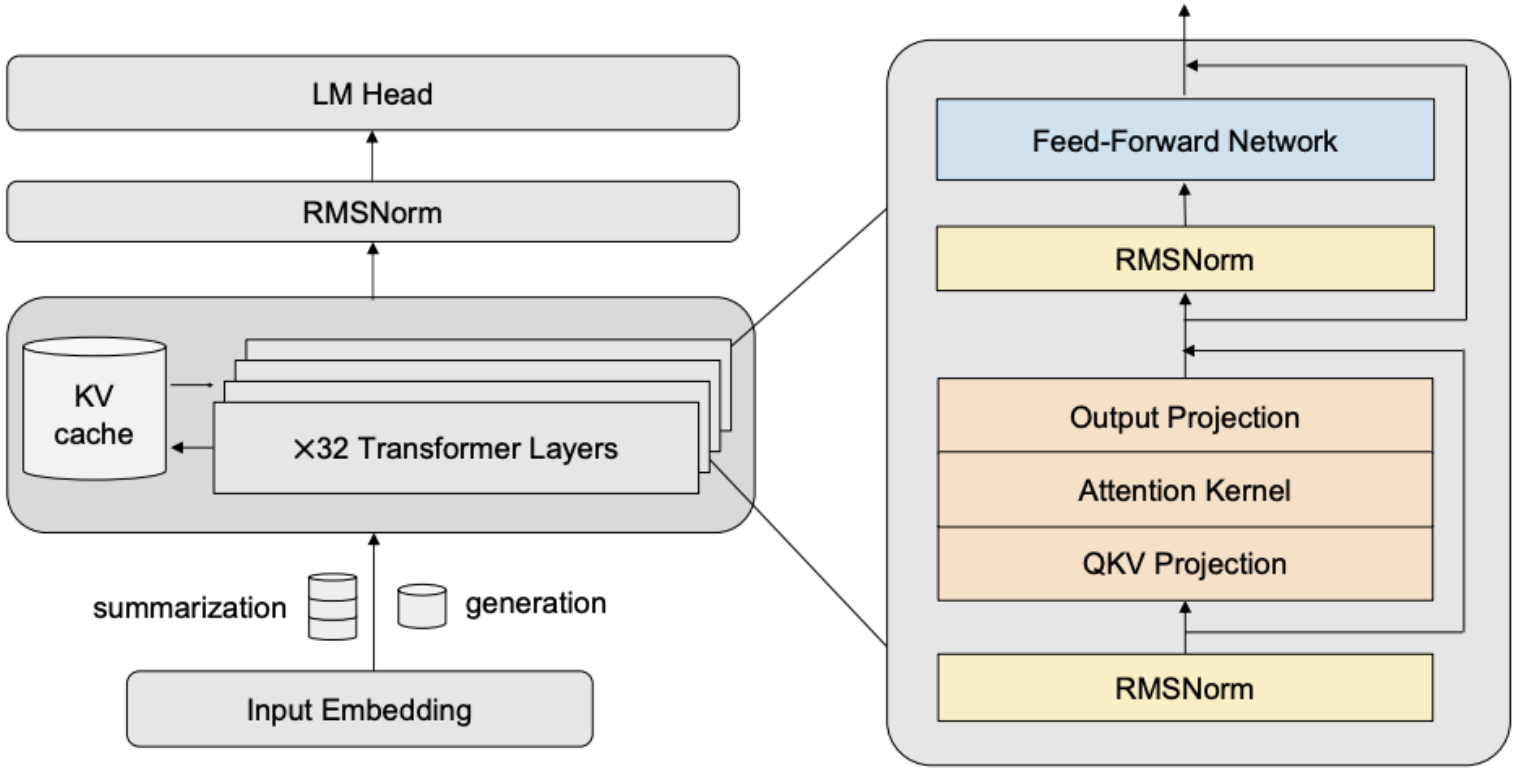
Scaling up Privacy-Preserving ML: A CKKS Implementation of Llama2-7B

Jaiyoung Park, Sejin Park, Jung Ho Ahn, Jung Hee Cheon,
Guillaume Hanrot, Jung Woo Kim, Jai Hyun Park, Minje Park,
Damien Stehle
Seoul National University, CryptoLab Inc

Llama Architecture



Llama-3-8B Model

Architecture	Decoder-only Transformer
Parameters	8 Billion
Hidden Dim	4,096
Layers	32 Transformer Blocks
Attention Heads	32
Vocab Size	32,000
Context Length	4,096 tokens

Challenges in FHE-based LLM Inference

Linear Algebra Operations	CCMM, CCMV,PCMM, PCMV operations, including batched variants
Activation Range Explosion	Large intermediate activations degrade bootstrapping precision → higher-degree Chebyshev approximations → amplified compute cost
Non-linear Function	SoftMax, SiLU, RMSNorm require polynomial approximation in FHE; minimizing degree without accuracy loss
System-level Challenges	GPU/ASIC acceleration, multi-device parallelism, memory management

Gadgets for LLM Inference

MM

Matrix–Matrix

- CCMM [JKLS18, Par25]
- PCMM [BCH+24, BCH+25]
- Batched CCMM [CKL25]
- **Batched PCMM ✦ NEW**
- Transposition

Mv

Matrix–Vector

- CCMV [HS24]
- PCMV [HYT+24]

Non-linear

Activation Functions

- SoftMax [CHK+24]
- RMSNorm [Chebyshev]
- SiLU [Chebyshev]
- **Slim poly evaluation ✦ NEW**

