TEAM ONE

Paper Review

BEYOND ATTENTION: BREAKING THE LIMITS OF TRANSFORMER CONTEXT LENGTH WITH RECURRENT MEMORY ()

DON'T DO RAG: WHEN CACHE-AUGMENTED GENERATION IS ALL YOU NEED FOR KNOWLEDGE TASKS

NCCU Computer Science
Senior Student
110703007 Jui-Hung Cheng
110703013 Yu-Chih Pan

TEAM ONE

Paper Topic

Beyond Attention:
Breaking the Limits of Transformer Context Length with Recurrent Memory
(AAAI 2025)

TEAM ONE

Task & Solution

- Task presented
  - Addressing the transformer limitation of handling long sequences due to quadratic computational complexity.
- Proposed Solution
  - Recurrent Memory Transformer (RMT) with token-based memory augmentation.

TEAM ONE

Why Is It Important?

- Enhance sequences from 128k tokens to 2 million tokens.
- Facilitates long-term dependency tasks in natural language understanding and generation.
- Improves scalability for memory-intensive applications.

TEAM ONE

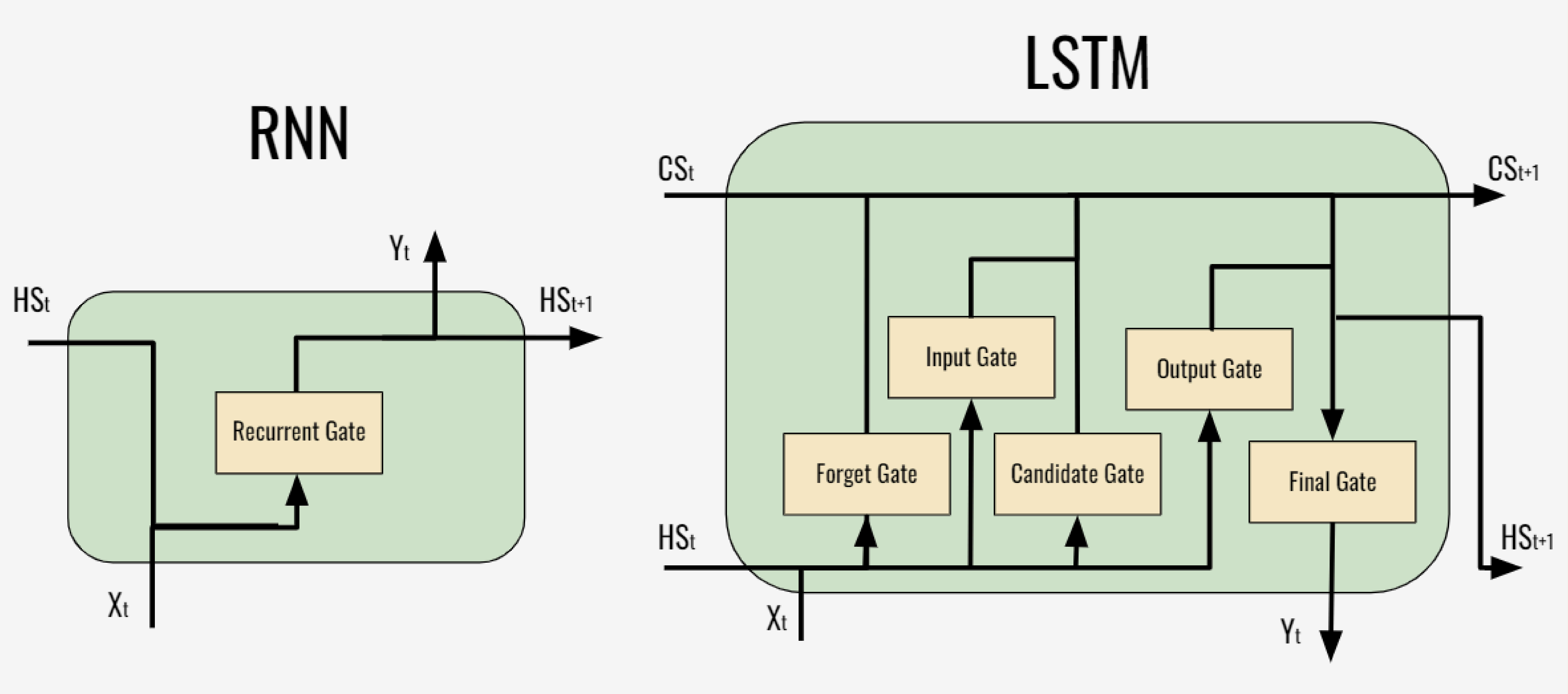
Prior Approaches

- **Memory-augmented neural networks**

(e.g., NTMs, LSTMs, Memory Networks).

