TEXAS
The University of Texas at Austin

# HELIOS: Adaptive Model and Early-Exit Selection for Efficient LLM Serving

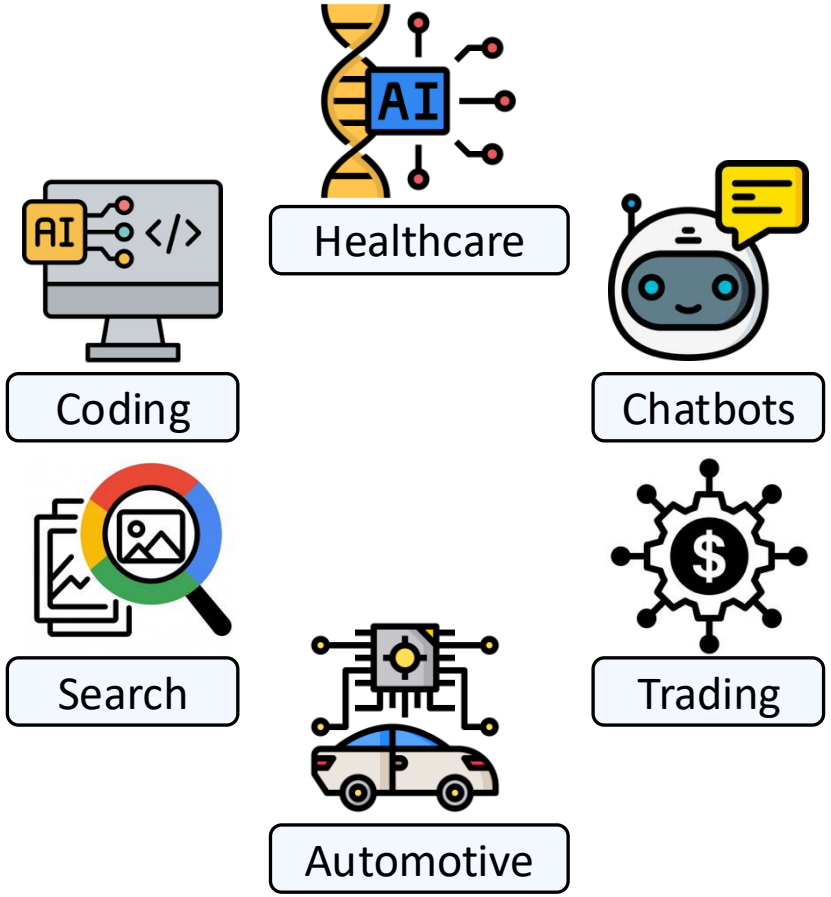
**Avinash Kumar***
Shashank Nag*
Jason Clemons#
Lizy John*
Poulami Das*

* The University of Texas at Austin
Chandra Department of Electrical and Computer Engineering
*Cockrell School of Engineering*

# NVIDIA®

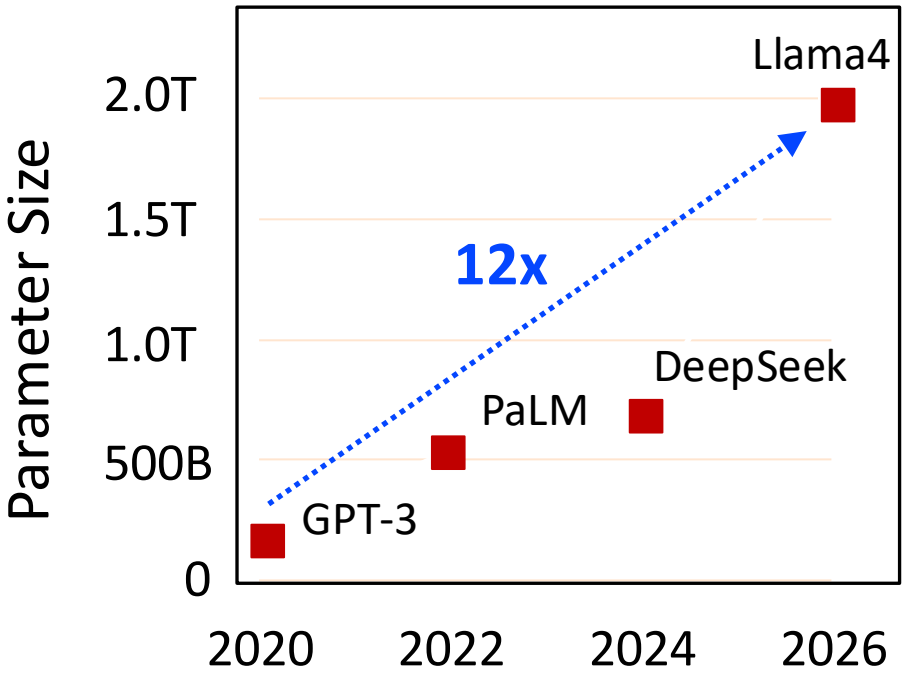
**MLSYS 2026**

# LLMs Are Becoming Increasingly Ubiquitous




**LLMs power a wide range of tasks**


**Increasing adoption across industries**



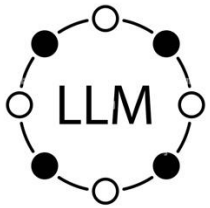

**Generally getting larger and better over recent years**

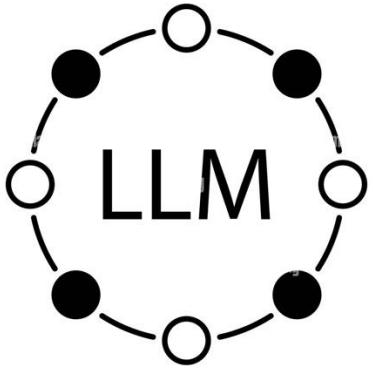


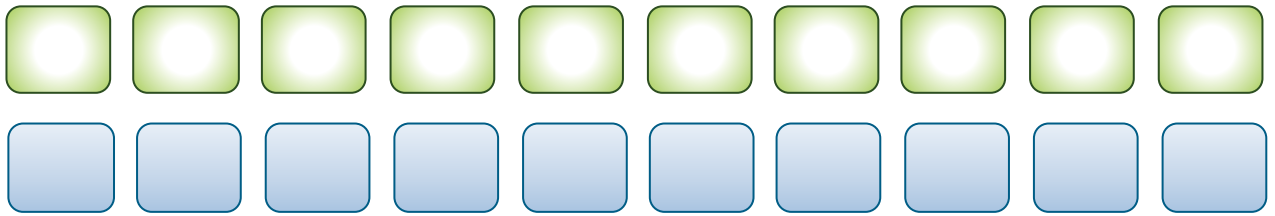
The widespread adoption and increasing complexity of LLMs pose critical deployment challenges

