StreamingDialogue: Prolonged Dialogue Learning via Long Context Compression with Minimal Losses

Jia-Nan Li[1*]   Quan Tu[1*]   Cunli Mao[2]   Zhengtao Yu[2†]   Ji-Rong Wen[1]   Rui Yan[1†]

[1] Gaoling School of Artificial Intelligence, Renmin University of China
[2] Kunming University of Science and Technology
{lijianan, quantu, jrwen, ruiyan}@ruc.edu.cn
maocunli@163.com, ztyu@hotmail.com

Abstract

Standard Large Language Models (LLMs) struggle with handling dialogues with long contexts due to efficiency and consistency issues. According to our observation, dialogue contexts are highly structured, and the special token of *End-of-Utterance* (EoU) in dialogues has the potential to aggregate information. We refer to the EoU tokens as "conversational attention sinks" (conv-attn sinks). Accordingly, we introduce StreamingDialogue, which compresses long dialogue history into conv-attn sinks with minimal losses, and thus reduces computational complexity quadratically with the number of sinks (i.e., the number of utterances). Current LLMs already demonstrate the ability to handle long context window, e.g., a window size of 200K or more. To this end, by compressing utterances into EoUs, our method has the potential to handle more than 200K of utterances, resulting in a prolonged dialogue learning. In order to minimize information losses from reconstruction after compression, we design two learning strategies of short-memory reconstruction (SMR) and long-memory reactivation (LMR). Our method outperforms strong baselines in dialogue tasks and achieves a 4 × speedup while reducing memory usage by 18 × compared to dense attention recomputation.[3]

1 Introduction
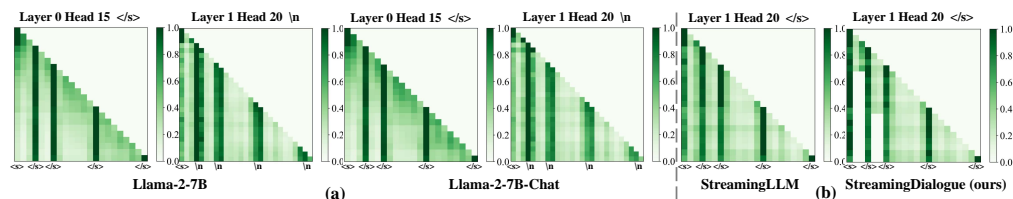
Figure 1: Attention map visualization. (a) Llama-2-7B/Chat with "</s>" and "\n" as EoU ("</s>" counts as one token, "\n" as two). (b) StreamingLLM versus StreamingDialogue attention on Llama-2-7B with "</s>" as EoU.

Large Language Models (LLMs) [1–5] are rapidly advancing. However, their performance is constrained by context size during pre-training. For example, with a context size of 4,096, the inference capability of LLaMA2 [6] sharply drops when the context length exceeds the preset limit. Moreover, the attention mechanism [7] leads to quadratic growth in computational complexity with

---

[*]The first two authors contributed equally.

[†]Corresponding authors: Rui Yan (ruiyan@ruc.edu.cn) and Zhengtao Yu (ztyu@hotmail.com).

[3]Code: https://github.com/JinaLeejnl/StreamingDialogue

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

text length, increasing GPU memory usage and slowing generation speed. As LLMs find widespread use in various conversational applications [8–10], these limitations become particularly severe for dialogue tasks [11–13], rendering standard LLMs infeasible for supporting prolonged dialogues with long conversation histories.

In order to support conversations with long contexts, a natural solution is to reduce the computation of inter-token correlations by modifying the implementation of attention. Beltagy et al. [14] proposed local attention, which confines attention within a k window size, reducing computational complexity to linear. However, when the text length exceeds k, the generation performance substantially declines. StreamingLLM [15] enhanced long context streaming by introducing the concept of "attention sinks." This approach builds on local attention, allowing initial tokens to be consistently attended to, which supports stable long-term interactions and efficient generation. However, StreamingLLM continuously updates the cached information within the fixed-size window, excluding initial tokens. These initial tokens are blind to subsequent tokens during the auto-regressive generation process. Consequently, as the dialogue context lengthens, historical information is progressively lost, which is detrimental to dialogue consistency and severely impacts the user experience.