- **Modified transformers like Transformer-XL, Compressive Transformer, and Big Bird.**
- **Attention mechanism redesigns to reduce computational complexity.**

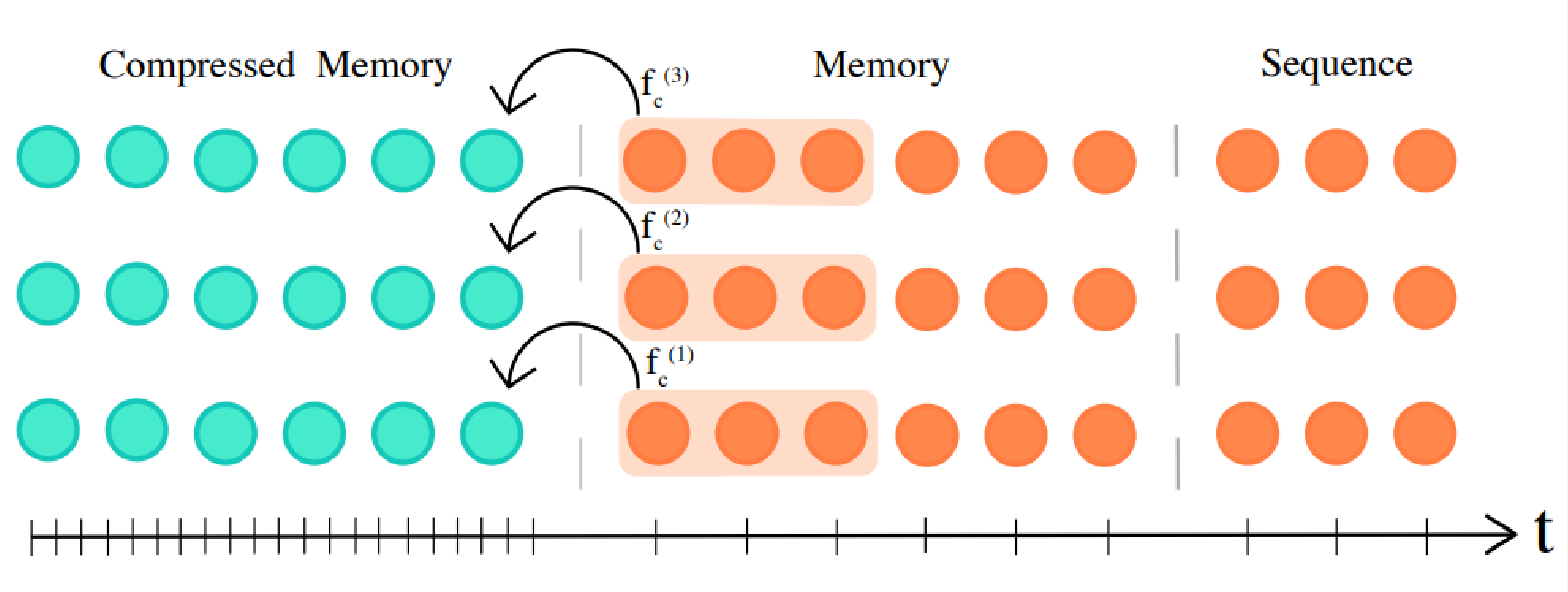
Prior Approaches -- LSTM



MEDIUM: RNN VS LSTM

TEAM ONE

Prior Approac



COMPRESSIVE TRANSFORMERS FOR LONG-RANGE SEQUENCE MODELLING

Limitations of Past Mehods

- Require significant architectural changes.

=> Hard to implement to other existing models

- Memory scaling is still limited by hardware constraints.