# Growing Concerns Regarding Throughput

Trade-off between accuracy and throughput



**Faster token processing**

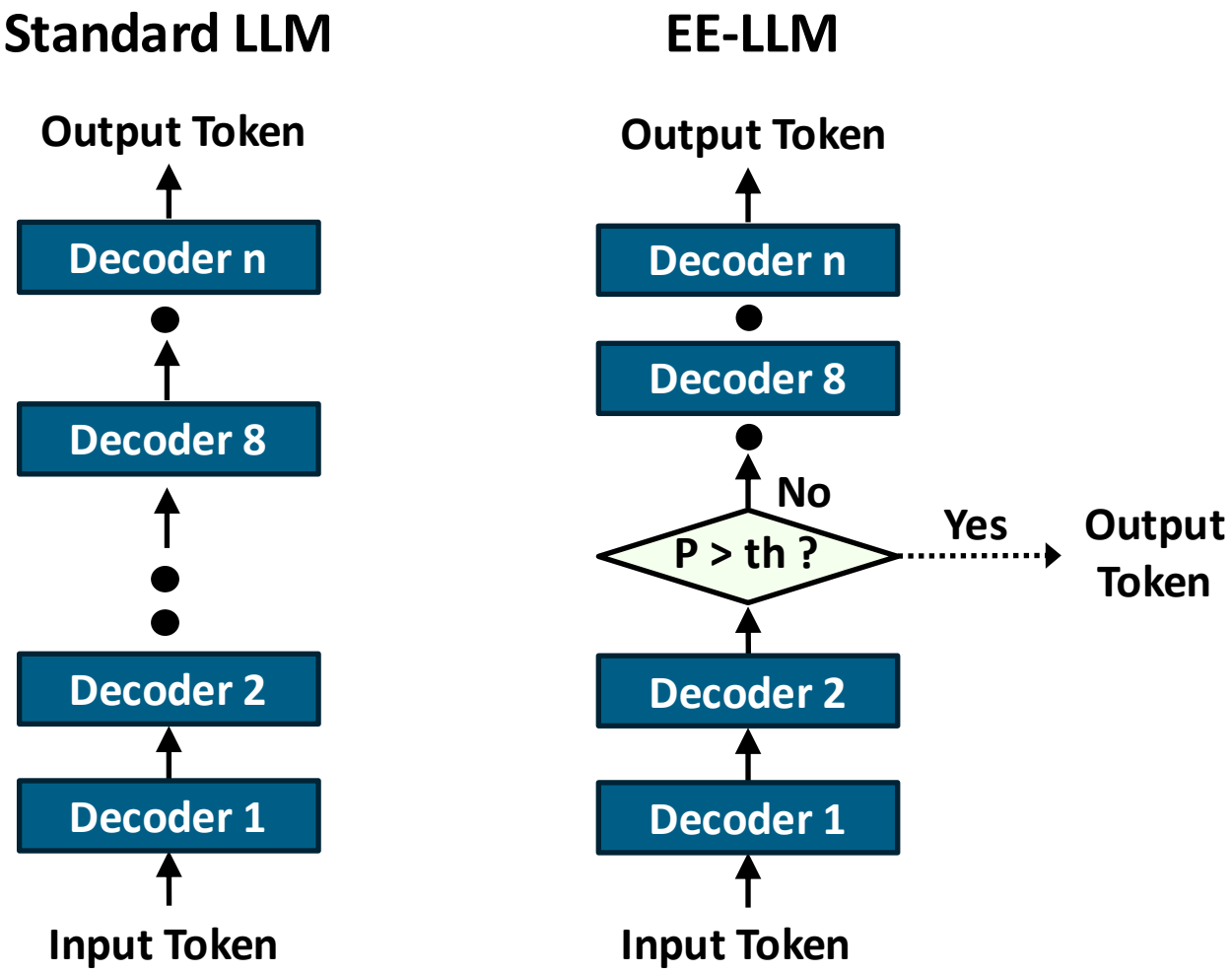
**More requests in parallel (higher batch sizes)**



*Ideally*, high throughput inference is desirable via faster token processing and large batch sizes

# Early-Exit (EE) LLMs

EE-LLMs are a variant of LLMs that allow tokens to *exit early* if they meet a confidence threshold



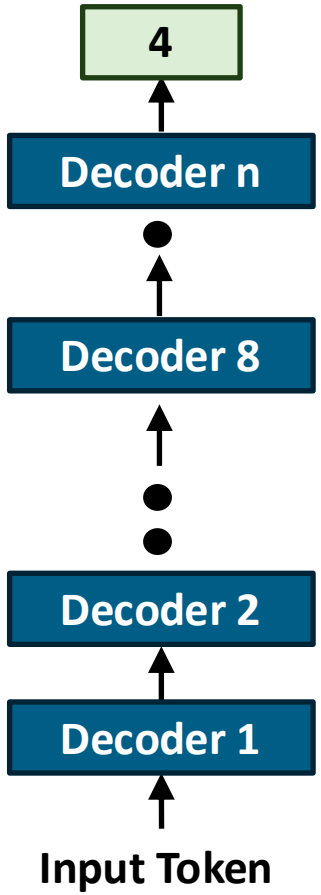


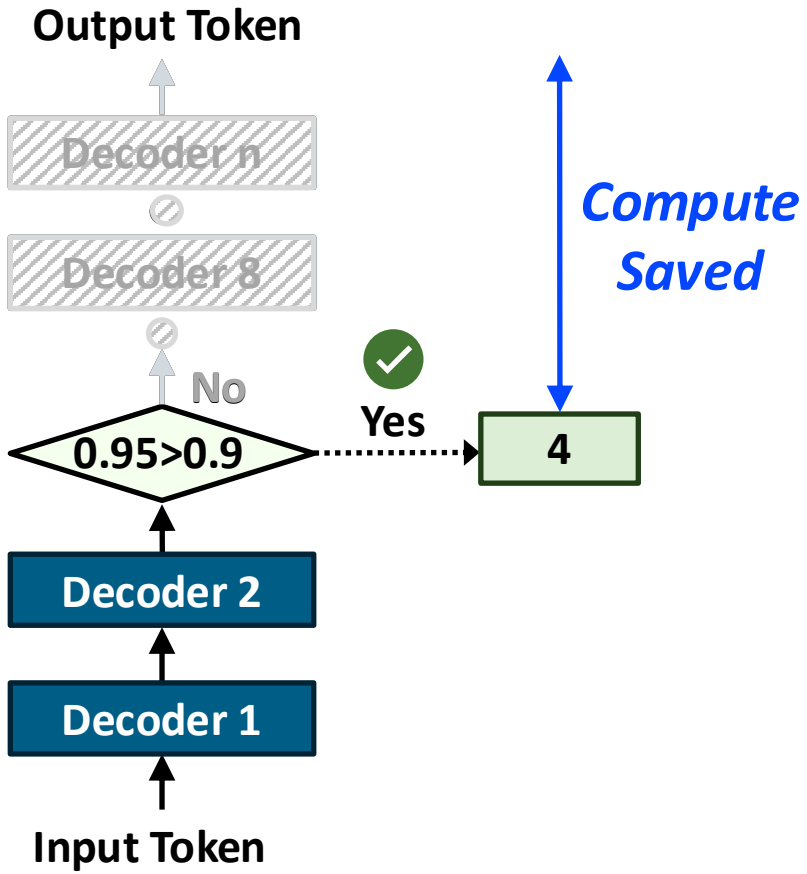

What are the challenges with serving EE-LLMs today ?

