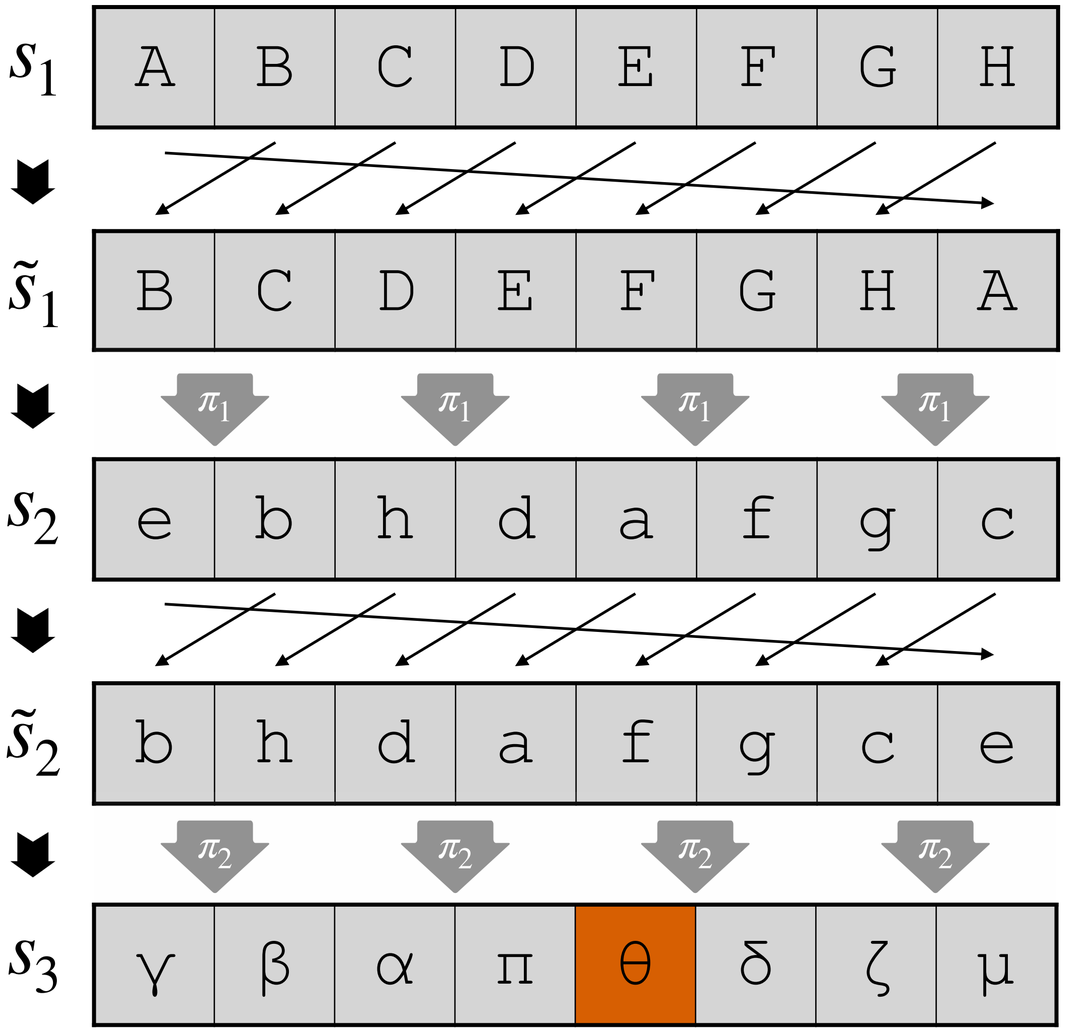
SFT

Prompt: Your are translating a sequence from language 1 to language 3. The input in language 1 is "A B C D E F G H".

Answer: The output in language 3 is "γ β α π θ δ ζ μ".

$$s_{d+1} = f_{\pi_d} \circ f_{\circlearrowleft} \circ f_{\pi_{d-1}} \circ f_{\circlearrowleft} \circ \cdots \circ f_{\pi_1} \circ f_{\circlearrowleft}(s_1)$$

Learning MLT from Input-Output is Hard



Circular Shift

Phrasebook π_1

BC → eb DE → hd

FG → af HA → gc

…

Circular Shift

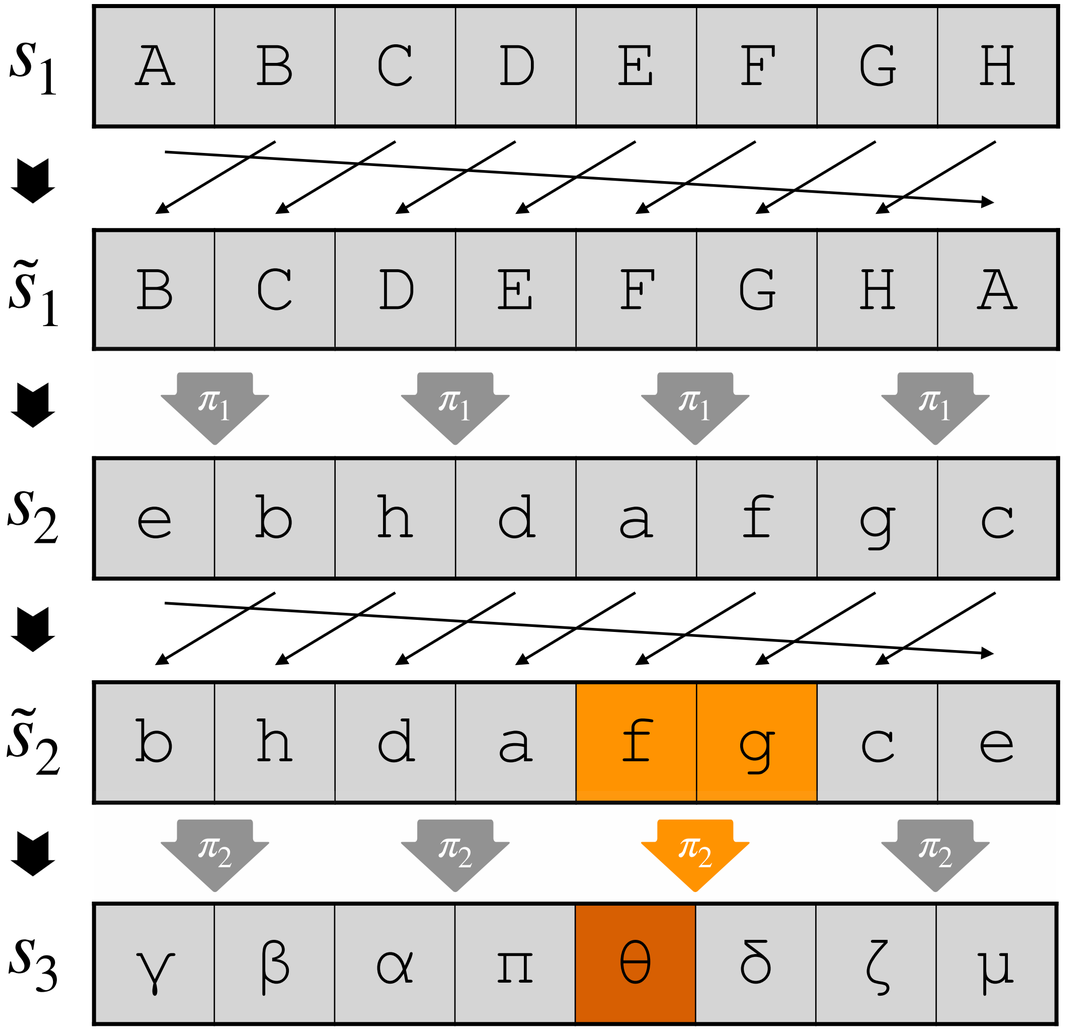
Phrasebook π_2

bh → γβ da → απ

fg → θδ ce → ζμ

…

Learning MLT from Input-Output is Hard



Circular Shift

Phrasebook π_1

BC → eb DE → hd

FG → af HA → gc

...

Circular Shift

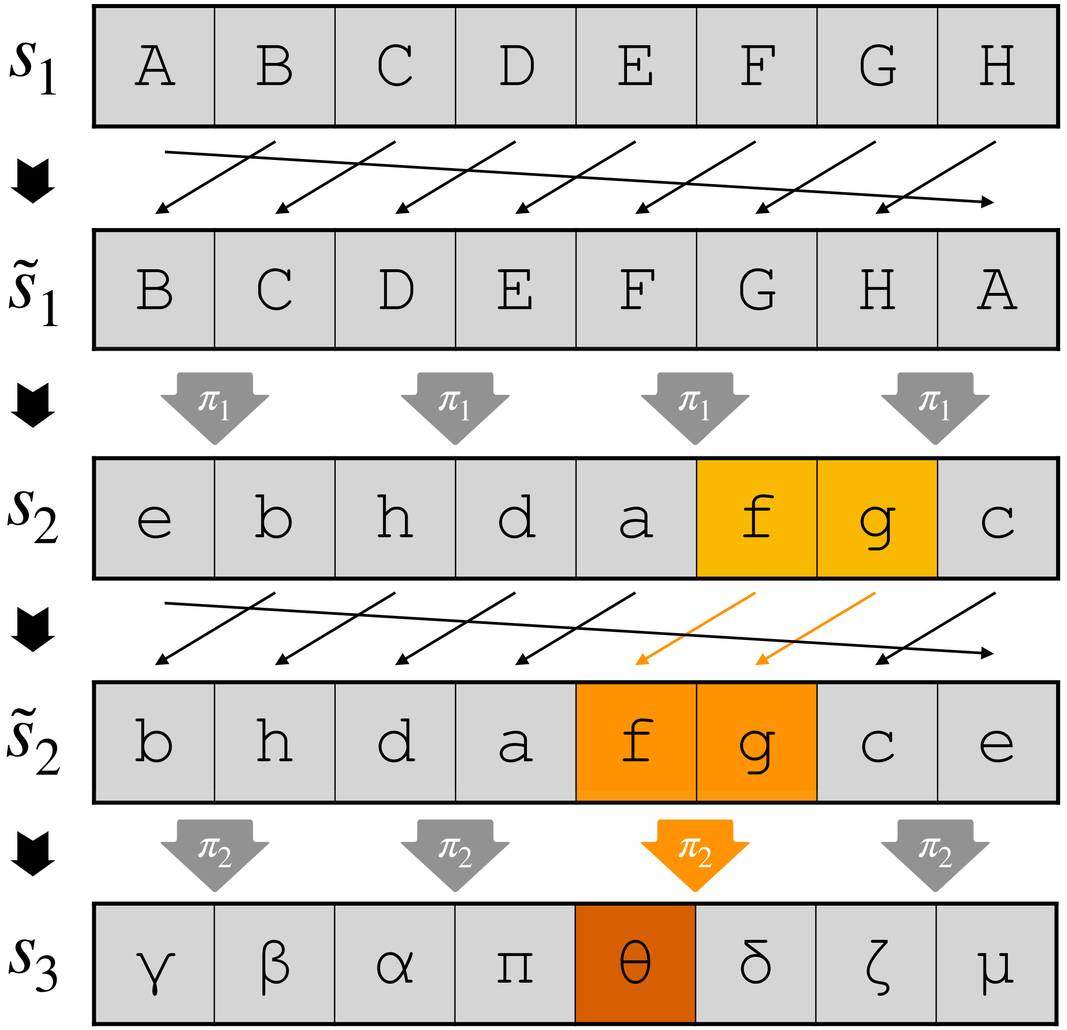
Phrasebook π_2

bh → γβ da → απ

fg → θδ ce → ζμ

...

