Long Context Language Models

Elaine Zhang & Zihan Zhao

Existing Language Models

Current Language Models are based on **Transformers**

- Transformers require increased memory and computation as input increases
    - Quadratic increase in computation complexity as the number of tokens grows
    - Linear increase in memory complexity with the number of tokens
- So, rely on small context windows to perform
    - See only a limited portion of input at a time

Long Context Language Models

- Essentially, an increase in the size of context windows
  - Space reserved for passing data to an LLM to base its response on
- A different, new approach compared to RAG which is more extensive
- Long-context comprehension or Chain of Thought reasoning
  - Maintain context over extended conversations or long-form queries (long text, video, audio, and images)
  - Generate more accurate answers or solutions for complex, multi-step problems

Lost in the Middle: How Language Models Use Long Contexts

Paper 1

Motivation

- Prior research has looked into long context windows
- How do these extended-context language models make use of their input contexts when performing downstream tasks?
- Provide insights into current LLMs when dealing with long sequences, which is crucial for improving their efficiency

Existing Works

- Ivgi et al. (2023)
  - Experiments with question answering model performance
    - Test when relevant paragraph is placed at beginning & random positions within the input context
- Do it differently by experimenting with finer-grained changes in the position of relevant information

Existing Works

- Papailiopoulos et al., 2023 & Li et al., 2023
  - Both use use natural language text for their retrieval experiments
- This paper removes linguistic structure by using random UUIDs
  - Retrieval performance is not influenced by language features
  - Focus on retrieval ability based on position/structure

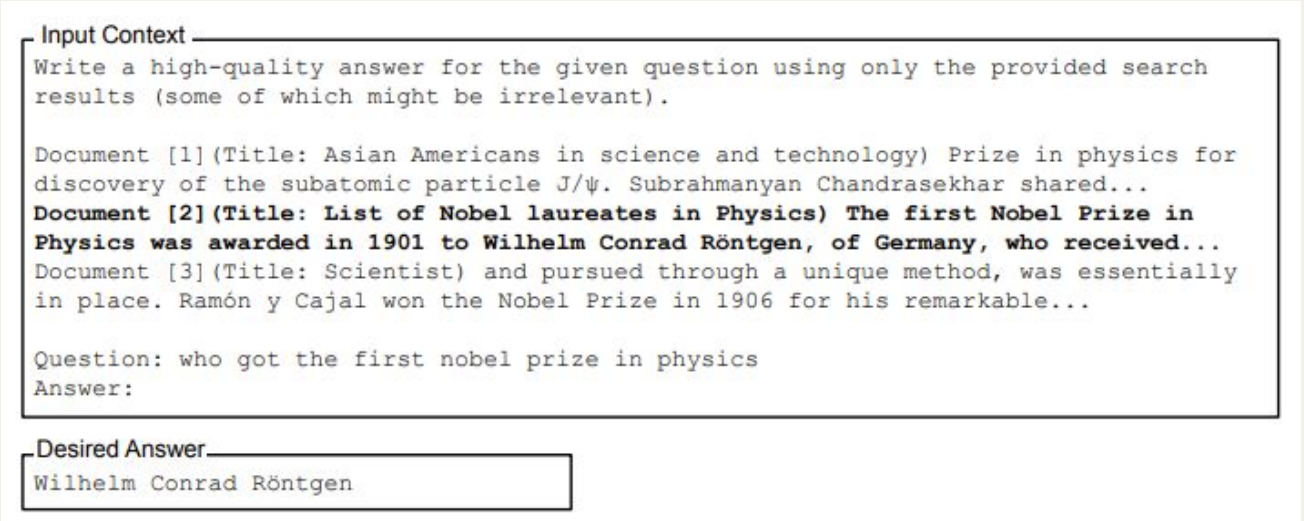
Multi-document Question Answering

- **Goal** of Multi-document Question and Answering
  - Model finds relevant information in input context and use it to answer a question
- Researchers will analyze model performance based on
  - Changes to length in input context
  - Position of relevant information (beginning, middle, or end)
- Analysis - How close is the model output to the actual answer to a question?

Multi-document Question Answering

Experiment Setup

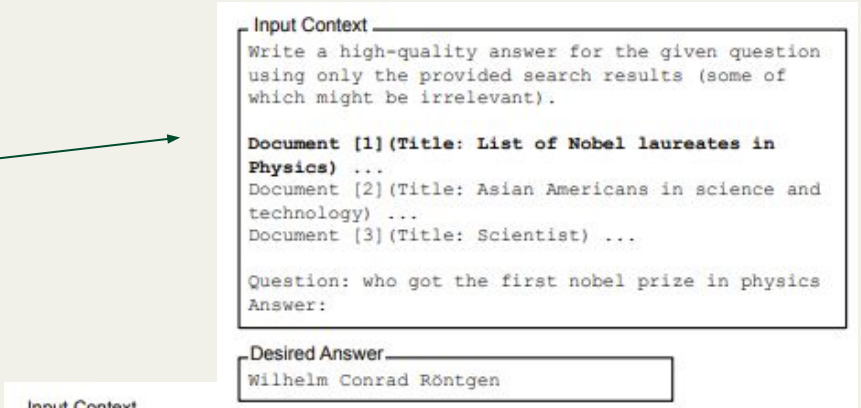
- Model inputs - question to answer, *k* documents where *1* document contains the answer and *k* - *1* documents are distractors

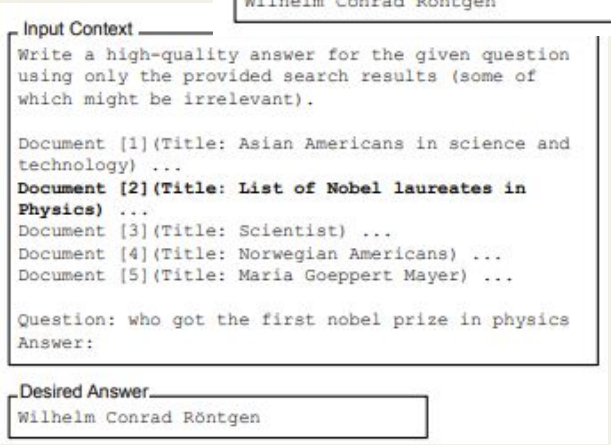


Multi-document Question Answering

Experiment Setup

- Change position of relevant information - change order of documents
- Change context length - add more distraction documents