We introduce StreamingDialogue, a method designed for efficient conversations with enhanced long-term memory capabilities. In dialogue contexts, we observe an interesting phenomenon: tokens used to separate utterances (namely End-of-Utterance, EoU), such as "</s>" and "\n," tend to aggregate more attention than other tokens (for details, refer to Figure 1 (a) and see §4.11 for further analysis). We refer to these separator tokens as **"conversational attention sinks"** (conv-attn sinks). Figure 1 (b) demonstrates that, in contrast to the highly dispersed attention pattern of StreamingLLM, StreamingDialogue maintains focus on critical positions like conv-attn sinks, thereby utilizing them to aggregate utterance information, compressing lengthy dialogues to only require caching conv-attn sinks' key-values to improve efficiency and reduce memory consumption.

Specifically, in the long-term generation, we preserve conv-attn sinks to memorize historical dialogues for retrieval. Additionally, caching both the first token and the previous and current utterances is crucial to ensure stable output beyond a certain inference length and to facilitate the smooth generation of consecutive replies. Beyond these measures, we introduce two self-learning strategies to better characterize the conv-attn sinks: (1) we devise a reconstruction task, where the reconstruction process can only attend to the conv-attn sink of the target utterance, thereby encouraging the conv-attn sink to restore information from the target sentence, namely **short-memory reconstruction** (SMR); (2) we propose a recall task, treating the final utterance as a query and attending solely to conv-attn sinks in the dialogue history to retrieve the matching response, thus prompting the model to reactivate information from lengthy dialogues, named as **long-memory reactivation** (LMR). These two tasks will be jointly optimized before dialogue learning.

Experiments on widely-used dialogue datasets demonstrate that our proposed method outperforms other sparse attention and memory-enhancement methods (in terms of evaluation metrics of Perplexity, BLEU, ROUGE, Distinct, USL-H, and Dial-M). In terms of efficiency, our method achieves a 4 $\times$ speedup and an 18 $\times$ reduction in memory usage compared to dense attention with recomputation. In particular, currently some LLMs support handling long contexts, such as Claude 2.1[4] with a 200K context window. In this way, leveraging our method with such long context LLMs enables the completion of numerous utterances within the conversation session, which indicates one big step towards prolonged dialogue learning with long contexts. In summary, our main contributions are as follows:

(1) We discover that EoU tokens have the potential to aggregate utterance information. By defining these separator tokens as "conv-attn sinks," we propose StreamingDialogue, which efficiently handles long context by only caching the first token, conv-attn sinks, and tokens from the most recent two utterances.

(2) We propose two learning strategies: short-memory reconstruction (SMR) and long-memory reactivation (LMR), enhancing the capability of conv-attn sinks to aggregate information and the ability to store historical information.

(3) We demonstrate that StreamingDialogue significantly reduces computational complexity experimentally, ensuring the efficiency of streaming conversations.

---

[4]https://www.anthropic.com/news/claude-2-1

2

2 Related work

StreamingDialogue efficiently handles long context, improving the model's long-term memory for conversation history. Existing methods for processing long context in transformer-based models broadly fall into three categories: efficient transformer design, long-term memory enhancement, and length extrapolation techniques.

2.1 Efficient transformers

Due to attention's computational bottleneck in transformers, some methods aim to explore efficient attention mechanisms. Solutions include trading accuracy for speed, e.g., Longformer [14] employs sliding window attention, expanding the receptive field with a dilated sliding pattern and optionally integrating global attention. BP-Transformer [16] balances complexity and capacity with fine-to-coarse attention across multiple scales using binary partitioning. Linformer [17] approximates self-attention with a low-rank matrix, simplifying operations to linear ones. LongLoRA [18] uses block-wise attention and token shifting to enhance communication between blocks. Another solution lies in system-level optimizations, e.g., FlashAttention [19, 20] optimizes memory access by perceptually reading and writing, improving efficiency without sacrificing accuracy. However, these methods don't preserve dialogue history or expand the context window sufficiently for prolonged dialogue with long-term memory.

2.2 Long-term memory

Some methods enhance models' long-term memory to improve long-text modeling. One approach is introducing recurrent mechanisms into attention, enabling the model to maintain information over long sequences. For example, Transformer-XL [21] introduces segment-level recurrence, reusing previous time step hidden states to model long dependencies. ∞-former [22] employs continuous-space attention for arbitrary context modeling with fixed computational cost. Another approach is utilizing existing models as interfaces to external knowledge bases, enhancing contextual input and long-term memory through reading and writing to these bases during inference [23], e.g., MemGPT [24] employs hierarchical memory for LLMs, optimizing information transfer between context windows and external storage. However, they require retraining LLMs from scratch or additional information retrieval, lacking efficiency.