Learning MLT from Input-Output is Hard



Circular Shift

Phrasebook π_1

BC → eb DE → hd

FG → af HA → gc

...

Circular Shift

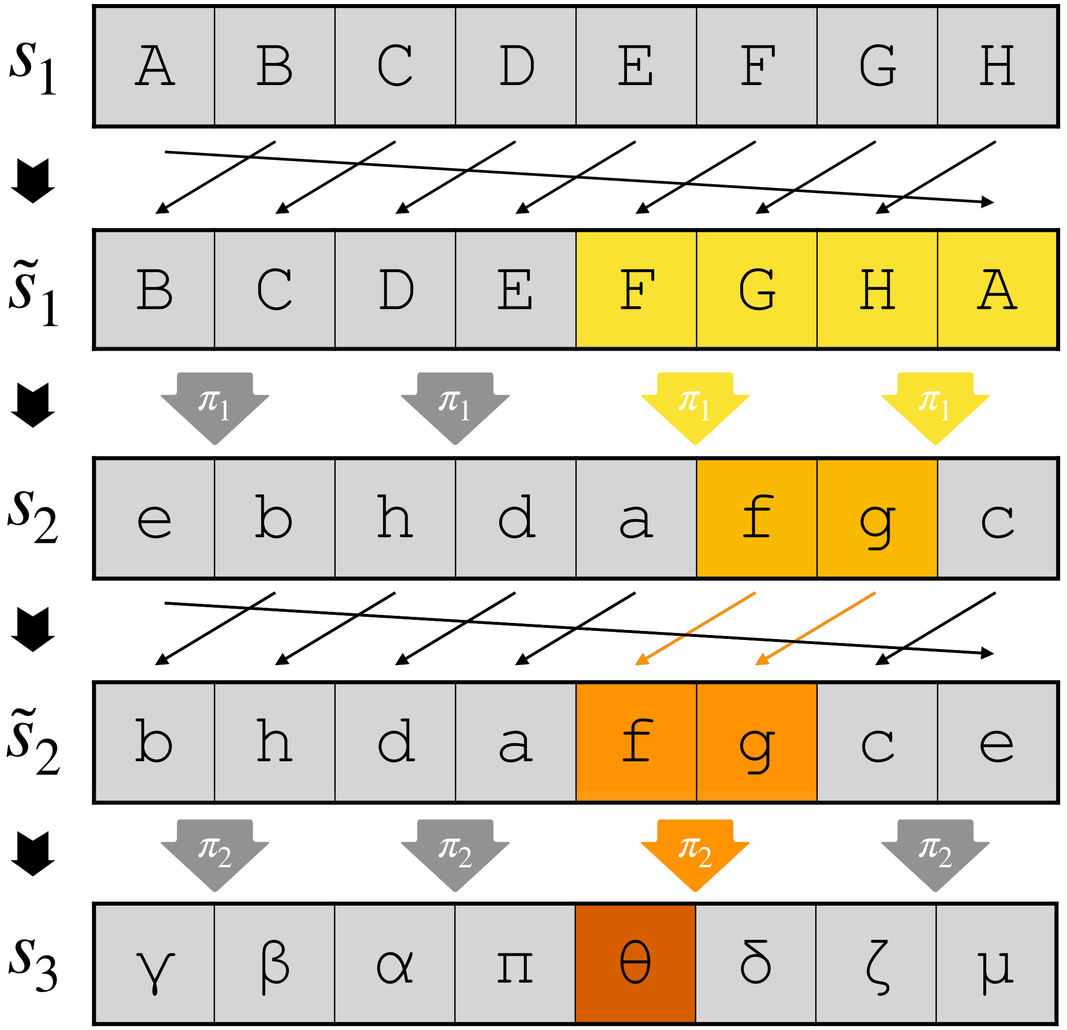
Phrasebook π_2

bh → γβ da → απ

fg → θδ ce → ζμ

...

Learning MLT from Input-Output is Hard



Circular Shift

Phrasebook π_1

BC → eb DE → hd

FG → af HA → gc

…

Circular Shift

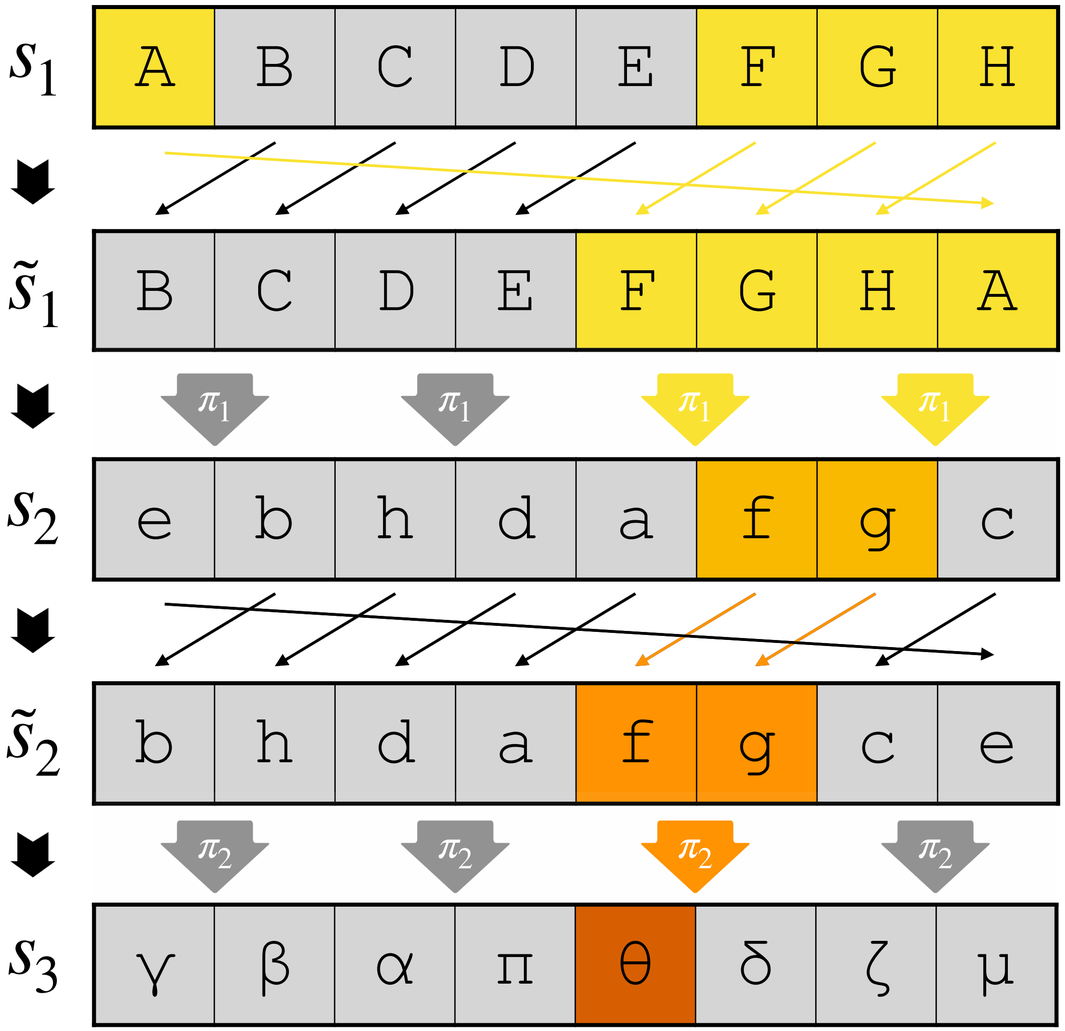
Phrasebook π_2

bh → γβ da → απ

fg → θδ ce → ζμ

…

Learning MLT from Input-Output is Hard



Circular Shift

Phrasebook π_1

BC → eb DE → hd

FG → af HA → gc

...

Circular Shift

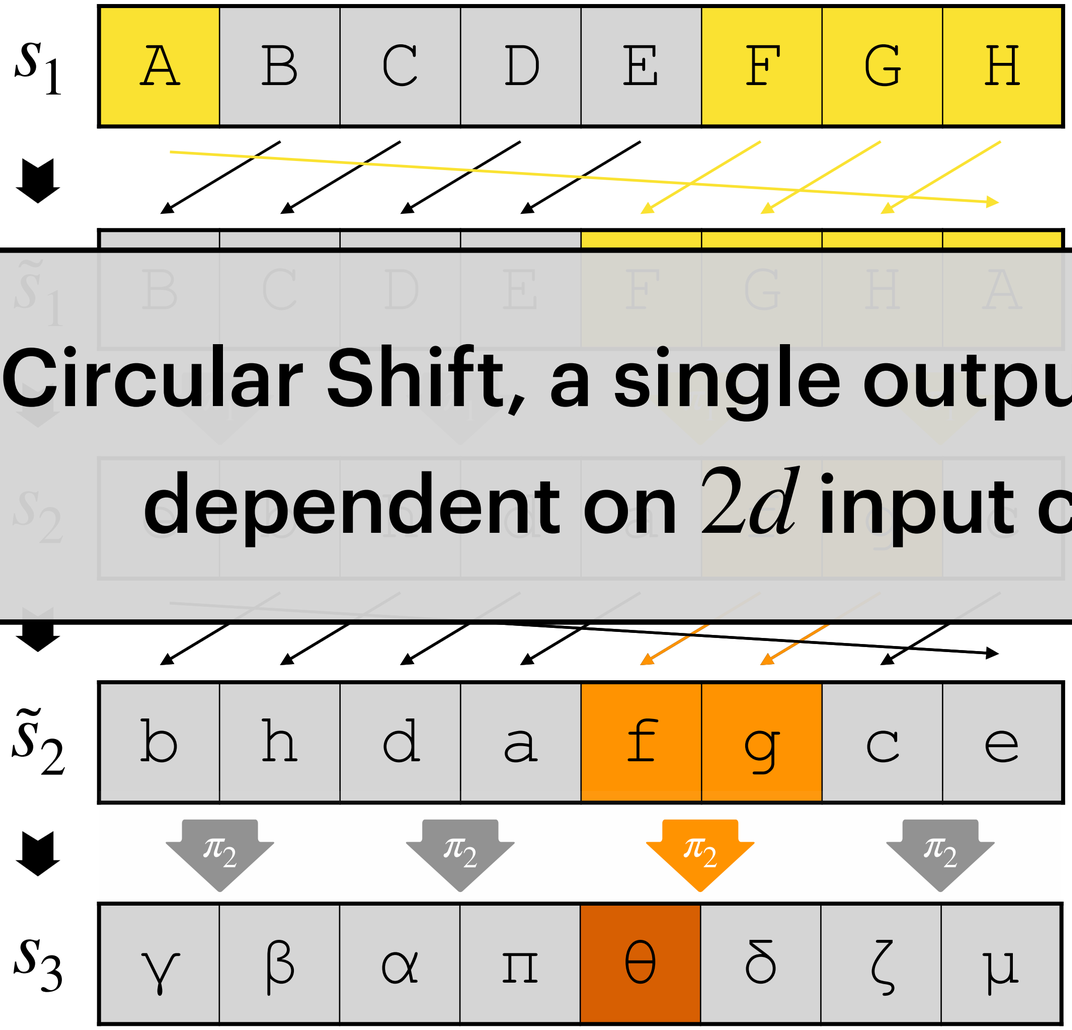
Phrasebook π_2

bh → γβ da → απ

fg → θδ ce → ζμ

...