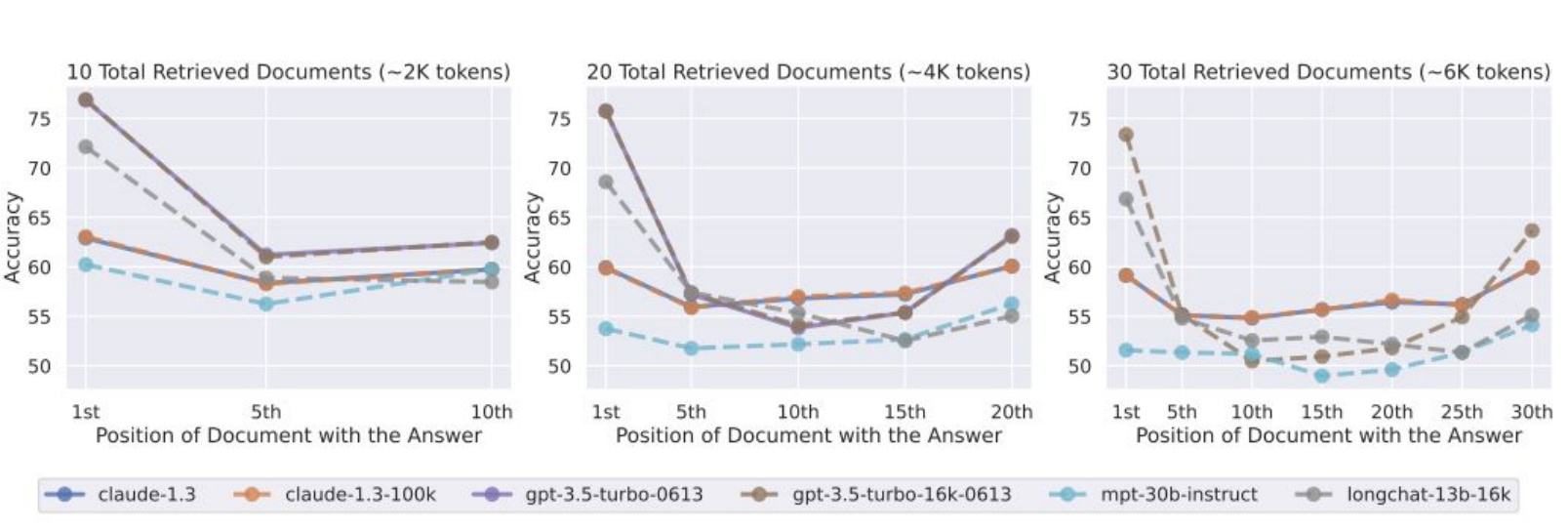
Multi-document Question Answering

Models Tested

- Open language models
  - MPT-30B Instruct - max context length of 8192 tokens
  - LongChat-13B (16K) - max context length of 16384 tokens
- Closed language models
  - GPT-3.5-Turbo (4K) & GPT-3.5-Turbo (16K)
  - Claude-1.3 (8K) & Claude-1.3 (100K)

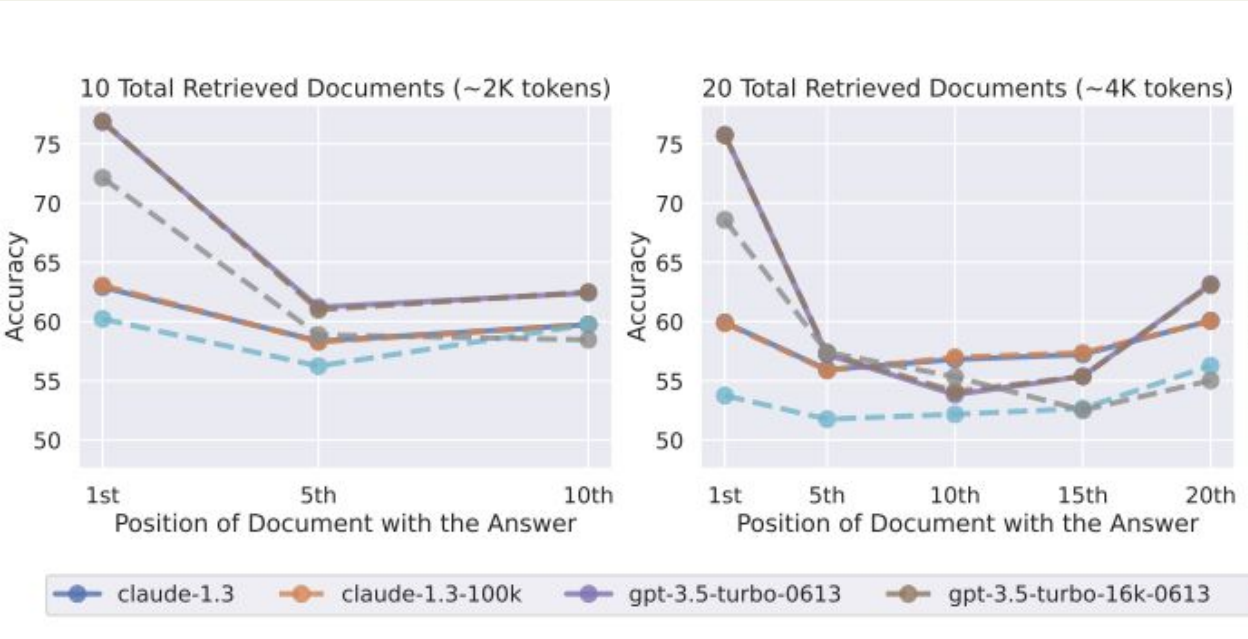
Results: Multi-document Question Answering

Model performance is **highest** when relevant information occurs at the beginning or end of its input context



Results: Multi-document Question Answering

Extended-context models are not necessarily better at using input context when comparing GPT-3.5-Turbo & GPT-3.5-Turbo (16K)



Retrieval Performance of Models

Motivation

- Already see that language models struggle to use information from middle of input
    - So, to what extent can they simply retrieve from input contexts?
- Experiment using a key-value retrieval task

