(Long-) Context Modeling and Beyond

Zecheng Tang

OpenNLG Lab, SUDA

ICML 2025

Background

Context Modeling is the task of modeling the probability distribution of sequences

General format of sequence modeling with **Neural ODE Function**

$$\begin{cases} \dfrac{dh(t)}{dt} = f(h(t), t) \\ h(t) = h(t_0) + \displaystyle\int_{t_0}^{t_1} f(h(t), t, \theta)dt \end{cases} \Rightarrow p(x) = \int p(h_0)p(x|h(t;h_0))dh_0$$

Joint probability distribution

Discrete State Modeling

Autoregressive Context Modeling

$$P(x_{1:T}) = \prod_{t=1}^{T} p(x_t|x_{1:t-1})$$

Non-autoregressive Context Modeling

$$P(x_{1:T}|z) = \prod_{t=1}^{T} p(x_t|z)$$

$P(x_{1:T})$ contains semantics and structure information of discrete sequence

Autoregressive Context Modeling

$$P(x_{1:T}) = \prod_{t=1}^{T} p(x_t | x_{1:t-1})$$

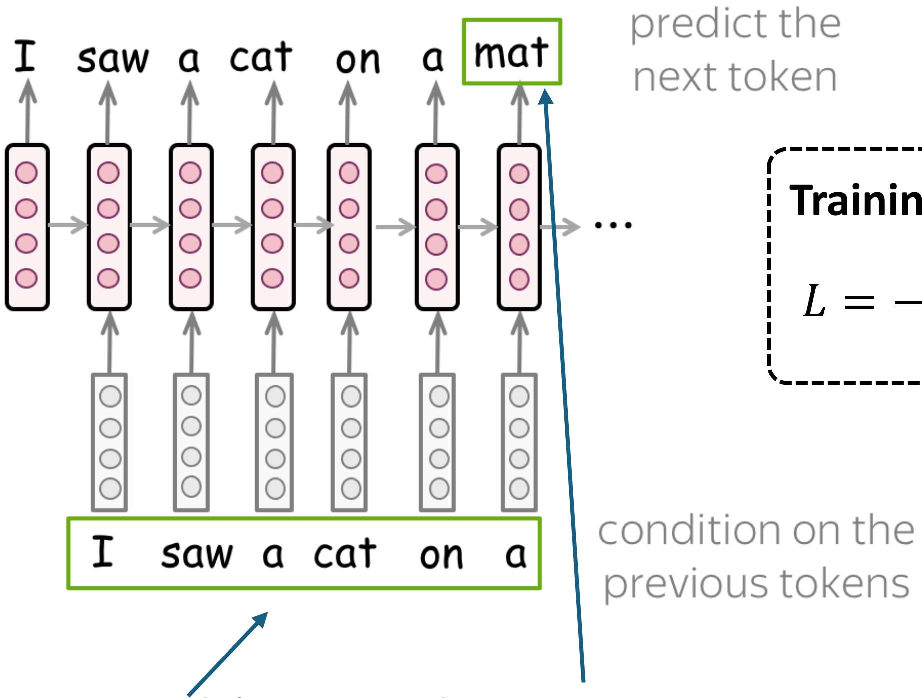
Next Token Prediction

$$P(x_t | x_{1:t-1}) = \text{Softmax}(Wh_t + b)$$

h_t is the context modeling results from neural network

Training Via Maximum Likelihood

$$L = -logP(x_{1:T}) = -\sum_{t=1}^{T} logP(x_t | x_{<t})$$



context modeling & prediction process

Context modeling is a fundamental capability of (Large) Language Models

Long-context Models are essential for AI development

Important scenarios in our daily life

- Book and document analysis
- Web content reading
- Code bases writing
- High-res images
- Audio recordings and Videos
- …

Some breakthrough moments in the AI field

- RL / Long-Cot
- Video Generation (World Model)
- Personal Agent
- ChatGPT moments (MCP, Model Context Protocol)
- …

<< MSTS Law >>

"Memory" More

✓ Parameter Scaling (?)