Learning MLT from Input-Output is Hard



With Circular Shift, a single output character can be dependent on $2d$ input characters

Learning MLT from Input-Output is Provably Hard

- The family of d-depth MLT has SQ-dimension $e^{\Omega(d)}$ under uniform inputs.
- There are MLT instances that **require** $e^{\Omega(d)}$ **samples** to learn from (input seq, output seq) pairs using SFT (**without intermediate steps**).[1]

[1] Edelman, Benjamin L., et al. "Pareto frontiers in neural feature learning: Data, compute, width, and luck."

Learning MLT from Input-Output is Provably Hard

- The family of d-depth MLT has SQ-dimension $e^{\Omega(d)}$ under uniform inputs.
- There are MLT instances that **require** $e^{\Omega(d)}$ **samples** to learn from (input seq, output seq) pairs using SFT (**without intermediate steps**).[1]

What if we provide the phrasebooks purely in-context during training?

[1] Edelman, Benjamin L., et al. "Pareto frontiers in neural feature learning: Data, compute, width, and luck."

Context-Enhanced Learning (CEL) Experiments for Learning MLT

Context-Enhanced Learning for MLT

SFT



Prompt: Your are translating a sequence from language 1 to language 3. The input in language 1 is "**A B C D E F G H**".

Answer: The output in language 3 is "**γ β α π θ δ ζ μ**".

CEL

Prompt: Your are translating a sequence from language 1 to language 3.

Use phrasebook 1: B C → e b, F G → a f, D E → h d, H A → g c and phrasebook 2: b h → γ β, f g → θ δ, d a → α π, c e → ζ μ.



The input in language 1 is "**A B C D E F G H**".

Answer: The output in language 3 is "**γ β α π θ δ ζ μ**".

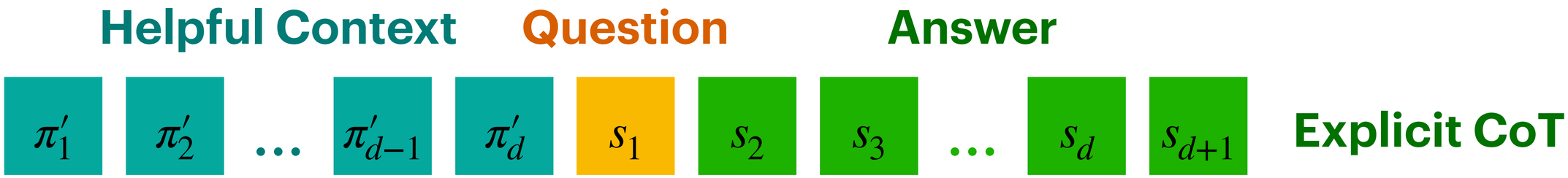
Step 1: Prepare an “ICL-capable” model

- Intuition: To benefit from a textbook while learning, human needs to “understand” the textbook described.
- To benefit from in-context information, the model needs to “understand” the information in-context.
- For MLT: start from a model that **knows how to read phrasebooks in context and apply them to translation**.

Step 1: Prepare an “ICL-capable” model

ICL-Capability Training

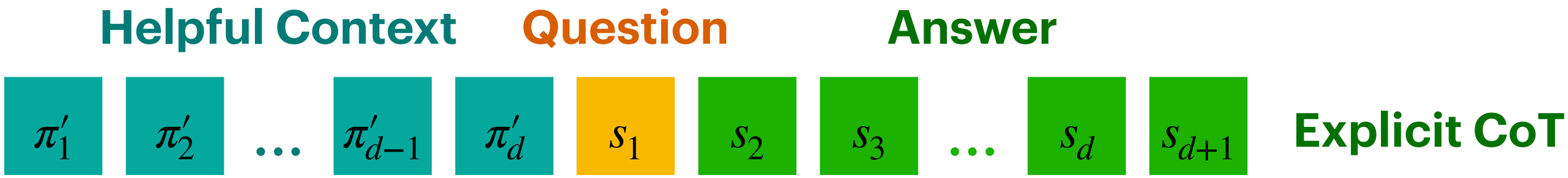
1. Sample random sets of phrasebooks and teach a Llama 3.2 3B model to do “open-book” translation task with explicit CoT.



Step 1: Prepare an “ICL-capable” model

ICL-Capability Training

1. Sample random sets of phrasebooks and teach a Llama 3.2 3B model to do “open-book” translation task with explicit CoT.
2. Internalize[1] intermediate CoT steps to abstract <T> tokens.

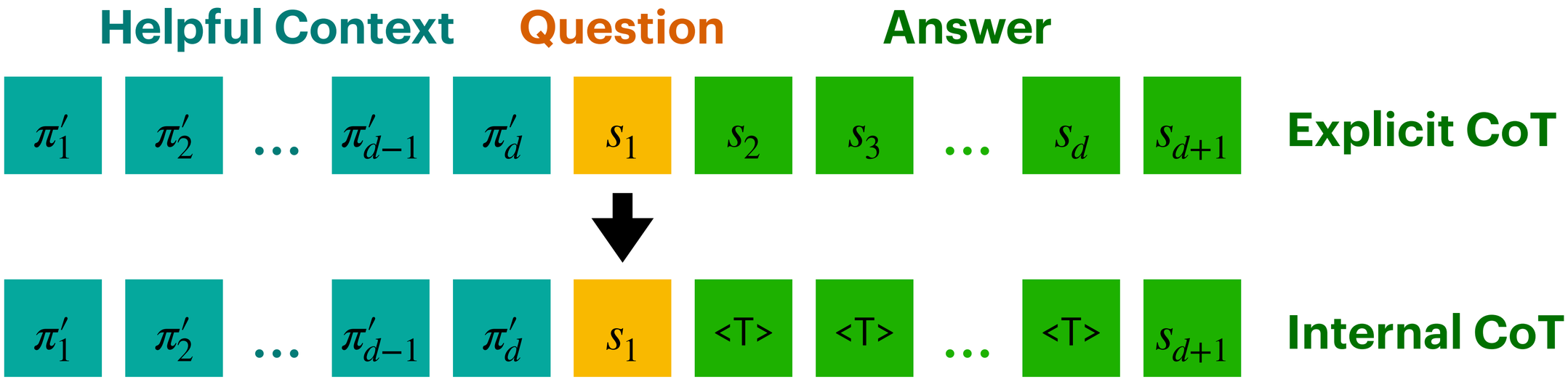


[1] Deng, Y., Choi, Y., and Shieber, S. From explicit cot to implicit cot: Learning to internalize cot step by step. (2024)

Step 1: Prepare an "ICL-capable" model

ICL-Capability Training

1. Sample random sets of phrasebooks and teach a Llama 3.2 3B model to do "open-book" translation task with explicit CoT.
2. Internalize[1] intermediate CoT steps to abstract <T> tokens.



[1] Deng, Y., Choi, Y., and Shieber, S. From explicit cot to implicit cot: Learning to internalize cot step by step. (2024)

Step 1: Prepare an “ICL-capable” model

ICL-Capability Training

1. Sample random sets of phrasebooks and teach a Llama 3.2 3B model to do “open-book” translation task with explicit CoT.
2. Internalize[1] intermediate CoT steps to abstract <T> tokens.

Step 1 gives a model that can do the translation perfectly with phrasebooks in context.

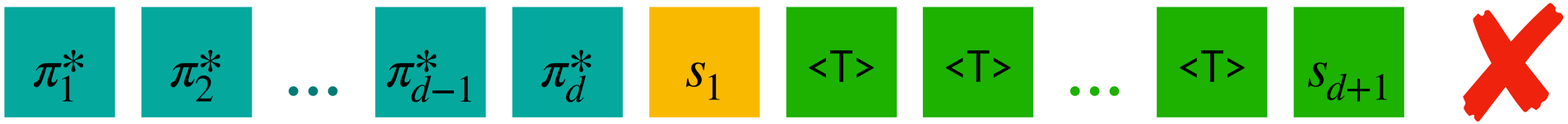
[1] Deng, Y., Choi, Y., and Shieber, S. From explicit cot to implicit cot: Learning to internalize cot step by step. (2024)

Step 2: CEL with ICL-Capable Model

- Now we fix d "target phrasebooks" $(\pi_1^*, \pi_2^*, \ldots, \pi_d^*)$, and hope the model can learn $f_{\pi_d^*} \circ f_{\circlearrowright} \circ f_{\pi_{d-1}^*} \circ f_{\circlearrowright} \circ \cdots \circ f_{\pi_1^*} \circ f_{\circlearrowright}$ faster via input-output pairs + in-context phrasebooks.



Open-book setting (from Step 1)



Closed-book setting

Step 2: CEL with ICL-Capable Model

- We want the model to learn the task without relying on the "open-book" materials.
- Simple solution: **random dropout** on the helpful context tokens.

CEL

Prompt: Your are translating a sequence from language 1 to language 3.

Use phrasebook 1: B C → e b, F G → a f, D E → h d, H A → g c and phrasebook 2: b h → γ β, f g → θ δ, d a → α π, c e → ζ μ.

The input in language 1 is "**A B C D E F G H**".

Answer: <T>...<T> The output in language 3 is "**γ β α π θ δ ζ μ**".

Next token prediction loss

Step 2: CEL with ICL-Capable Model

- We want the model to learn the task without relying on the "open-book" materials.
- Simple solution: **random dropout** on the helpful context tokens.

CEL

Prompt: Your are translating a sequence from language 1 to language 3.

Use phrasebook 1: [redacted] **F G → a f, D E → h d, H A → g c**
and phrasebook 2: b h → γ β, f g → θ δ, d a → α π, c e → ζ μ.

The input in language 1 is "**A B C D E F G H**".

Answer: <T>...<T> The output in language 3 is "**γ β α π θ δ ζ μ**".

Next token prediction loss

Step 2: CEL with ICL-Capable Model

- We want the model to learn the task without relying on the "open-book" materials.
- Simple solution: **random removal** on the helpful context tokens.

CEL

Prompt: Your are translating a sequence from language 1 to language 3.

Use phrasebook 1: B C → e b, F G → a f, [redacted] H A → g c
and phrasebook 2: b h → γ β, [redacted] d a → α π, c e → ζ μ.

The input in language 1 is "**A B C D E F G H**".

Answer: <T>...<T> The output in language 3 is "**γ β α π θ δ ζ μ**".

Next token prediction loss

Step 2: CEL with ICL-Capable Model

- We want the model to learn the task without relying on the "open-book" materials.
- Simple solution: **random dropout** on the helpful context tokens.

CEL

Prompt: Your are translating a sequence from language 1 to language 3.

Use phrasebook 1: B C → e b, [redacted] D E → h d, H A → g c
and phrasebook 2: b h → γ β, f g → θ δ, d a → α π, c e → ζ μ.

The input in language 1 is "**A B C D E F G H**".

Answer: <T>...<T> The output in language 3 is "**γ β α π θ δ ζ μ**".

Next token prediction loss

Step 2: CEL with ICL-Capable Model

- We want the model to learn the task without relying on the "open-book" materials.
- Simple solution: **random dropout** on the helpful context tokens.

CEL

Prompt: Your are translating a sequence from language 1 to language 3.