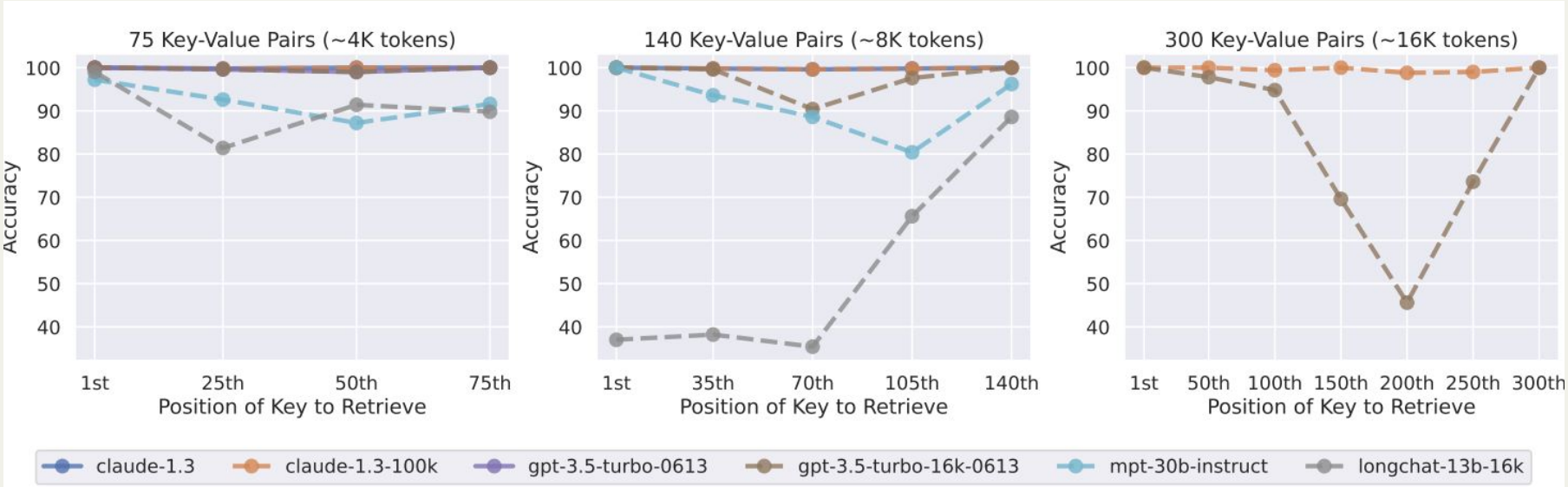
Retrieval Performance of Models

Experiment

- A JSON object containing *k* unique key-value pairs
  - One key-value pair is relevant, *k - 1* are distractions
- Goal
  - A query key from the JSON object is given → model returns the corresponding value
- Analysis
  - Accuracy - does the model retrieve the correct value?

Results: Retrieval Performance of Models

Claude-1.3 and Claude-1.3 (100K) have perfect accuracy, but other models perform highest when context is at beginning or end.



Why Do LMs Struggle with Information Positioning?

- **Decoder-Only Models** process text sequentially and only attend to past tokens
    - Model can’t leverage full information
- **Encoder-Decoder Models** use bidirectional encoders, allowing them to analyze the full input before generating output

Why Do LMs Struggle with Information Positioning?

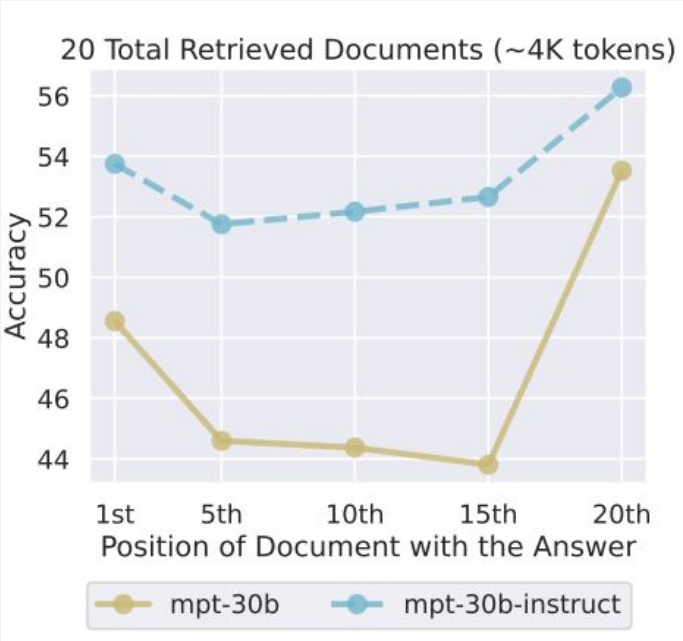
- Encoder-decoder models perform consistently with shorter context, but performance drops when key information is in the middle of long inputs
    - Encoder-decoder models can rank relevance better, but struggle with very long contexts
    - Rank relevance better due to its bidirectional architecture

Why Do LMs Struggle with Information Positioning?

- Decoder only models process data sequentially and in prior experiments researchers placed query after input
- Query-aware contextualization (placing the query before and after the documents)
    - **Improved** performance on the key-value retrieval task
    - No significant difference for multi-document task

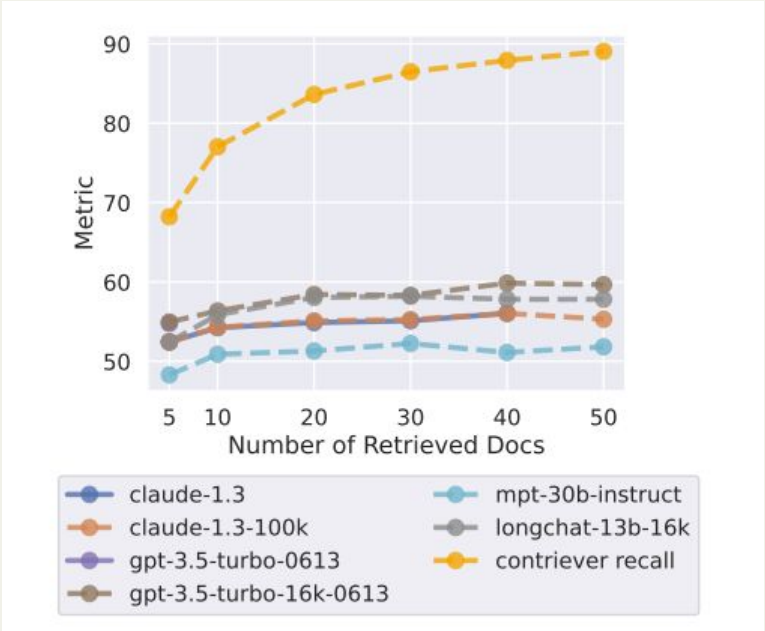
Why Do LMs Struggle with Information Positioning?

- **Instruction fine-tuning improves the model performance**
  - Why? Tune for specific tasks, understand the task more
- Compare mpt-30b which is before model is instruction fine-turned w/ mpt-30b-instruct
- Model still has U-shaped performance curve



Is More Context Better?

Even if a language model can take in 16K tokens, is it actually beneficial to provide 16K tokens of context?



Major Contributions

- The paper examines how language models, particularly decoder-only models, handle the position of relevant information in the input context
- Decoder-only models struggle with information located in the middle of long sequences
- Architecture, Query-Aware Contextualization, & Instruction Fine-Tuning all influence how the model performs
- More context does not always mean better performance
- **Results provide a better understanding of how LMs use their input context and provides new evaluation protocols for future work**

Limitations & Future Work

- Based on the results, models perform better when relevant information appears at the start or end of the input context
  - To solve this problem, future work can look into
    - Effective reranking: Push relevant information closer to the start of the input context
    - Ranked list truncation: Retrieve fewer documents when appropriate, preventing the model from being overwhelmed with irrelevant data
- Did not explore scalability of their approaches
  - Real-world performance analysis, data diversity

LLM Maybe LongLM: Self-Extend LLM Context Window Without Tuning

Paper 2

Motivations

- LLMs are trained at a fixed context length (32k), but users go beyond that
  - An extremely long multi-turn chat
  - An agent that requires long-term memory
  - A Large Multimodal Model (LMM), *like LLaVA-OneVision (2024)*
    - An image === 7k text tokens
    - A 32s video === 6k text tokens (compression applied)

Motivations

- LLMs won’t crash when users go beyond the context length, but performance degrades
  - Perplexity (PPL) explodes ===> lower accuracy
  - Positional *Out-of-Distribution* (O.O.D) Issue
    - Unpredictable if sequence length > context length
      - Positional encoding gives relative and/or absolute position information of tokens in a sequence

Existing Methods

- Fine-tuning
  - Resource and time-intensive

Table 1: Maximum path lengths, per-layer complexity and minimum number of sequential operations for different layer types. n is the sequence length, d is the representation dimension, k is the kernel size of convolutions and r the size of the neighborhood in restricted self-attention.