"See" More

✓ Long context modeling 🌟

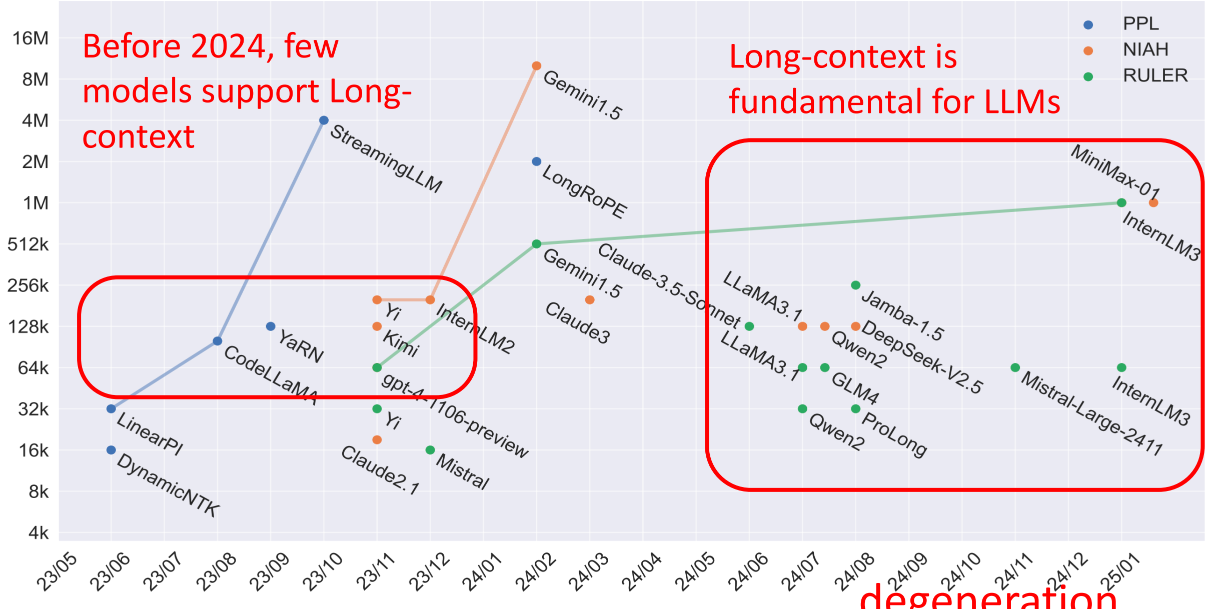
"Think" More

✓ Reasoning Model 🌟

"Speak" More

✓ Long-CoT / Multimodal 🌟

Yes,



degeneration

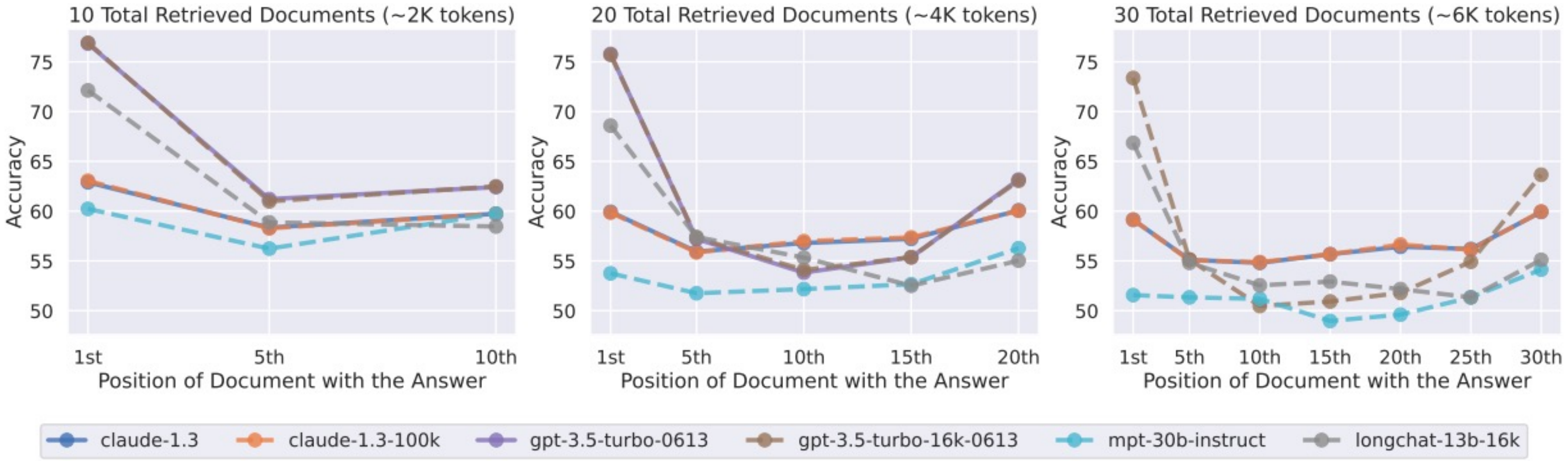
But,

Models	Claimed Length	Effective Length	4K	8K	16K	32K	64K	128K	Avg.
Llama2 (7B)	4K	-	85.6						
Gemini-1.5-Pro	1M	>128K	96.7	95.8	96.0	95.9	95.9	94.4	95.8
GPT-4	128K	64K	96.6	96.3	95.2	93.2	87.0	81.2	91.6
Llama3.1 (70B)	128K	64K	96.5	95.8	95.4	94.8	88.4	66.6	89.6
Qwen2 (72B)	128K	32K	96.9	96.1	94.9	94.1	79.8	53.7	85.9
Command-R-plus (104B)	128K	32K	95.6	95.2	94.2	92.0	84.3	63.1	87.4
GLM4 (9B)	1M	64K	94.7	92.8	92.1	89.9	86.7	83.1	89.9
Llama3.1 (8B)	128K	32K	95.5	93.8	91.6	87.4	84.7	77.0	88.3
GradientAI/Llama3 (70B)	1M	16K	95.1	94.4	90.8	85.4	80.9	72.1	86.5
Mixtral-8x22B (39B/141B)	64K	32K	95.6	94.9	93.4	90.9	84.7	31.7	81.9
Yi (34B)	200K	32K	93.3	92.2	91.3	87.5	83.2	77.3	87.5
Phi3-medium (14B)	128K	32K	93.3	93.2	91.1	86.8	78.6	46.1	81.5
Mistral-v0.2 (7B)	32K	16K	93.6	91.2	87.2	75.4	49.0	13.8	68.4
LWM (7B)	1M	<4K	82.3	78.4	73.7	69.1	68.1	65.0	72.8
DBRX (36B/132B)	32K	8K	95.1	93.8	83.6	63.1	2.4	0.0	56.3
Together (7B)	32K	4K	88.2	81.1	69.4	63.0	0.0	0.0	50.3
LongChat (7B)	32K	<4K	84.7	79.9	70.8	59.3	0.0	0.0	49.1
LongAlpaca (13B)	32K	<4K	60.6	57.0	56.6	43.6	0.0	0.0	36.3

Background:

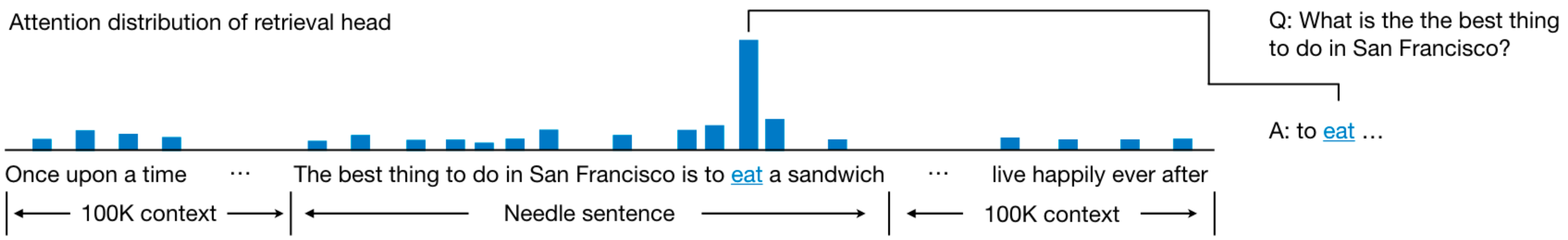
Long-context Modeling

Phenomena: Lost In the Middle of LLMs [Liu N. F. et al., 2023]



- Information occurs at the very start or end of the context: ***Highest***
- Information in the middle: ***Rapidly Degrade***

Theory I: Retrieval Head Explains Long-Context Factuality [Wu. et al, 2024]

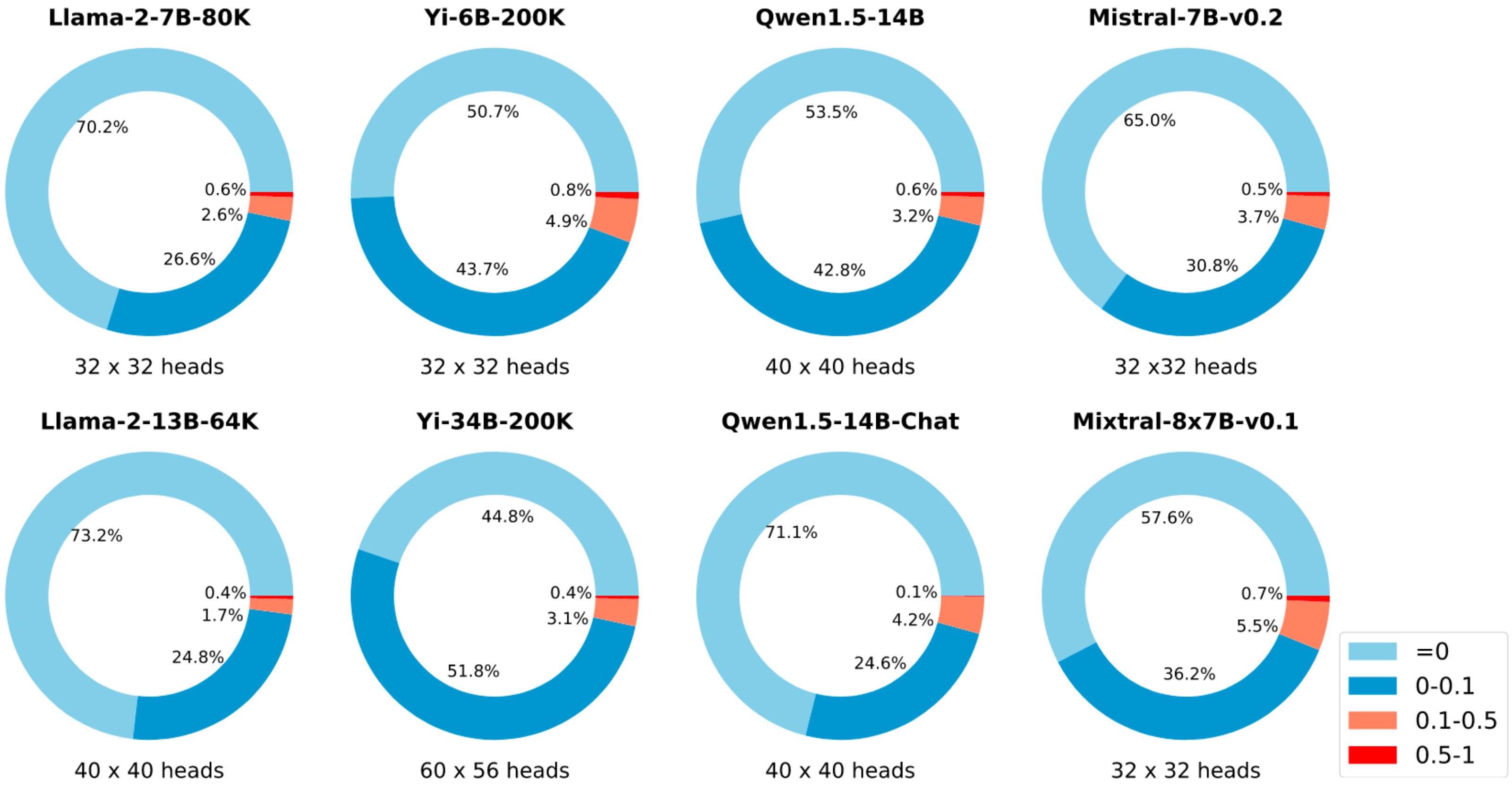


Retrieval Score Measure how often a head performs copy-paste from the input (needle) to the output