Our contributions Prior work

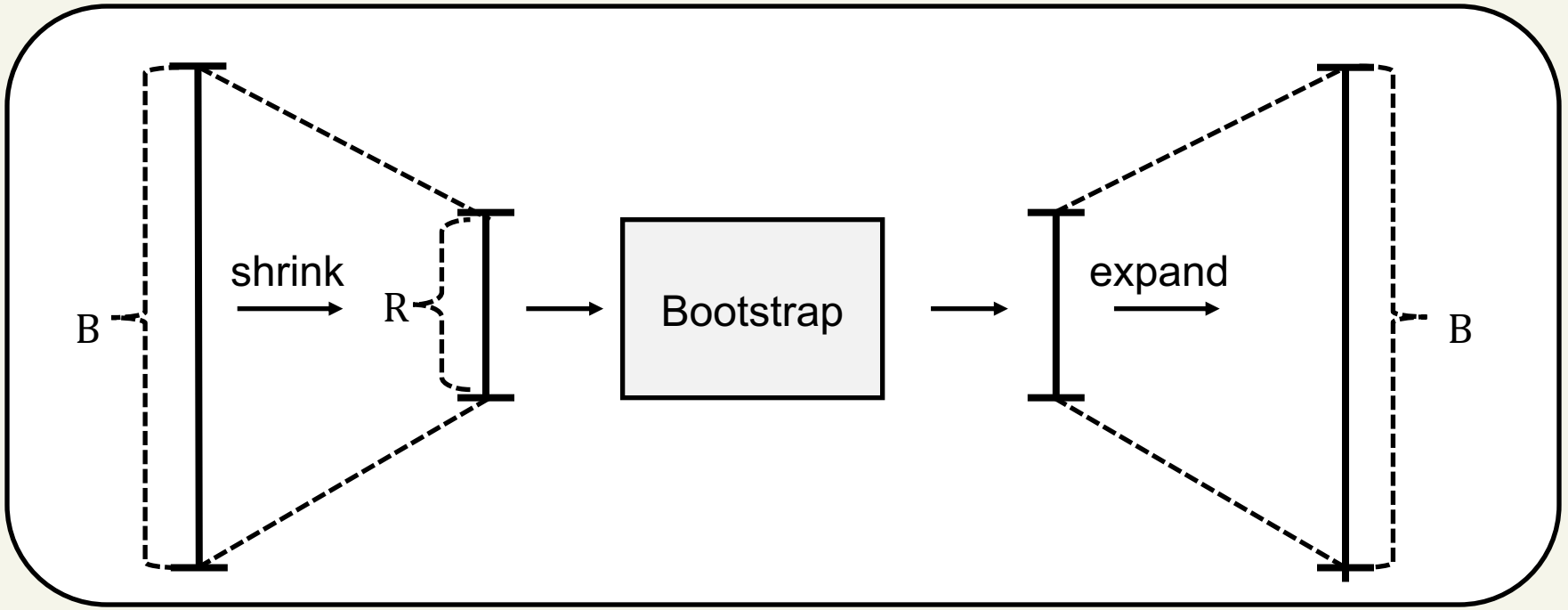
LLM Activation Interval

Input interval B

→ Bootstrapping precision $\propto 1/\mathrm{B}$

ML optimization technique

→ post training quantization (PTQ)



Outlier Phenomenon in LLM Activations

Outlier Presence

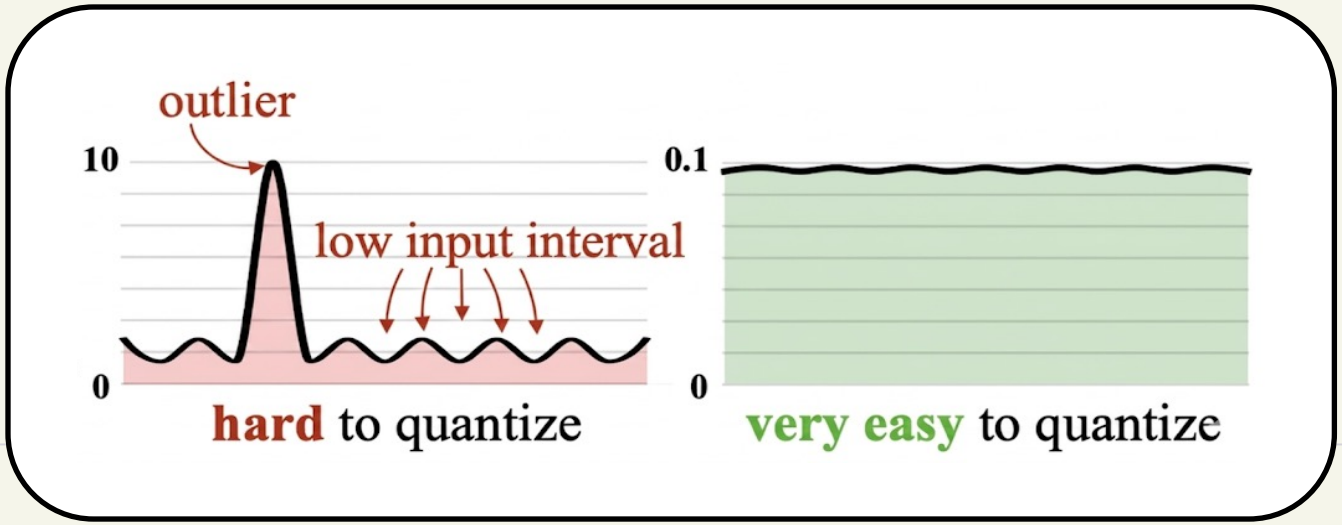
→ LLM activations contain large outlier values concentrated in specific channels across tokens

Interval Expansion

→ These outliers significantly widen the activation interval, making quantization difficult

Impact on FHE Inference

→ Large intervals degrade bootstrapping precision and polynomial approximation accuracy, increasing compute cost



Rotate Activation to Distribute Outlier Values

Goal

Apply a rotation to activations to distribute outlier energy across all dimensions

Formulation

$$x' = Rx$$

(R: orthogonal matrix)