TEAM ONE

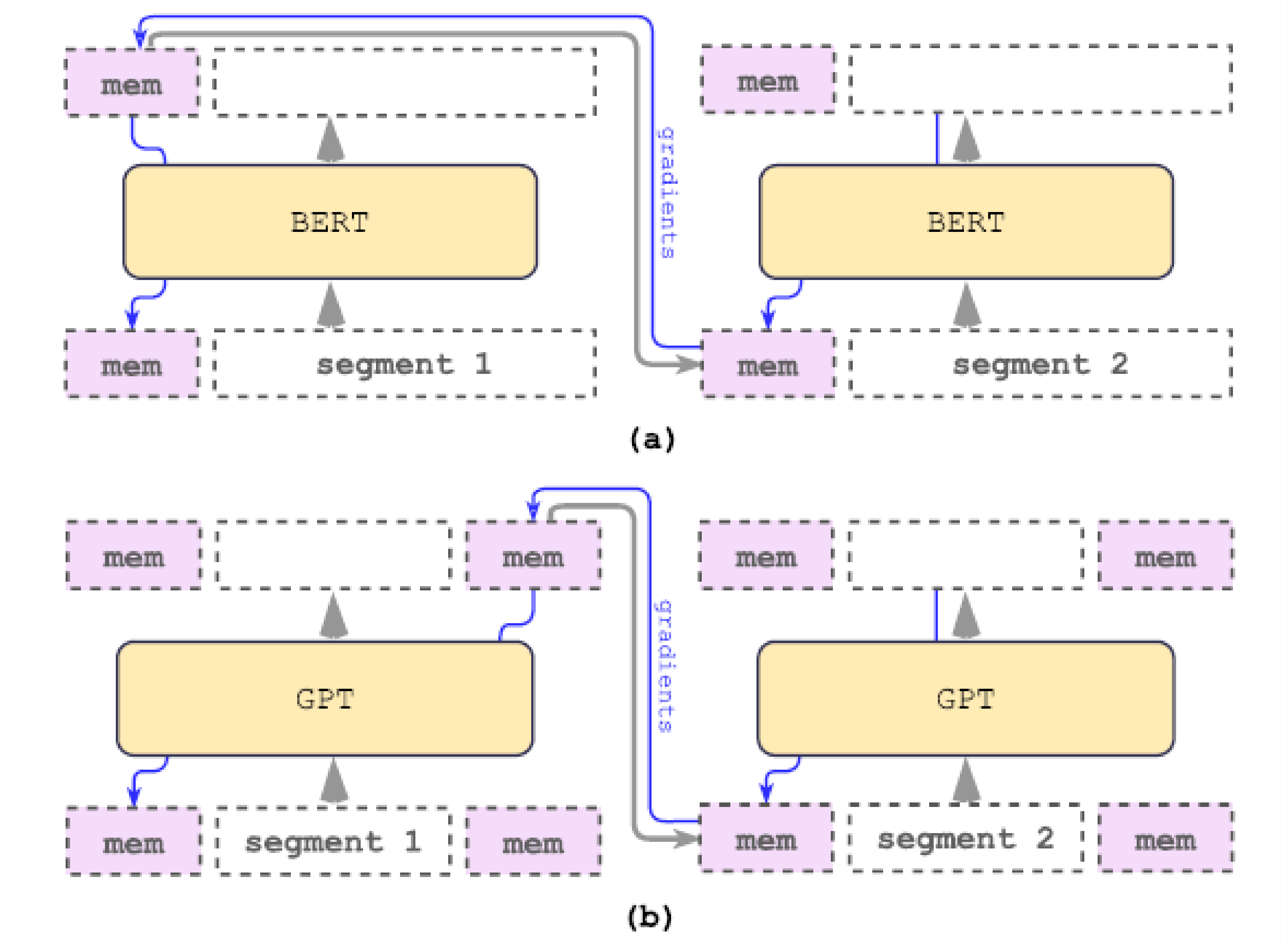
Connection to RMT

- Avoid architectural modifications
- Seamlessly integrate with pre-trained models
- Achieve linear computational scaling
- Solving memory and computational bottlenecks.

Method: Design

- Uses “memory tokens” prepended to input or output sequences.
- Processes long sequences by segmenting inputs and passing memory recurrently.
- Implemented as a plug-and-play wrapper for transformers like BERT and GPT-2.