2.3 Length extrapolation

Length extrapolation in models refers to their ability to maintain good performance beyond the training length during inference. A mainstream solution is based on position encoding. LLMs [25–27] employ rotary position embedding (RoPE) [28–31] for length extrapolation without fine-tuning. Initially introduced by Chen et al. [32], position interpolation proportionally extends the inference length by reducing rotation angles. NTK-aware[5] and NTK-by-parts[6] interpolations balance high and low-frequency information to optimize performance. YaRN [33] combines NTK-by-parts interpolation with an attention distribution correction strategy, reducing rotation angles for low frequencies and adjusting attention distribution. Additionally, randomized position encoding [34] extends context exposure by decoupling pre-training length from inference length, utilizing random positions during training for broader context coverage. Due to current methods' inability for infinite length extrapolation, they're unsuitable for prolonged dialogue in streaming applications.

3 StreamingDialogue

3.1 Empirical observation

StreamingLLM [15] focuses on initial tokens as attention sinks, i.e., initial tokens attract a significant amount of attention. After visualizing the attention maps of all layers and heads for both Llama-2-7B and Llama-2-7B-Chat, we observe a similar phenomenon in structured texts such as multi-turn

---

[5]https://www.reddit.com/r/LocalLLaMA/comments/14lz7j5/ntkaware_scaled_rope_allows_llama_models_to_have/

[6]https://github.com/jquesnelle/yarn/pull/1

3
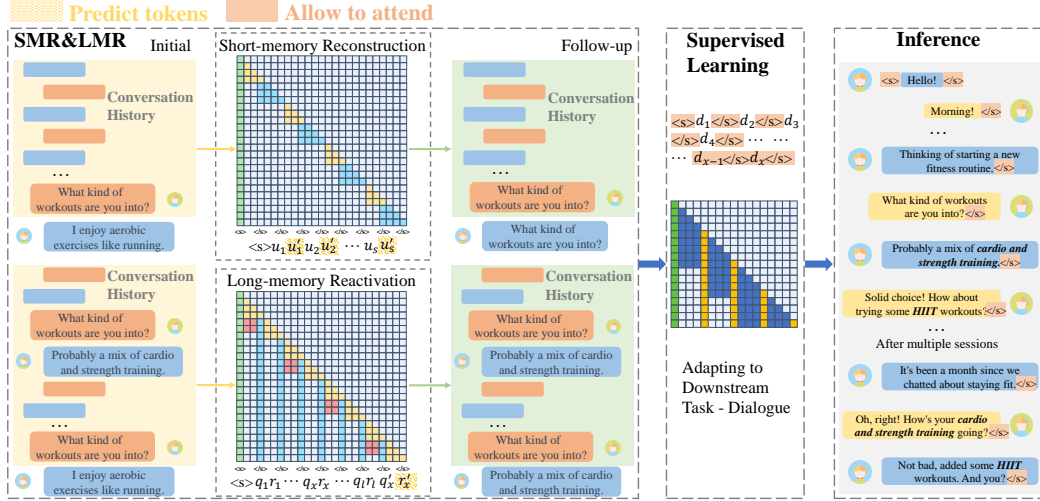
Figure 2: StreamingDialogue framework. SMR & LMR strategies co-train the model by adjusting attention mechanisms. In supervised learning, the SMR & LMR-trained model is fine-tuned with dialogue datasets. During inference, only specific tokens are cached, with critical historical dialogue information in bold italics for clarity.

dialogue, where LLMs tend to attend more to tokens used to separate dialogue when speakers switch (i.e., the end symbol "</s>," newline symbol "\n," or other symbols, known as End-of-Utterance), and their attention aggregation is even greater than that of initial tokens (shown in Figure 1). Based on the attention map, we suggest that each EoU captures the information of the current utterance, and EoUs are visible to subsequent utterances. These EoUs imply the potential for information aggregation, which we refer to as "conversational attention sinks" (conv-attn sinks).

According to the observation, rather than caching entire utterances to retain information as done in dense attention, we cache conv-attn sinks as a replacement. Let T represent the number of utterances, and L denote the average length of each utterance. By caching only the corresponding conv-attn sinks, the space complexity reduces from $O(TL)$ in dense attention to $O(T)$, and the time complexity from $O(T^2L^2)$ to $O(T^2L)$. Moreover, given the infrastructure of LLMs based long context modeling, our method is capable of efficiently handling dialogues with prolonged conversational histories. To this end, conv-attn sinks matter because they memorize the context information not only effectively, but also efficiently.