Use phrasebook 1: B C → e b, F G → a f, D E → h d, [redacted]
and phrasebook 2: b h → γ β, f g → θ δ, d a → α π, c e → ζ μ.

The input in language 1 is "**A B C D E F G H**".

Answer: <T>...<T> The output in language 3 is "**γ β α π θ δ ζ μ**".

Next token prediction loss

Evaluating the CEL Model

- Just like SFT! Evaluate without any additional context.

CEL

Prompt: Your are translating a sequence from language 1 to language 3.

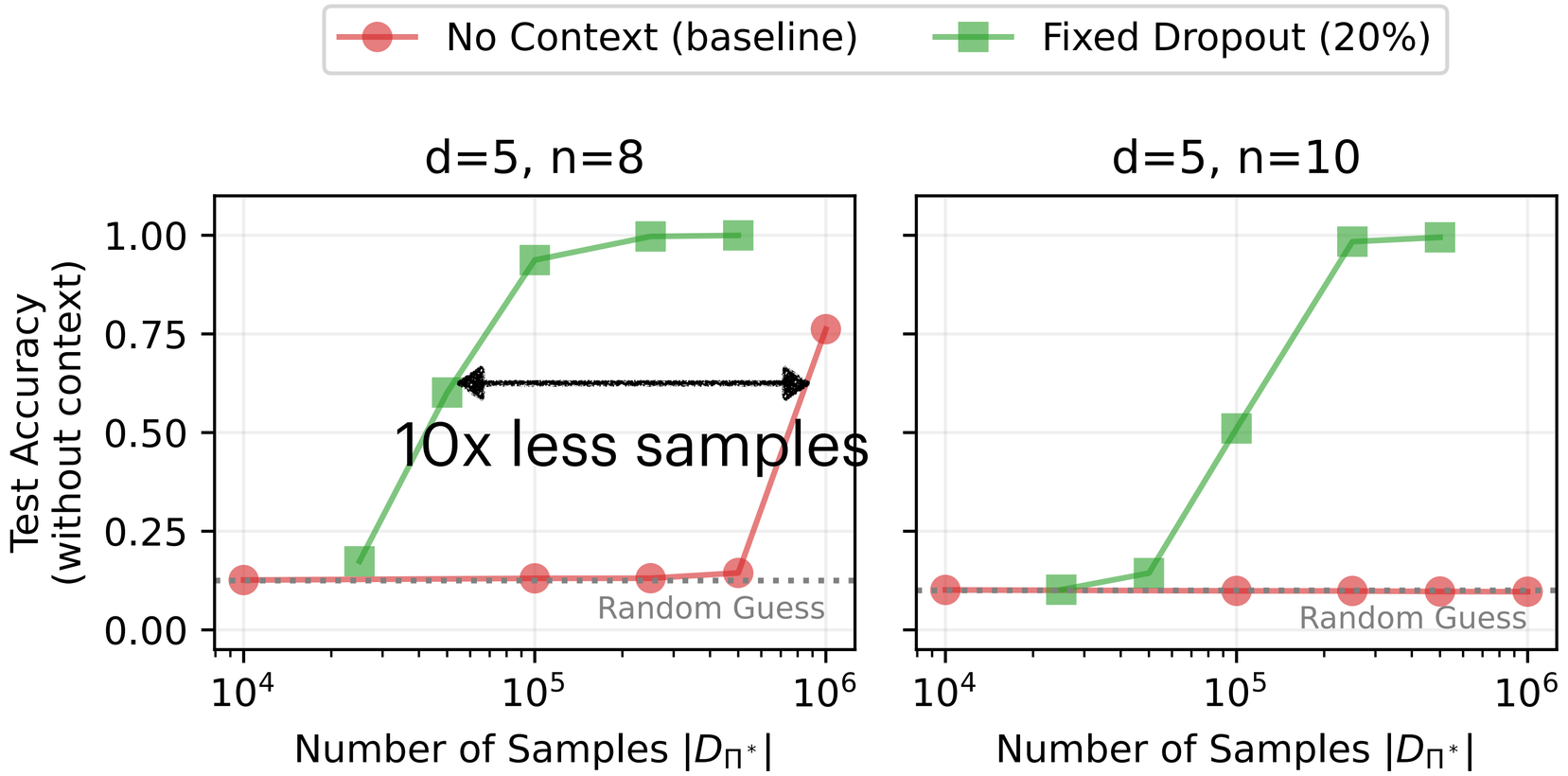


The input in language 1 is “**A B C D E F G H**”.

Answer: <T>...<T> The output in language 3 is “**γ β α π θ δ ζ μ**”.

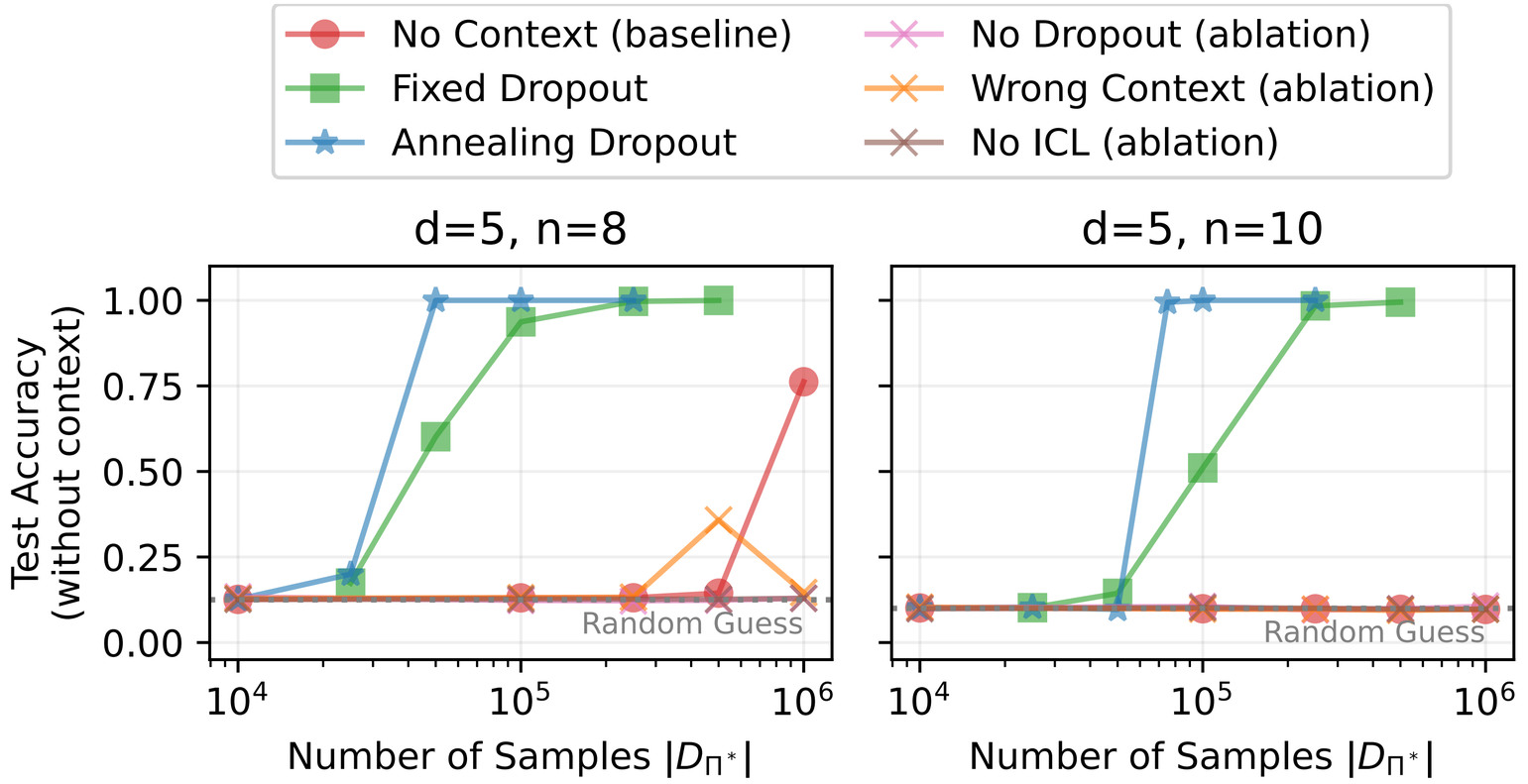
Sample Efficiency of CEL on Learning MLT

- Experiment on Llama 3.2 3B, d = 5, with 8 or 10 characters per vocab.



Sample Efficiency of CEL on Learning MLT

- Experiment on Llama 3.2 3B, d = 5, with 8 or 10 characters per vocab.



- Better dropout curriculum leads for even faster learning.
- Dropout is required during training.
- ICL capability is required before training.

MLT Experiment Takeaways

How powerful can CEL be (vs. vanilla SFT)?

Empirically: 10x more sample efficient with proper context removal

Does CEL require certain capability of the base model?

Yes. The model needs to start with corresponding ICL capability

Does contextual removal need to be 100% in training?

No. Removing 20% in training generalizes to removing 100% @ eval

Little Verbatim Memorization

CEL

Prompt: Your are translating a sequence from language 1 to language 3.

Use phrasebook 1: B C → e b, F G → a f, D E → h d, H A → g c and phrasebook 2: b h → γ β, f g → θ δ, d a → α π, c e → ζ μ.

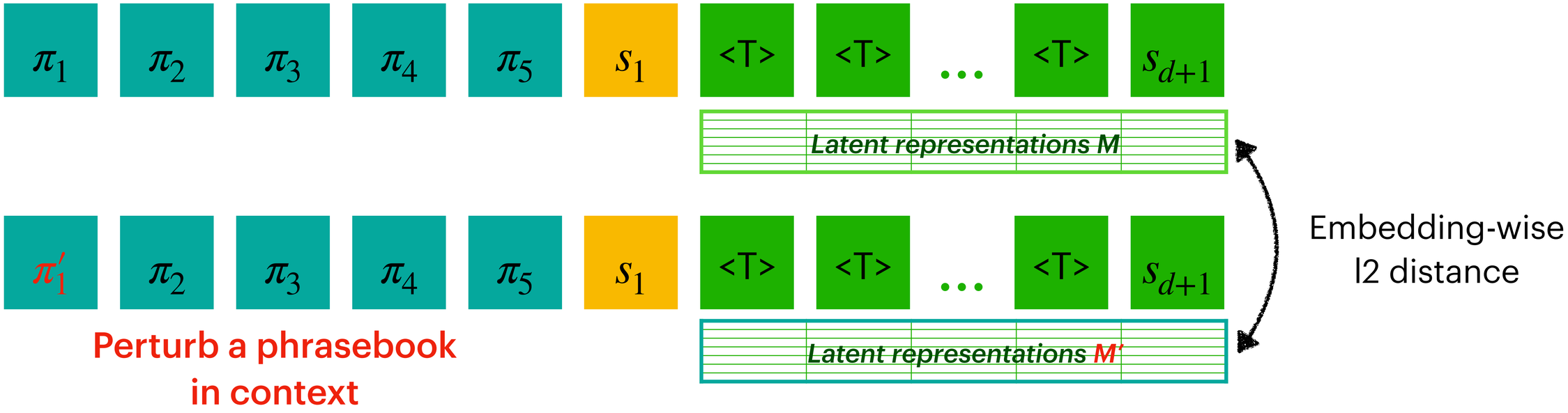
Table 2. Recovery Success Rate (Rounded to 2 decimals)