- During decoding:
  - Let w be the token being generated
  - Let $a \in R^{|x|}$ be the attention scores for a head
- A head is considered to copy-paste w if
  - $w \in k$ → w is in the needle sentence
  - $x_j = w, j = \text{argmax}(a), j \in i$ → the most attended input token matches w and is from the needle
- Retrieval Score $|g_h|$ → set of tokens copied by head h

$$h = \frac{|g_h \cap k|}{|k|}$$

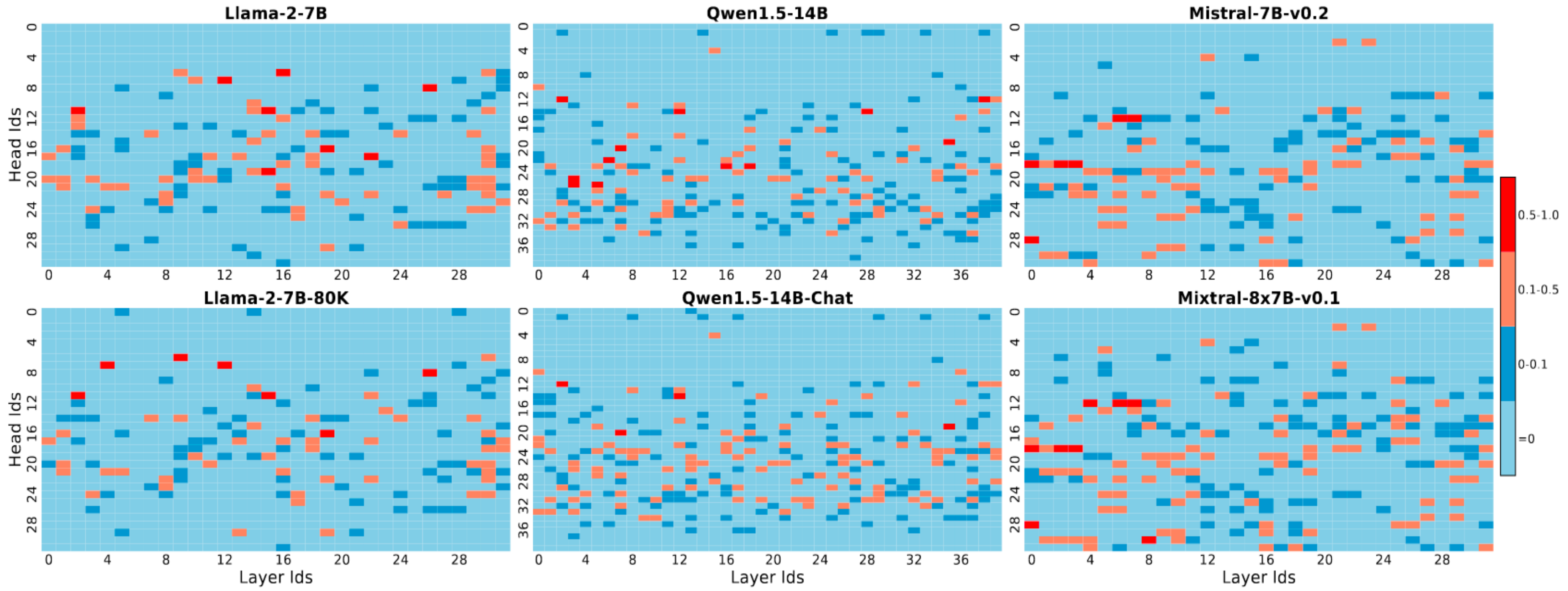
Theory I: Retrieval Head Mechanistically Explains Long-Context Factuality



Retrieval heads are universal and sparse across model family and scale.

- less than 5% of the attention heads are activated more than 50% of the time (with a retrieval score higher than 0.5) when retrieval is required

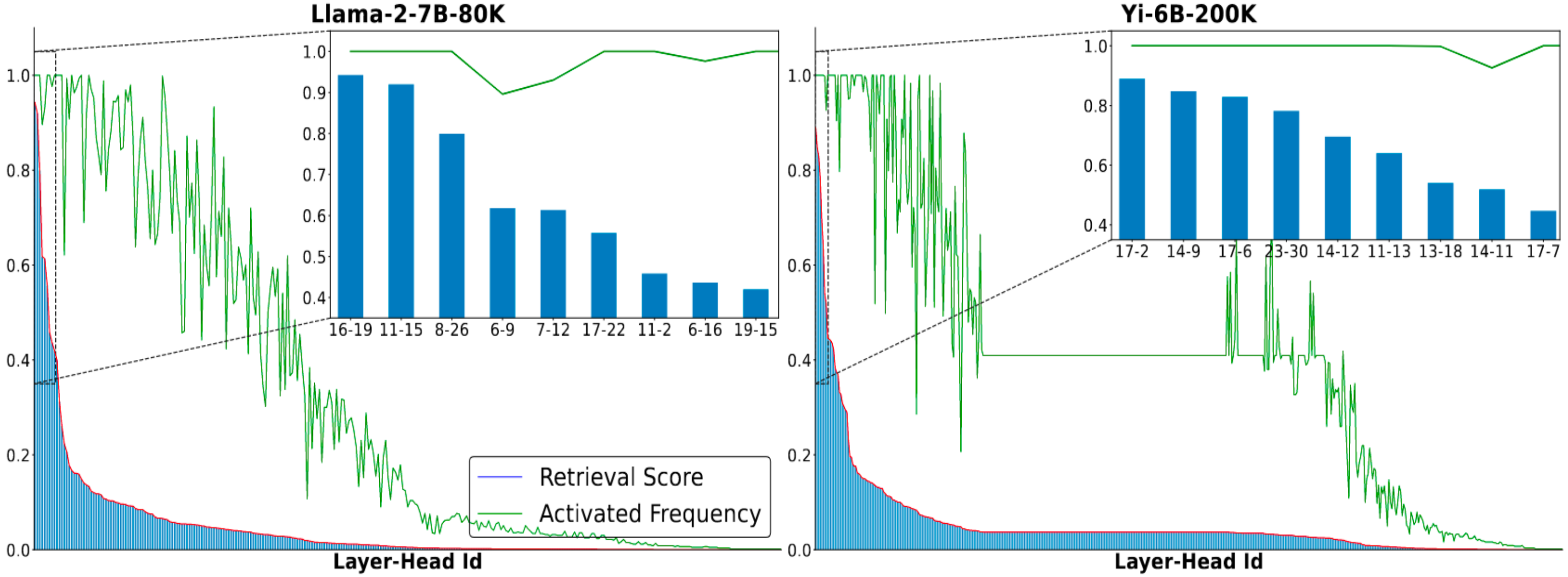
Theory I: Retrieval Head Mechanistically Explains Long-Context Factuality



Most retrieval heads are concentrated in the upper-middle layers

- Sparse distribution
- Lower layers focus on local feature extraction
- Upper layers perform information aggregation

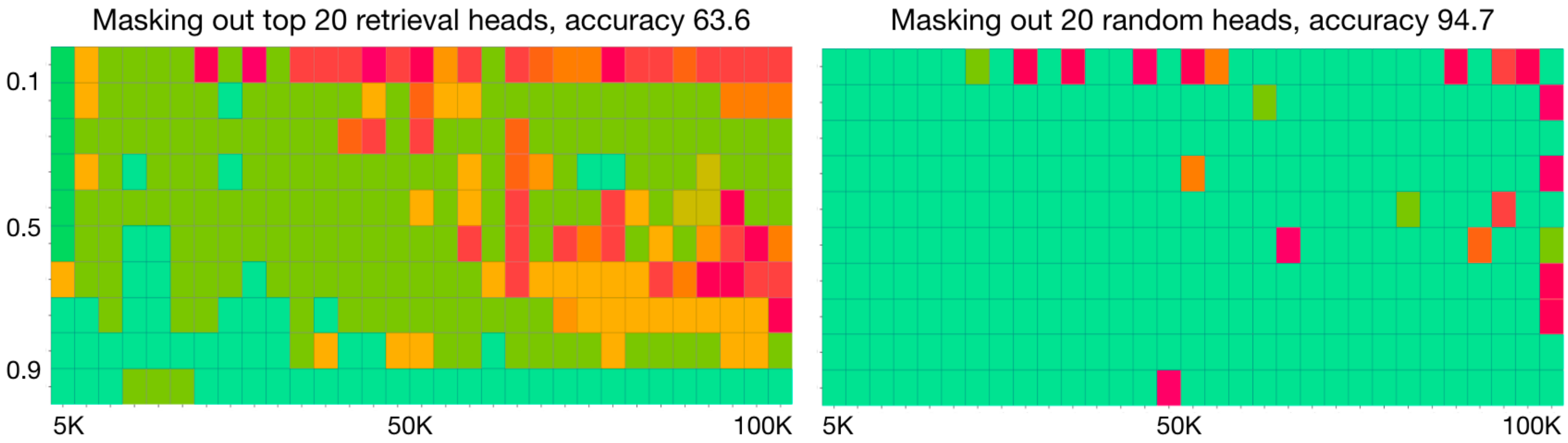
Theory I: Retrieval Head Mechanistically Explains Long-Context Factuality