Properties

$\max|Rx| < \max|x| \rightarrow$ smaller activation interval

Example

Before (with outlier)

```
x = [0.3, -0.2, 0.1, 9.7]
```

↓ Apply Rotation R

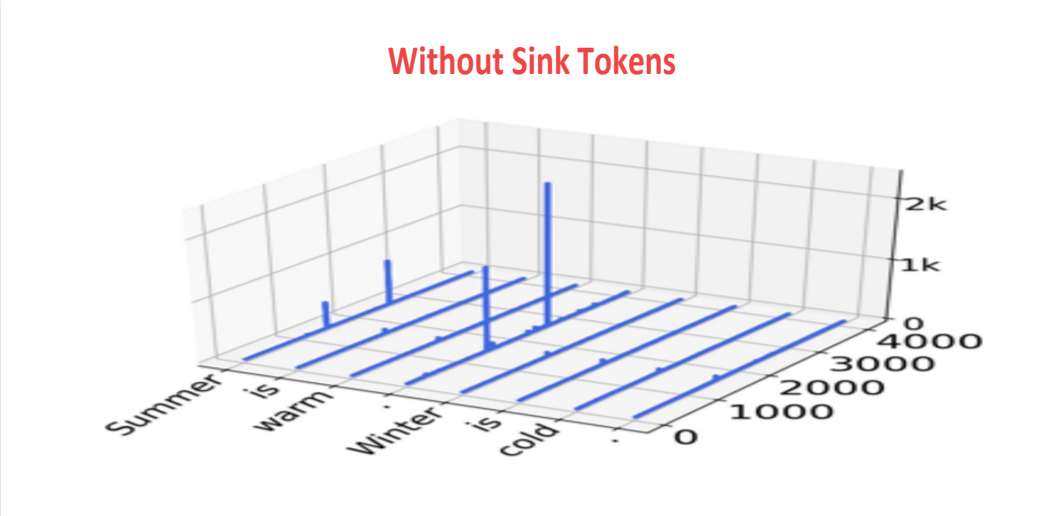
After (distributed)

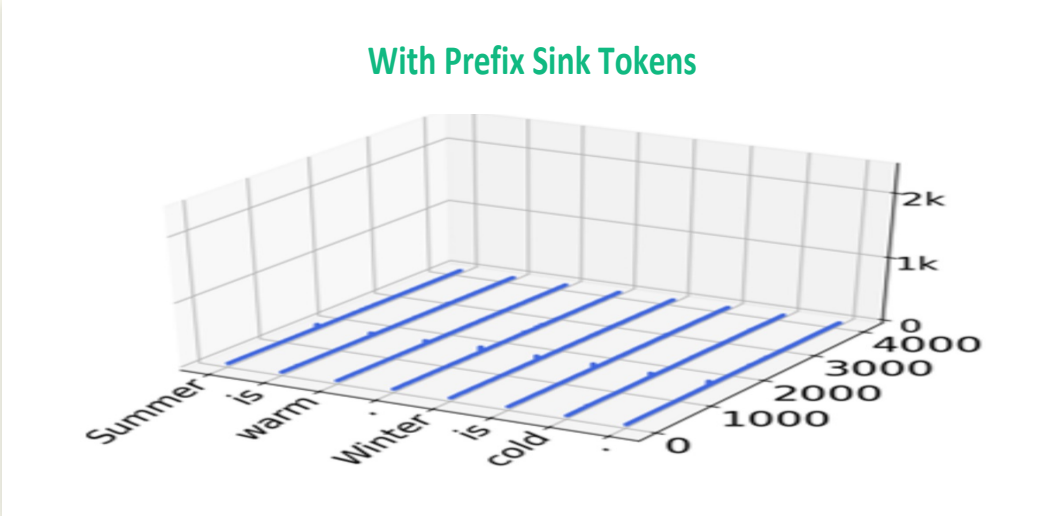
```
x' = Rx = [2.1, 2.5, 1.8, 2.4]
```

Attention Sink in LLMs

- Certain tokens attract disproportionate attention across layers (*attention sinks*)
- Prepending sink tokens absorbs excess attention from the main sequence
- Redistributed attention suppresses activation outliers

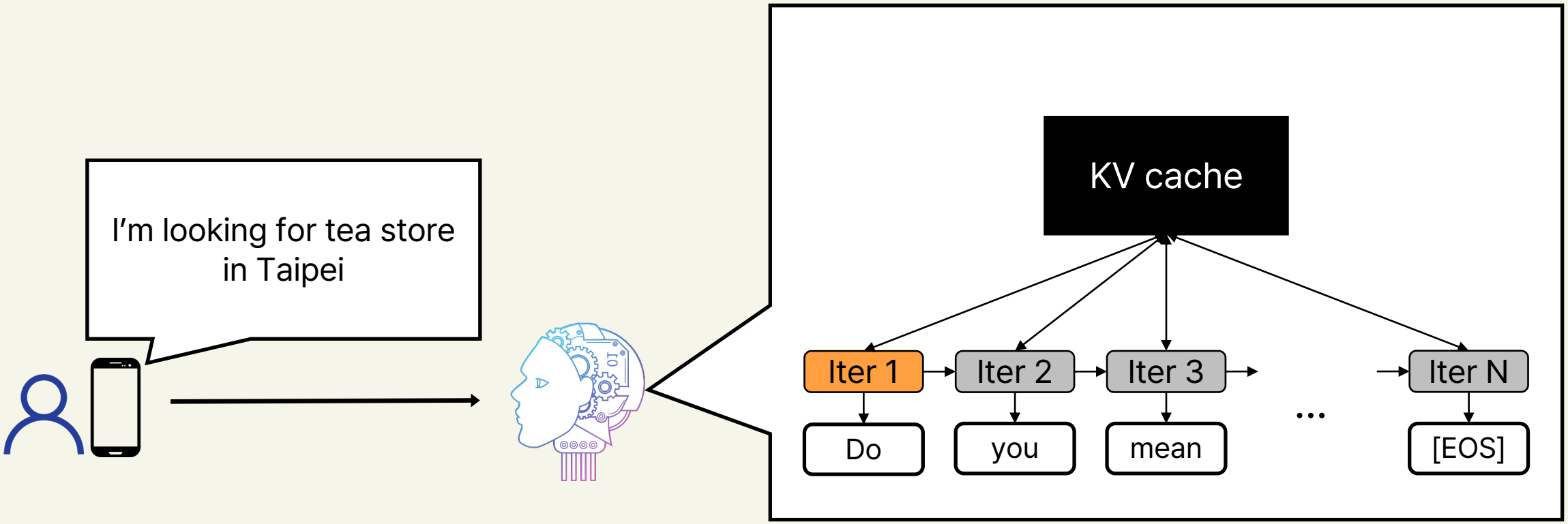
→ **Controlled sinks reduce the activation interval for FHE inference**





Private LLM Inference

Does a single-turn query provide enough context for accurate LLM responses?