Recurrent memory mechanism

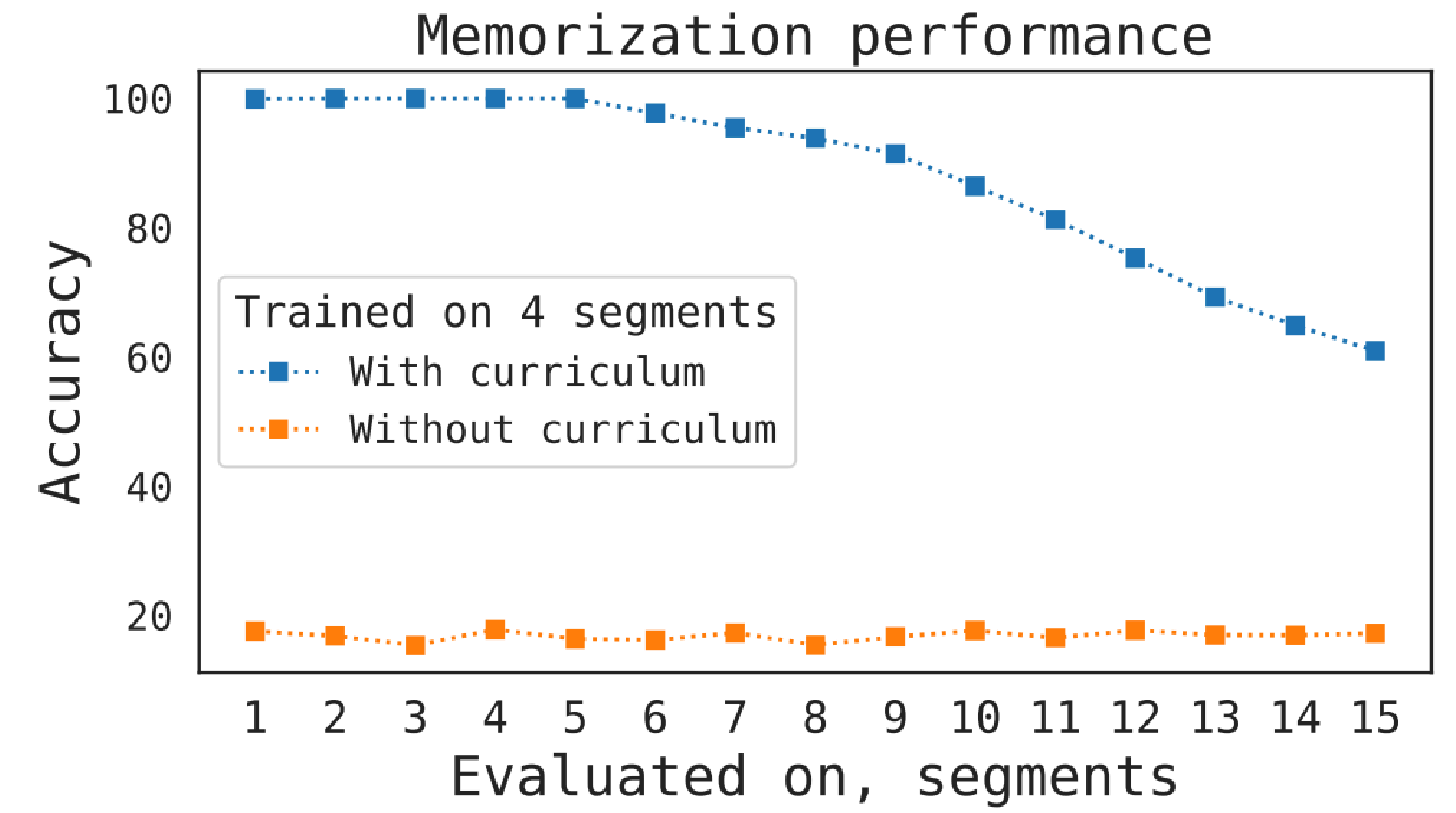


TEAM ONE

Curriculum Learning

Optimize RMT by curriculum learning

- Starts with short sequences and progressively increases input length.
- Focuses on stable fine-tuning of pre-trained models, eg: BERT, GPT2.



Advantages & Significance

- Simplifies integration with existing pre-trained models and close source LLM.
- Scaling the input lengths without sacrificing computational efficiency.
- Even short sequence model can apply this method and effectively adapted tasks involves longer sequences

TEAM ONE

Experiment & Paper

Exploring the value of long context in the Era of RAG

(Don't Do RAG: When Cache-Augmented Generation is All You Need for Knowledge Tasks)

Problem Statement

Why is this experiment important?

1. Question Answering Task in specific field.
2. The popular of Retrieval Augmented Generation
3. Is long context still important for Language Model?

Question Answering

Given 1. one or many question prompt (queries)
2. given or not-given relevant knowledge