Token Filter	Greedy Decoding		Sampling	
	$\pi_1^* - \pi_4^*$	π_5^*	$\pi_1^* - \pi_4^*$	π_5^*
None	0.00%	0.20%	0.00%	0.89%
No <THINK>	0.00%	0.20%	0.00%	0.90%
Only from A_{i+1}	1.66%	0.20%	1.28%	0.94%

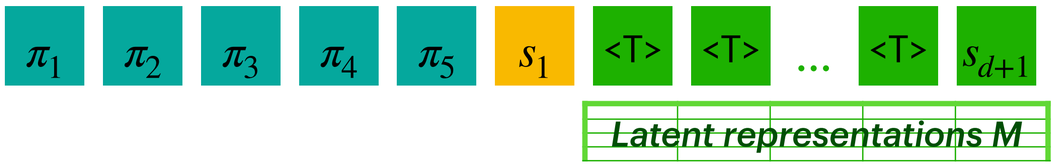
Understanding CEL Mechanistically

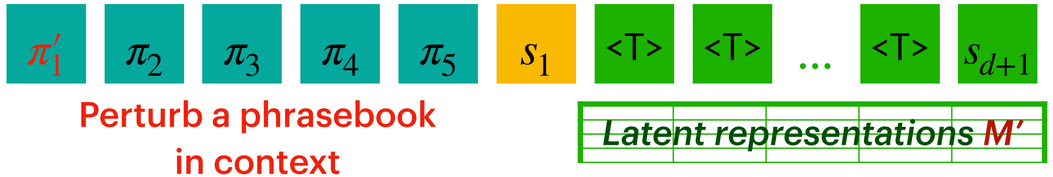
Mechanism of ICL-capable Model

- How does the ICL-capable transformer (after step 1) perform the MLT translation task with in-context phrasebooks?
- We inspect the difference in latent representations of **output tokens** when perturbing different phrasebooks in context.

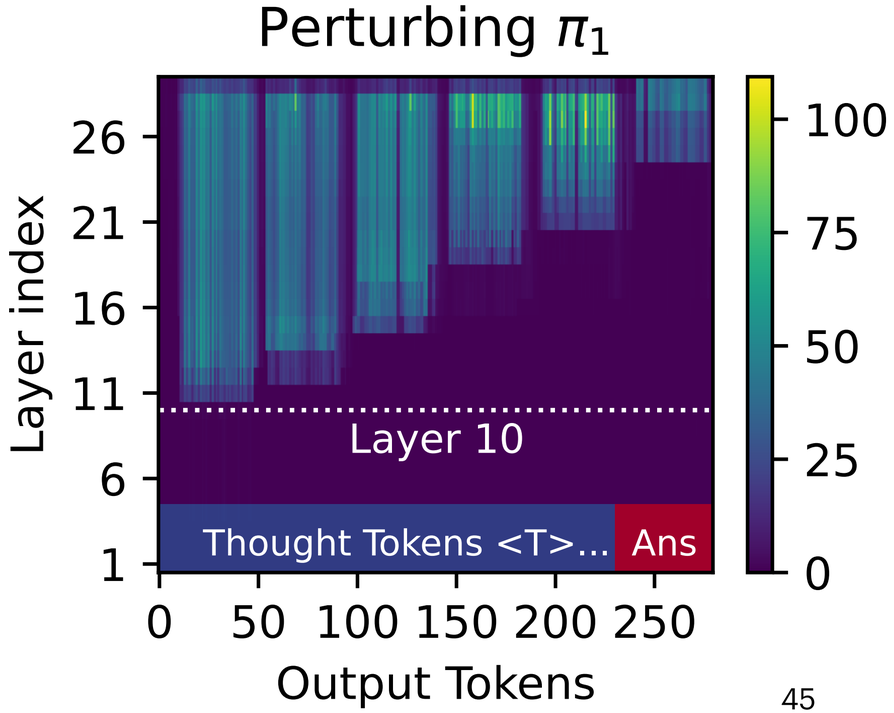


Mechanism of ICL-capable Model



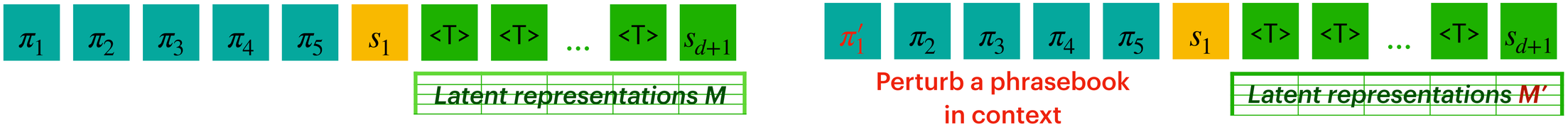


Embedding-wise l2 distance between representations ***M*** and ***M'***

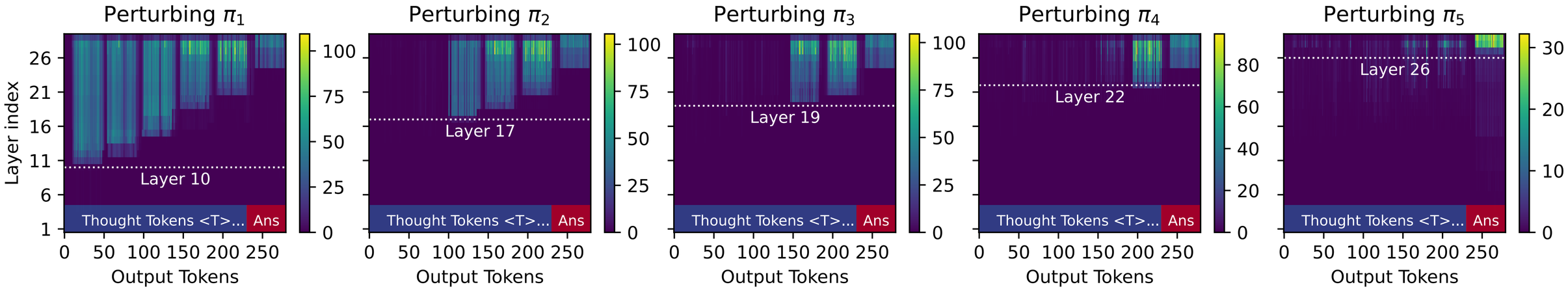


Phrasebook π_1 starts to be involved in the translation calculation starting @ layer 10

Mechanism of ICL-capable Model



Embedding-wise l2 distance between representations ***M*** and ***M'***

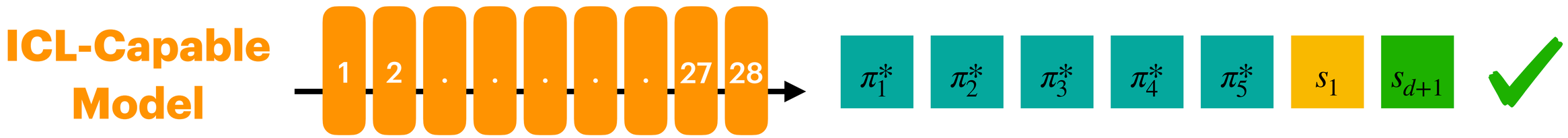


Phrasebook π_1 in context has been used in translation computation starting at Layer 10

Phrasebook π_2: Layer 17 Phrasebook π_3: Layer 19 Phrasebook π_4: Layer 22 Phrasebook π_5: Layer 26

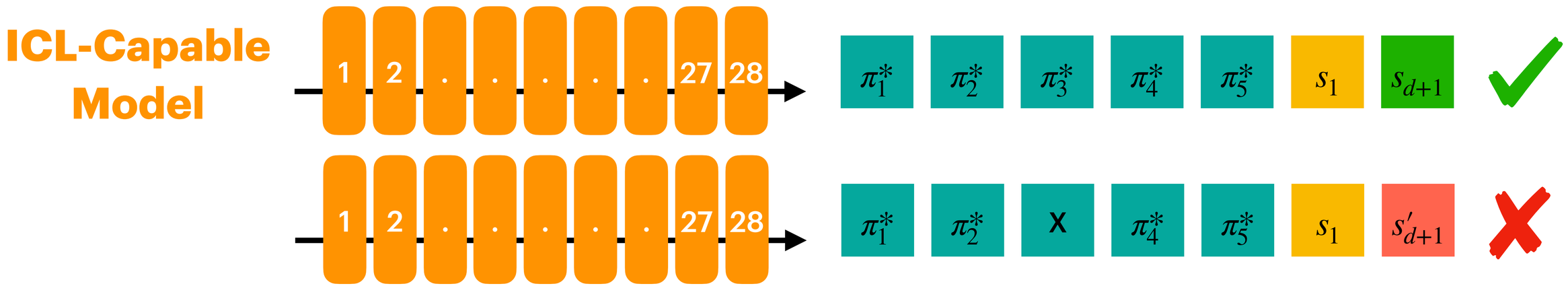
Mechanism of CEL-trained Model

- After CEL training, the model internalizes $(\pi_1^*, \pi_2^*, \pi_3^*, \pi_4^*, \pi_5^*)$
- Where does the internalization happen?



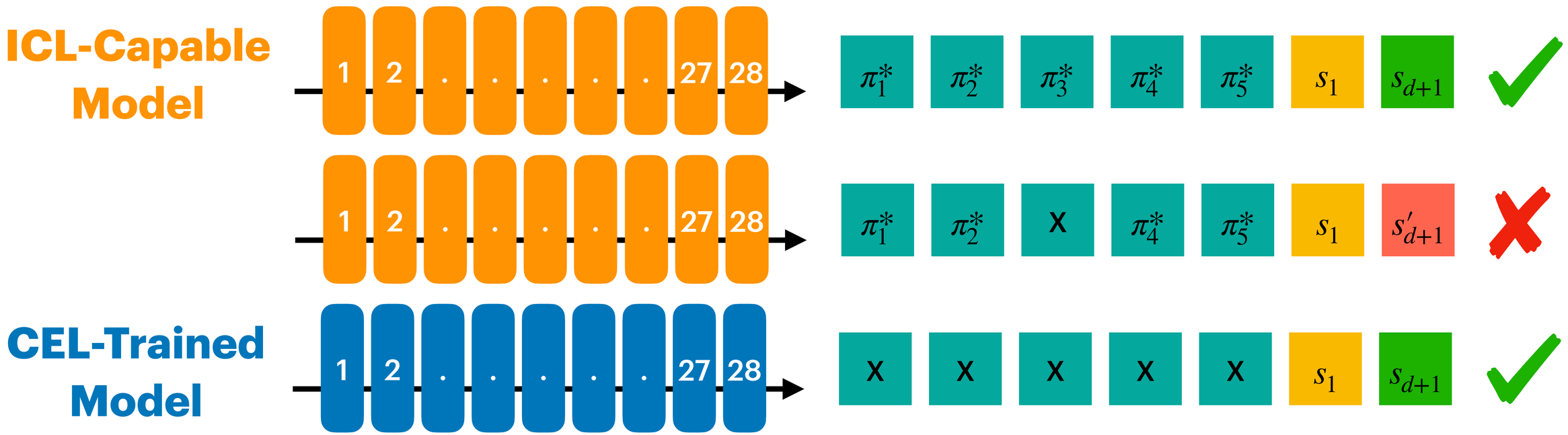
Mechanism of CEL-trained Model

- After CEL training, the model internalizes $(\pi_1^*, \pi_2^*, \pi_3^*, \pi_4^*, \pi_5^*)$
- Where does the internalization happen?