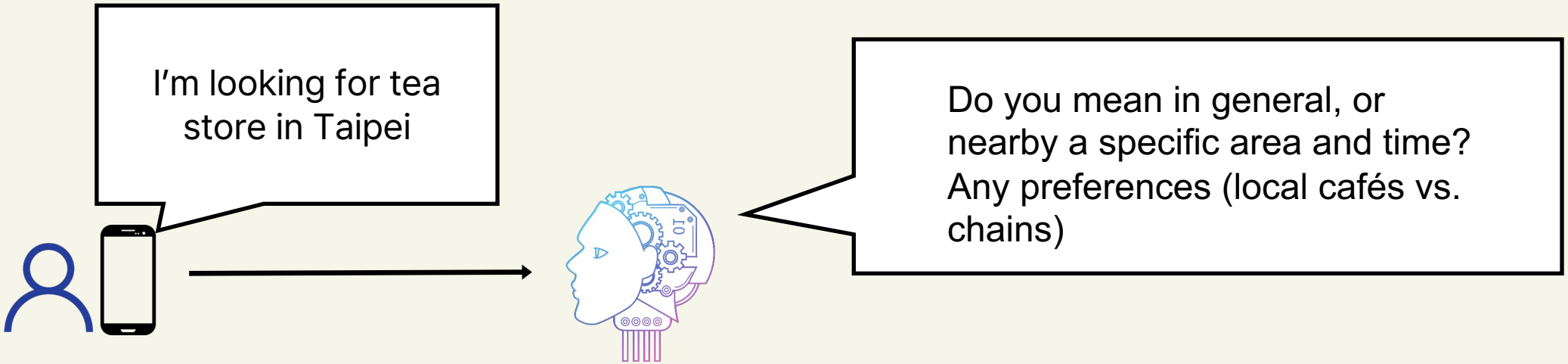
Recent Trends in LLM Usage

- Multi-turn conversation 💬
  - LLM requires conversational context to refine responses
- Retrieval-Augmented Generation (RAG) 📚
  - LLM knowledge is incomplete; external retrieval fills the gap
- Chain-of-Thought (CoT) 🧠
  - LLM generates intermediate reasoning steps, expanding the token sequence
- Multimodal Input (Documents, Images) 🖼️
  - Documents and images are tokenized into the same attention sequence, further extending context

➡ Encrypting the entire pipeline end-to-end is impractical — we need selective encryption

LLM Inference Challenges

- Single-turn queries lack conversational context and reasoning history, often failing to fully capture user intent



Privacy-preserving LLM Inference in Action

Multi-turn (Plaintext)

I'm looking for tea stores in Taipei

Traditional tea houses; a calm atmosphere with seating would be nice.

I'll be near *Yongkang Street, today 14:00–16:00*
(🔒 Secured from here)

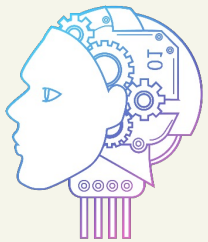
Do you mean in general, or nearby a specific area and time? Any preferences (traditional tea houses vs. bubble tea)?

Tentative Search: local cafés in Lyon with iced coffee + outdoor seating

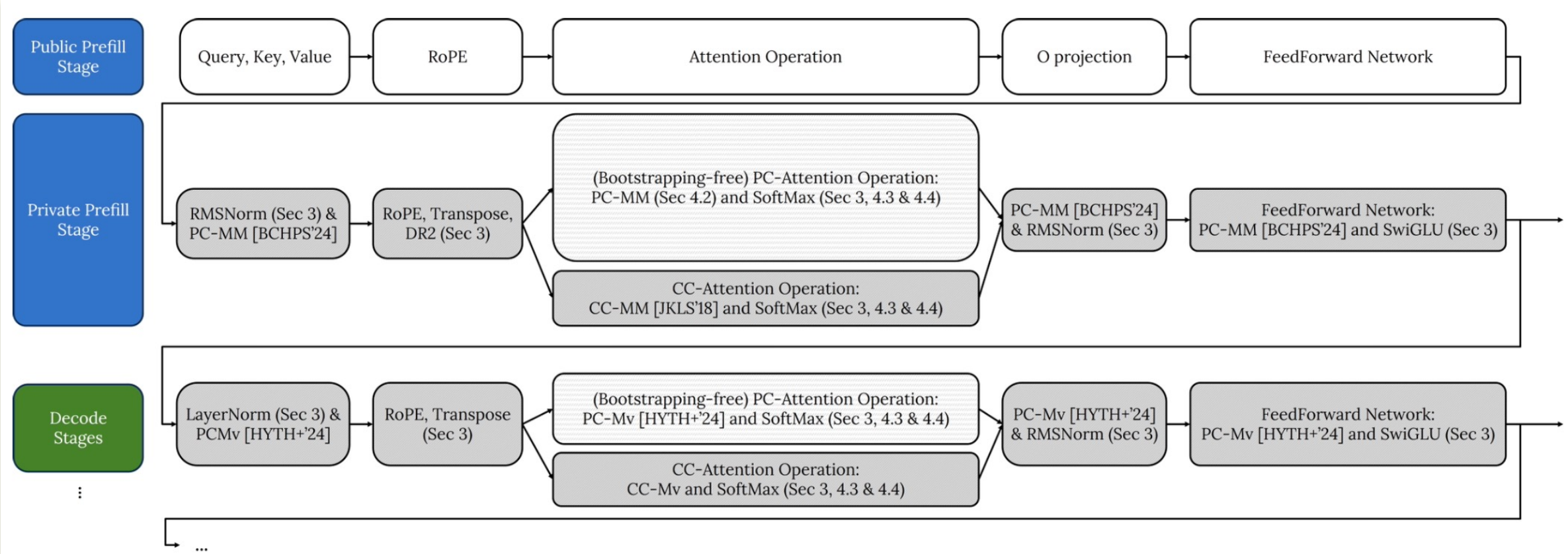
If you share approximate area and time window, I can be specific.