AF: Activation Frequency RS: Retrieval Score

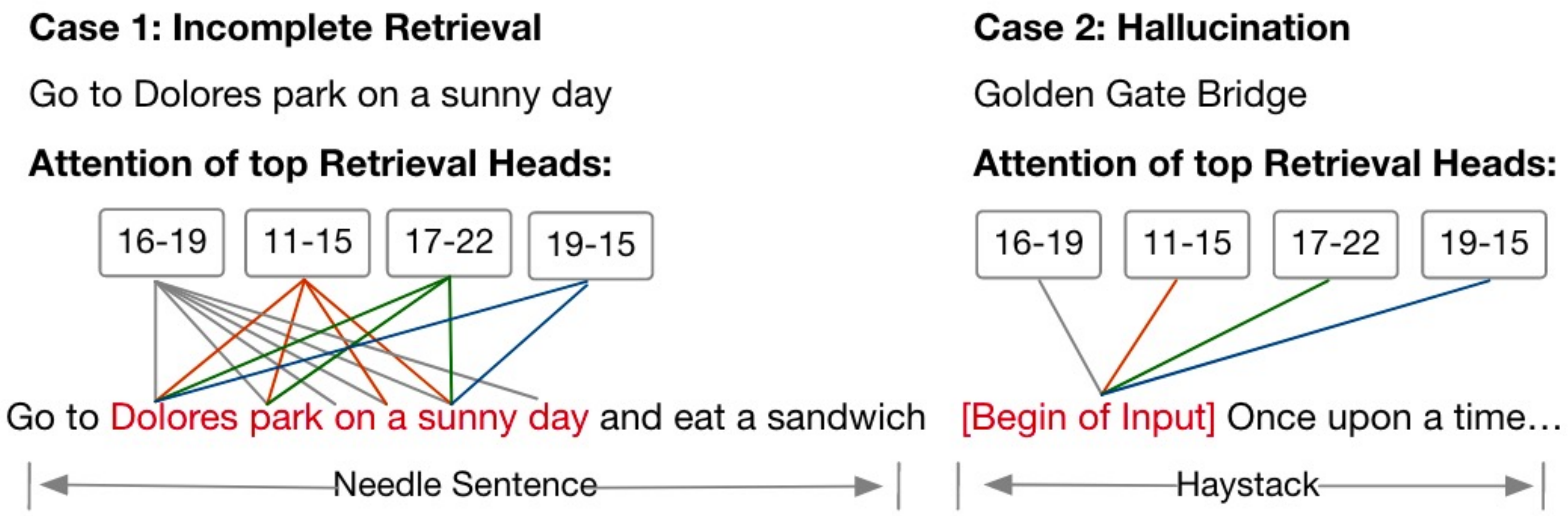
- Head of high AF and RS → Retrieval Head
- Head of high AF but low RS → Bias Head: Activated on certain tokens
- Head of low AF and RS → Useless Head (*can be pruned for compression*)

Theory I: Retrieval Head Mechanistically Explains Long-Context Factuality



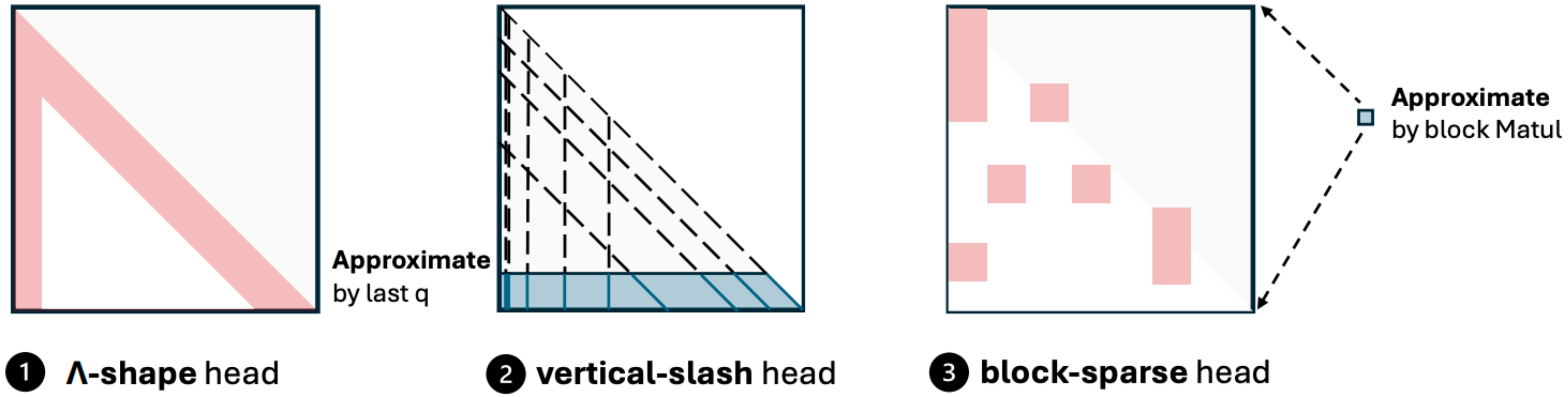
- ➢ Masking out the top retrieval heads, performance drops significantly, and the model hallucinates during decoding.
- ➢ Masking out random non-retrieval heads does not influence the model's retrieval behavior.

Theory I: Retrieval Head Mechanistically Explains Long-Context Factuality



- **Incomplete Retrieval**: The retrieval heads fail to capture partial information (e.g., *"eat a sandwich"*).
- **Hallucination**: The retrieval heads incorrectly attend to initial tokens (attention sink).

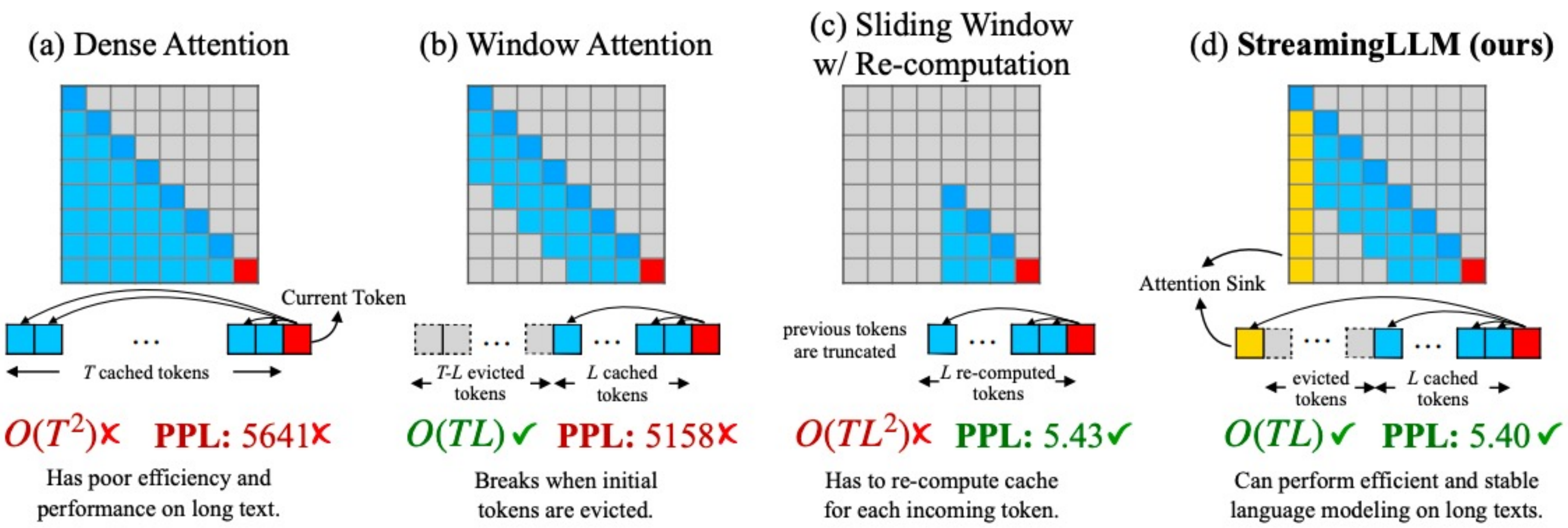
Theory II: Three Attention Patterns Exist in LLMs



Minference 1.0 [Jiang et al. 2024] summaries three attention patterns in LLMs

- **A-Shape Pattern**
  - Focus on **initial tokens** and **local windows**
  - Exhibits relatively **higher stability** compared to other patterns.
- **Vertical-Slash (VS) Pattern**
  - **Specific tokens** (*vertical lines*)
  - **Fixed-interval tokens** (*slash lines*).
- **Block-Sparse Pattern**
  - **Dynamic and dispersed** distribution.
  - **Spatial clustering** (concentrate near top-K neighbors).