Goal: Return the answer of the question

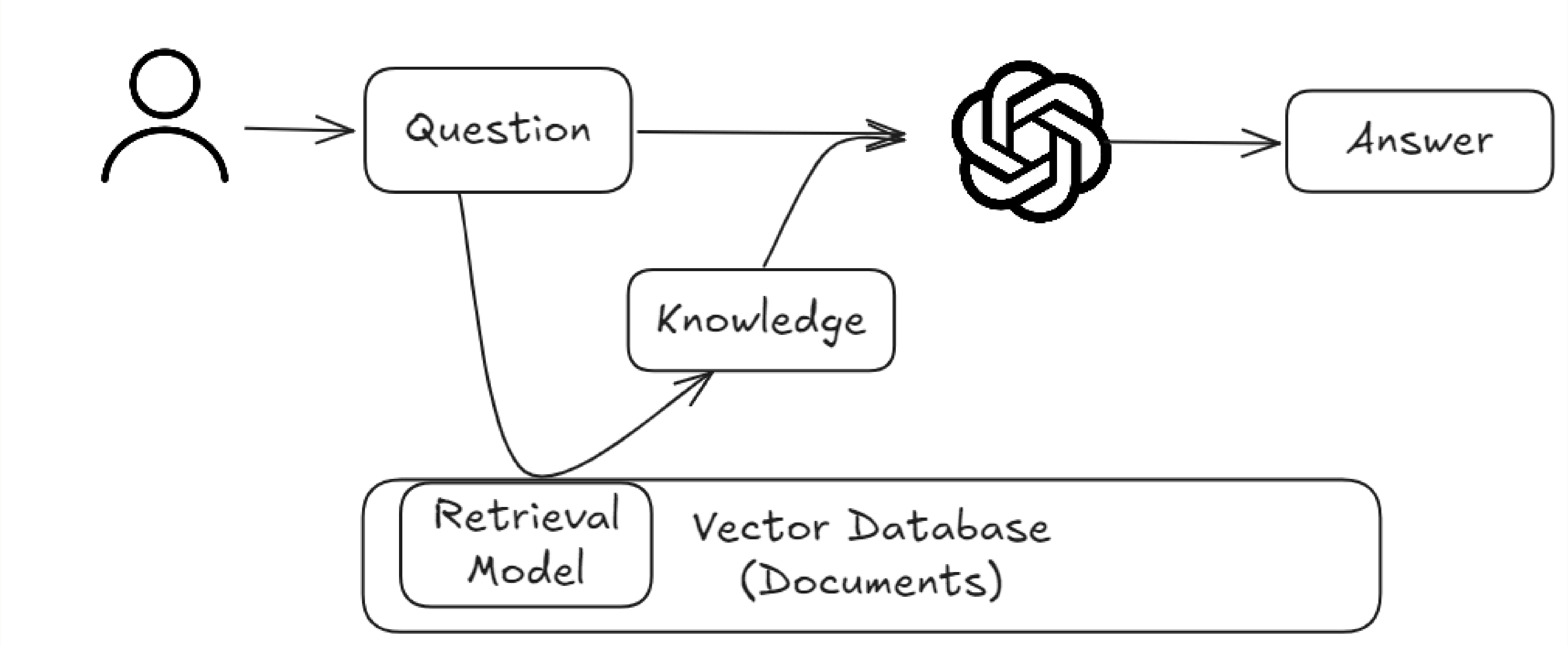
Popular Dataset for QA:

1. SQuAD: Stanford Question Answering Dataset
2. HotpotQA: 113k wiki-pedia QAs
3. TriviaQA: 662k wikipedia and web QAs

TEAM ONE

The popular of RAG

RAG = Retrieval Augmented Generation



TEAM ONE

Experiment Parameter

Language Model :

- Llama 3.1 8b-instruct

Retrirval Model :

- BM25 (Sparse Retrieval)
- OpenAI (Dense Retrieval)

RAG Framework :

- Llama-index

Experiment Dataset: 21k 30k 50k

(Fixed Random seed)

Source	Size	# Docs	# Tokens	# QA Pairs
HotPotQA	Small	16	21k	1,392
	Medium	32	43k	1,056
	Large	64	85k	1,344
SQuAD	Small	3	21k	500
	Medium	4	32k	500
	Large	7	50k	500

(DON'T DO RAG)

TEAM ONE

Experiment Model

Model	av.	2k	4k	8k	16k	32k	64k	96k	125k
o1-preview-2024-09-12	0.763	0.582	0.747	0.772	0.787	0.799	0.831	0.824	0.763
o1-mini-2024-09-12	0.731	0.566	0.728	0.754	0.772	0.777	0.769	0.778	0.704
gpt-4o-2024-05-13	0.709	0.467	0.671	0.721	0.752	0.759	0.769	0.769	0.767
claude-3-5-sonnet-20240620	0.695	0.506	0.684	0.723	0.718	0.748	0.741	0.732	0.706
claude-3-opus-20240229	0.686	0.463	0.652	0.702	0.716	0.725	0.755	0.732	0.741