# Early-Exit (EE) LLMs

EE-LLMs are a variant of LLMs that allow tokens to *exit early* if they meet a confidence threshold

***Q: What is 2+2 ?***



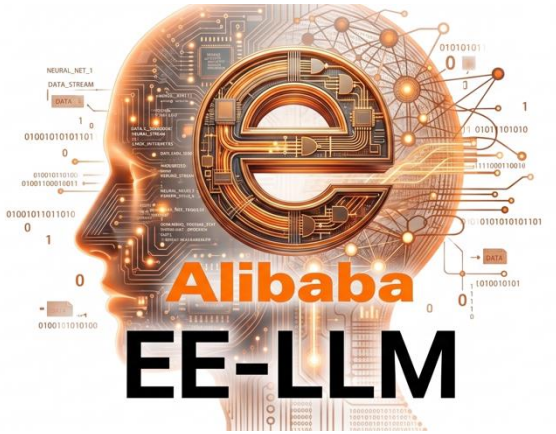

Meta AI Research

LayerSkip*

Enabling Early Exit Self-Speculative Decoding

Jun 14, 2024

Learn more




#

*Layerskip, ACL 2024, https://aclanthology.org/2024.acl-long.681.pdf

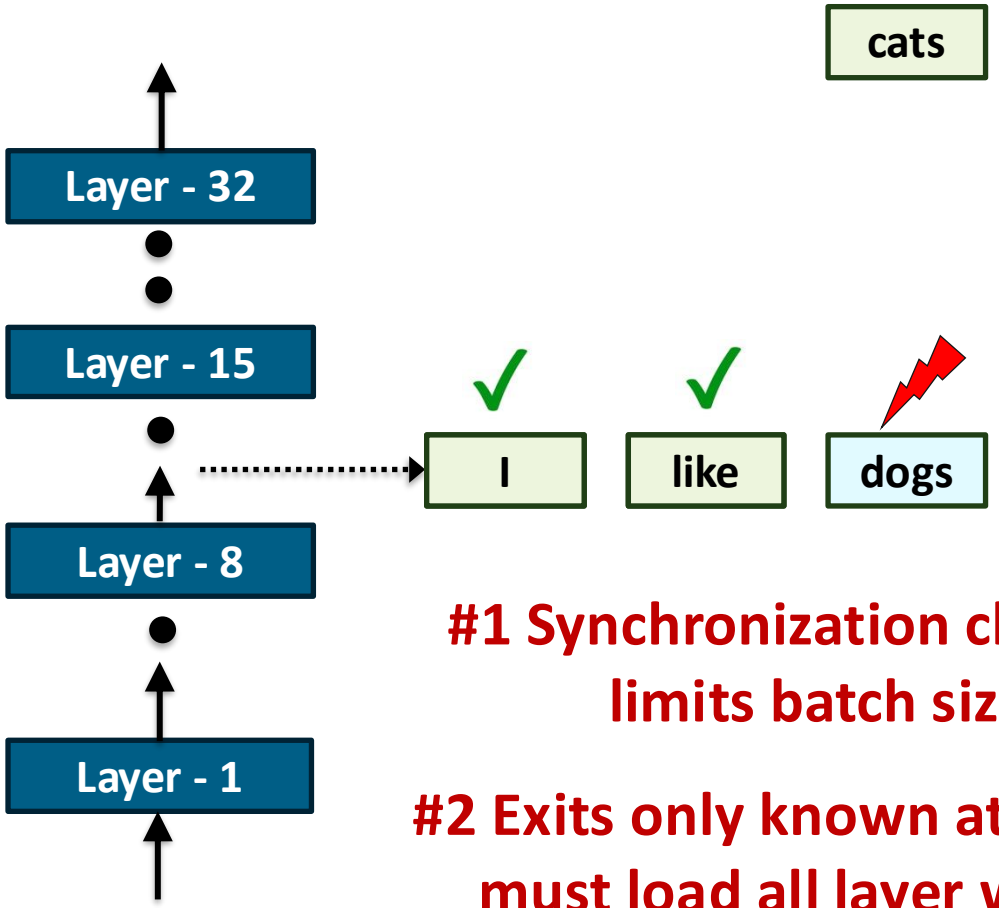
#EE-LLM, ICML 2024, https://arxiv.org/pdf/2312.04916



What are the challenges with serving EE-LLMs today ?