3.2 Framework overview

3.2.1 Fine-tuning LLMs with conv-attn sinks

We compress the content of each utterance into the subsequent conv-attn sink and recall historical information during the dialogue by attending to conv-attn sinks. To achieve this, we adjust the attention pattern $A \in \{0,1\}^{N \times N}$, where N represents the conversation length and 0 indicates masked attention values, i.e., specifying the specific keys and values that a query can attend to. Each token within an utterance can focus on the first token to uphold stable output during extended conversations, all preceding conv-attn sinks to extract historical context, and tokens from both the previous and current utterances to ensure continuity with the preceding utterance. Formally, we denote an utterance as $u = d$</s>, where d represents the dialogue content and </s> denotes the EoU token, a.k.a., conv-attn sink. Thus, a conversation can be organized as $D =$ <s>$u_1 u_2 ... u_t$, where t is the number of utterances. Attention mask matrix A is defined as:

$$
A_{ij} = \begin{cases} 1, & j = kl \leq i \, (k \in \mathbb{N}), \, 0 \leq i < N \\ 1, & 1 \leq j \leq i \leq l \\ 1, & j \neq kl \, (k \in \mathbb{N}), \, (\lceil \frac{i}{l} \rceil - 2) \cdot l < j \leq i < N \\ 0, & \text{otherwise}, \end{cases}
$$

4

where l denotes the average length of each utterance, $\mathbb{N}$ represents a non-negative integer, and $\lceil \cdot \rceil$ represents the ceiling function. During fine-tuning, all tokens are treated as predicted tokens to participate in the loss calculation.

While this method excels in managing long contexts compared to sparse attention methods like StreamingLLM, it falls short in characterizing short-term memories. To learn towards a more robust model with balanced memory capacity for both long and short memories, we propose two learning strategies to co-train the model and address these issues: short-memory reconstruction (SMR) and long-memory reactivation (LMR). The final version of StreamingDialogue is conducted in three stages, including SMR & LMR, supervised learning, and inference, as illustrated in Figure 2.

3.2.2 Short-memory reconstruction

We propose a learning strategy to guide model behavior, enabling conv-attn sinks to consciously aggregate information through short-memory reconstruction (SMR). We reorganize data formats, modify the attention pattern, and adjust the loss function. More specifically, training samples are organized as $D = \texttt{<s>}u_1 u_1' u_2 u_2' \ldots u_s u_s'$, where $u_1, u_2, \ldots, u_s$ are randomly selected from the original dataset, and s represents the number of randomly selected utterances. Each "uu'" pair can be regarded as a reconstruction task, where tokens in u can attend to $\texttt{<s>}$ and tokens that appear before the token in the current utterance. u' can additionally attend to the conv-attn sink in u. The task objective is to reconstruct u in u', encouraging the conv-attn sink in u to aggregate information from u for utterance reconstruction. The attention pattern A in SMR is defined as:

$$
A_{ij} = \begin{cases} 1, & j = 0 \\ 1, & \lceil \frac{i}{l} \rceil = 2k \ (k \in \mathbb{N}^*), (\lceil \frac{i}{l} \rceil - 1) \cdot l \leq j \leq i < N \\ 1, & \lceil \frac{i}{l} \rceil = 2k + 1 \ (k \in \mathbb{N}), (\lceil \frac{i}{l} \rceil - 1) \cdot l < j \leq i < N \\ 0, & \text{otherwise,} \end{cases}
$$

where $\mathbb{N}^*$ represents a positive integer.

Since the goal is to reconstruct the contents of $u_1', u_2', \ldots, u_s'$ into $u_1, u_2, \ldots, u_s$, the loss calculation is defined as:

$$
\mathcal{L}_{\text{SMR}} = - \sum_{(x,y) \in \mathcal{Z}} \sum_{t=1}^{|y|} \log \left(P_\Phi \left(y_t \mid x, y_{<t} \right) \right),
$$

where $\mathcal{Z} = \{(x_i, y_i)\}_{i=1,\ldots,N}$ denotes the set of (u, u') pairs. x denotes the target utterance u, and y represents the reconstructed utterance u'. The model learns to aggregate information into conv-attn sinks during SMR with minimal training.

3.2.3 Long-memory reactivation

The model is required to both aggregate dialogue information into conv-attn sinks and retrieve information from them. To ensure consistency in multi-turn dialogue, our proposed model must efficiently extract long context information during dialogue generation. Therefore, we introduce long-memory reactivation (LMR) to enhance its long-term memory capability.