Layer Type	Complexity per Layer	Sequential Operations	Maximum Path Length
Self-Attention	$O(n^2 \cdot d)$	$O(1)$	$O(1)$
Recurrent	$O(n \cdot d^2)$	$O(n)$	$O(n)$
Convolutional	$O(k \cdot n \cdot d^2)$	$O(1)$	$O(log_k(n))$
Self-Attention (restricted)	$O(r \cdot n \cdot d)$	$O(1)$	$O(n/r)$

Attention is all you need (2017)

Heuristics

- **LLMs should have inherent capabilities to handle long contexts**
  - Kids never learned to read a lengthy book
  - Humans understand texts by general location and order
  - Adjacent texts have similar meanings

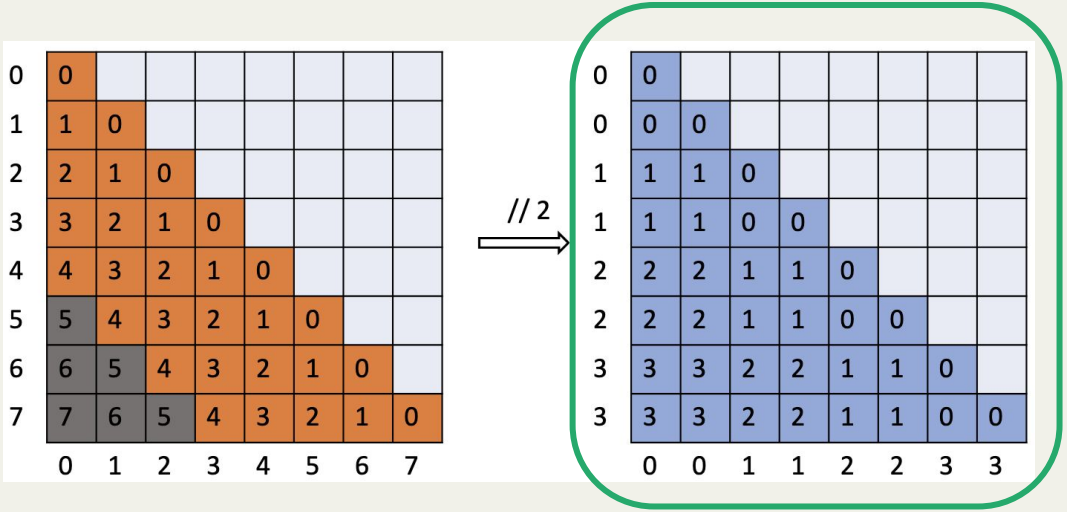
Positional *Out-of-Distribution (O.O.D)* Issue

Method - SelfExtend

- **Map unseen large relative positions to known positions**
  - Grouped Attention
  - Standard/Neighbor Attention

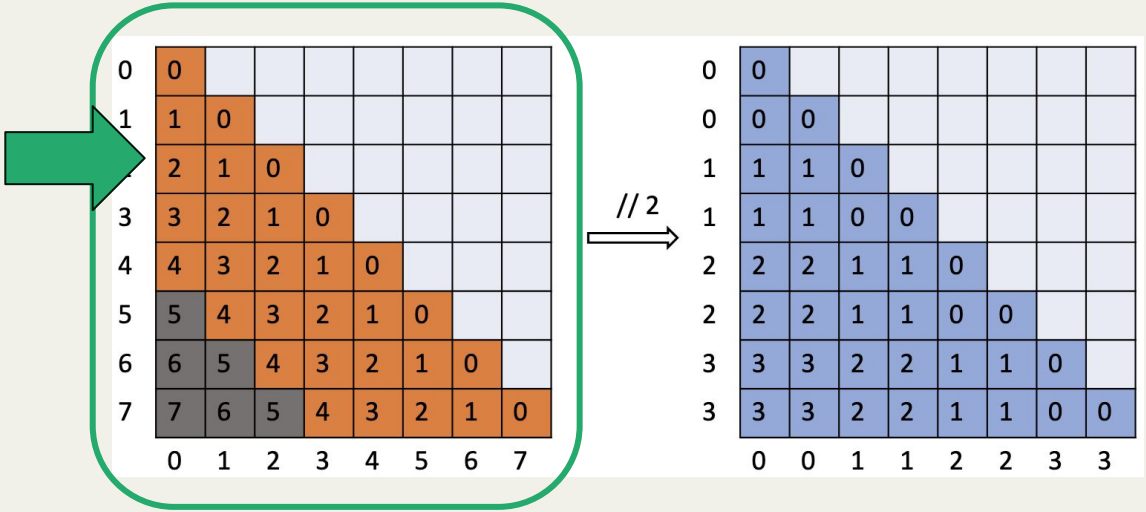
Method - Grouped Attention

- Applies a floor operation to the positions to manage **long-distance** relationships between tokens



Method - Standard/Neighbor Attention

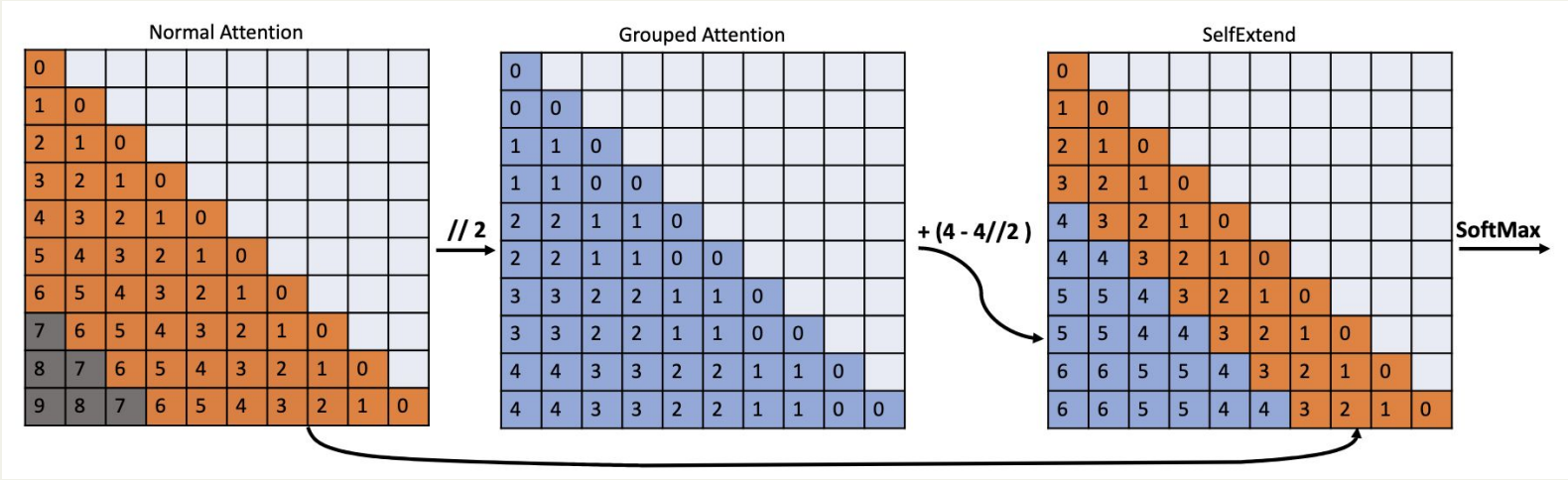
- Sticks to the original positional encoding within a specified range