# Challenge With EE-LLM Serving

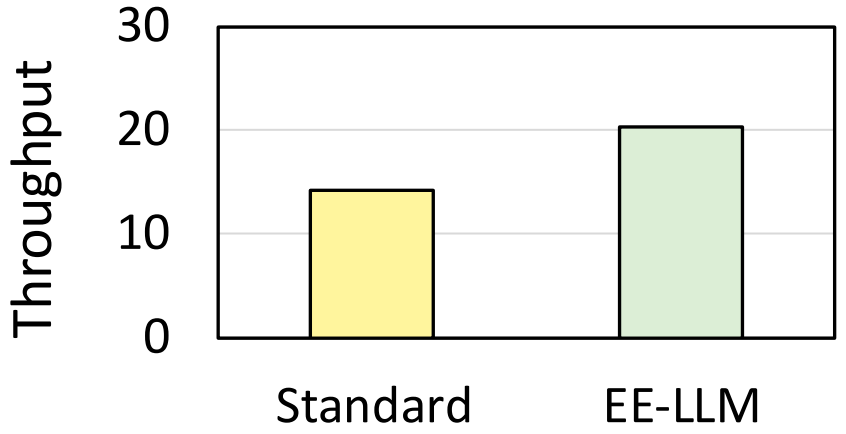
***Q: Which animal do you like ?***



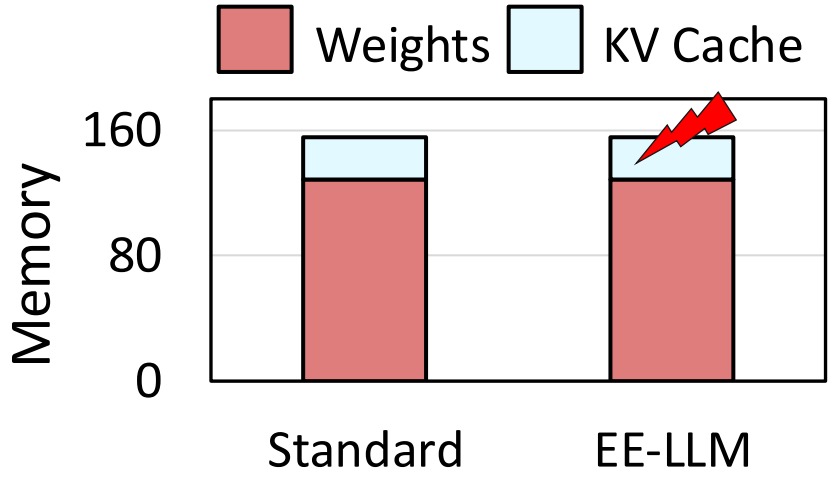

**#1 Synchronization challenge limits batch sizes**

**#2 Exits only known at runtime, must load all layer weights**

**Only limited throughput improvements**




**Memory footprint remains identical**





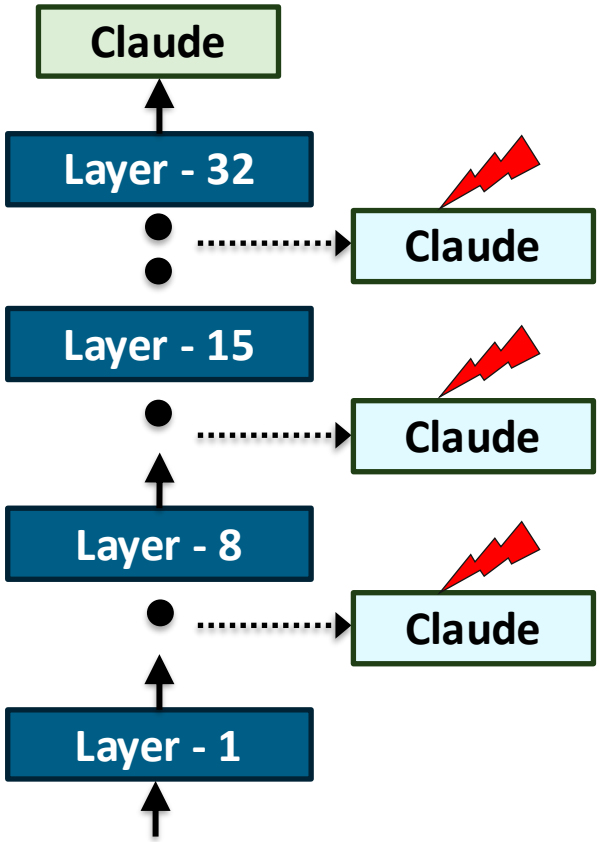

EE-LLMs offer only limited throughput benefits

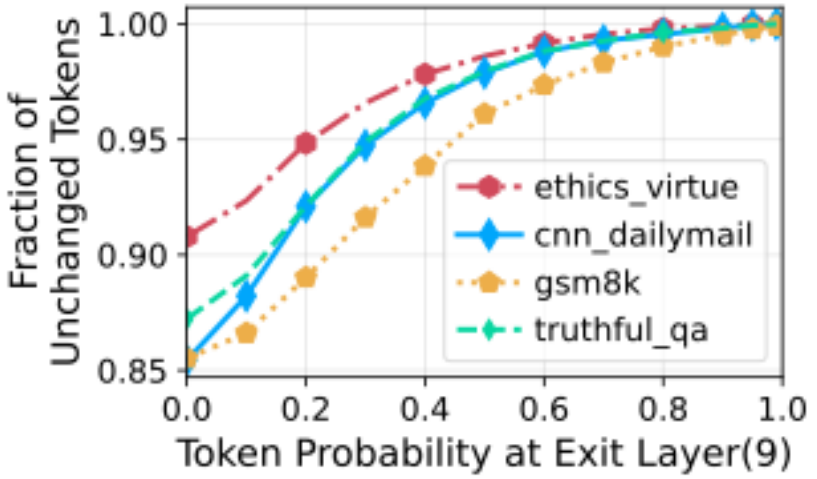
# Outline

- ❑ Background & Motivation
- ❑ **HELIOS: Insights and Design**
- ❑ Evaluation Methodology & Results
- ❑ Conclusion

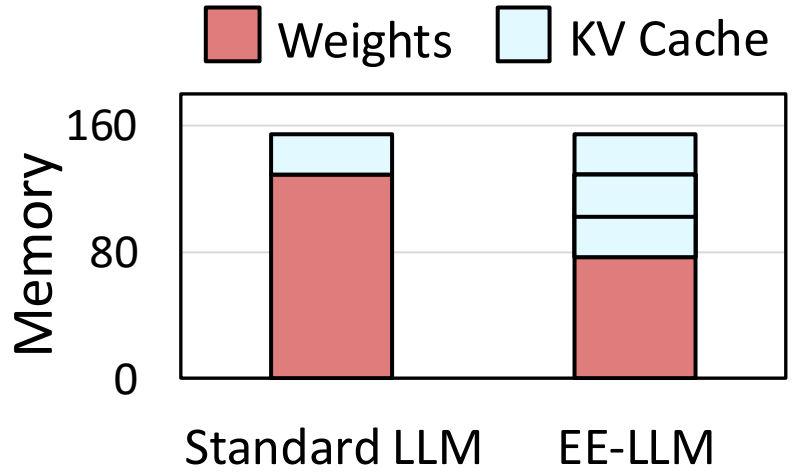
# Key Insight #1: Not Meeting Confidence Is Okay

***Q: What is your name ?***




95% of low confidence tokens could exit at layer-9




Predicted token *often unchanged despite additional layer traversal*

Greedily exit early → *only* load most likely to be used layers

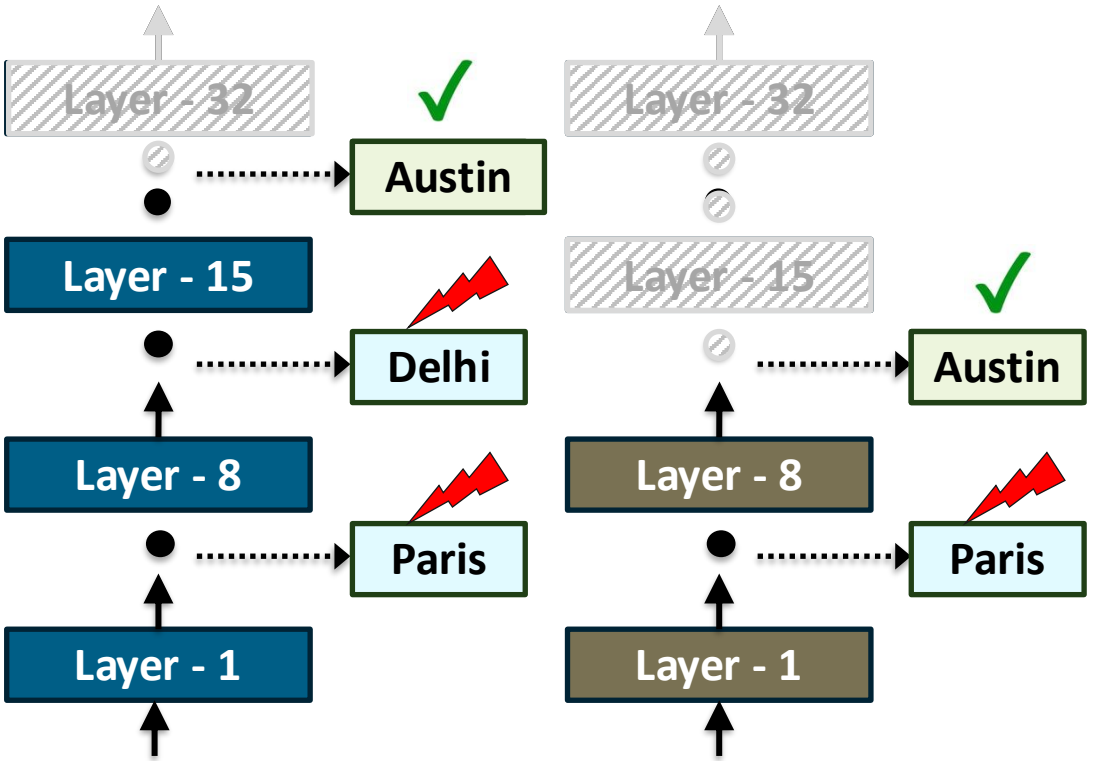
Re-purpose memory saved → *support* more requests