Each pair of utterances in the dialogue dataset, denoted as qr, represents a query-response pair with q and r from distinct roles. We organize training samples as $D = \texttt{<s>}q_1 r_1 \ldots q_x r_x \ldots q_l r_l q_x' r_x'$, with l denoting the number of training pairs. Each pair $q_x' r_x'$ at the end of the sample is randomly selected from historical dialogues.

We design a response recall task where the goal is to recall r_x' from the historical context $q_x r_x$ given query q_x'. Concurrently, we adjust A so that each utterance can only attend to the first token, all conv-attn sinks, and itself. Moreover, each response in a training pair can attend to the corresponding query, while the conv-attn sink of the response is restricted to attending only to the response itself,

5

ensuring that the conv-attn sink aggregates information solely from its associated utterance, i.e.,

$$A_{ij} = \begin{cases} 1, & j = 0 \\ 1, & \lceil \frac{i}{l} \rceil = 2k \ (k \in \mathbb{N}^*), (\lceil \frac{i}{l} \rceil - 2) \cdot l < j \le i < N \\ 1, & j = kl \ (k \in \mathbb{N}^*) \ \text{or} \ j > \lceil \frac{i}{l} \rceil - 1, \ j \le i < N \\ 0, & \text{otherwise.} \end{cases}$$

Since the objective is to evoke historical dialogue within the response of the last training pair in the sample, the loss function of LMR is defined as:

$$\mathcal{L}_{LMR} = - \sum_{(m,n)} \sum_{t=1}^{|n|} \log\left(P_\Phi\left(n_t \mid m, n_{<t}\right)\right),$$

where m represents $\texttt{<s>}q_1 r_1 \ldots q_x r_x \ldots q_l r_l q_x'$, and n denotes r_x'.

The model now effectively utilizes historical information through LMR. We co-train the model using SMR and LMR, fully harnessing the information aggregation potential of conv-attn sinks and enhancing both short and long-term memory capability of the proposed model.

Following SMR & LMR, the model requires additional refinement through fine-tuning on dialogues using the methods outlined in Section 3.2.1.

Table 1: Main results on the PersonaChat and MSC datasets. ↓ indicates lower values are better, while ↑ indicates the opposite. The best result for each metric is presented in bold, while the second-best one is underlined. * indicates significance ($p < 0.05$) via pairwise t-test compared to other methods. "PC" denotes PersonaChat and "StrLLM" represents StreamingLLM.

Data	Method	PPL ↓	BLEU (%)			ROUGE (%)			Distinct (%)			USL-H (%) ↑	Dial-M ↓
			B-avg ↑	B-1 ↑	B-2 ↑	R-1 ↑	R-2 ↑	R-L ↑	D-1 ↑	D-2 ↑	D-3 ↑		
PC	Dense	8.41*	13.15	49.30*	20.05*	13.98	3.07	13.44	16.37*	41.61*	63.36*	14.21*	2.38*
	Local	11.59*	13.01	50.78	20.13*	13.83	2.69	13.29	12.49*	32.17*	51.12*	17.35*	2.07
	Big Bird	9.00*	12.93*	50.00*	20.52*	13.78	2.64	13.33	11.83*	32.46*	52.17*	16.95*	2.37*
	StrLLM	8.96	13.16	50.15	20.68	13.94	2.73	13.36	12.00*	32.64*	52.36*	17.63*	2.30*
	MemBART	13.15*	11.18*	46.63*	17.65*	13.11	2.56	12.78	12.86*	30.87*	48.86*	12.23*	2.49*
	Ours	8.71	**13.63**	**51.27**	**20.77**	13.96	3.05	13.43	14.43	37.23	58.07	**17.96**	2.10
MSC	Dense	7.58	19.47	52.22	28.41	16.93	2.92	15.48	12.85*	37.75*	57.51*	90.11*	1.94*
	Local	8.92*	13.34*	41.14*	20.44*	13.48*	1.88*	12.61*	7.89*	22.71*	35.89*	76.68*	2.15*
	Big Bird	8.42*	16.54*	46.63*	24.77*	15.32*	2.34	14.15*	8.72*	25.81*	40.34*	85.30*	1.72
	StrLLM	8.38*	16.76*	47.54*	25.08*	15.25*	2.44*	14.21*	9.18*	26.93*	41.62*	86.91*	**1.71**
	MemBART	13.73*	17.11*	49.78*	25.82*	14.93*	2.61	13.76*	10.86*	30.55*	47.37*	85.13*	1.97*
	Ours	7.99	19.33	51.49	28.12	**17.18**	2.77	**15.86**	11.54	32.58	50.27	**90.48**	1.76