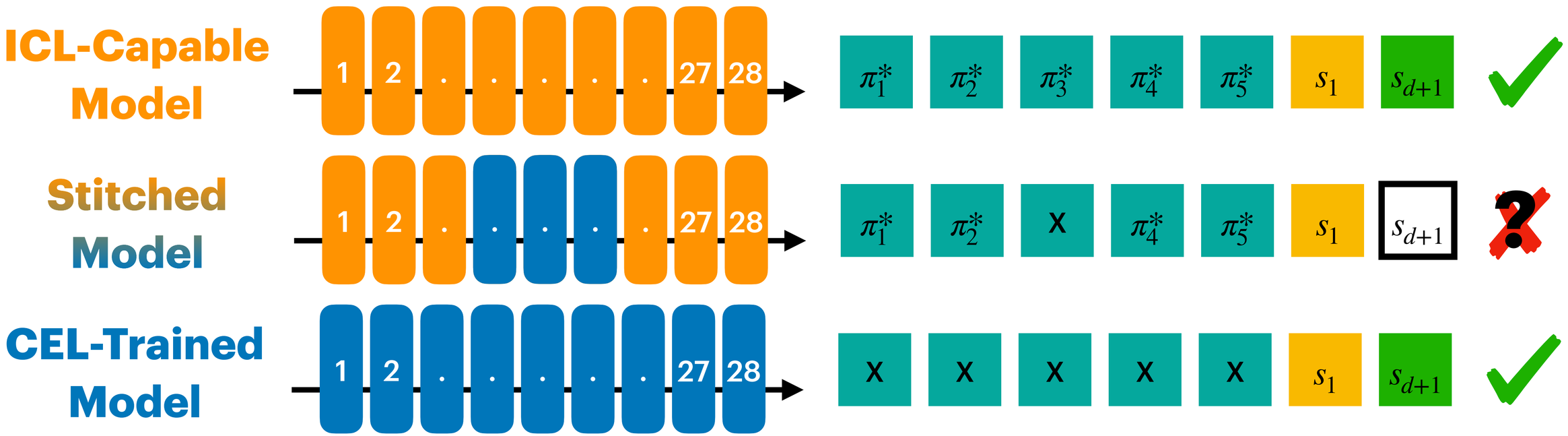
Mechanism of CEL-trained Model

- After CEL training, the model internalizes $(\pi_1^*, \pi_2^*, \pi_3^*, \pi_4^*, \pi_5^*)$
- Where does the internalization happen?



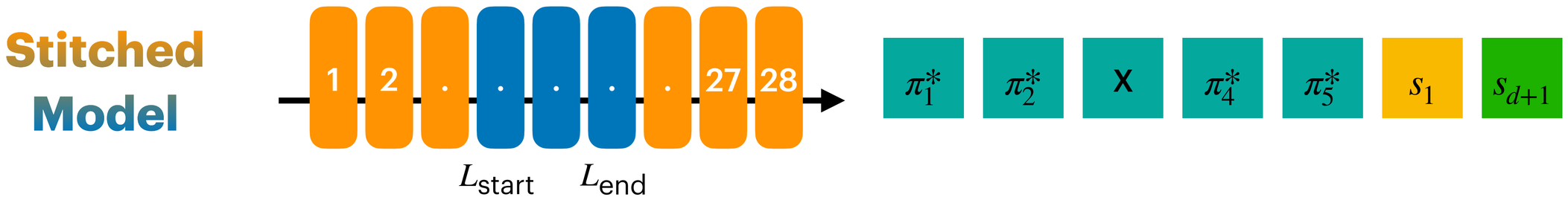
Mechanism of CEL-trained Model

- After CEL training, the model internalizes $(\pi_1^*, \pi_2^*, \pi_3^*, \pi_4^*, \pi_5^*)$
- Where does the internalization happen?

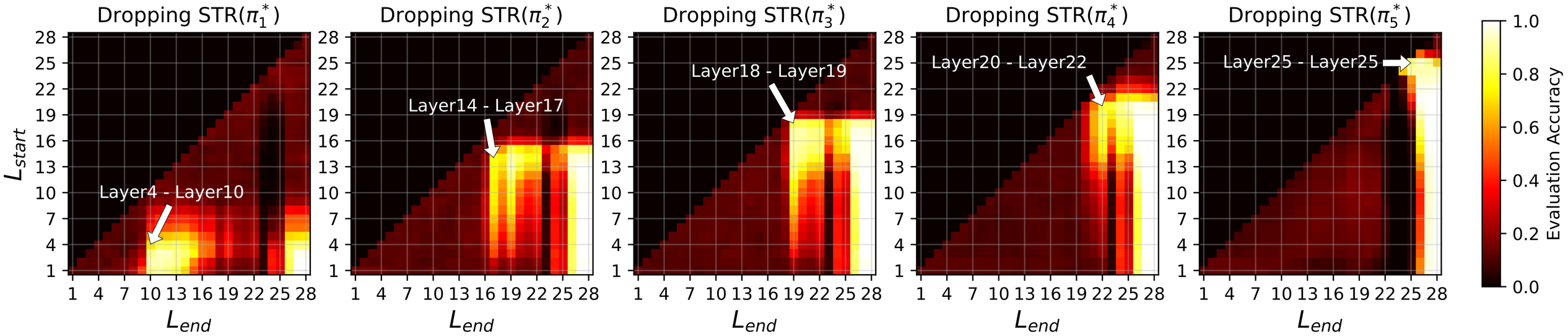


If a small set of layers from the CEL model compensates for the missing phrasebook in context, we argue the missing phrasebook is stored in these layers.

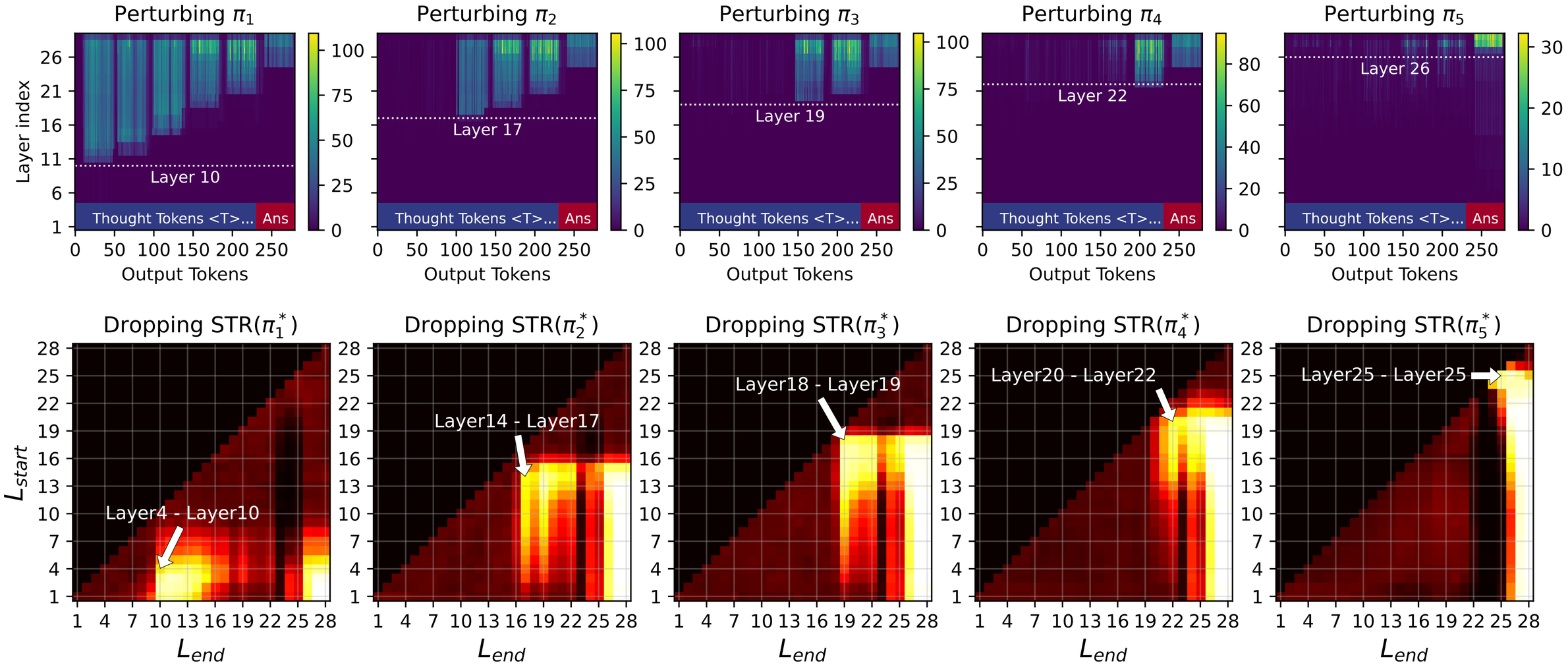
Mechanism of CEL-trained Model



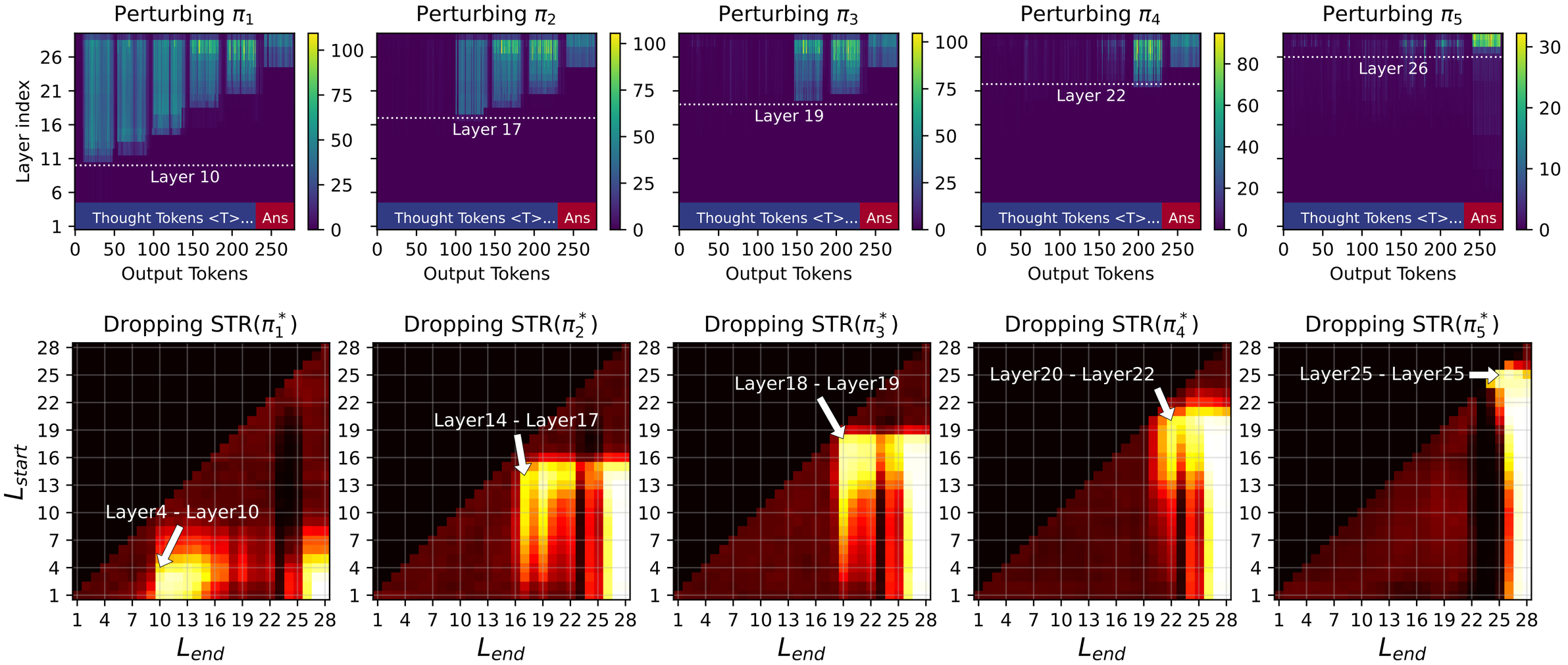
Evaluation Accuracy of Stitched Models when Dropping Different Phrasebooks



Distributed Storage of Phrasebook Information



Distributed Storage of Phrasebook Information



Understanding CEL Theoretically

Sample Complexity Separation for Toy Model

Lower Bound: SFT

There exists an d-step MLT instance such that any gradient-based training algorithm with uniformly random input-output pairs requires at least $e^{\Omega(d)}$ gradient updates.