claude-3-haiku-20240307	0.649	0.466	0.666	0.678	0.705	0.69	0.668	0.663	0.656
qwen2-72b-instruct	0.637	0.469	0.628	0.669	0.672	0.682	0.683	0.648	0.645
gpt-4o-mini-2024-07-18	0.61	0.424	0.587	0.624	0.649	0.662	0.648	0.646	0.643
gpt-4-turbo-2024-04-09	0.588	0.465	0.6	0.634	0.641	0.623	0.623	0.562	0.56
gemini-1.5-pro	0.584	0.368	0.51	0.55	0.58	0.595	0.634	0.636	0.622
claude-3-sonnet-20240229	0.569	0.432	0.587	0.662	0.668	0.631	0.525	0.559	0.485
gpt-4-0125-preview	0.568	0.466	0.614	0.64	0.664	0.622	0.585	0.505	0.452
llama-3.1-405b-instruct	0.55	0.445	0.591	0.615	0.623	0.594	0.587	0.516	0.426
gemini-1.5-flash	0.505	0.349	0.478	0.517	0.538	0.534	0.522	0.52	0.521
llama-3-70b-instruct	0.48	0.365	0.53	0.546	0.555	0.562	0.573	0.583	0.593
mixtral-8x7b-instruct	0.469	0.414	0.518	0.506	0.488	0.417	-	-	-
llama-3.1-70b-instruct	0.45	0.403	0.526	0.527	0.478	0.469	0.444	0.401	0.353
dbrx-instruct	0.447	0.438	0.539	0.528	0.477	0.255	-	-	-
gpt-3.5-turbo	0.44	0.362	0.463	0.486	0.447	-	-	-	-
llama-3.1-8b-instruct	0.411	0.368	0.547	0.536	0.523	0.485	0.383	0.296	0.15

Table S3: LLM answer correctness up to 125k tokens. Same data as Fig. 1.