4 Experiments

4.1 Experimental setup

Datasets & baselines We conduct experiments on PersonaChat [35], Multi-Session Chat (MSC) [36], Topical-Chat [37] and MultiWOZ [38] datasets. MSC, known for its extended conversational context, differs from PersonaChat, which is single-session. MSC's training set includes up to 4 sessions, and the test set comprises 5 sessions. Please refer to Appendix A for detailed information on the datasets. We use only the dialogue portions of all datasets and explore the impact of additional knowledge from various datasets in Appendix F.

Baseline methods include dense attention [7] for capturing all information, local attention [14] with a fixed window size restriction, Big Bird [39] combining sliding window, global, and random attention, and StreamingLLM [15] attending to attention sinks alongside the recent fixed window. Additionally, we compare our method with several methods that support long contexts, i.e., MemBART [40], a memory-augmented Transformer encoder-decoder model, and HRED [41] and VHRED [42], which utilize hierarchical encoders capable of encoding conversations of arbitrary length[7].

---

[7]Appendix B provides attention features and implementation details for both the baselines and StreamingDialogue

6

Evaluation metrics   We evaluate model performance on the dialogue generation task using several metrics: BLEU (B-avg / B-1 / B-2, where B-avg is the average of BLEU-1 to BLEU-4 scores) [43], ROUGE (R-1 / R-2 / R-L) [44], and Distinct (D-1 / D-2 / D-3) [45]. Additionally, we utilize two reference-free metrics specifically designed for dialogue quality assessment: USL-H [46] and Dial-M [47]. Perplexity (PPL) is also computed. Furthermore, we report the C scores [48] for various models on PersonaChat to assess the consistency of the generated dialogues.

4.2   Main results

Table 1 shows evaluation results for each method on the test sets (partial results related to baselines HRED and VHRED, metric C score, and datasets Topical-Chat and MultiWOZ, are available in Appendix C). For generation, we use the last utterance from each test set episode as the target for the model to generate. We also calculate the overall PPL for the entire test sets.

Our method outperforms sparse attention and memory-augmented baselines, achieving higher scores in BLEU, ROUGE, Distinct, and USL-H, while maintaining lower PPL and Dial-M metrics. For example, in MSC, StreamingDialogue demonstrates significant improvements over the second-best baseline, StreamingLLM, with B-avg increasing from 16.76% to 19.33% and R-L rising from 14.21% to 15.86%. In PersonaChat, specifically in D-2, StreamingDialogue increases from 32.64% to 37.23% compared to StreamingLLM. The notable superiority of StreamingDialogue can be attributed to its focus on conv-attn sinks, which compress historical information into them and cache them to enhance long-term memory, unlike baselines that rely on local attention windows and cannot handle extended dialogues effectively.

Furthermore, StreamingDialogue exhibits comparable performance to dense attention, e.g., in PersonaChat, the difference in ROUGE is less than 0.02%. It also outperforms dense attention in terms of R-1 and R-L in MSC and achieves better BLEU scores on PersonaChat. This validation highlights the more accurate information conveyance capabilities of text generated by our method.

	Win	Tie	Loss
Fluency	34%	35%	31%
Coherence	41%	22%	37%
Consistency	44%	27%	29%
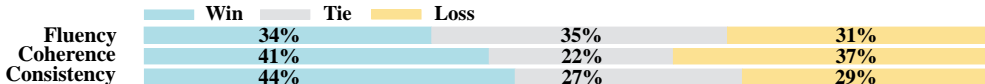
Figure 3: Fluency, coherence, and consistency in human evaluations: ours vs StreamingLLM.

4.3   Human evaluation

In human evaluation, we generate dialogues from 100 randomly selected episodes of the MSC test set. Four crowdsource evaluators compare our method with StreamingLLM in fluency, coherence, and consistency, categorizing the outcome as win, tie, or loss. Figure 3 demonstrates our method's superiority across all metrics, particularly in consistency, showcasing StreamingDialogue's superior long-term memory capacity. We apply Fleiss' kappa [49] to measure the agreement among four annotators, yielding a result of 52.51%. This indicates that the inter-annotator agreement is moderate ($\kappa \in [0.4, 0.6]$). More details on human evaluation are in Appendix D.

Table 2: Ablation results on MSC with different learning strategies. "Base" denotes the model fine-tuned without SMR and LMR learning.