Apply Attention Patterns For Better (Long-)context Modeling



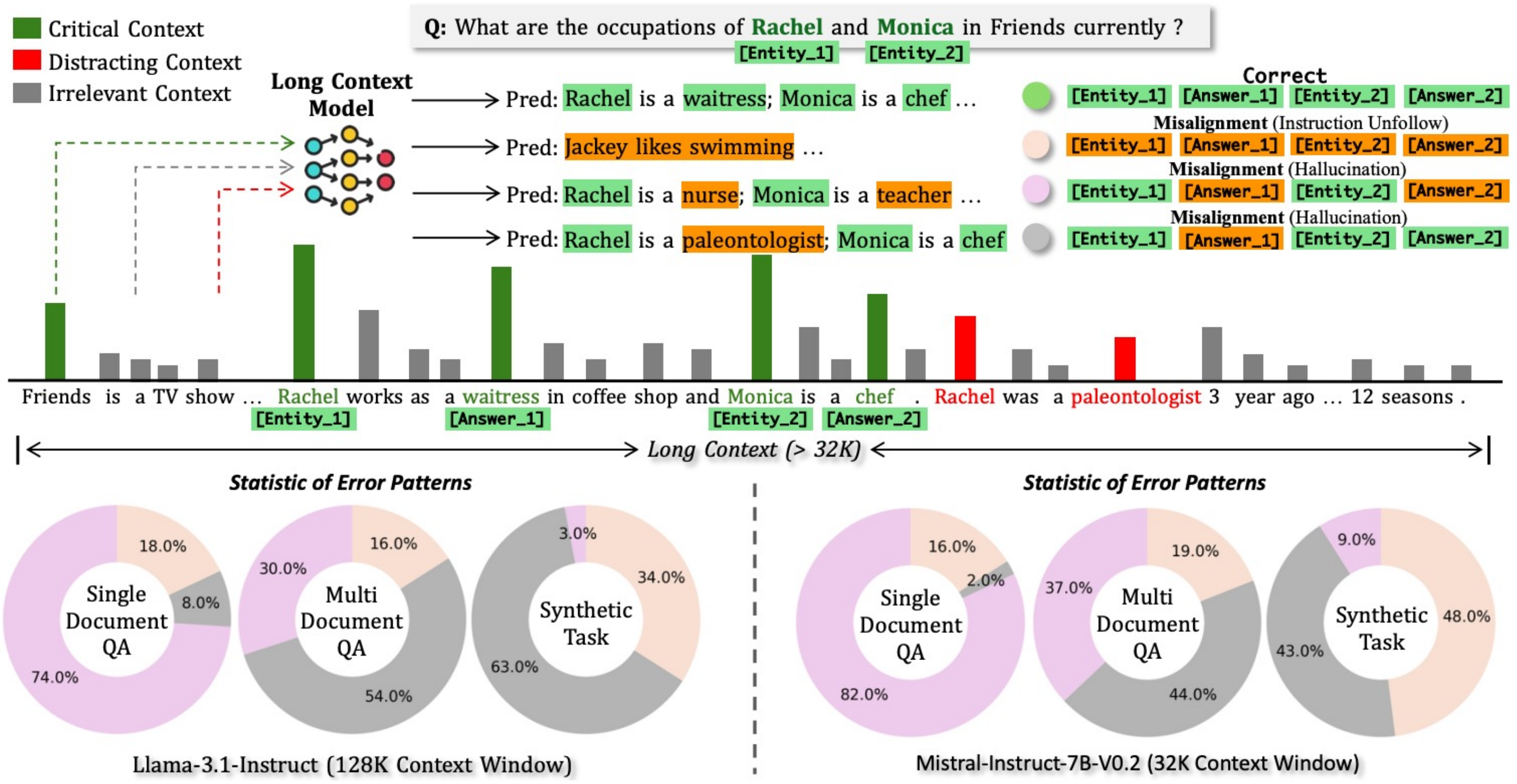
StreamingLLM [Xiao et al., 2023]

- **Attention Sink**: Retains initial tokens (as "sinks") to stabilize attention computation.
- **Recent Tokens**: Combines sinks with the most recent tokens for efficient context processing.
- Computationally efficient for streaming/extended text generation.

Issue : Imbalanced Modeling and Generation

- Precise information retrieval
- Deficient generation capability

Issue 1: Imbalanced Context Modeling and Generation



- **Good** retrieval capability and low "PPL" score
- **Poor** downstream task performance, e.g., reasoning

Modeling Approach: Preference-Optimized Context Modeling

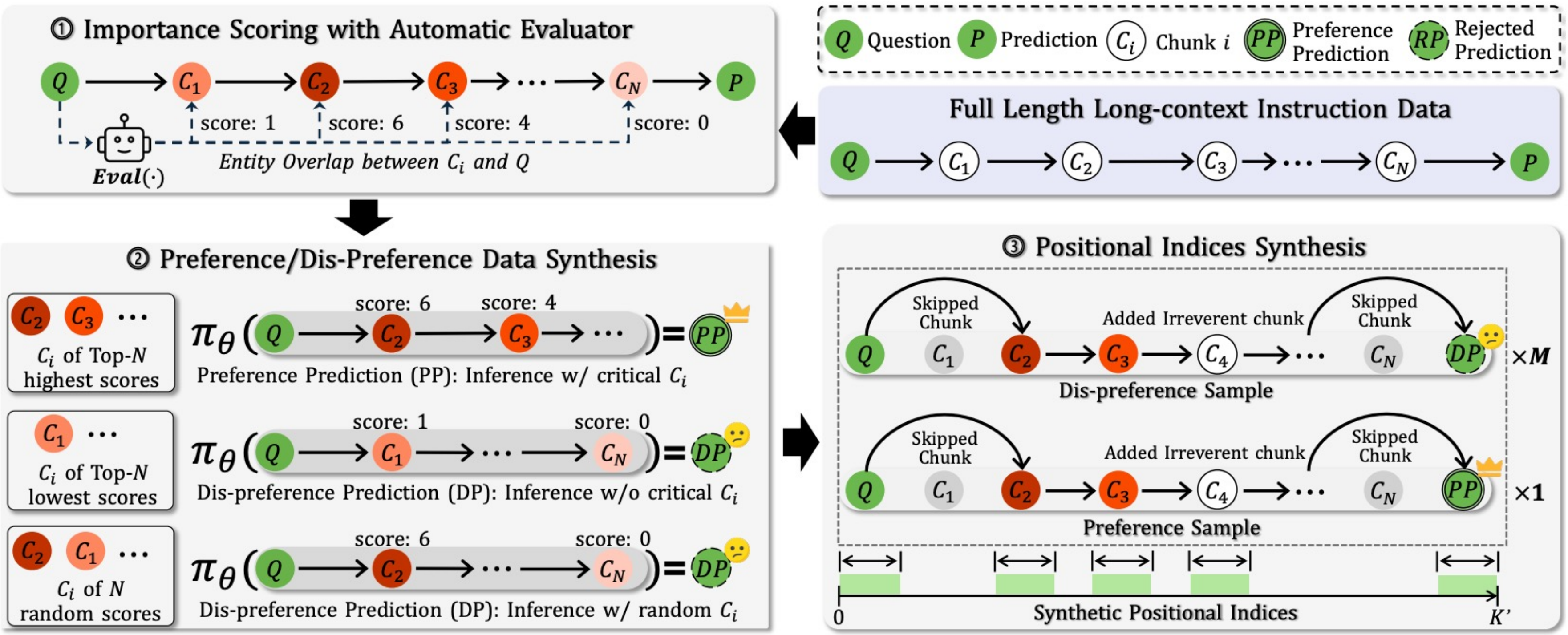
LOGO -- Long cOntext aliGnment via efficient preference Optimization [Tang et al., 2024]

$$\mathcal{L}_{\text{LOGO}}(\pi_\theta) = -\mathbb{E}_{(x, y_w, y_l^{(1\cdots M)})}\left[\log\sigma\left(\underbrace{\frac{\beta}{|y_w|}\log\pi_\theta(y_w|x)}_{\text{Win Response}} - \underbrace{\frac{\beta}{M|y_l|}\sum_{j=1}^{M}\log\pi_\theta(y_l^{(j)}|x)}_{\text{Lose Response}} - \gamma\right)\right]$$

Motivation: activate the model's capability to ***effectively utilize captured critical information for prediction*** through preference optimization.