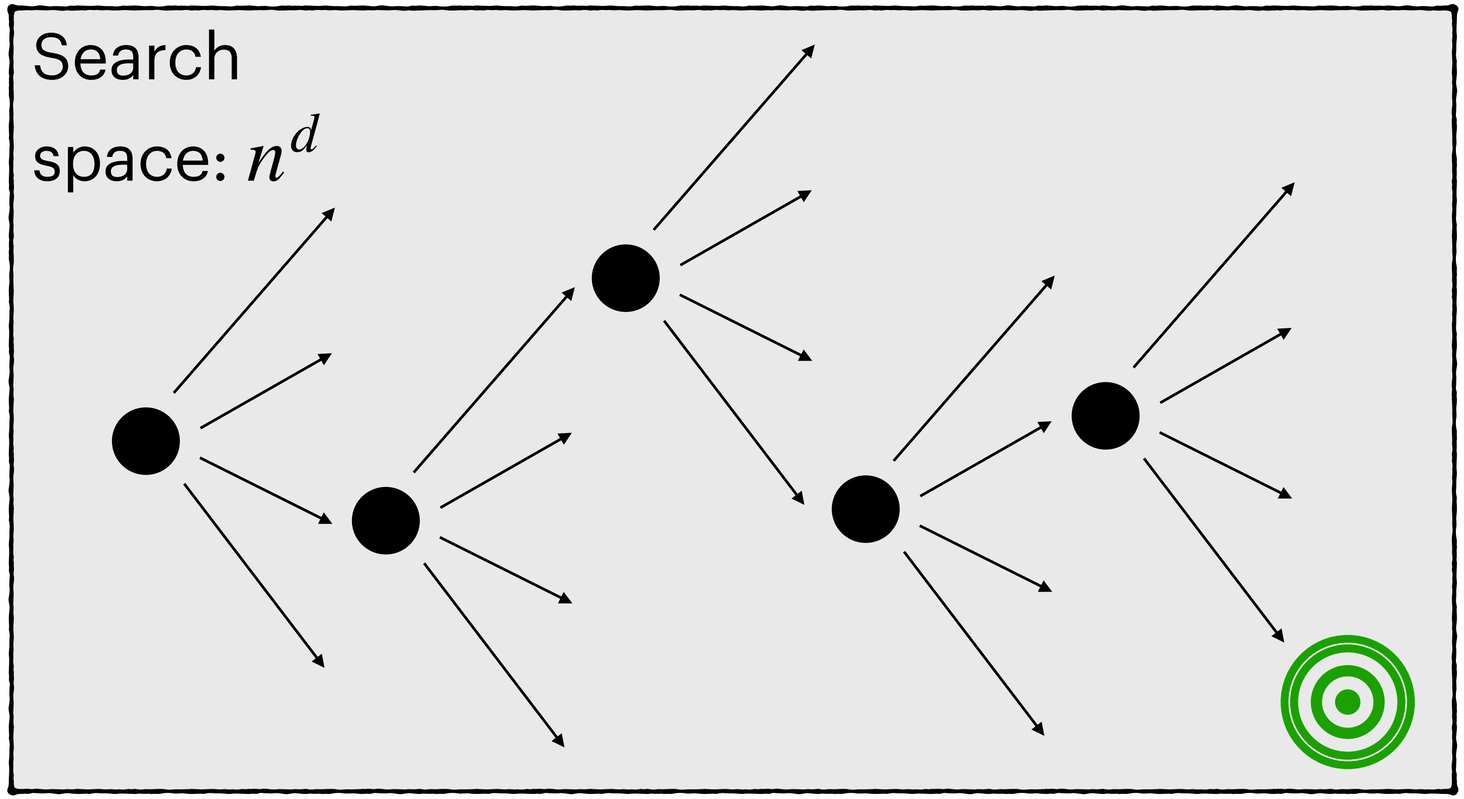
Upper Bound: CEL

There exists an layer-wise search algorithm*, accompanied with a dropout curriculum on the context information, that can perfectly learn any d-step MLT task with $O(d \log d)$ samples.

* Layer-wise GD results provable for $d = 2$, full parameter GD results empirically verified.

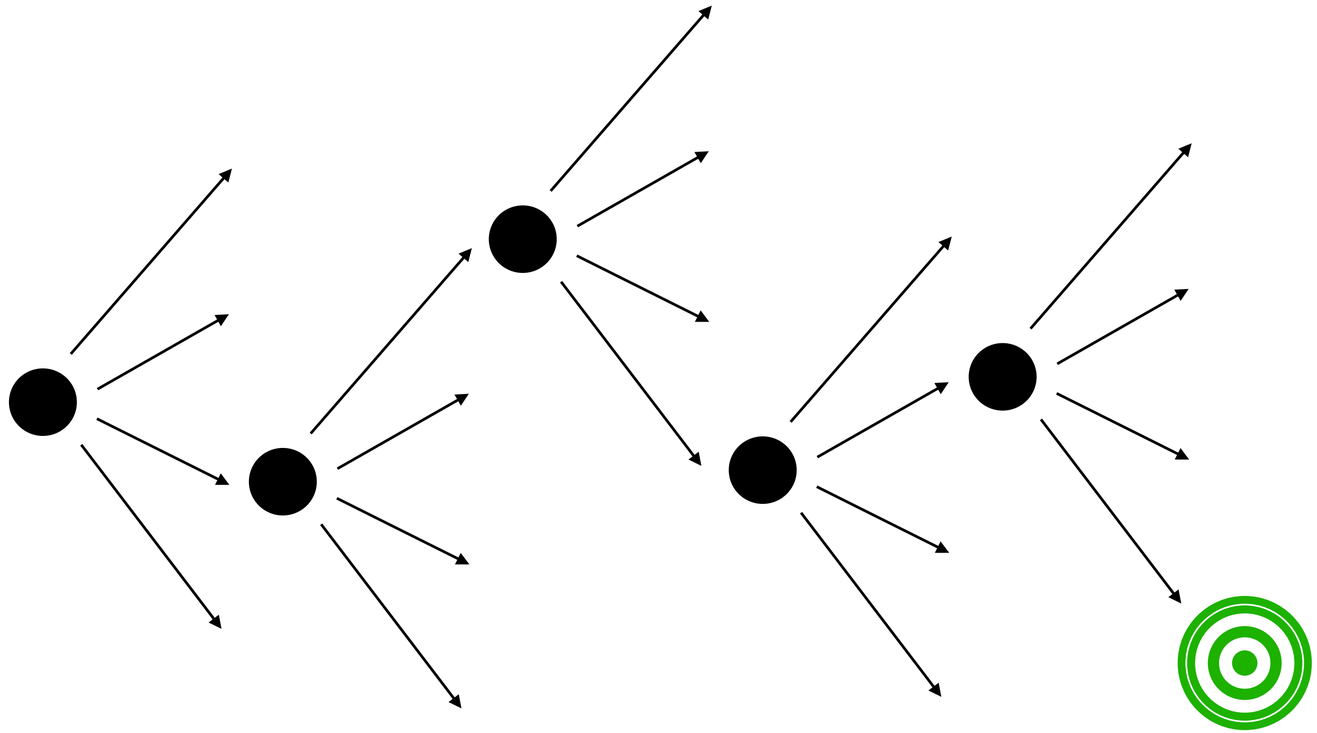
Intuition: Search Space Factorization

- For a d-step task with size-n action space for each step, the search space could be n^d large. Learning signal could be very noisy (requiring n^d samples)



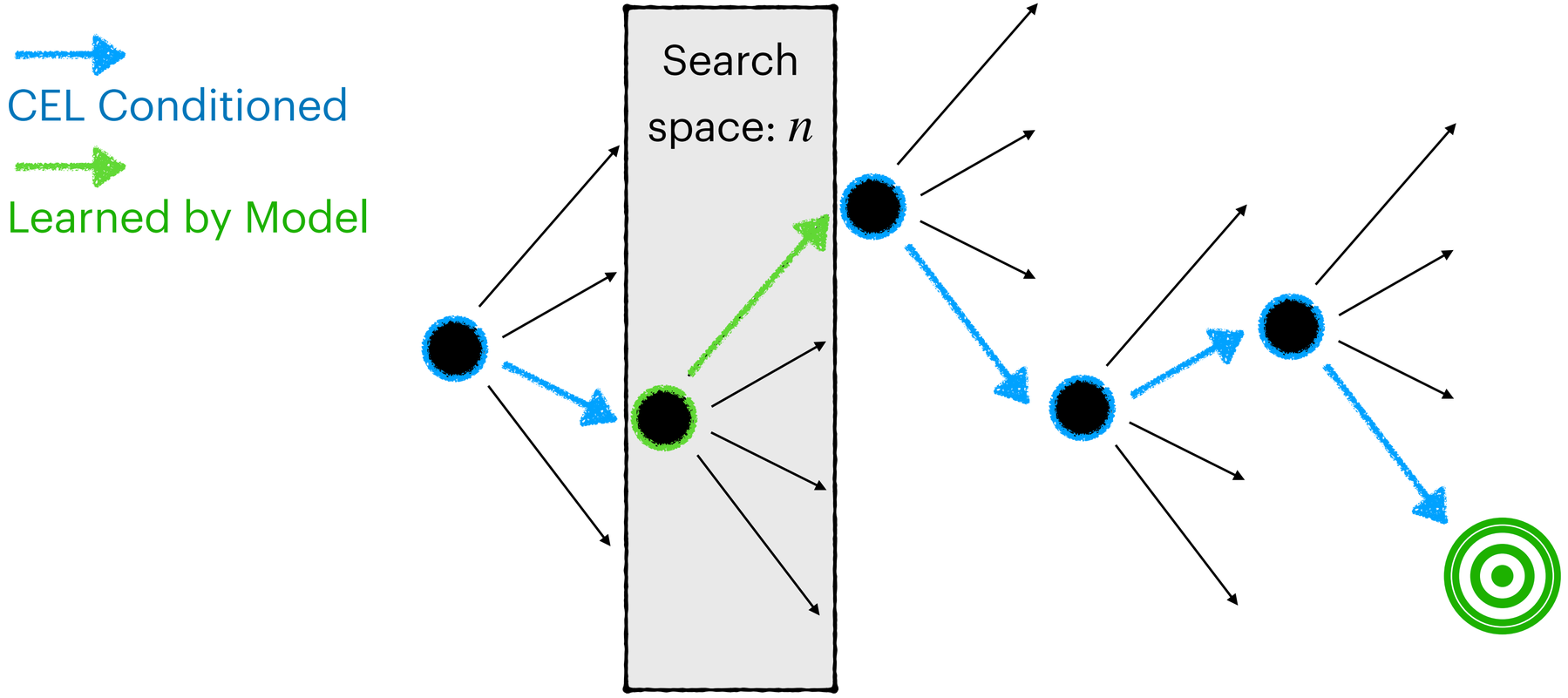
Intuition: Search Space Factorization

- When model has certain ICL capability: In-context information + dropout can condition on most steps and limit the search space to the dropped out steps: allowing the model to “focus” on learning the dropout steps.