Model	PPL	BLEU-avg	ROUGE-L	Distinct-3
Ours	7.99	19.33	15.86	50.27
Base	8.21	17.32	10.25	46.15
LMR	8.01	18.87	15.66	49.44
SMR	8.40	18.25	15.24	48.57

4.4   Ablation results

We conduct an experiment to test the effectiveness of SMR and LMR. Results, shown in Table 2, highlight a significant decline in model performance when either strategy is ablated, indicating the

7

importance of both strategies. The absence of SMR results in prominent declines in BLEU and ROUGE scores, indicating inadequate information aggregation in conv-attn sinks. Consequently, the model struggles to extract valuable information from conv-attn sinks during lengthy conversations, resulting in reduced text quality.

Similarly, without LMR, the model's performance declines significantly, indicating that relying solely on SMR leads to excessive guidance, limiting the model's ability in extended conversations. Thus, both SMR and LMR are crucial for enhancing information gathering and text extraction across conversations of long contexts.
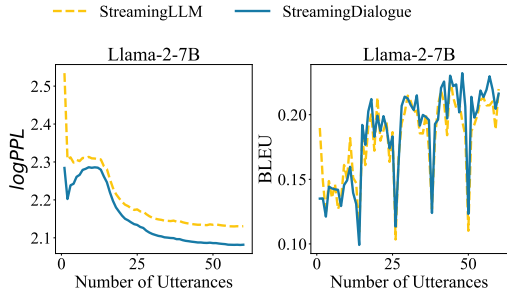
Figure 4: Average perplexity and BLEU for StreamingLLM and StreamingDialogue on the MSC test set across varying utterance counts.
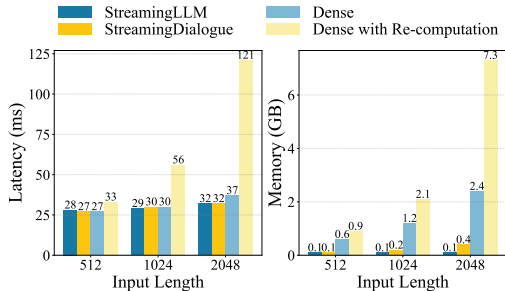
Figure 5: Per-token latency and memory usage by method on MSC for varying input lengths, with memory reported as total minus fixed.

4.5 Performance on different context length

We evaluate the model performance across different conversation lengths in terms of perplexity and BLEU under varying context length (i.e., the number of utterances in the dialogue context) during inference, as shown in Figure 4. As dialogue length increases, StreamingDialogue exhibits greater superiority over StreamingLLM, with perplexity stabilizing and nearing convergence, and BLEU improving. Furthermore, StreamingDialogue maintains stable perplexity even with prolonged conversations over 25K tokens in inference (see Figure 6). This highlights our method's stability in handling long dialogues and emphasizes the importance of conv-attn sinks in enhancing long-term memory.
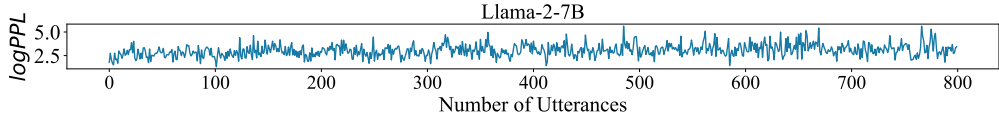
Figure 6: The perplexity for StreamingDialogue under the concatenated MSC test set, evaluating approximately 25K tokens.

Table 3: Results under the non-training setting on the MSC test set.

Model	Method	BLEU-avg	BLEU-1	BLEU-2	ROUGE-1	ROUGE-2	ROUGE-L
Llama-2-7B-Chat	StreamingLLM	20.16	51.18	29.99	15.90	1.92	14.26
	Ours	20.19	51.55	30.03	16.46	2.11	15.00
Llama-3-8B-Instruct	StreamingLLM	16.48	39.68	24.63	16.88	1.93	15.47
	Ours	16.77	40.10	24.88	17.11	2.01	15.85
Mistral-7B	StreamingLLM	12.75	42.86	19.99	12.58	1.83	11.73
	Ours	13.33	44.08	20.65	13.40	1.98	12.58

4.6 Performance under the non-training setting

We validate the performance of StreamingLLM and our method under a non-training setting on the MSC test set using Llama-2-7B-Chat, Llama-3-8B-Instruct [50] and Mistral-7B [51]. As the table

