Limits of Transformer Language Models on Learning to Compose Algorithms

Jonathan Thomm[1,2*]
jthomm@ethz.ch

Giacomo Camposampiero[1,2]
giacomo.camposampiero1@ibm.com

Aleksandar Terzic[1,2]
aleksandar.terzic1@ibm.com

Michael Hersche[1]
michael.hersche@ibm.com

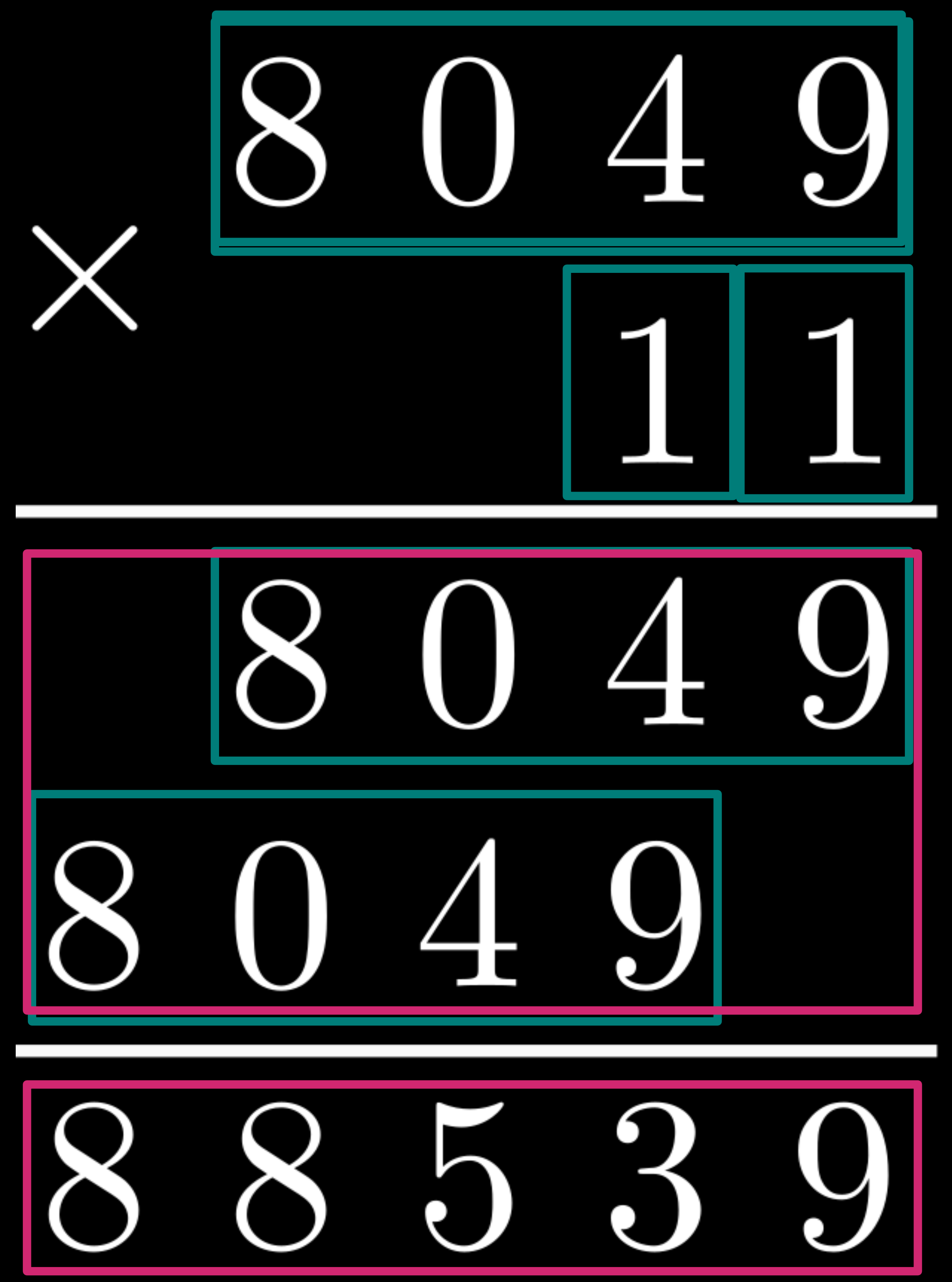
Bernhard Schölkopf[2,3]
bs@tuebingen.mpg.de