1. Wistaria Tea House – historic Japanese-era house
2. Cha Cha Thé – modern courtyard seating
3. (other matches…)

RAG, CoT (Plaintext)



Three-stage Transformer Inference

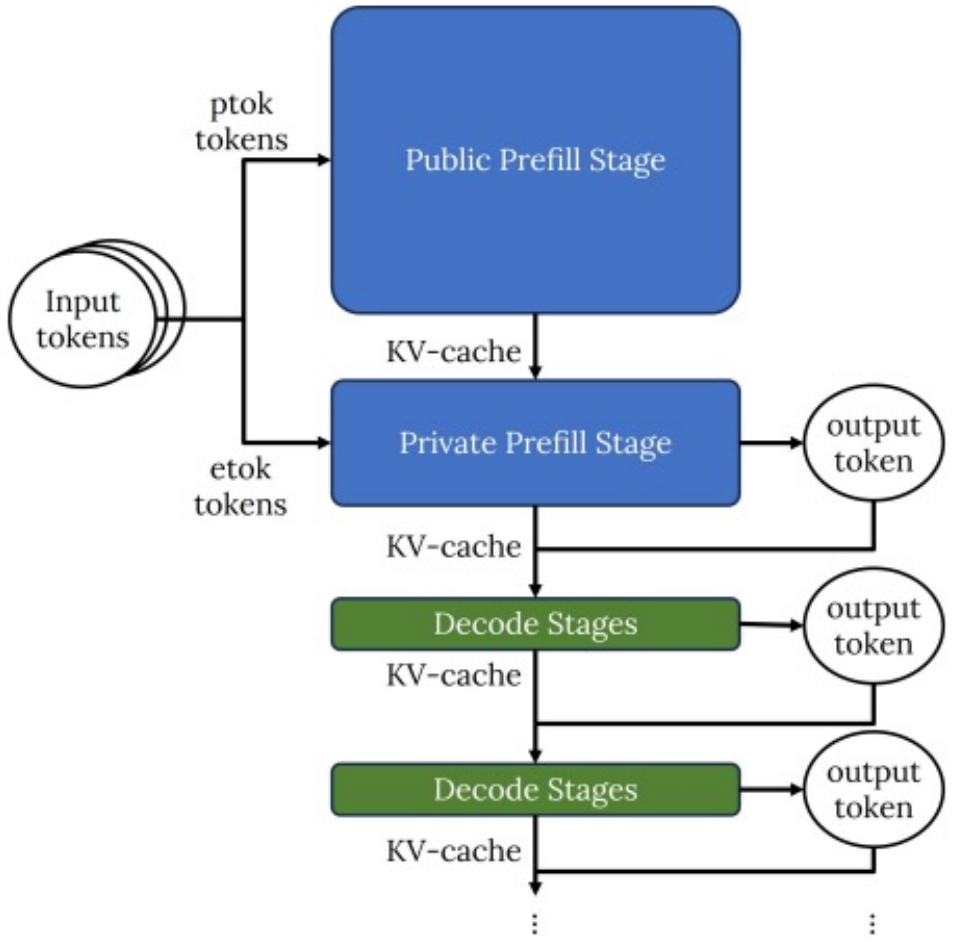


Public Prefill Stage
Standard plaintext processing

Private Prefill Stage
Encrypted summarization

Decode Stage
Encrypted generation

Two Prefill Stages: Plain then Encrypted



01

Public Prefill

- Process public input tokens in plaintext (no encryption overhead)
- Execute prefill to generate the initial KV-cache

02

Private Prefill

- Process private tokens (128 tokens) under CKKS encryption
- Update KV-cache and produce the first output token

FHE-based LLM Inference Result

15s
TTFT

1s
TBT

8×
RTX-5090

9×
Performance

	GPU	# tokens	Prefill
Ours	8 x RTX5090	128	15s
CERIUM[1]	8 x B200	128	134s

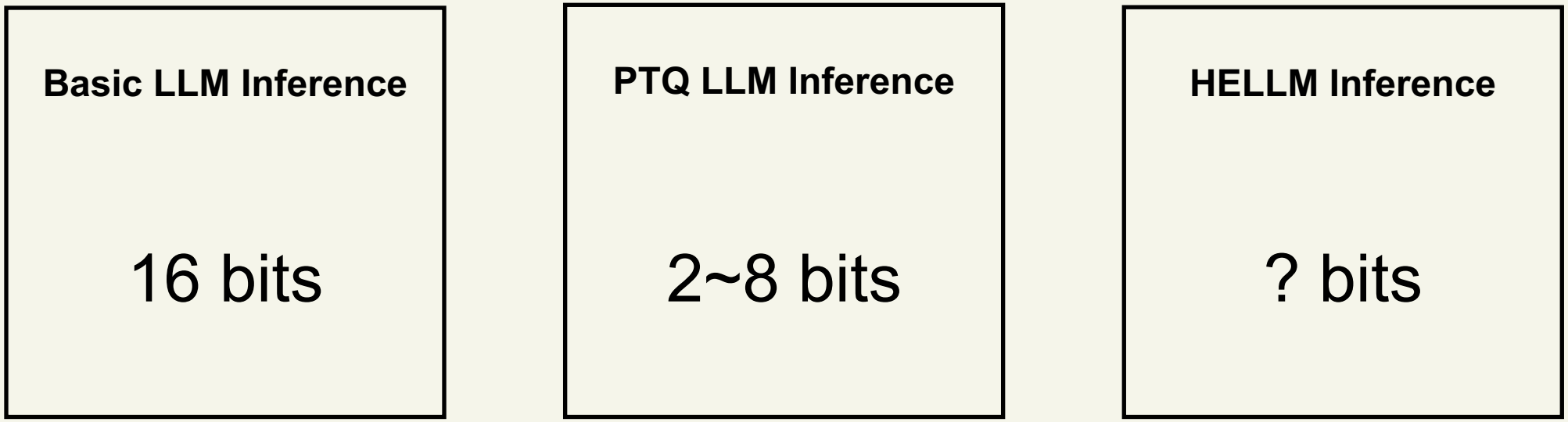
[1] Jayashankar et al. "A Scalable Multi-GPU Framework for Encrypted Large-Model Inference", arXiv:2512.11269

Backup Slides

Precision Analysis of LLM Inference

- Typical LLM inference
  - uses 16-bit floating-point operations (BF16, FP16)
  - can be reduced to 2~8 bits with post-training quantization (PTQ) techniques

- Key Question
  - How precise must HELLM be to perform LLM inference correctly?