LONG CONTEXT RAG PERFORMANCE OF LARGE LANGUAGE MODELS

TEAM ONE

Experiment Design -- RAG

Retrieval Model

- BM25 (sparse)
- OpenAI (dense)

Retrieval Chunks Num

- Top-k = 1
- Top-k = 3
- Top-k = 5
- Top-k = 10

```
PROMPT = F"""
    <|BEGIN_OF_TEXT|>
    <|START_HEADER_ID|>SYSTEM<|END_HEADER_ID|>
    YOU ARE AN ASSISTANT FOR GIVING SHORT
ANSWERS BASED ON GIVEN CONTEXT.<|EOT_ID|>
    <|START_HEADER_ID|>USER<|END_HEADER_ID|>
    CONTEXT INFORMATION IS BELLOW.
    ---------------------------------------------
    {KNOWLEDGE}
    ---------------------------------------------
    {ANSWER_INSTRUCTION}
    QUESTION:
    {QUESTION}
    <|EOT_ID|>

<|START_HEADER_ID|>ASSISTANT<|END_HEADER_ID|>
    """
```

TEAM ONE

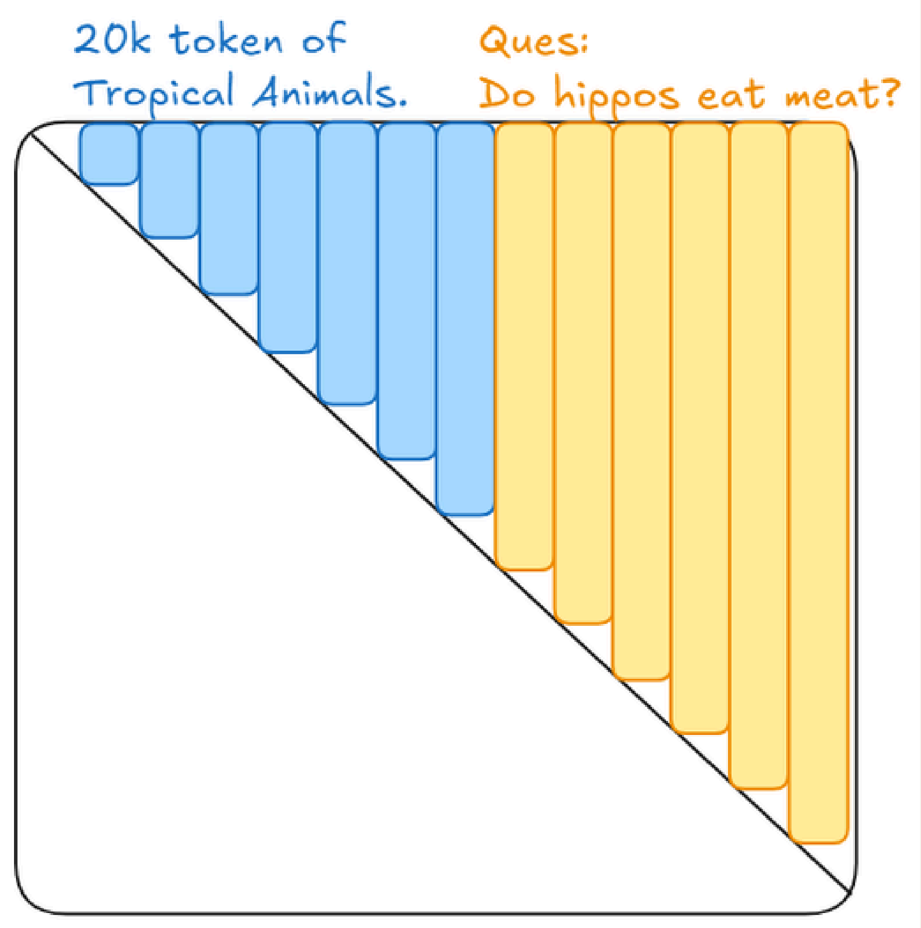
Experiment Design -- CAG

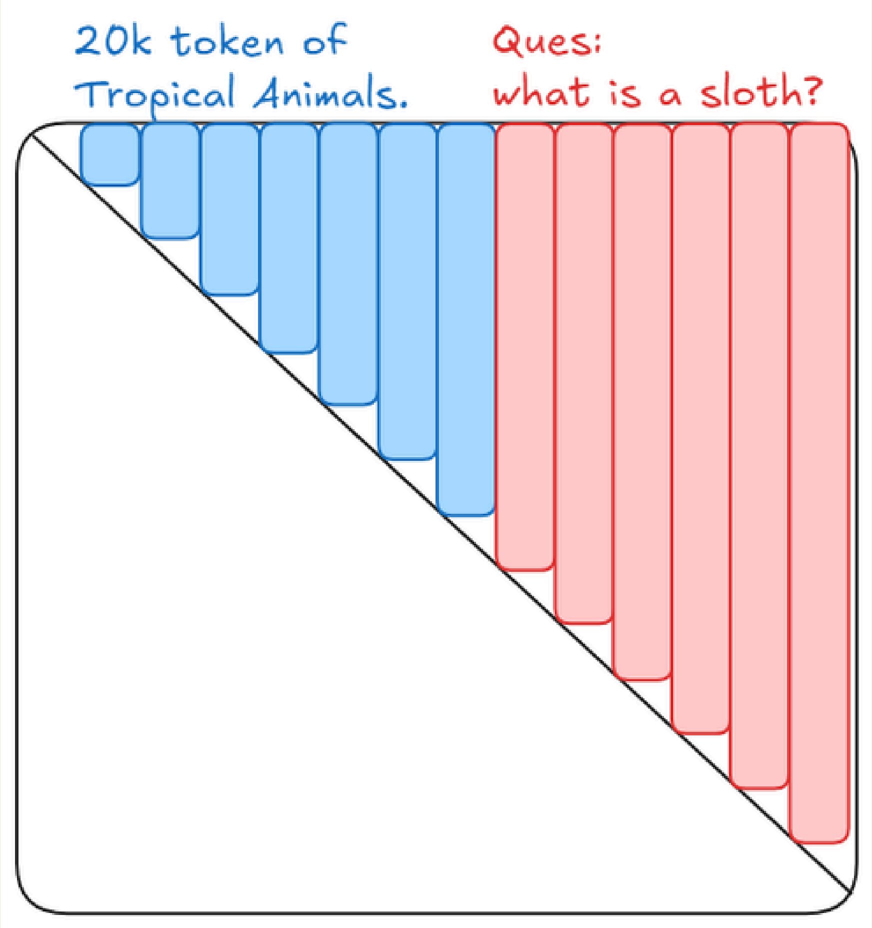
CAG

- **Cache Augmented Generation**
- using kvcache to accelerate LLM generation

Method:

- store the kvcache of knowledge with pytorch function
- read the past kvcache instead of recomputing it everytime.