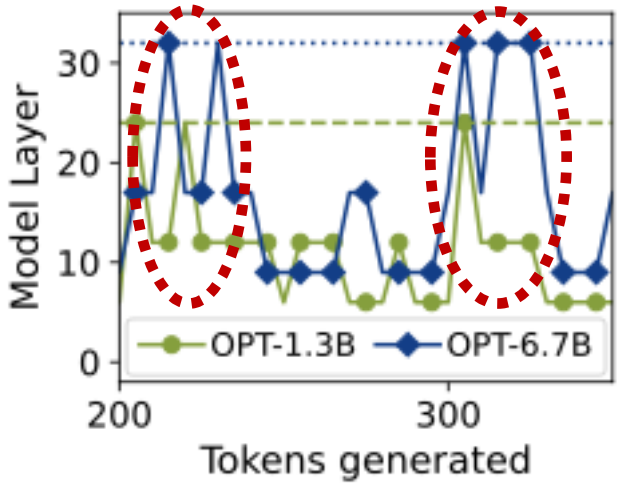


Challenge: How do we know what are the most likely to be used layers ?

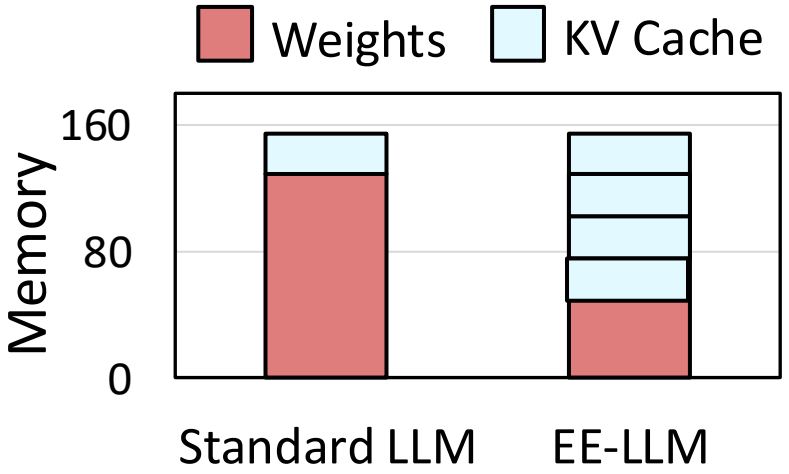
# Key Insight #2: Early-Exits Are Often Complementary

***Q: Where are the longhorns from?***




57% of tokens unable to exit on OPT-1.3B, exit early on OPT-6.7B



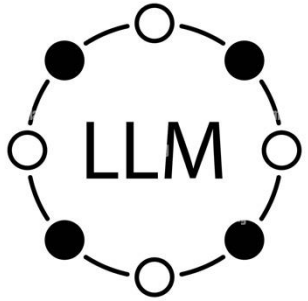

Tokens which require more layers to exit in one model can *exit earlier* in another

Employ multiple LLMs → collectively maximize early-exits

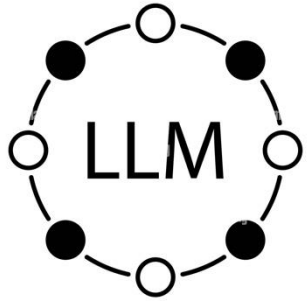


Challenge: How do we know which models to use and when to use a different model ?

# Challenges In Exploiting Our Key Insights


Too many models → How do you choose?


Early exits unknown → task and LLM-dependent

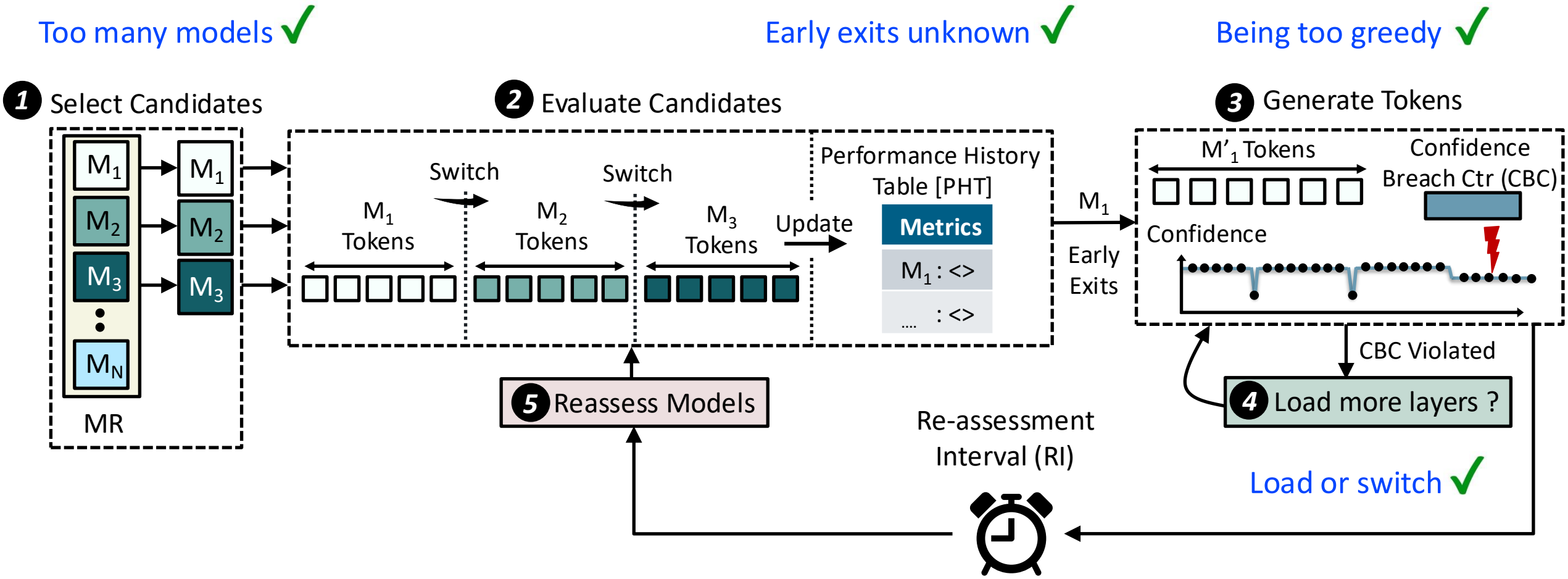
Being too greedy → lowers accuracy

Load more layers or switch to another?