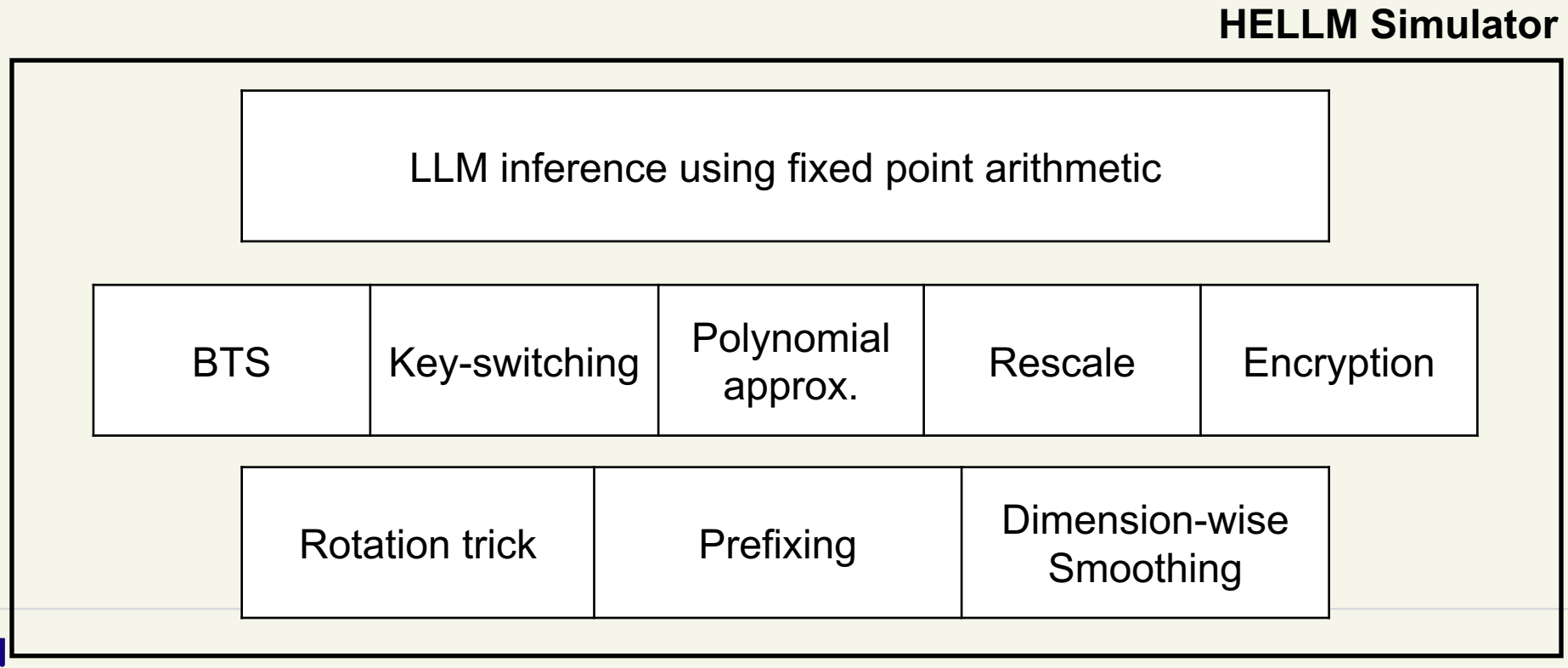


(PTQ = quantization + dequantization + …)

Simulating HELLM Precision in PyTorch

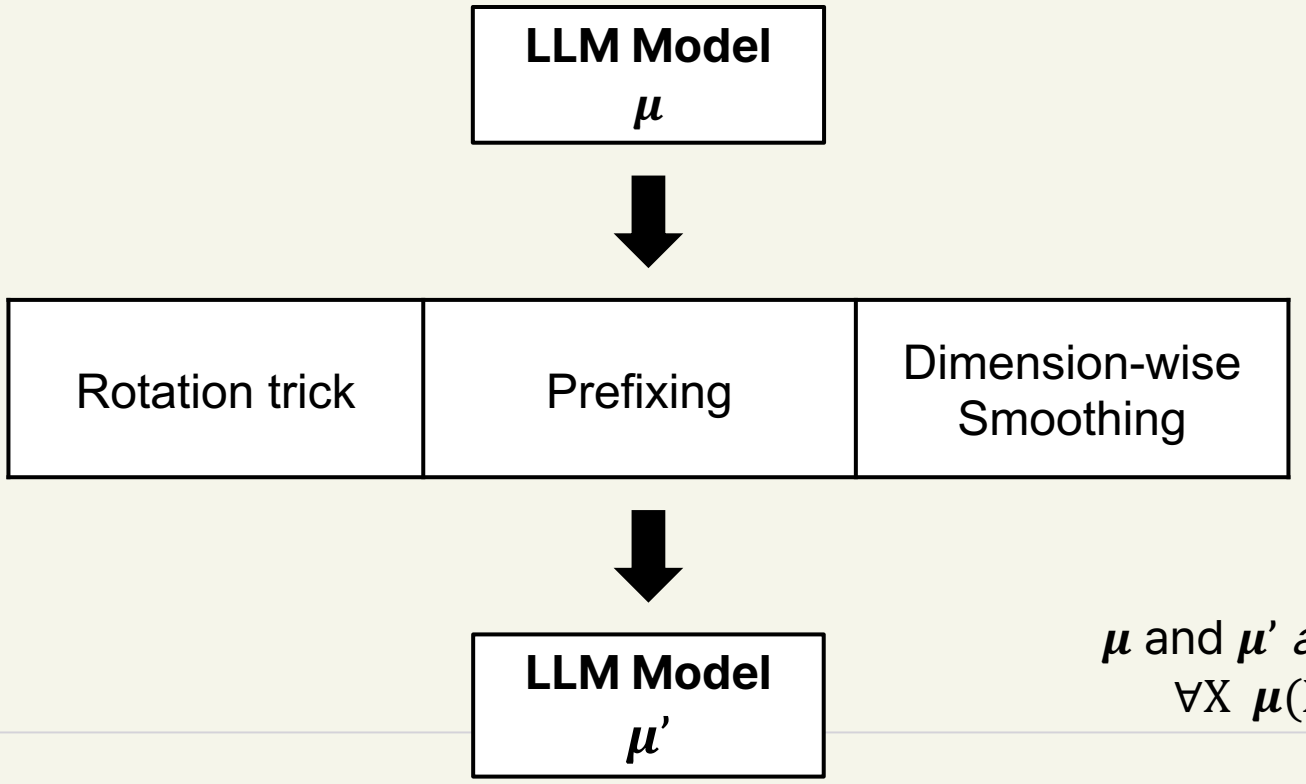
Method

- Implement fixed point arithmetic
- Introduce 5 key error sources
- Apply 3 quantization techniques