Method - SelfExtend Formulation

- Position Shift
  - $\mathbf{w_n}$ for **window size** for neighbor tokens
  - $\mathbf{G_s}$ for **group size** for grouped attention

$$w_n - w_n//G_s$$



Method - SelfExtend Formulation

- Extended context length
  - $\mathbf{w_n}$ for **window size** for neighbor tokens
  - $\mathbf{G_s}$ for **group size** for grouped attention
  - **L** for **original context length**

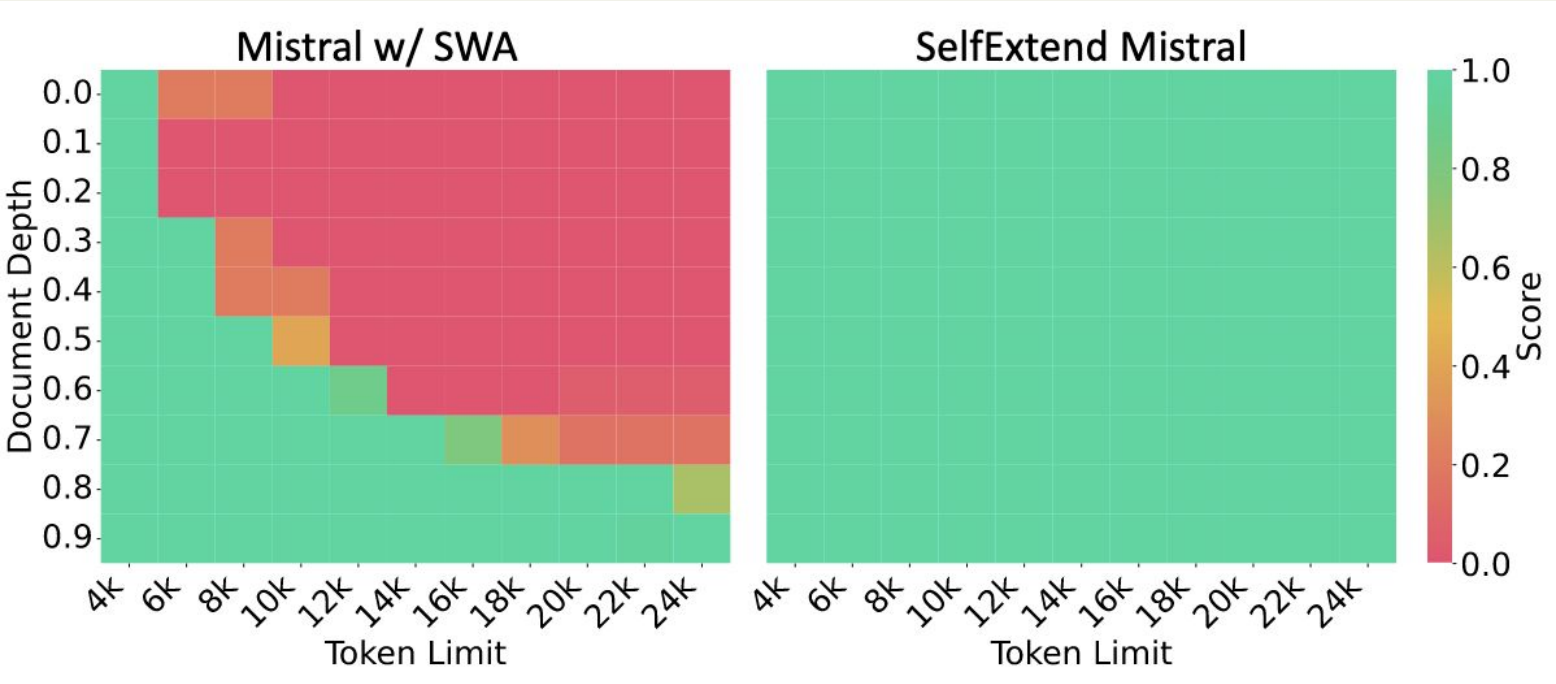
$$(L - w_n) * G_s + w_n$$

Results - Perplexity

Table 1. Perplexity on dataset PG19 with Llama-2-7b-chat and Mistral-7b-instruct-0.1. We report the PPL of with&without Sliding Window Attention (SWA) for Mistral.

Model	Evaluation Context Window Size						
Name	**4096**	**6144**	**8192**	**10240**	**12288**	**14336**	**16384**
Llama-2-7b-chat	9.181	$> 10^3$	$> 10^3$	$> 10^3$	$> 10^3$	$> 10^3$	$> 10^3$
SelfExtend-Llama-2-7b-chat	8.885	8.828	9.220	8.956	9.217	9.413	9.274
Mistral-7b-instruct-0.1 w/ SWA	9.295	9.197	9.532	9.242	9.198	9.278	9.294
Mistral-7b-instruct-0.1 w/o SWA	9.295	9.205	10.20	55.35	$> 10^3$	$> 10^3$	$> 10^3$
SelfExtend-Mistral-7b-instruct-0.1	9.272	9.103	9.369	9.070	8.956	9.022	9.128