We propose HELIOS that addresses these challenges

# HELIOS: Design Overview






HELIOS greedily loads layers and orchestrates multiple EE-LLMs to maximize early-exits

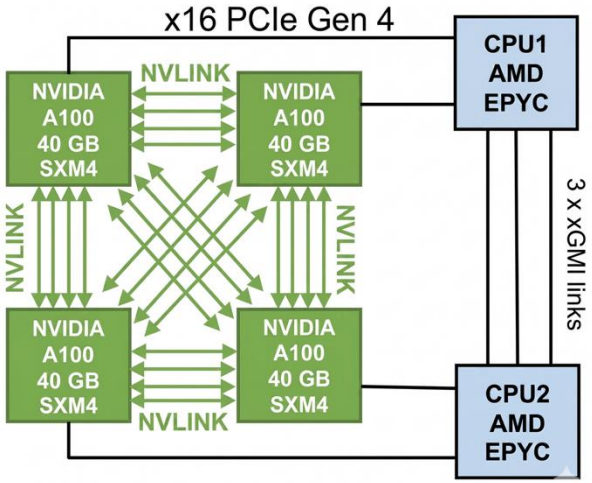
# Outline

- ❑ Background & Motivation
- ❑ HELIOS: Insights and Design
- ❑ **Evaluation Methodology & Results**
- ❑ Conclusion

# Evaluation Methodology

## *Setup*

4xNVIDIA A100 GPUs & AMD EPYC CPU




## *Models*

OPT


Llama

CodeLlama

## *Datasets*

Standard benchmarks:

- ShareGPT
- CNN/Dailymail
- HumanEval

…

## *Figure of Merit*

$$\text{Throughput} = \frac{\text{Tokens}}{\text{Second}}$$

*(Higher is Better)*

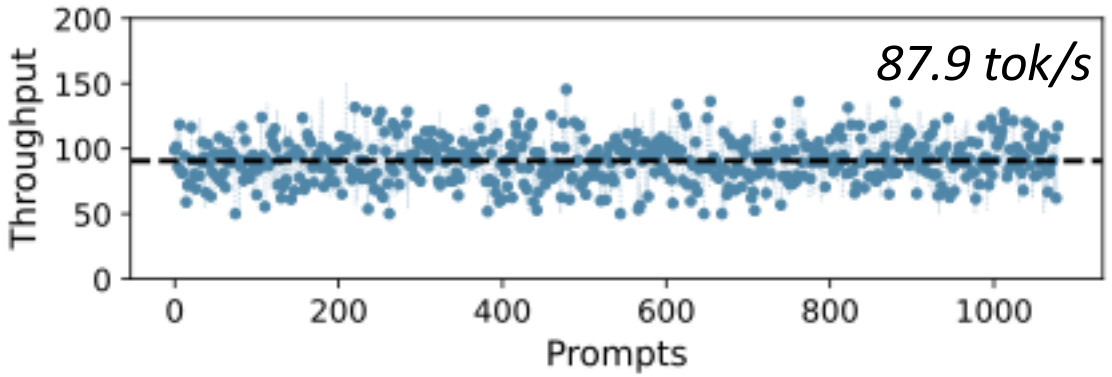
$$\text{Accuracy} = \text{Perplexity}$$

*(Higher is Better)*

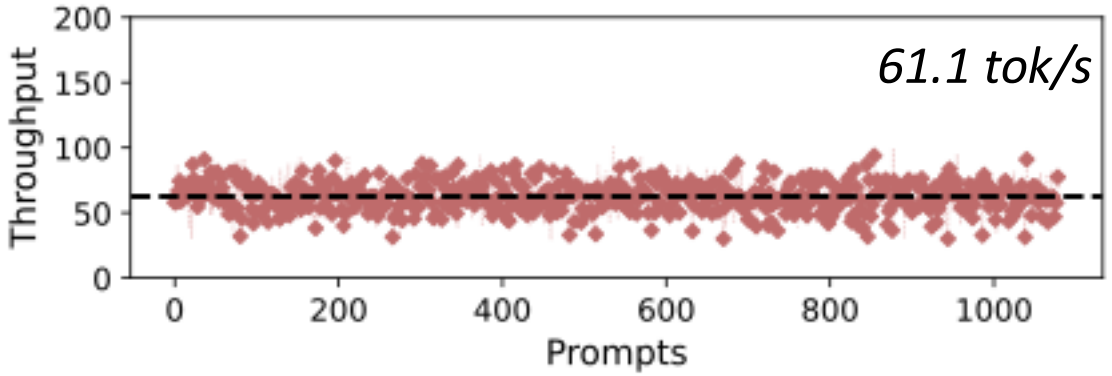


Our methodology ensures that our evaluations are consistent with prior works

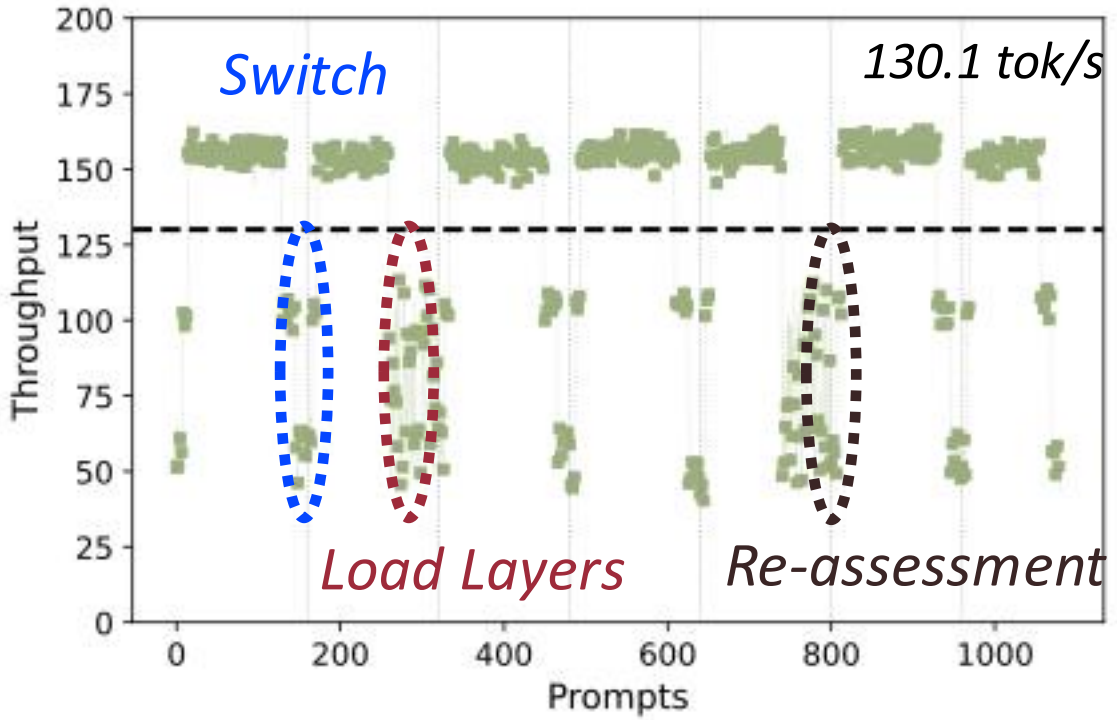
# HELIOS Enables Faster Token Processing