Abbas Rahimi[1]
abr@zurich.ibm.com

[1] IBM Research [2] ETH zürich [3] MAX PLANCK INSTITUTE FOR INTELLIGENT SYSTEMS

NEURAL INFORMATION PROCESSING SYSTEMS

We investigate how well transformer language models can learn algorithmic compositional tasks

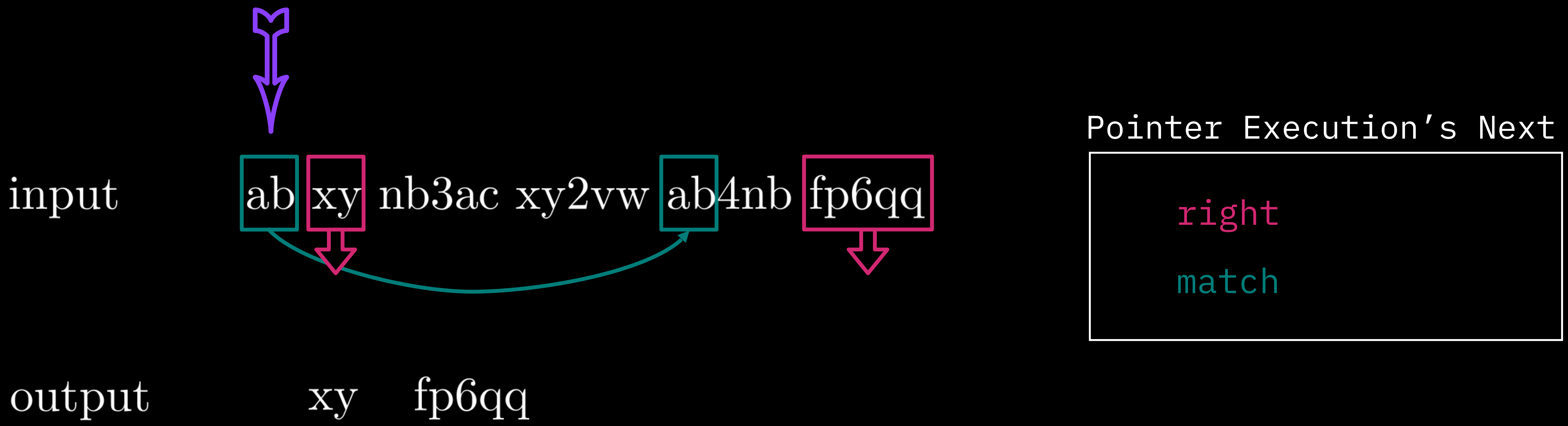


Step-by-step multiplication

digit multiplication

addition

We design new tasks based on pointer execution[1, 2] to benchmark compositional learning



[1] Abnar et al. Adaptivity and Modularity for Efficient Generalization Over Task Complexity. ArXiv, 2023
[2] Zhang et al. Pointer value retrieval: A new benchmark for understanding the limits of neural network generalization. ArXiv, 2021.

We outline four possible hypotheses to characterise more formally the sample efficiency of the learning process