Simulating HELLM Precision in PyTorch

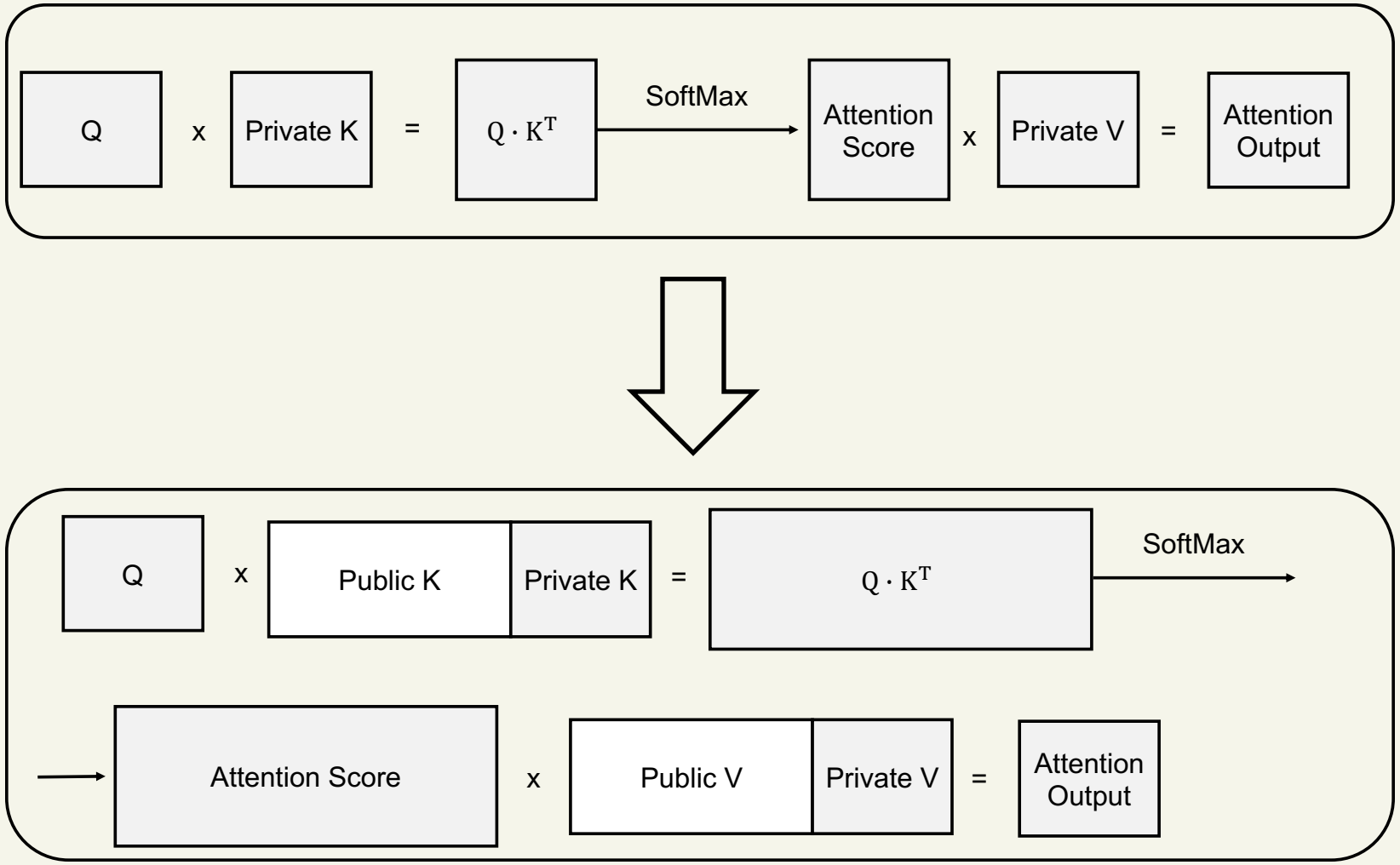
- Implement fixed point arithmetic
- Introduce 5 key error sources
- Apply 3 quantization techniques