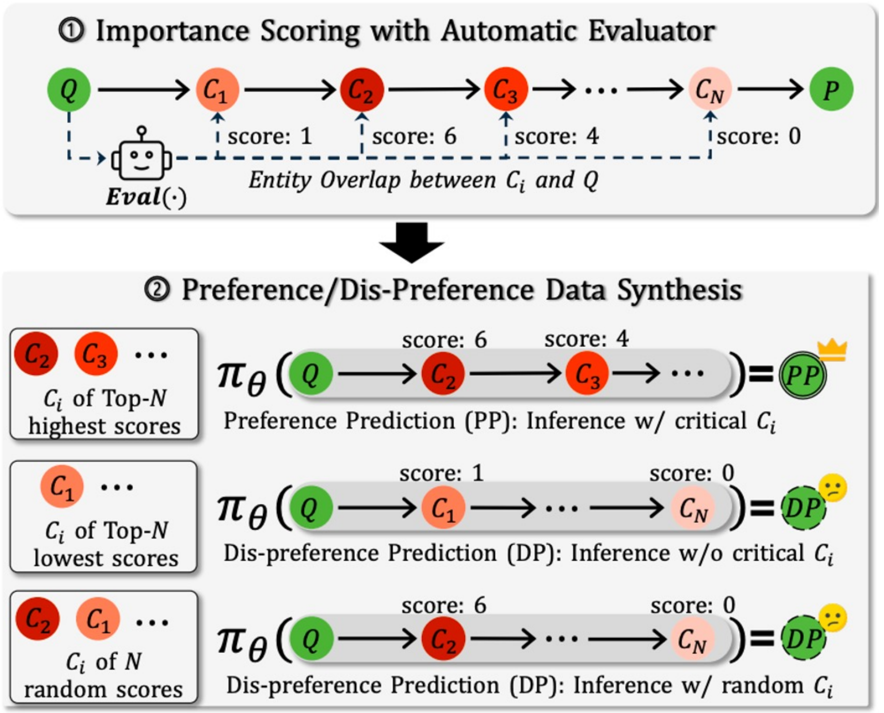
- ***Challenge 1***: Hard to distinguish win and lose respond
- ***Challenge 2***: Expensive to train with long-context RL

Observation I: Model response varies with the density of critical information



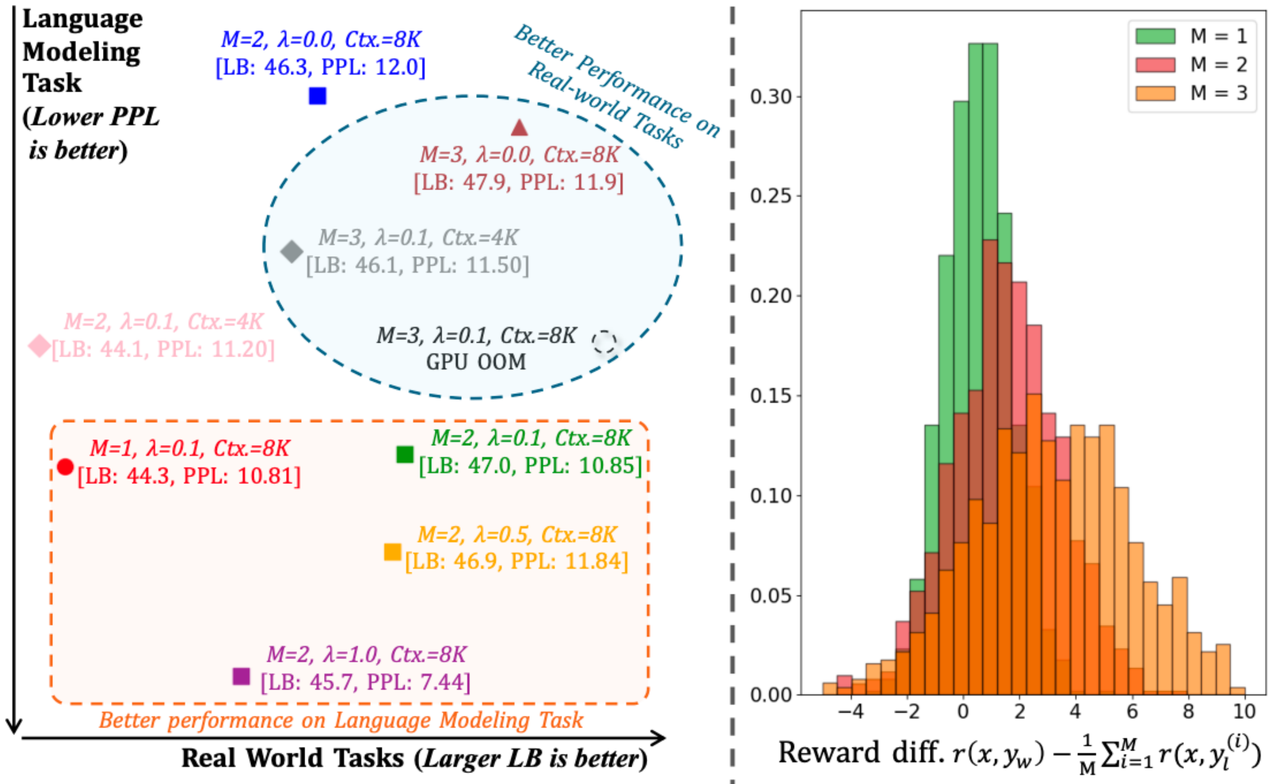
- ***High Information-Density Contexts***
  - Responses exhibit high correctness probability
  - (Model effectively leverages concentrated key information)
- ***Low Information-Density Contexts***
  - Responses show lower correctness probability
  - (Performance degrades due to sparse/noisy signal)

Method: Synthesizing preference pairs with reverse generation



- **Stage 1: Context Filtering**
  - Locate salient chunks with *Entity Overlap Score*
- **Stage 2: Reverse Generation**
  - Generate response based on *filtered context*
    - ✓ Win response: All salient chunks
    - ✓ Lose response: Partial / No salient chunks

Observation II: Scaling Rejection Perception Field



(a) Language Modeling and Real-world Tasks

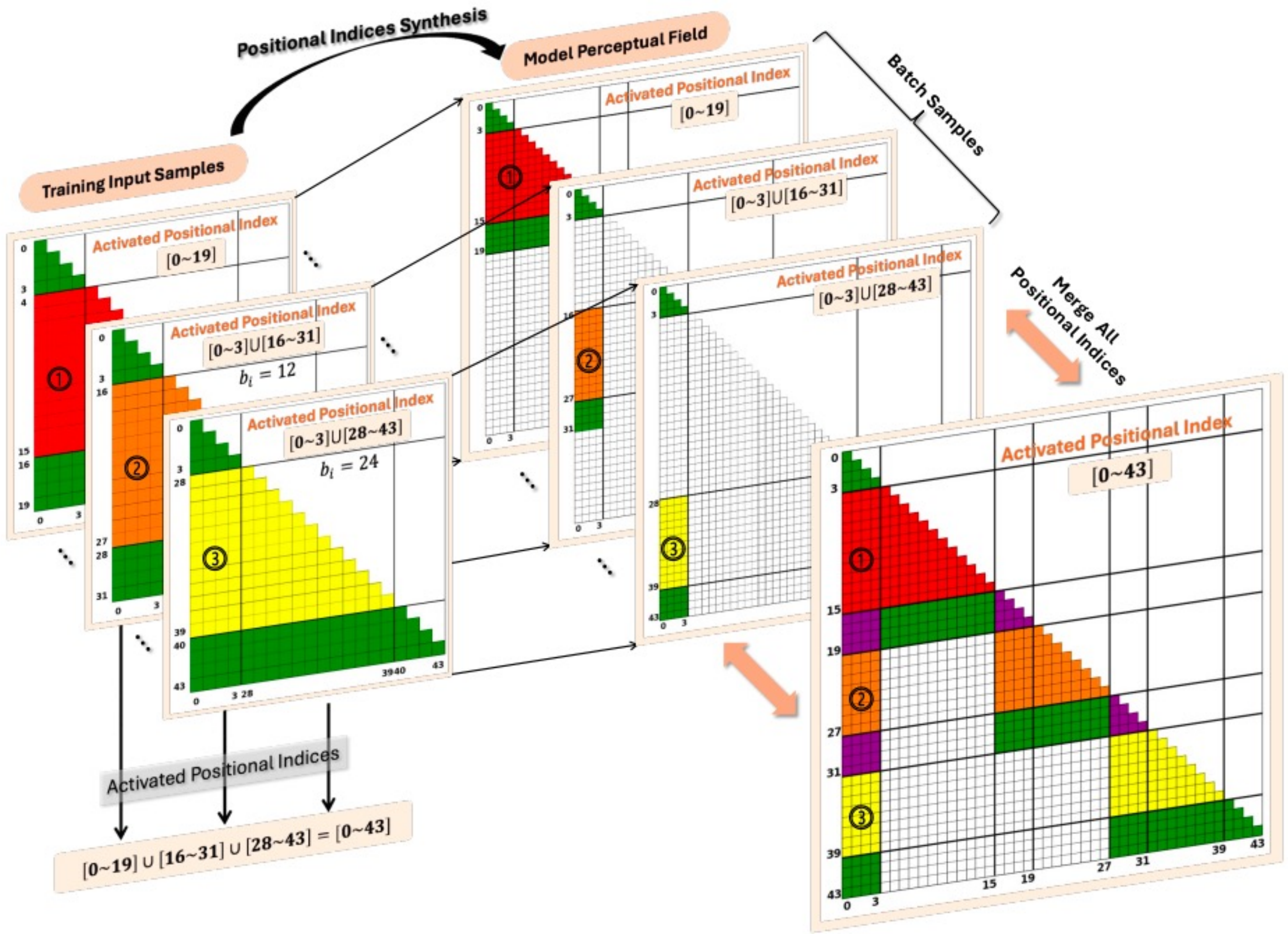
(b) Reward diff. distribution

Beter performance with larger rejection perception field

$$\mathcal{L}_{\text{LOGO}}(\pi_\theta) = -\mathbb{E}_{(x, y_w, y_l^{(1\cdots M)})}\left[\log\sigma\left(\underbrace{\frac{\beta}{|y_w|}\log\pi_\theta(y_w|x)}_{\text{Win Response}} - \underbrace{\frac{\beta}{M|y_l|}\sum_{j=1}^{M}\log\pi_\theta(y_l^{(j)}|x)}_{\text{Scaled Lose Response}} - \gamma\right)\right]$$