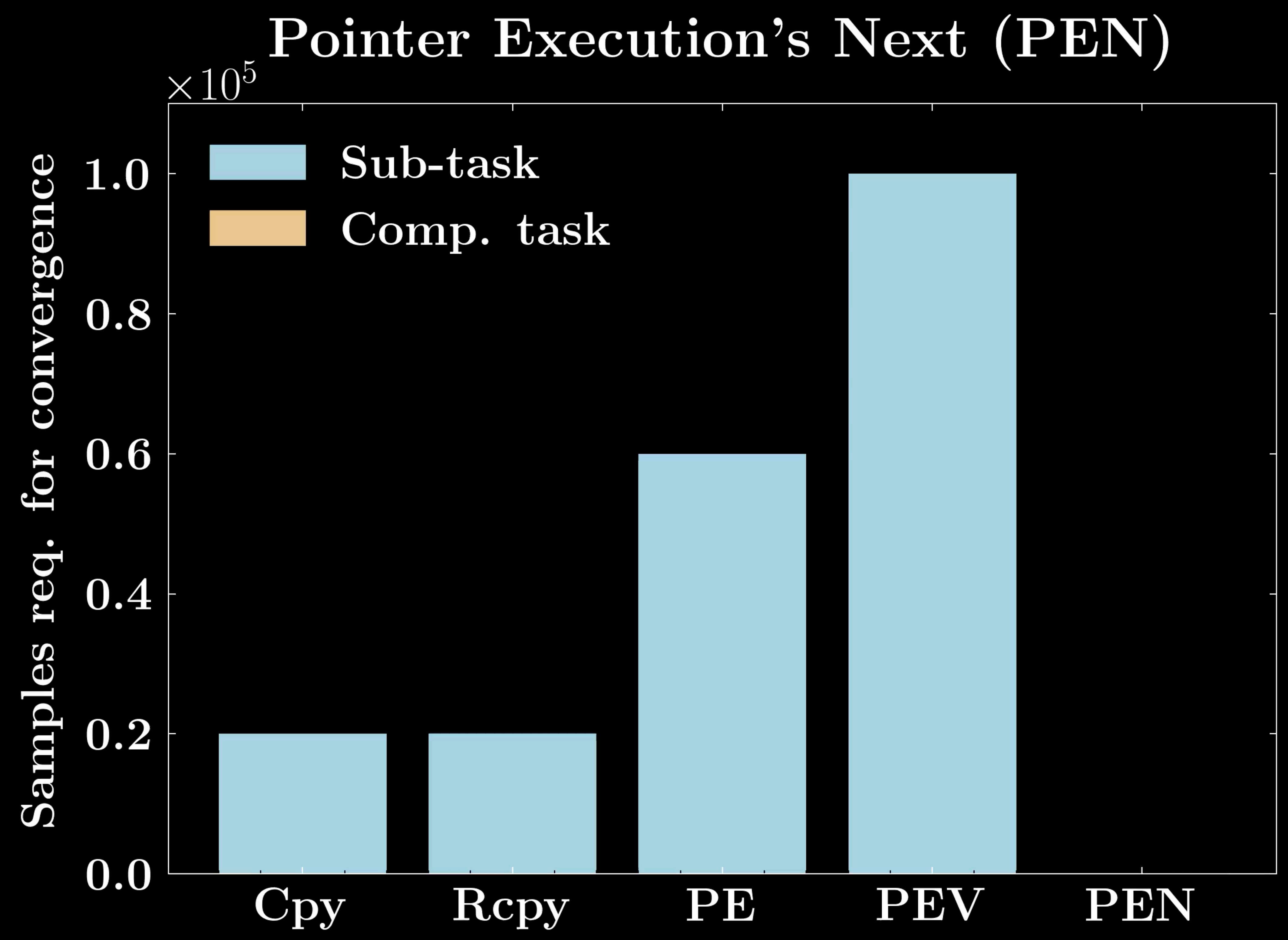
$\mathcal{H}_1$ constant number of samples

$\mathcal{H}_2$ fewer samples than those required to learn the most difficult sub-task

$\mathcal{H}_3$ fewer samples than the sum of samples needed to learn every sub-task