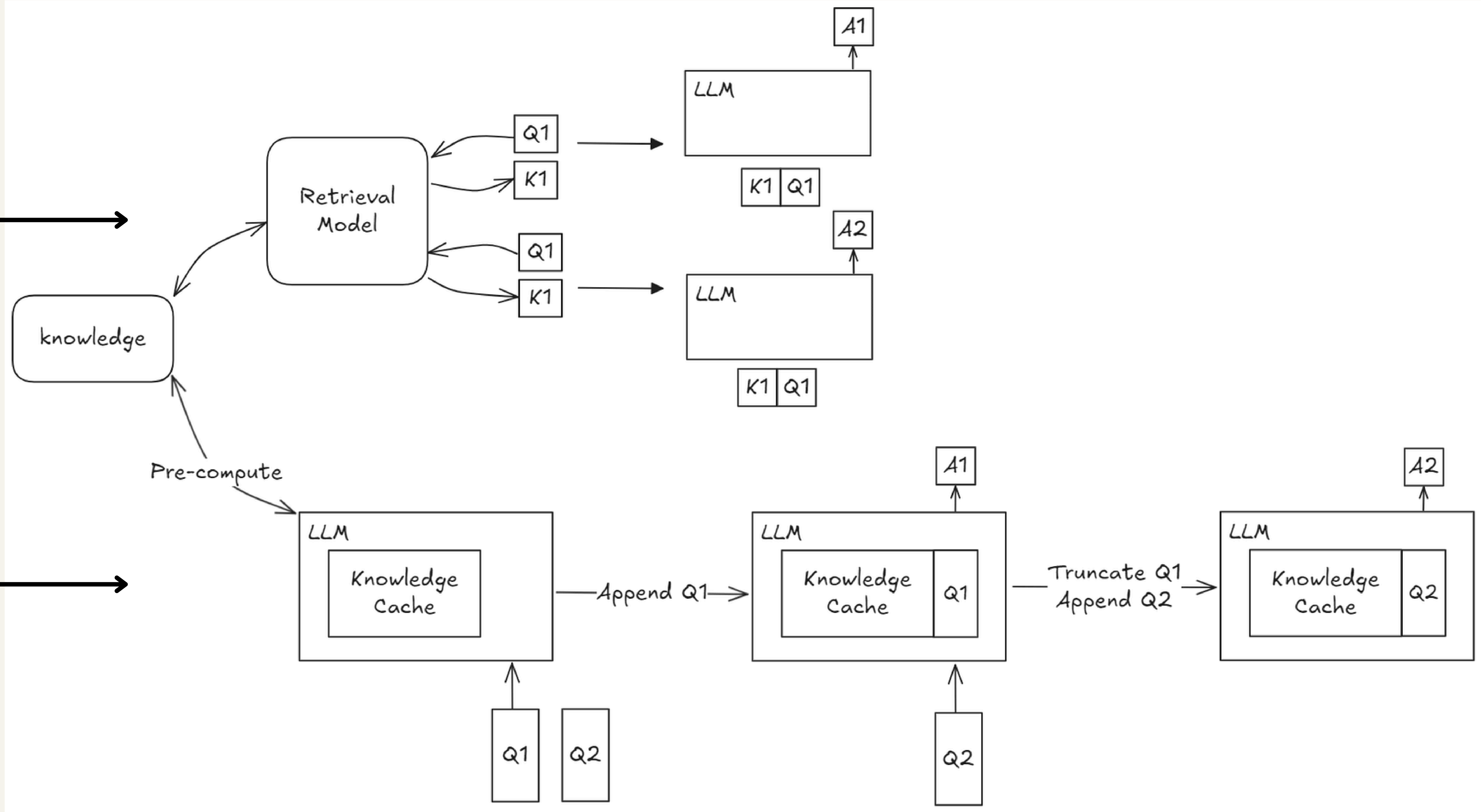


TEAM ONE

Structure of RAG and CAG

RAG

CAG



TEAM ONE

Results -- Performance

METRIC: BERT SCORE

Table 2: Experimental Results

Size	System	Top-*k*	HotPotQA BERT-Score	SQuAD BERT-Score
Small	Sparse RAG	1	0.0673	0.7469
		3	0.0673	0.7999
		5	0.7549	0.8022
		10	0.7461	0.8191
	Dense RAG	1	0.7079	0.6445
		3	0.7509	0.7304
		5	0.7414	0.7583
		10	0.7516	0.8035
	CAG (Ours)		**0.7759**	**0.8265**
Medium	Sparse RAG	1	0.6652	0.7036
		3	0.7619	0.7471
		5	0.7616	0.7467
		10	0.7238	0.7420
	Dense RAG	1	0.7135	0.6188
		3	0.7464	0.6869
		5	0.7278	0.7047
		10	0.7451	0.7350
	CAG (Ours)		**0.7696**	**0.7512**
Large	Sparse RAG	1	0.6567	0.7135
		3	0.7424	0.7510
		5	0.7495	0.7543
		10	0.7358	0.7548
	Dense RAG	1	0.6969	0.6057
		3	0.7426	0.6908
		5	0.7300	0.7169
		10	0.7398	0.7499
	CAG (Ours)		**0.7527**	**0.7640**

(DON'T DO RAG)

TEAM ONE

Results -- Acceleration

Table 3: Comparison of Generation Time

Dataset	Size	System	Generation Time (s)
HotpotQA	Small	CAG	0.85292
		w/o CAG	9.24734
	Medium	CAG	1.66132
		w/o CAG	28.81642
	Large	CAG	2.32667
		w/o CAG	94.34917
SQuAD	Small	CAG	1.06509
		w/o CAG	10.29533
	Medium	CAG	1.73114
		w/o CAG	13.35784
	Large	CAG	2.40577
		w/o CAG	31.08368

(DON'T DO RAG)

TEAM ONE

Experiment Paper

Don't Do RAG:
When Cache-Augmented Generation is All You Need for Knowledge Tasks
(Jui-Hung, Cheng. NCCU)

[https://arxiv.org/pdf/2412.15605]

[Youtube]

TEAM ONE

Citation

1. Beyond Attention: Breaking the Limits of Transformer Context Length with Recurrent Memory
2. Don't Do RAG: When Cache-Augmented Generation is All You Need for Knowledge Tasks
3. Long Context RAG Performance of Large Language Models
4. Medium:Building a Neural Network Zoo From Scratch: The Long Short-Term Memory Network
5. COMPRESSIVE TRANSFORMERS FOR LONG-RANGE SEQUENCE MODELLING
6.

TEAM ONE

The End

THANK YOU FOR LISTENING