OPT-1.3B (Small)




OPT-6.7B (Large)



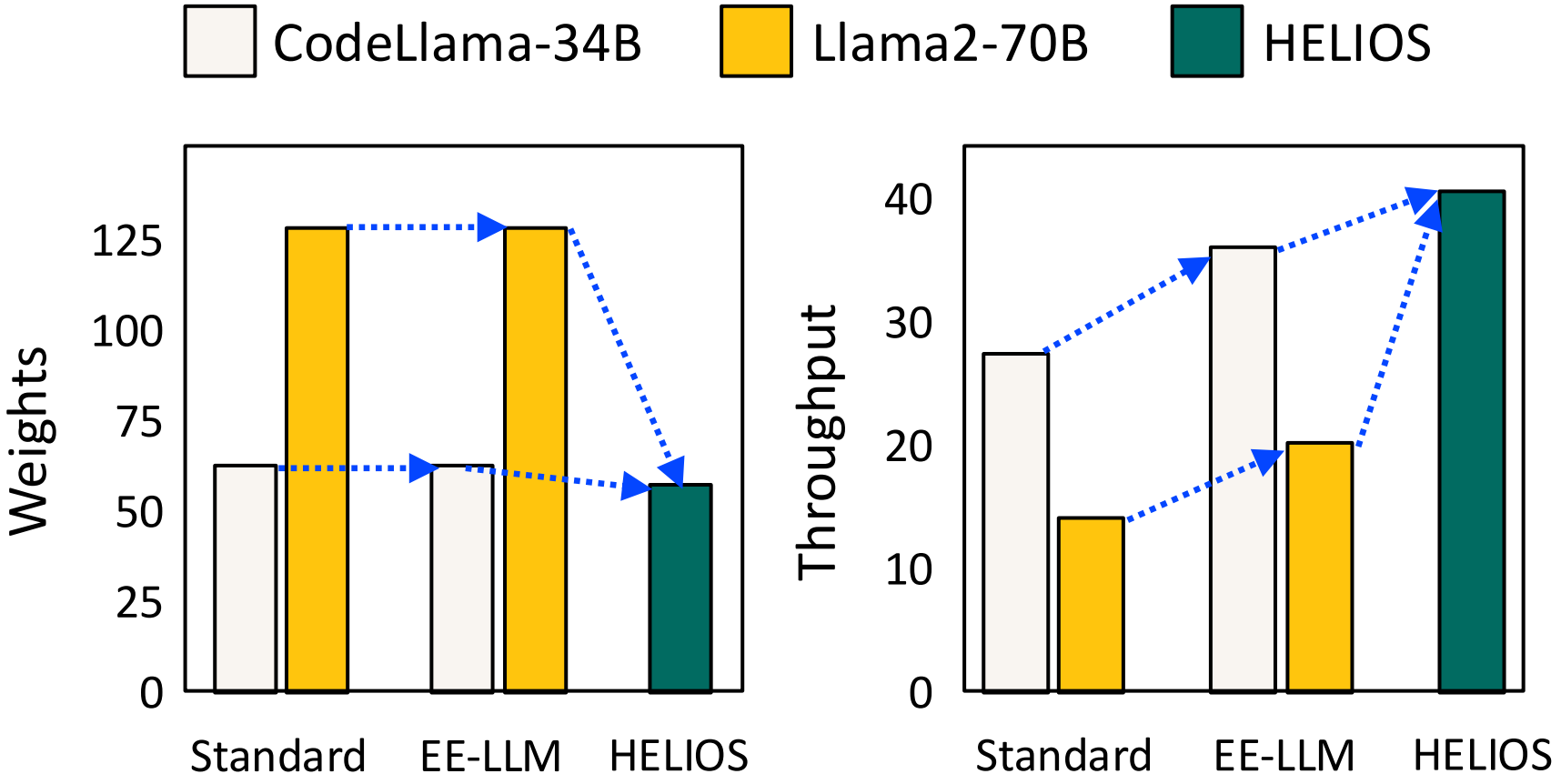

HELIOS

HELIOS achieves accuracy comparable to OPT-6.7B and throughput higher than OPT-1.3B



HELIOS enables faster token processing speed while preserving accuracy

# HELIOS Improves Batch Sizes




*Varies across tasks*


Llama3-8B (Total 32 Layers)

*Higher for larger models*


Llama2-70B (Total 80 Layers)



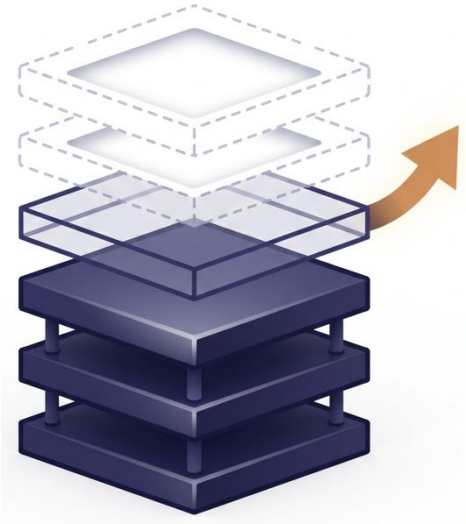
HELIOS improves both token processing latency and batch sizes

# Outline

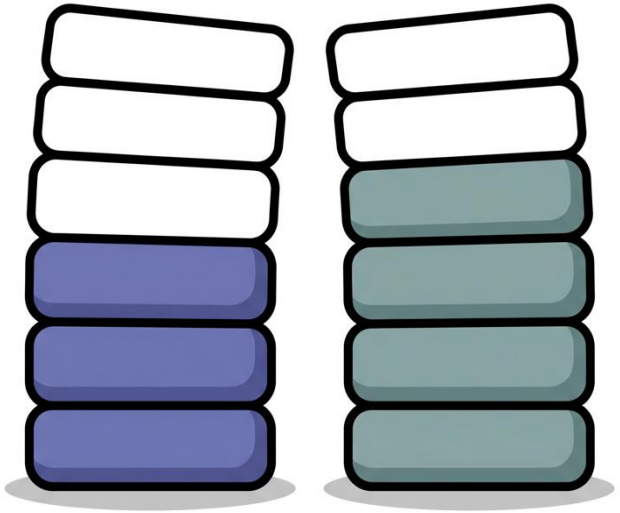
- ❑ Background & Motivation
- ❑ HELIOS: Insights and Design
- ❑ Evaluation Methodology & Results
- ❑ **Conclusion**