Method: Positional Index Synthesis can relieve the training burden

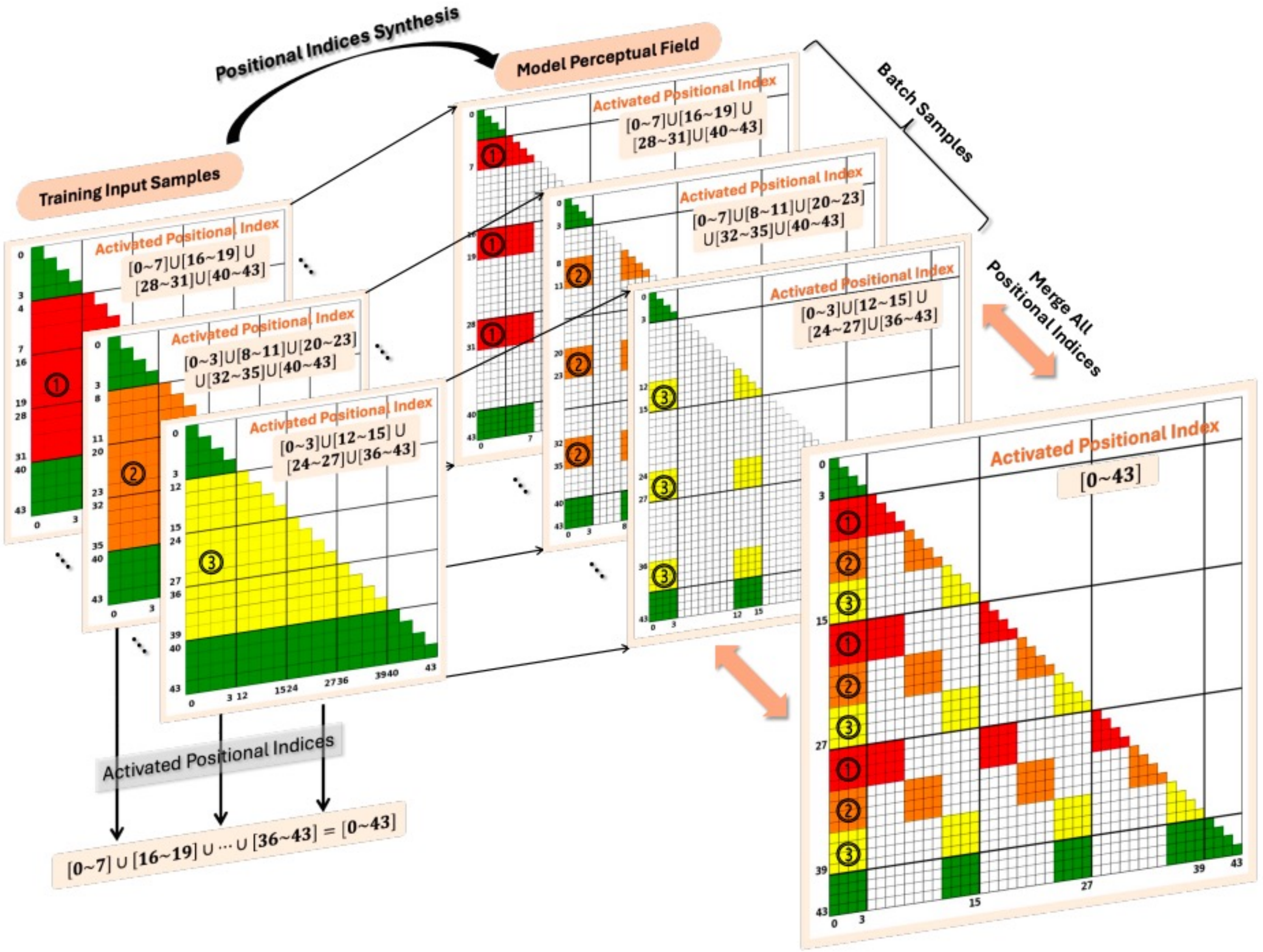
Context Sparse → Positional Index Sparse



A-Shape Pattern + Vertical-Slash (VS) Pattern

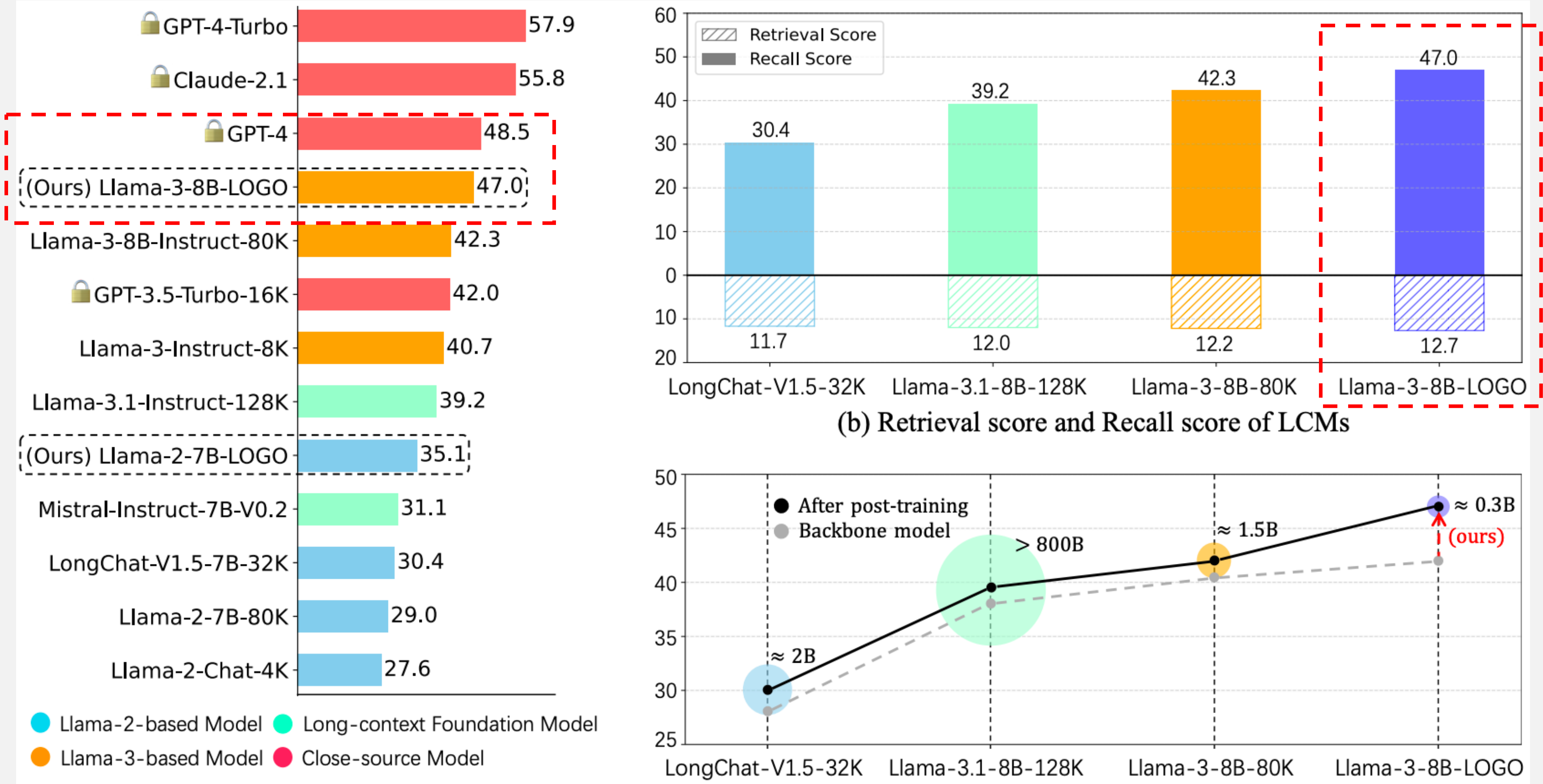
Method: Positional Index Synthesis can relieve the training burden