Results - Passkey Retrieval Task Score



Results

	LLMs[a]	Single-Document QA			Multi-Document QA			Summarization			Few-shot Learning			Synthetic		Code	
		NarrativeQA	Qasper	MultiField-en	HotpotQA	2WikiMQA	Musique	GovReport	QMSum	MultiNews	TREC	TriviaQA	SAMSum	PassageCount	PassageRe	Lcc	RepoBench-P
SelfExtend	Llama-2-7B-chat-4k*	18.7	19.2	**36.8**	25.4	32.8	9.4	27.3	20.8	25.8	61.5	77.8	40.7	2.1	**9.8**	52.4	43.8
	SE-Llama-2-7B-chat-16k+	**21.69**	25.02	35.21	34.34	30.24	14.13	27.32	**21.35**	25.78	**69.50**	**81.99**	40.96	**5.66**	5.83	**60.60**	**54.33**
	SE-Llama-2-7B-chat-25k+	21.37	**26.68**	34.63	**35.47**	**30.46**	**15.51**	**27.51**	21.30	**25.87**	68.50	78.79	**41.29**	3.90	3.50	59.69	53.83
	Mistral-7B-ins-0.1-16k w/ SWA+	19.40	34.53	37.06	42.29	32.49	14.87	27.38	22.75	26.82	65.00	**87.77**	42.34	**1.41**	28.50	**57.28**	53.44
	Mistral-7B-ins-0.1-8k w/o SWA+	20.46	35.36	39.39	34.81	29.91	11.21	24.70	21.67	26.67	68.00	86.66	41.28	0.18	24.00	56.94	**55.85**
	SE-Mistral-7B-ins-0.1-16k+[b]	**23.56**	**39.33**	**49.50**	**45.28**	**34.92**	**23.14**	**30.71**	**24.87**	**26.83**	**69.50**	86.47	**44.28**	1.18	**29.50**	55.32	53.44
	Phi-2-2k+	4.46	7.01	19.98	**9.43**	8.55	**4.62**	25.64	14.32	**24.03**	50.50	74.55	**1.71**	**2.83**	**4.17**	**58.96**	54.14
	SE-Phi-2-8k+	**12.04**	**12.10**	**20.15**	8.22	**9.68**	3.89	**27.90**	**14.58**	22.13	**61.00**	**82.82**	1.40	2.37	2.83	57.87	**56.42**
	SOLAR-10.7B-ins-4k+	16.50	24.06	46.76	44.03	**36.05**	22.76	**31.39**	19.81	**26.36**	70.00	87.91	42.49	**4.5**	26.5	41.04	54.36
	SE-SOLAR-10.7B-ins-16k+	**22.63**	**32.49**	**47.88**	**46.19**	34.32	**27.88**	30.75	**22.10**	25.62	**74.50**	**89.04**	**42.79**	4.0	**28.0**	**53.73**	**56.47**
Other Methods	LongChat1.5-7B-32k*	16.9	27.7	41.4	31.5	20.6	9.7	30.8	22.7	26.4	63.5	82.3	34.2	1.0	30.5	53.0	55.3
	together/llama-2-7b-32k+	15.65	10.49	33.43	12.36	12.53	6.19	29.28	17.18	22.12	71.0	87.79	43.78	1.0	23.0	63.79	61.77
	CLEX-7B-16k*	18.05	23.68	44.62	28.44	19.53	9.15	32.52	22.9	25.55	68	84.92	42.82	0	11.5	59.01	56.87
	CodeLLaMA-7B-16k*	22.93	30.69	43.37	33.05	27.93	14.2	28.43	24.18	26.84	70	84.97	43.43	2	13.5	64.35	55.87
	SE-Llama-2-7B-chat-16k+	21.69	25.02	35.21	34.34	30.24	14.13	27.32	21.35	25.78	69.50	81.99	40.96	5.66	5.83	60.60	54.33
	SE-Llama-2-7B-chat-25k+	21.37	26.68	34.63	35.47	30.46	15.51	27.51	21.30	25.87	68.50	78.79	41.29	3.90	3.50	59.69	53.83
	Vicuna1.5-7B-16k*	19.4	26.1	38.5	25.3	20.8	9.8	27.9	22.8	27.2	71.5	86.2	40.8	6.5	4.5	51.0	43.5
	SE-Vicuna1.5-7B-16k+	21.88	35.16	42.00	31.14	22.51	13.33	28.47	22.24	26.70	69.50	86.31	40.54	3.56	7.50	60.16	44.07
	SE-Vicuna1.5-7B-25k+	22.46	34.42	42.58	30.95	24.33	12.72	27.75	22.26	27.21	72.00	84.02	40.38	3.01	7.00	58.86	43.86
	MistralLite-16k+	32.12	47.02	44.95	58.5	47.24	31.32	33.22	26.8	24.58	71.5	90.63	37.36	3	54.5	66.27	65.29
	SE-Mistral-7B-ins-0.1-16k+	23.85	37.75	46.93	45.35	34.54	23.28	30.45	23.58	26.94	69.50	85.72	43.88	0.59	28.50	54.92	53.44
Fixed Models	GPT-3.5-Turbo-16k*	23.6	43.3	52.3	51.6	37.7	26.9	29.5	23.4	26.7	68.0	91.4	41.7	4.5	71.0	54.7	53.6
	XGen-7B-8k*	18	18.1	37.7	29.7	21.1	10.3	27.3	20.5	26.2	65.5	77.8	25.3	2.1	8.5	38.6	38.6
	InternLM-7B-8k*	12.1	16.7	23.4	28.7	22.8	9.0	9.7	15.9	22.8	52.0	77.8	21.2	3.0	6.0	44.1	28.8
	ChatGLM2-6B-32k*	21.1	31.5	46.2	45.1	34.0	21.9	32.4	24.0	26.5	62.5	78.7	36.3	1.5	77.0	55.6	49.9
	ChatGLM3-6B-32k*	26.0	43.3	51.7	54.4	44.9	40.4	36.8	23.9	27.9	79.0	87.1	38.2	2.0	99.0	57.66	54.76
	Baichuan-13B-4k*	0.07	17.55	17.28	3.29	15	0.1	6.8	1.71	23.1	20.05	20.06	5.77	0.06	0.5	47.98	16.58
	ALiBi-7B-4k*	0.04	8.13	17.87	2.73	8	1.33	5.31	1.64	25.55	9.25	8.83	4.67	0	1.27	46.69	18.54