$\mathcal{H}_4$ more samples than $\mathcal{H}_3$

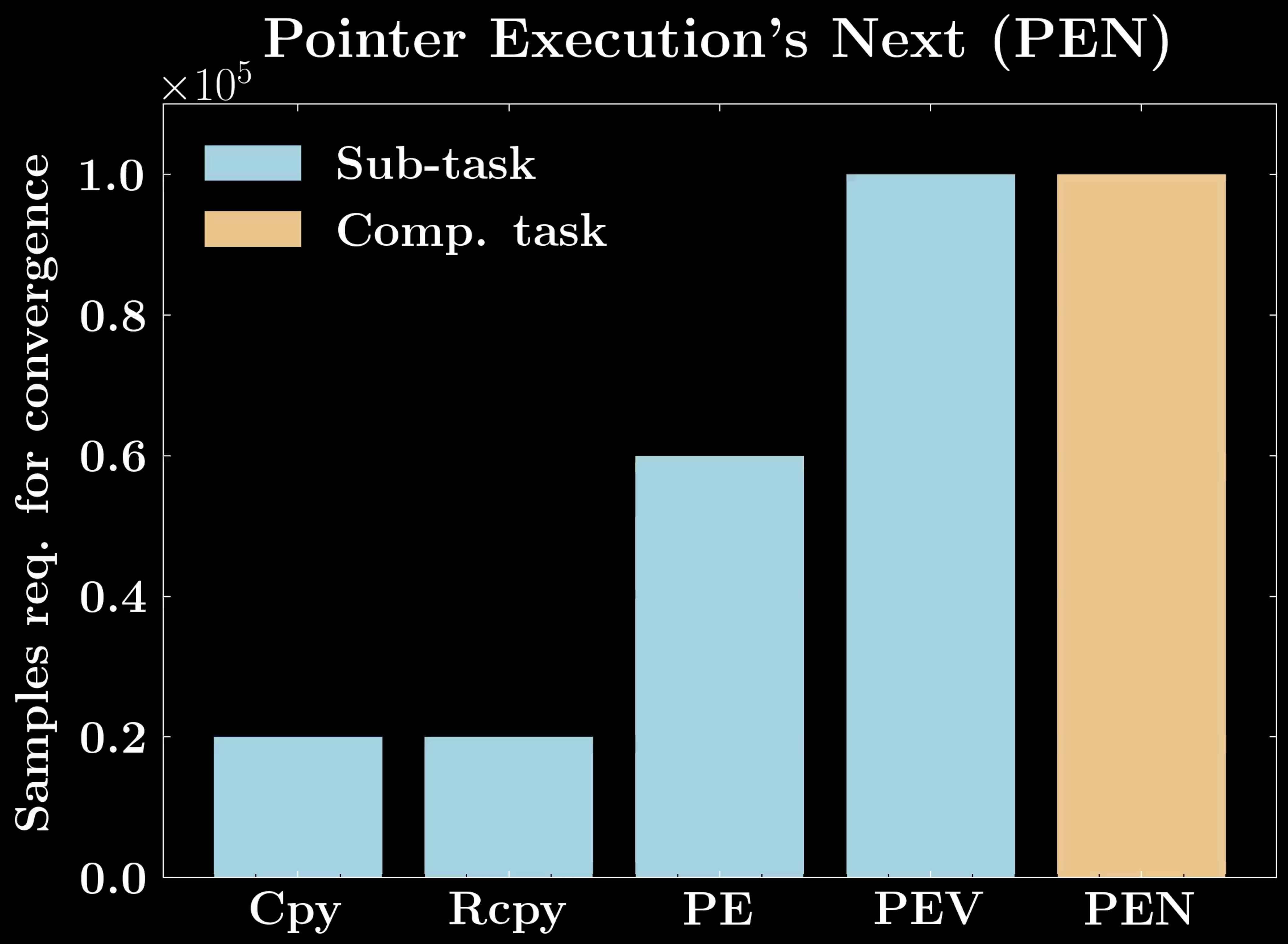
Transformer language models require an exponential number of samples to learn the composition of primitives ($\mathcal{H}_4$)