Context Sparse → Positional Index Sparse



A-Shape Pattern + Block-Sparse Pattern

Result I: 8B model achieves comparable results with GPT-4



(a) Performance on real-world long-context tasks

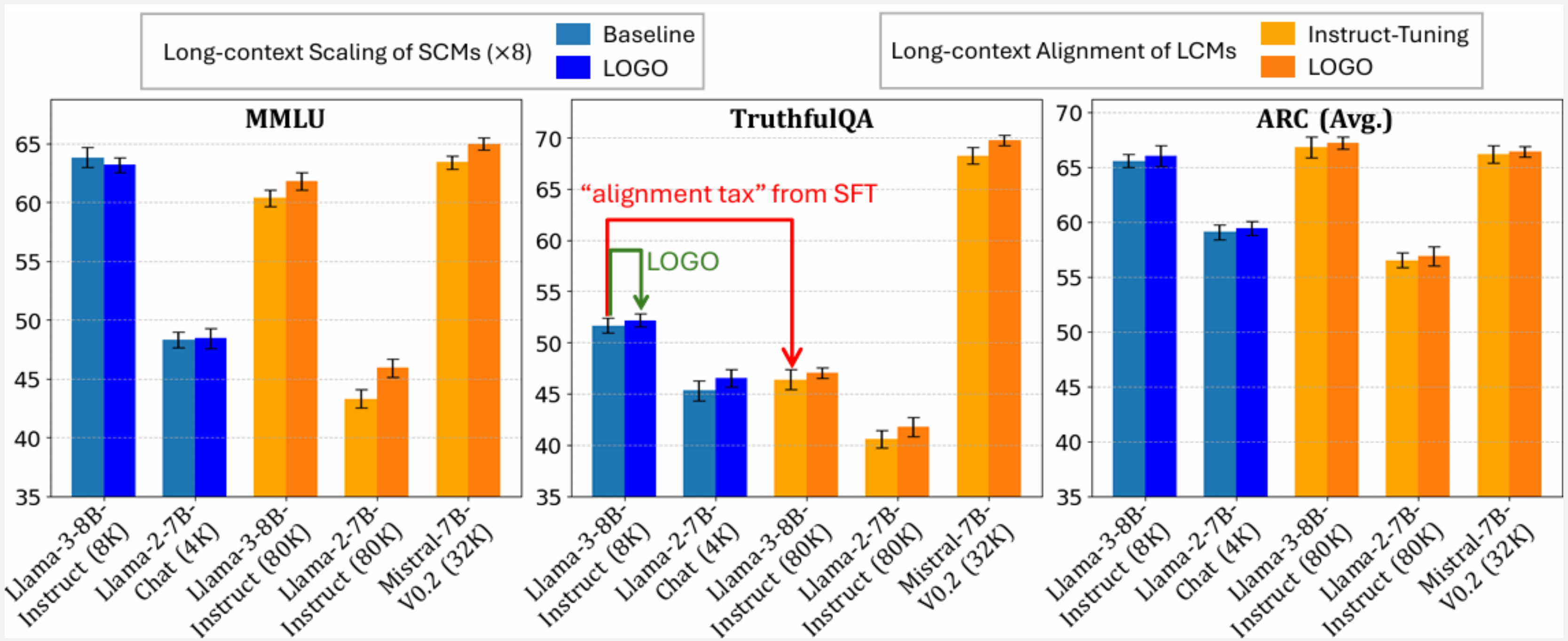
(b) Retrieval score and Recall score of LCMs

(c) Model Performance V.S. Long-context Training Data Size

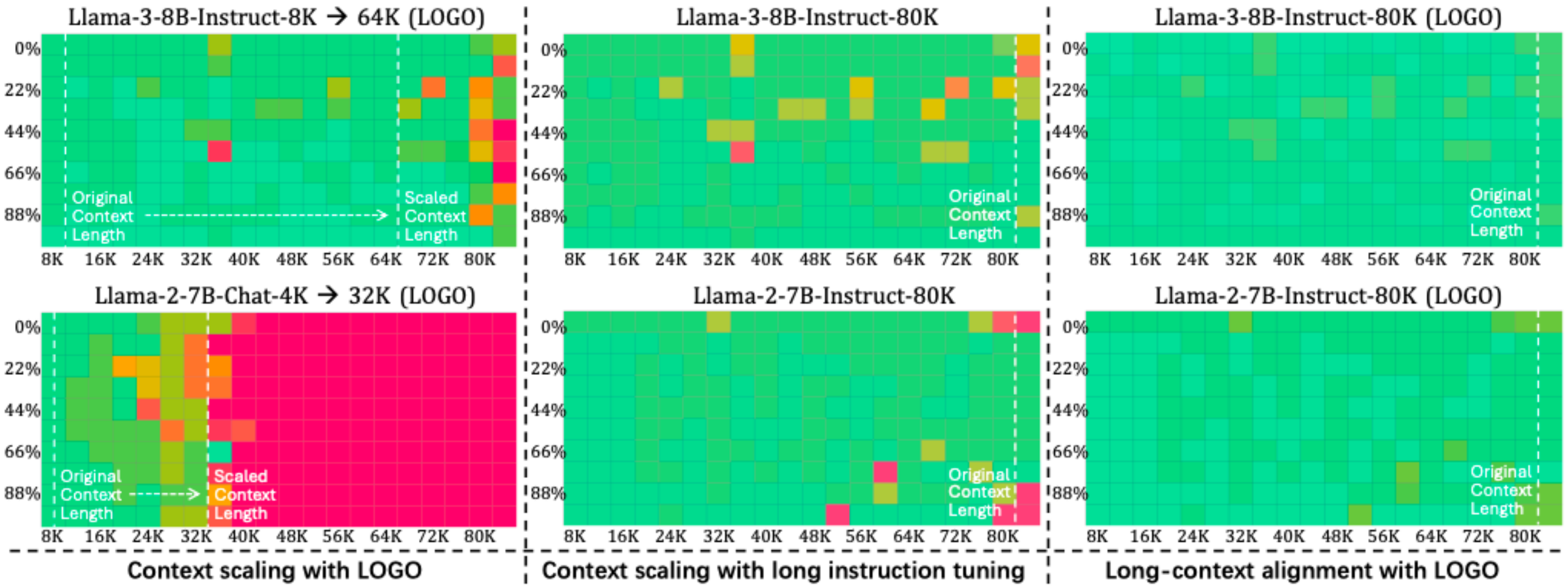
Result I: LOGO can generalize to all Long-context training settings

Models	Type	S-Doc QA	M-Doc QA	Summ	Few-shot	Synthetic	Avg.
GPT-3.5-Turbo-16K	-	39.8	38.7	26.5	67.1	37.8	42.0
GPT-4	-	45.1	55.0	28.3	72.3	41.8	48.5
LongChat-v1.5-7B-32k	-	28.7	20.6	26.7	60.0	15.8	30.4
LLama-3.1-8B-Instruct-128K	-	23.9	15.8	28.9	69.8	57.5	39.2
Results on SCMs *(scaling ×8 context window)*							
Llama-3-8B-Instruct-8K	-	39.3	36.2	24.8	63.5	39.9	40.7
+ YaRN-64K (Peng et al., 2023b)	Free	38.0	36.6	27.4	61.7	40.9	40.9
+ PoSE-64K (Zhu et al., 2023)	SFT	34.9	31.4	18.7	59.3	44.2	37.7
+ LOGO-64K	DPO	**39.8**	**36.7**	**28.8**	**65.4**	**49.0**	**43.9**
Llama-2-7B-Chat-4K	-	24.9	22.6	24.7	60.0	5.9	27.6
+ Data-Engineering-80K (Fu et al., 2024)	SFT	**26.9**	**23.8**	21.3	**65.0**	7.9	29.0
+ LOGO-32K	DPO	26.7	23.3	**26.3**	63.1	**11.1**	**30.1**
Results on LCMs *(preserving original context window)*							
Llama-3-8B-Instruct-80K	-	43.0	39.8	22.2	64.3	46.3	42.3
+ LongLoRA (Chen et al., 2023b)	SFT	39.3	36.2	26.8	63.5	48.0	42.8
+ SimPO (Meng et al., 2024)	DPO	43.2	40.7	23.5	66.7	48.4	44.5
+ LOGO-80K	DPO	**44.0**	**41.2**	**28.1**	**68.6**	**53.0**	**47.0**
Llama-2-7B-64K	-	28.3	33.2	13.4	62.3	6.1	28.7
+ LongAlign (Bai et al., 2024)	SFT	29.9	32.7	26.5	63.8	16.5	33.9
+ LOGO-64K	DPO	**33.6**	**28.0**	**29.4**	**65.1**	**24.5**	**36.1**
Mistral-Instruct-7B-V0.2-32K	-	31.7	30.6	16.7	58.4	17.9	31.1
+ FILM-32K (An et al., 2024)	SFT	37.9	34.9	25.3	64.7	31.2	38.8
+ LOGO-32K	DPO	**38.3**	**37.6**	**26.1**	**67.0**	**31.5**	**40.1**

Result II: Preserve performance on short-context tasks



Result III: Stress testing on long-context synthesis tasks



Pass all NIAH testing from 8K → 96K context length