Results

Model	Tokens	Coursera	GSM	QuALITY	TOEFL	CodeU	SFiction	Avg.
Claude1.3-100k	100k	60.03	88.00	73.76	83.64	17.77	72.65	65.97
GPT-4-32k	32k	75.58	96.00	82.17	84.38	25.55	74.99	73.11
Turbo-16k-0613	16k	63.51	84.00	61.38	78.43	12.22	64.84	60.73
Chatglm2-6b-8k	2k	43.75	13.00	40.59	53.90	2.22	54.68	34.69
XGen-7b-8k (2k-4k-8k)	2k	26.59	3.00	35.15	44.23	1.11	48.43	26.41
Chatglm2-6b-8k	8k	42.15	18.00	44.05	54.64	2.22	54.68	35.95
Chatglm2-6b-32k	32k	47.81	27.00	45.04	55.01	2.22	57.02	39.01
XGen-7b-8k	8k	29.06	16.00	33.66	42.37	3.33	41.40	27.63
MPT-7b-65k	8k	25.23	8.00	25.24	17.84	0.00	39.06	19.22
Llama2-7b-chat	4k	29.21	19.00	37.62	51.67	1.11	60.15	33.12
Longchat1.5-7b-32k	32k	32.99	18.00	37.62	39.77	**3.33**	57.02	31.45
Llama2-7b-NTK	16k	32.71	19.00	33.16	52.78	0.00	**64.84**	33.74
SE-Llama2-7B-chat+	16k	**35.76**	**25.00**	**41.09**	**55.39**	1.11	57.81	**36.02**
Vicuna1.5-7b-16k	16k	**38.66**	19.00	39.60	**55.39**	**5.55**	60.15	36.39
SE-Vicuna1.5-7B+	16k	37.21	**21.00**	**41.58**	**55.39**	3.33	**63.28**	**36.96**
Llama2-13b-chat	4k	35.75	39.00	**42.57**	60.96	1.11	54.68	39.01
Llama2-13b-NTK	16k	36.48	11.00	35.64	54.64	1.11	**63.28**	33.69
Llama2-13b-NTK(Dyn)	16k	30.08	**43.00**	41.58	64.31	1.11	35.15	35.87
SE-Llama2-13B-chat+	16k	**38.95**	42.00	41.09	**66.17**	1.11	**63.28**	**42.10**
Mistral-7b-ins-0.1 w/ SWA+	16k	44.77	44.00	46.53	60.59	2.22	**64.06**	43.70
Mistral-7b-ins-0.1 w/o SWA+	8k	43.60	49.00	45.05	60.59	**4.44**	60.94	43.94
MistralLite+	16k	29.23	32.00	46.04	17.47	3.33	14.06	23.69
SE-Mistral-7b-ins-0.1+	16k	**45.20**	**51.00**	**48.02**	**64.68**	3.33	59.38	**45.27**
Phi-2+	2k	38.37	64.00	**42.08**	55.76	3.33	**52.34**	42.64
SE-Phi-2+	8k	**42.44**	**65.00**	41.08	**62.83**	**4.44**	**52.34**	**44.69**
SOLAR-10.7b-Instruct-v1.0+	4k	48.84	**72.00**	59.90	77.32	**4.44**	69.53	55.34
SE-SOLAR-10.7b-v1.0+	16k	**50.44**	**72.00**	**70.30**	**79.18**	**4.44**	**73.44**	**58.30**

Results

- SelfExtend achieves **comparable or better** performance, compared to methods that requires further fine-tuning

Trade-offs

- Grouped Attention
  - Large group size ===> Coarse position information
  - Small group size ===> Under-trained relative positions
- Neighbor Attention
  - Large neighbor window size ===> Large group size
  - Small neighbor window size ===> Small group size

Takeaways

- LLMs struggle with long contexts due to positional encoding (O.O.D)
- LLMs should have inherent capabilities to handle long contexts
- SelfExtend uses a **bi-level attention mechanism** to evoke this
- SelfExtend requires **no further finetuning**
- SelfExtend achieves **comparable or better** performance
- SelfExtend requires **minor** code modification **at inference**
- Choice of group size and neighbor window size needs to be careful

Retrieval meets Long Context Large Language Models

Paper 3

Motivation

- Recent advancements in long context LLMs with exact attention
  - Due to development of faster GPU with more memory and memory-efficient exact attention
- Increase the context window of LLMs **vs** Retrieval Augmentation methods
  - Combine the benefits of the two to improve performance

Models

- **NeMo GPT-43B** *(Nvidia, proprietary)*
  - 70% English corpus and 30% multilingual and code data
  - 4K context length with RoPE embeddings
- **Llama 2-70B** *(Meta, open-sourced)*
  - 90% English corpus
  - 4K context length with RoPE embeddings

Zero Shot Evaluations

- Single document QA
  - Reason over a single document
- Multi document QA
  - Reason over multiple documents
- Query-based summarization
  - Summarize as instructed

Zero Shot Evaluations - Datasets

Datasets	Tasks	Key Challenge
Qasper	Single Document QA	Giving short answers from long NLP papers
NarrativeQA	Single Document QA	Giving short answers from noisy books/movies
QuALITY	Single Document QA	Multi-choice over stories/articles
HotpotQA	Multi Document QA	Linking multiple Wikipedia sources
MuSiQue	Multi Document QA	Harder and less cheatable than HQA
MultiFieldQA-en	Multi Document QA	Reasoning over gov reports, legal docs, etc.
QMSum	Query-based Summarization	Extracting relevant summary from meetings

Sample Data - Single Document QA

NarrativeQA


```
{
    "document": {
        "id": "23jncj2n3534563110",
        "kind": "movie",
        "url": "https://www.imsdb.com/Movie%20Scripts/Name%20of%20Movie.html",
        "file_size": 80473,
        "word_count": 41000,
        "start": "MOVIE screenplay by",
        "end": ". THE END",
        "summary": {
            "text": "Joe Bloggs begins his journey exploring...",
            "tokens": ["Joe", "Bloggs", "begins", "his", "journey", "exploring"],
            "url": "http://en.wikipedia.org/wiki/Name_of_Movie",
            "title": "Name of Movie (film)"
        },
        "text": "MOVIE screenplay by John Doe\nSCENE 1..."
    },
    "question": {
        "text": "Where does Joe Bloggs live?",
        "tokens": ["Where", "does", "Joe", "Bloggs", "live", "?"]
    },
    "answers": [
        {"text": "At home", "tokens": ["At", "home"]},
        {"text": "His house", "tokens": ["His", "house"]}
    ]
}
```

Sample Data - Multi Document QA

HotpotQA

Paragraph A: Reginald Engelbach

Reginald Engelbach (July 9, 1888 – February 26, 1946) was an English Egyptologist and engineer. He is mainly known for his works in the Egyptian Museum of Cairo, above all the compilation of a register of artifacts belonging of the museum.