$\mathcal{H}_1$ constant ✗

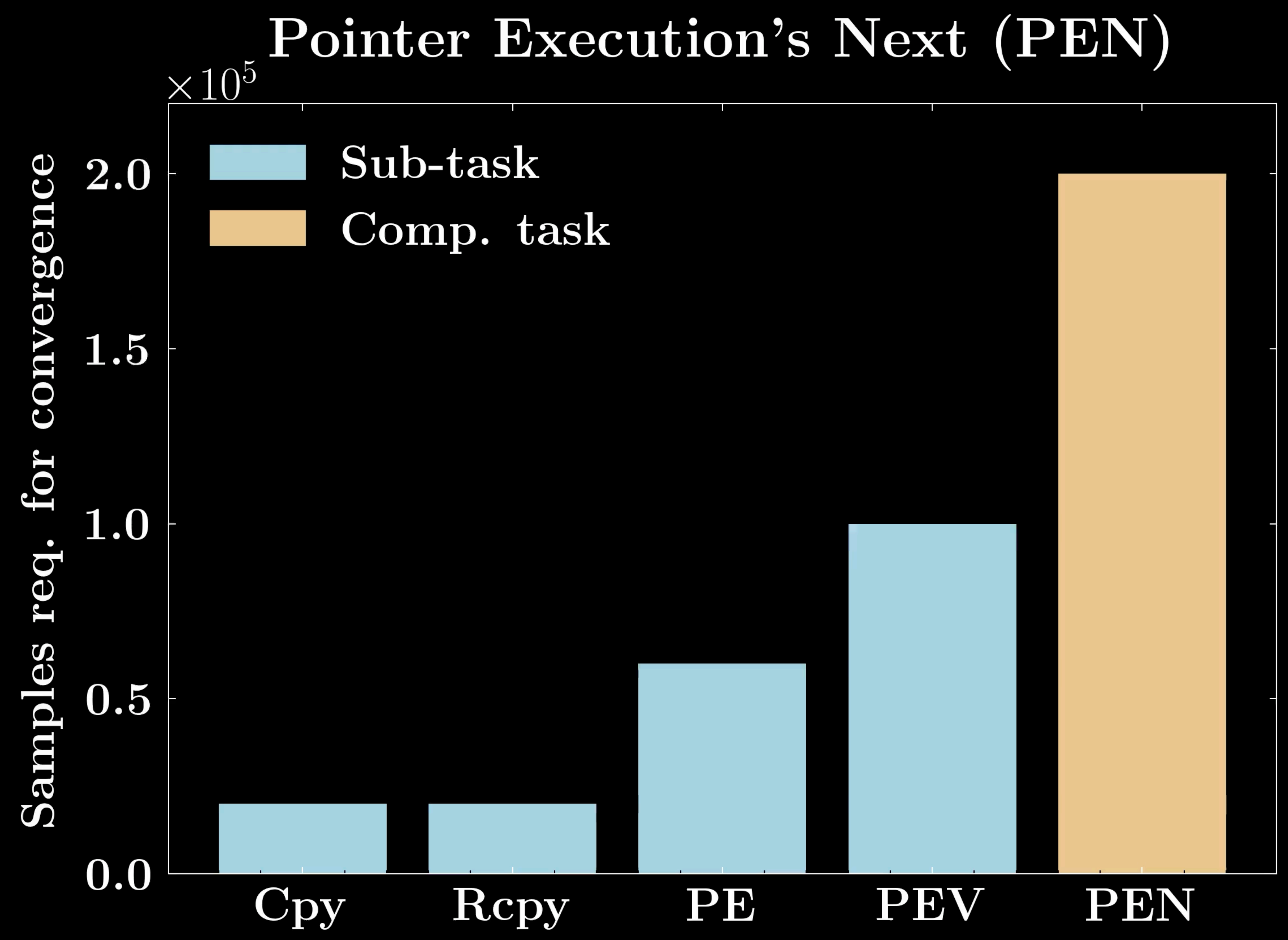
$\mathcal{H}_2$ < most difficult subtask ✗

Transformer language models require an exponential number of samples to learn the composition of primitives ($\mathcal{H}_4$)



$\mathcal{H}_1$	constant	✗
$\mathcal{H}_2$	< most difficult subtask	✗
$\mathcal{H}_3$	< subtasks sum	✗

Transformer language models require an exponential number of samples to learn the composition of primitives ($\mathcal{H}_4$)