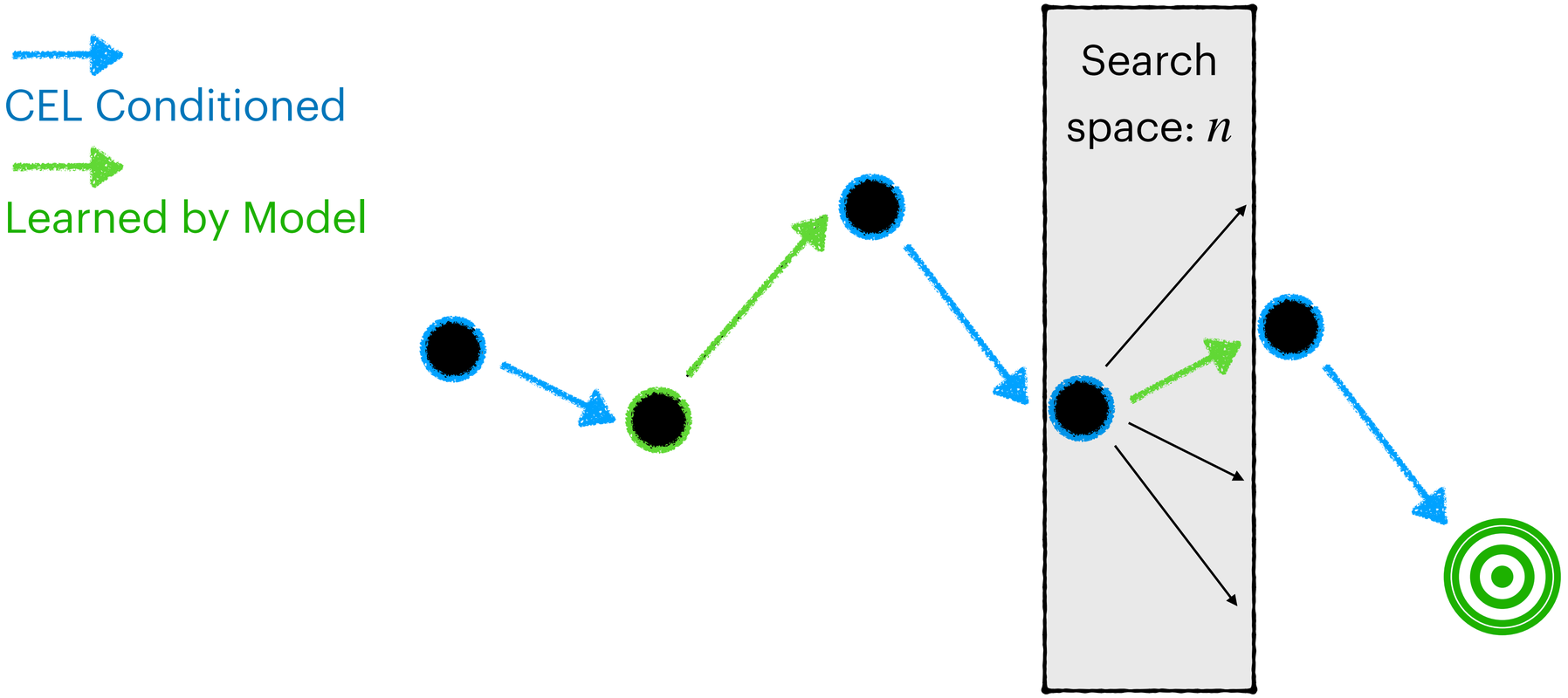
Intuition: Search Space Factorization

- When model has certain ICL capability: In-context information + dropout can condition on most steps and limit the search space to the dropped out steps: allowing the model to “focus” on learning the dropout steps.



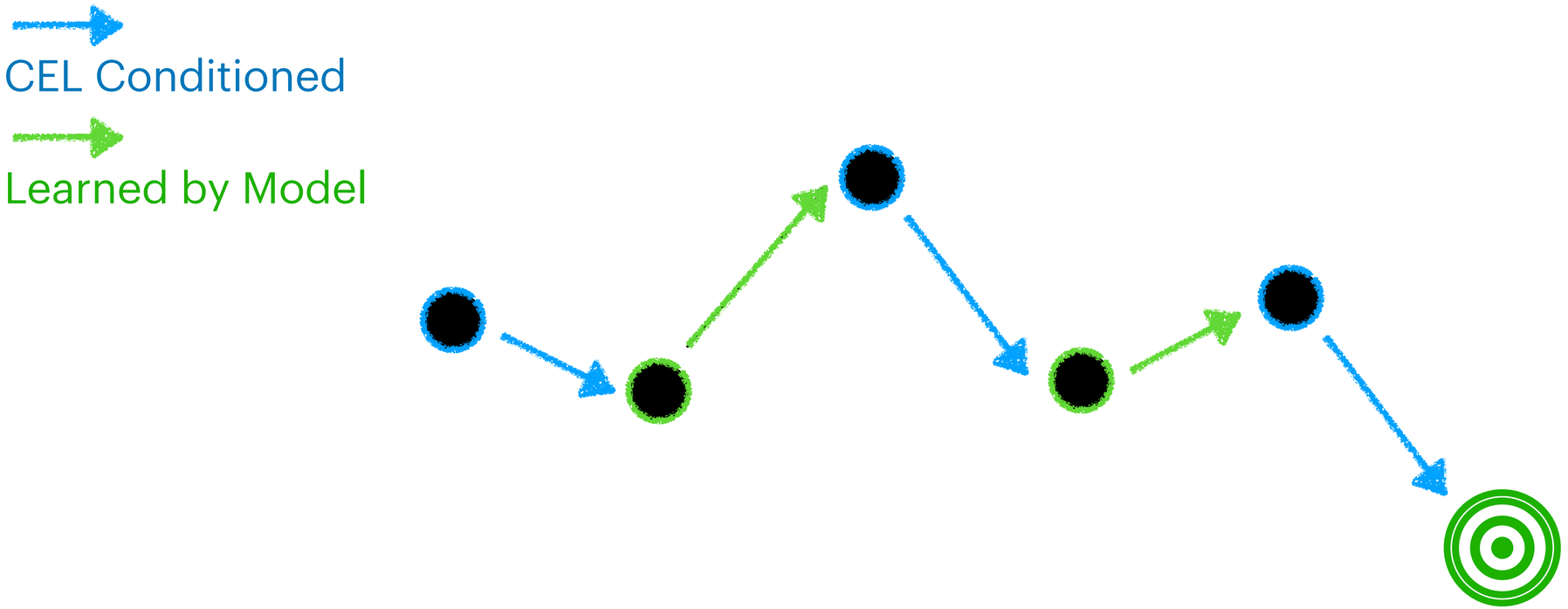
Intuition: Search Space Factorization

- By applying different dropouts, we form a natural curriculum requiring d meta-steps of sample complexity polylog(n)



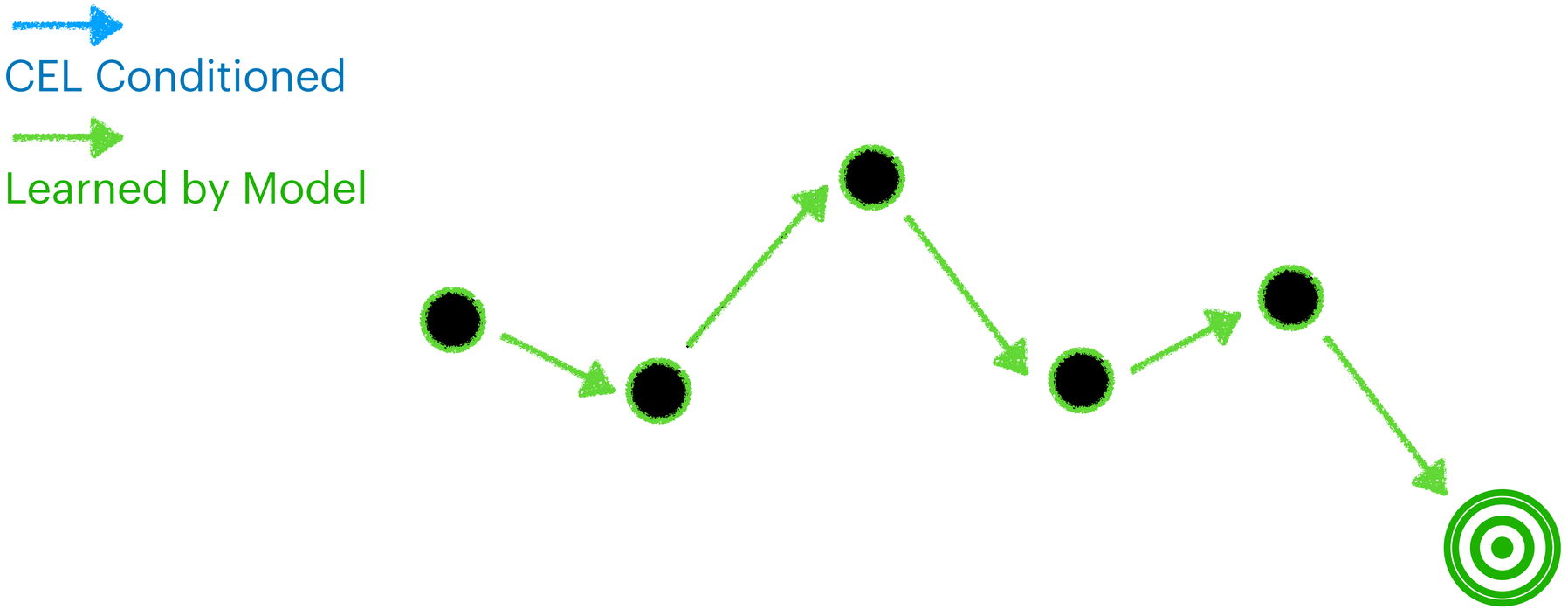
Intuition: Search Space Factorization

- By applying different dropouts, we form a natural curriculum requiring d meta-steps of sample complexity polylog(n)



Intuition: Search Space Factorization

- By applying different dropouts, we form a natural curriculum requiring d meta-steps of sample complexity polylog(n)



Future Directions

- Empirically:
  - Test on real world data: what could be good in-context curriculums?
- Theoretically:
  - Formalize the optimization landscape benefit provided by the in-context information.

Thanks!

- We thank Yun Cheng, Simon Park, Tianyu Gao, Yihe Dong, Zixuan Wang, Haoyu Zhao, and Bingbin Liu for discussions, suggestions, and proof-reading at various stages of the work.