# Conclusion


Current EE-LLMs: no memory savings → limited throughput

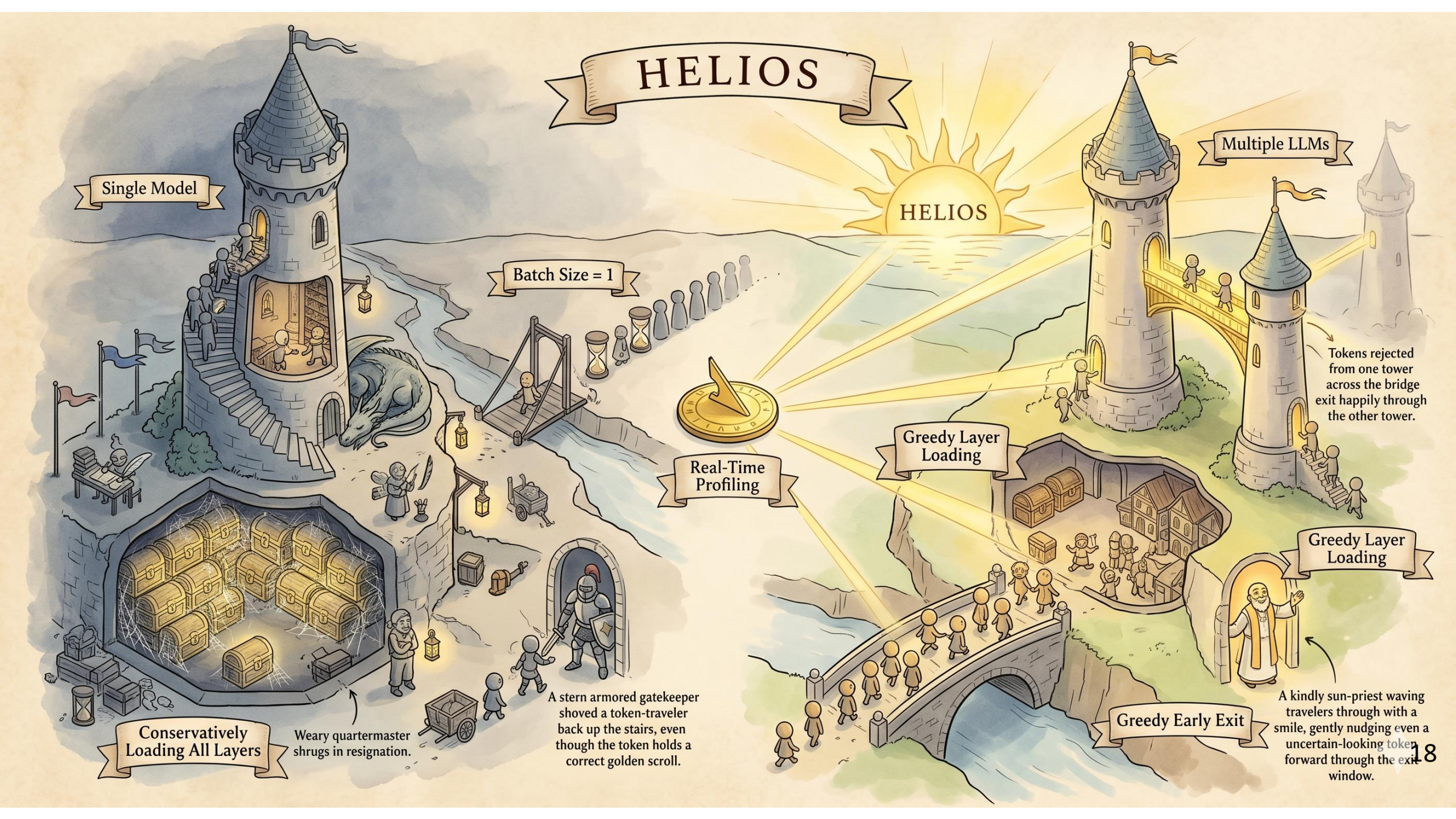


O*nly* loads most likely to be used layers



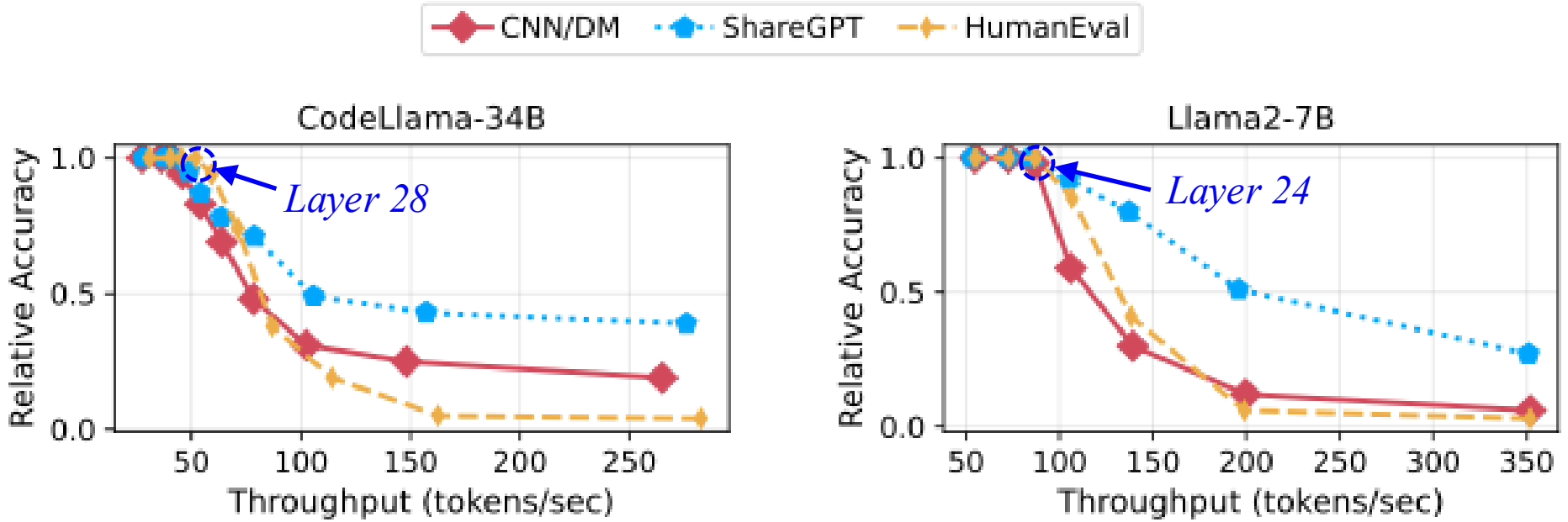
Uses *multiple LLMs* to maximize early exits

Improves both token processing latency and batch sizes


HELIOS
Single Model
Batch Size = 1
HELIOS
Multiple LLMs
Tokens rejected from one tower across the bridge exit happily through the other tower.
Real-Time Profiling
Greedy Layer Loading
Greedy Layer Loading
Conservatively Loading All Layers
Weary quartermaster shrugs in resignation.
A stern armored gatekeeper shoved a token-traveler back up the stairs, even though the token holds a correct golden scroll.
Greedy Early Exit
A kindly sun-priest waving travelers through with a smile, gently nudging even a uncertain-looking token forward through the exit window.

# Backup Slides

# HELIOS Preserves Downstream Task Accuracy




- Our experiments show that accuracy remains identical even with fewer layers
- This observation is consistent with prior work*

*Enabling Early Exit Inference and Self-Speculative Decoding, ACL 2024
The Unreasonable Ineffectiveness of the Deeper Layers, ICLR 2025



HELIOS dynamically finds the minimum number of layers needed to serve the request stream