μ and μ' *are equivalent*

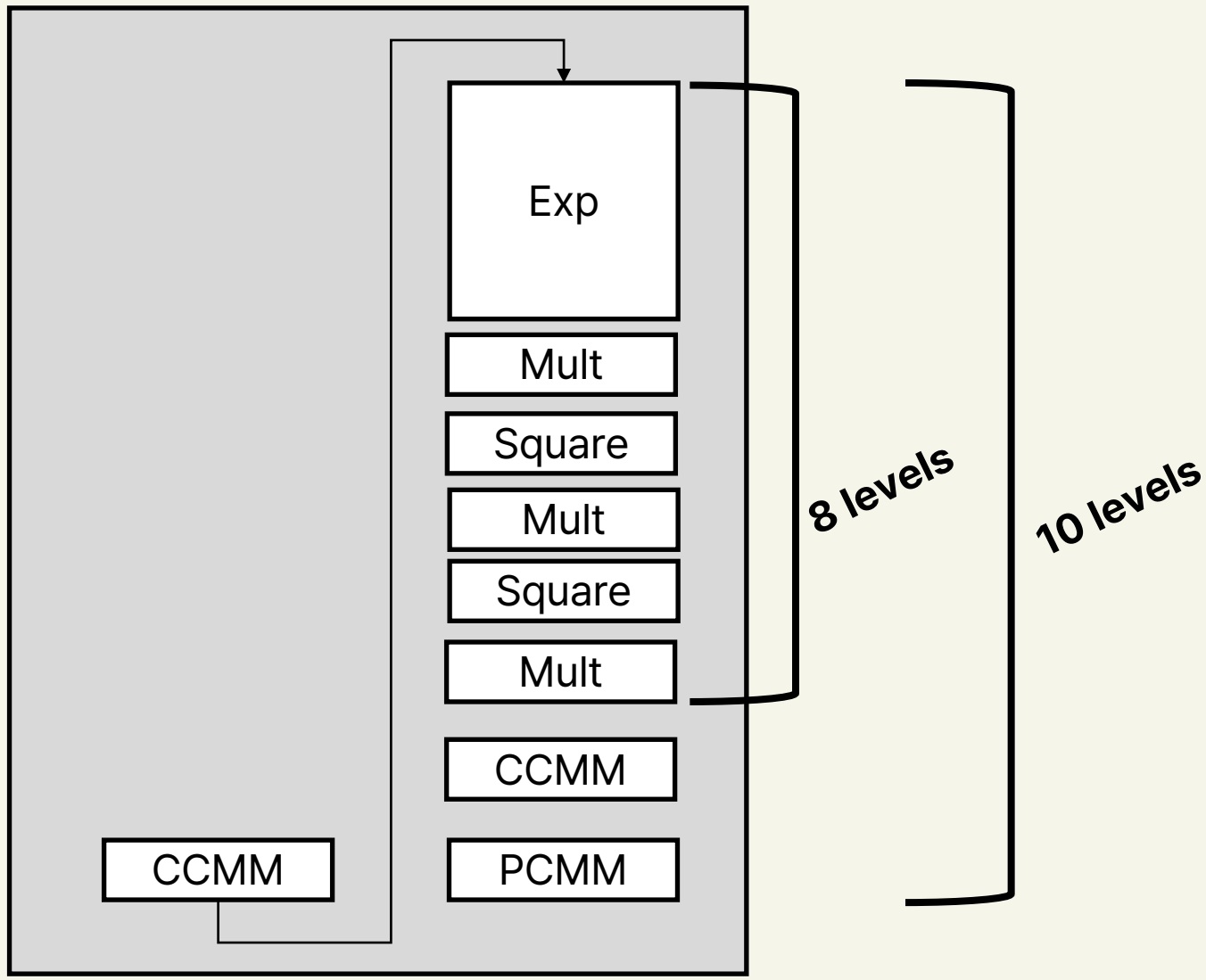
$\forall X \ \mu(X) = \mu'(X)$

Modified HELLM Computation

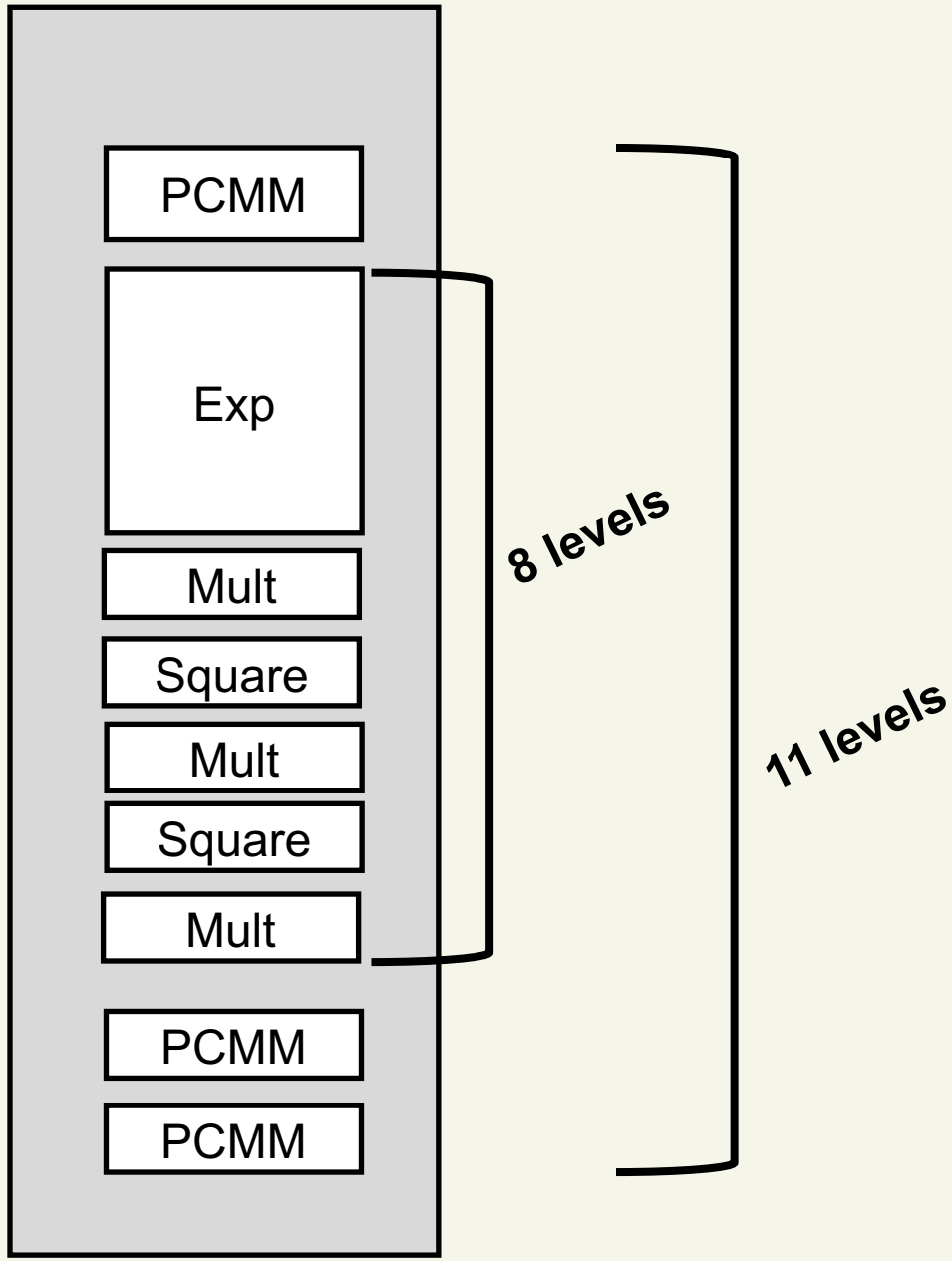


Level Plan in Attention

- New SoftMax
  - requires 8 levels
  - assumes intervals [-96,0]
  - exploits slim polynomial approximation method



Level Plan in Attention



Thank you!