Paragraph B: Cairo

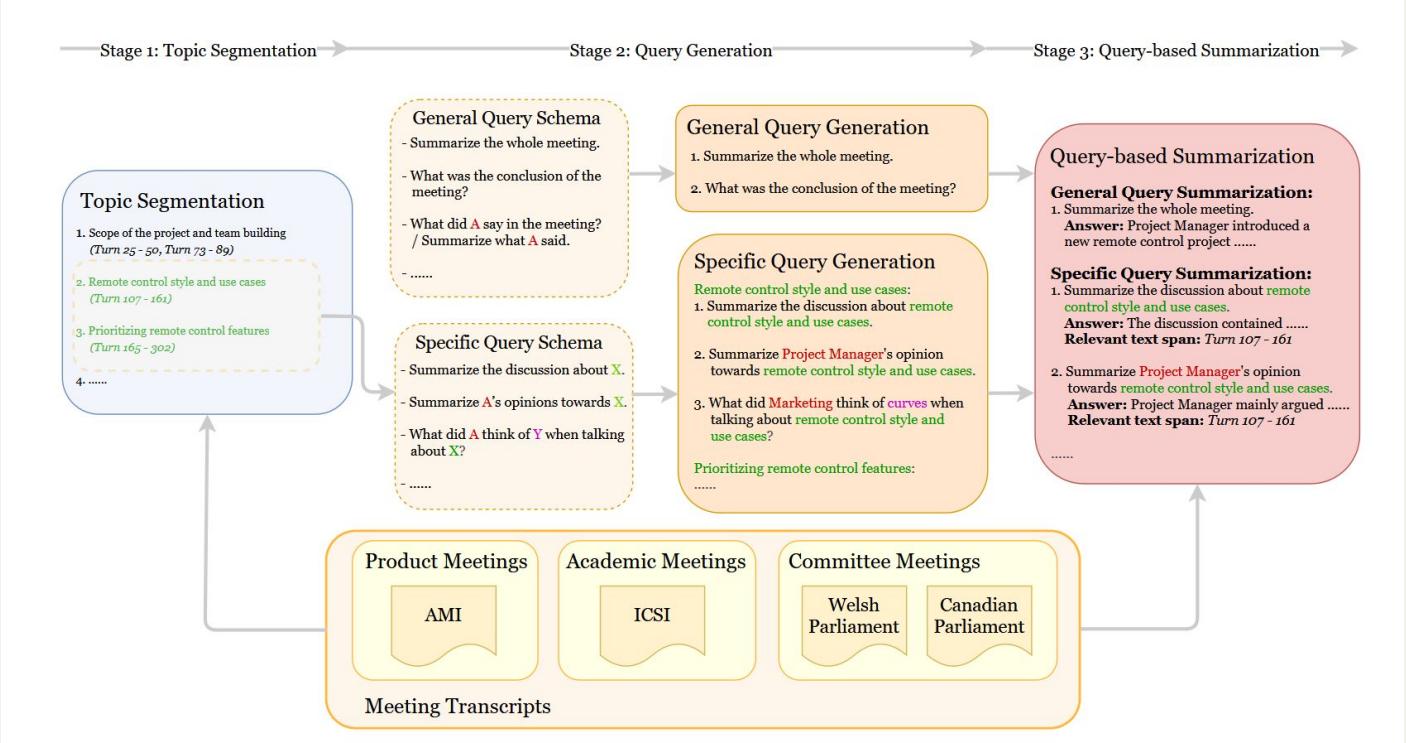
Cairo (; Arabic: القاهرة "al-Qāhirah ", , Coptic: "Kahire ") is the capital and largest city of Egypt. The city's metropolitan area is the largest in the Middle East and the Arab world, and the 15th-largest in the world, and is associated with ancient Egypt, as the famous Giza pyramid complex and the ancient city of Memphis are located in its geographical area. Located near the Nile Delta, modern Cairo was founded in 969 CE by the Fatimid dynasty, but the land composing the present-day city was the site of ancient national capitals whose remnants remain visible in parts of Old Cairo. Cairo has long been a center of the region's political and cultural life, and is titled "the city of a thousand minarets" for its preponderance of Islamic architecture. Cairo is considered a World City with a "Beta +" classification according to GaWC.

Q: Which English Egyptologist is known mainly for his works in the Egyptian Museum that is named after the capital of Egypt?

A: Reginald Engelbach

Sample Data - Query-based Summarization

QMSum



Retrievers

- Dragon
  - A *dual encoder*: one for queries and one for documents
- Contriever
  - *Contrastive learning*: pulling semantically similar documents closer and pushing irrelevant documents apart
- OpenAI Embedding
  - Input: max 8191 tokens
  - Output: 1536-dimensional vector

Other Preparations

- Context window extension
  - 4k ===> 16k and 32k
- Instruction tuning

Results

Model	Seq len.	Avg.	QM	QASP	NQA	QLTY	MSQ	HQA	MFQA
GPT-43B	4k	26.44	15.56	23.66	15.64	49.35	11.08	28.91	40.90
+ ret	4k	29.32	16.60	23.45	19.81	51.55	14.95	34.26	44.63
GPT-43B	16k	29.45	16.09	25.75	16.94	50.05	14.74	37.48	45.08
+ ret	16k	**29.65**	15.69	23.82	21.11	47.90	15.52	36.14	47.39
Llama2-70B	4k	31.61	16.34	27.70	19.07	63.55	15.40	34.64	44.55
+ ret	4k	36.02	17.41	28.74	23.41	70.15	21.39	42.06	48.96
Llama2-70B	16k	36.78	16.72	30.92	22.32	**76.10**	18.78	43.97	48.63
+ ret	16k	37.23	**18.70**	29.54	23.12	70.90	23.28	44.81	50.24
Llama2-70B	32k	37.36	15.37	**31.88**	23.59	73.80	19.07	49.49	48.35
+ ret	32k	**39.60**	18.34	31.27	**24.53**	69.55	**26.72**	**53.89**	**52.91**
Llama2-7B	4k	22.65	14.25	22.07	14.38	40.90	8.66	23.13	35.20
+ ret	4k	**26.04**	16.45	22.97	18.18	43.25	14.68	26.62	40.10
Llama2-7B	32k	**28.20**	16.09	23.66	19.07	44.50	15.74	31.63	46.71
+ ret	32k	27.63	17.11	23.25	19.12	43.70	15.67	29.55	45.03

Results - Lost in the Middle

- Lost in the middle is a phenomenon we discussed in paper 1
- Longer context models (e.g., 32K) perform better at using retrieved evidence than shorter context models (e.g., 4K)
    - But lost in the middle phenomenon still occurs in long context models
- Hypothesis as to why
    - Longer context windows help models retain and process more relevant information at a time

Results - Comparison with OpenAI Models

Model	Trec	SAMSum
GPT-3.5-turbo-16k	68	41.7
Llama2-70B	73	46.5
Llama2-70B-ret	76	47.3

Table 6: Comparison of Llama2-70B to GPT-3.5-turbo-16k with two few-shot learning tasks from LongBench. Retrieval is helpful for few-shot learning as well.

Major Contributions

- Conducted a comprehensive study with two state-of-the-art LLMs
  - Proprietary 43B pretrained GPT
  - Llama2-70B
- Focused on **9** downstream tasks
- Impact of Retrieval-Augmentation
  - Retrieval-augmentation improves performance for 4K context LLMs
  - Performance is comparable to 16K long context LLMs

Major Contributions

- Enhancing Long-Context Models:
  - Retrieval improves performance for 16K/32K context models, especially for Llama2-70B
  - Best model: Llama2-70B-32K-ret outperforms GPT-3.5-turbo-16K
  - Outperforms non-retrieval Llama2-70B-32K
  - 4x faster generation speed on NarrativeQA

Limitations

- Lost in the Middle phenomenon
  - Did not explore mitigation strategies
- Computational costs of extending the context window in real world scenarios
- Grabbed the top-5 chunks from the retriever model
  - No analysis of the quality of the received chunks
  - Test grabbing different number of chunks

Future Work

- Extend the context window beyond 32K
  - 64K
- Mitigate the "Lost-in-the-Middle" Phenomenon
  - Continuing pre-training with UL2 loss
- Develop advanced methods for existing pre-trained large language models
  - Memory or hierarchical attention
  - Non-trivial

Questions?