$\mathcal{H}_1$ constant ✗

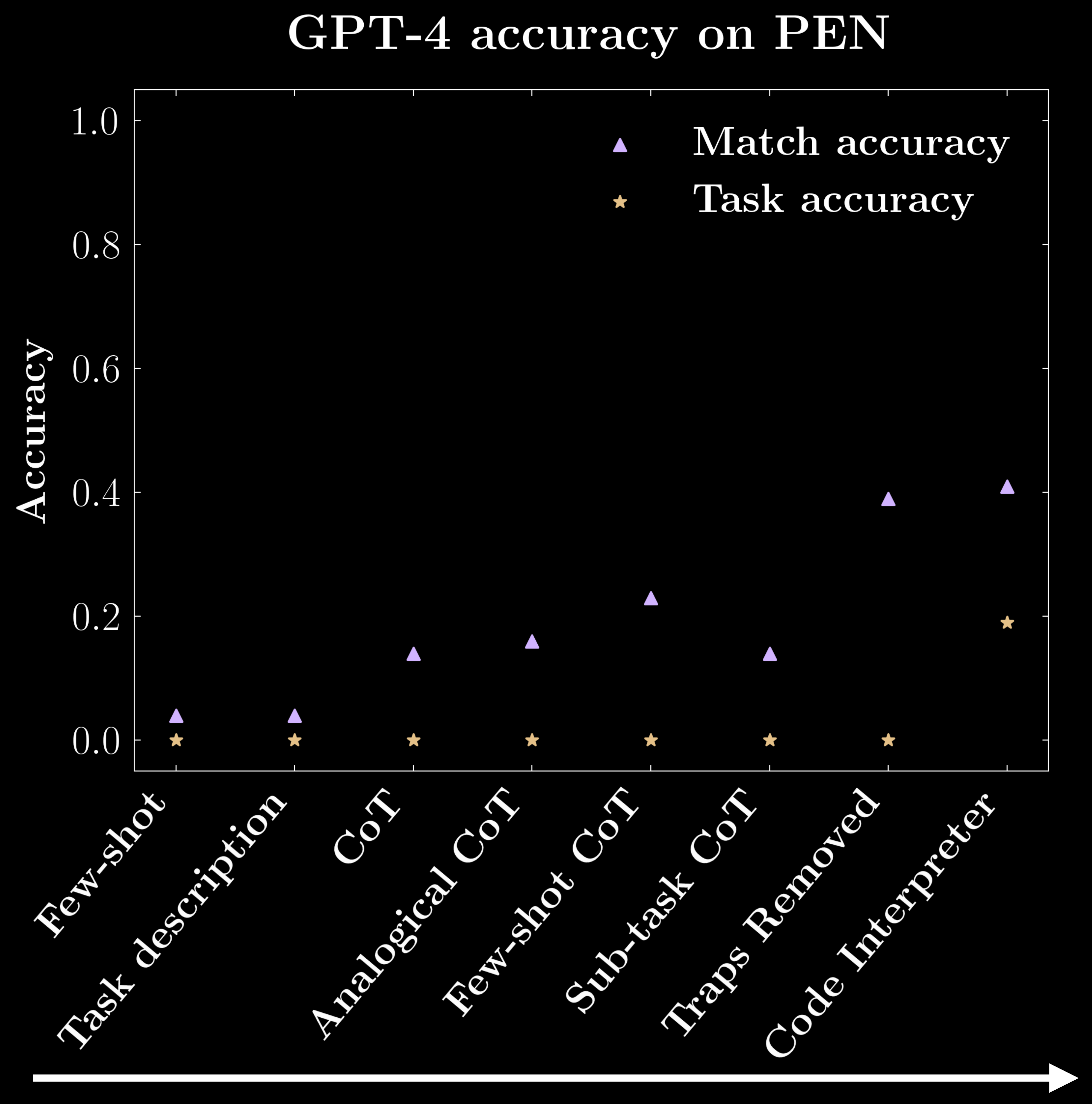
$\mathcal{H}_2$ < most difficult subtask ✗

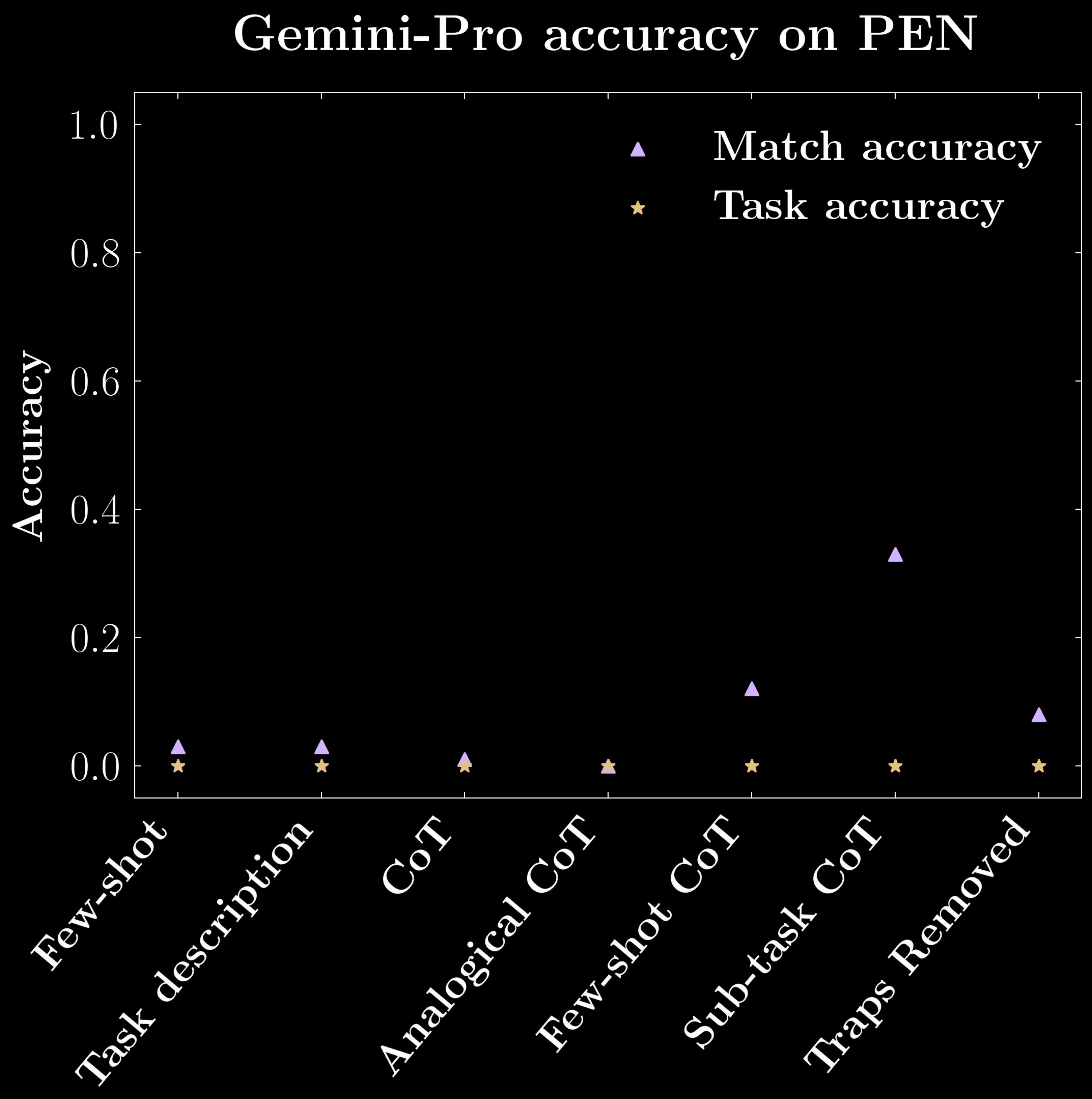
$\mathcal{H}_3$ < subtasks sum ✗

$\mathcal{H}_4$ > H_3 ✓

We observe the same trend across a wide range of compositional algorithmic datasets

Pre-trained LLMs struggle on these tasks as well





increasing prompt engineering complexity

Message: Transformer LMs are inefficient learners of compositions of tasks, requiring more training samples than the sum of those required to learn each task individually.

Paper https://arxiv.org/abs/2402.05785

Code https://github.com/IBM/limitations-lm-algorithmic-compositional-learning

Paper

Code