8

Layer 3 Head 1   Layer 11 Head 1   Layer 12 Head 8
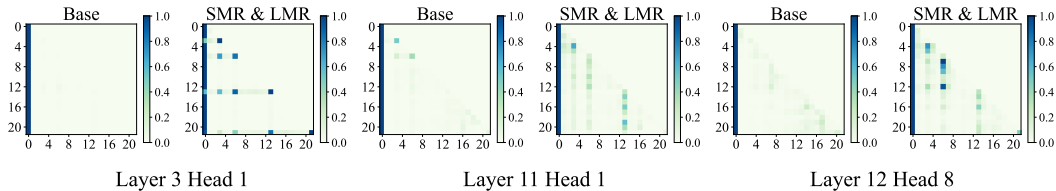
Figure 7: Comparison of attention maps before and after learning. "Base" denotes Llama-2-7B, while "SMR & LMR" represents the model obtained post co-training with SMR and LMR on Llama-2-7B. The "</s>" positions in the encoded sentences are: 3, 6, 13, and 21.

3 shows, our StreamingDialogue, thanks to the conv-attn sinks, retains more complete historical information. Consequently, it consistently outperforms StreamingLLM. This demonstrates that applying conv-attn sinks for modeling long contexts remains effective even without any training.

Table 4: Dialogue reconstruction performance.

BLEU-avg	BLEU-1	BLEU-2	ROUGE-1	ROUGE-L
68.02	89.19	76.83	76.79	72.94

4.7 Information preservation

To assess our method's information compression loss, we use SMR-trained models to reconstruct dialogue content from the MSC test set, leveraging only the conv-attn sink of each utterance. Randomly selecting 6,000 utterances from the MSC test set, we present the average results in Table 4. Our method achieves a BLEU-1 score of 89.19%, signifying effective compression of dialogue information with minimal losses.

4.8 Speedup for inference

Figure 5 depicts the average per-token latency and memory usage during dialogue generation with NVIDIA A100 GPU using various methods. As input lengths increase, StreamingDialogue shows minimal growth in memory usage for caching conv-attn sinks, with latency exhibiting linear growth. This suggests that as dialogue length increases, StreamingDialogue's advantage becomes more promising. At an input length of 2,048, our method demonstrates a $6 \times$ improvement in memory usage compared to dense attention and an $18 \times$ improvement compared to dense attention with re-computation. In terms of per-token latency, our method shows a $4 \times$ improvement compared to dense attention with re-computation. More-

Figure 8: The generated dialogues by StreamingLLM and StreamingDialogue for the same input dialogue history from an MSC episode, with an average utterance length of $L = 32$ tokens. Bold italic indicates key information in the dialogue.

over, our method maintains similar latency and memory usage as StreamingLLM as context length varies.

4.9 Impacts of SMR & LMR learning

Since the motivation of SMR and LMR learning is to improve conv-attn sinks aggregation capability, we examine the attention maps after SMR and LMR co-training, comparing them with the base model to confirm enhancement. Results are illustrated in Figure 7. Guided by SMR and LMR, the model's attention patterns transform into maps that sharply concentrate on conv-attn sinks, showcasing our effective enhancement of their information aggregation ability.

9

4.10 Case study: memory capacity

To validate StreamingDialogue's effectiveness in enhancing long-term memory, we conduct a case study comparing it with StreamingLLM. Figure 8 illustrates content generated by both methods. StreamingLLM responds solely based on recent utterances, lacking connection to distant context and coherence, thus reaffirming its unsuitability for open-domain dialogue. In contrast, StreamingDialogue effectively recalls distant historical information (e.g., 44 utterances ago), demonstrating the model's enhanced ability to remember long conversations through SMR and LMR.

4.11 Generality of conv-attn sinks

To establish the generality of the conv-attn sink phenomenon, where separators attract more attention than other tokens within dialogues, we conduct both qualitative and quantitative analyses. Qualitatively, we visualize attention patterns across different training methods, attention mechanisms, and types of dataset constructions, demonstrating that this phenomenon persists regardless of these variables. Quantitative measures further support these findings, with a set threshold indicating significantly higher attention on separators than on other tokens. Detailed results, including visualizations and statistical data, are provided in Appendix E. This comprehensive analysis confirms the robustness of the conv-attn sink phenomenon across different settings and models.

4.12 Analysis of EoU tokens' information aggregation capability

To assess the effectiveness of EoU tokens in capturing dialogue information, we conduct experiments using an untrained Llama-2-7B-Chat model. The model focuses solely on EoU tokens and the last complete utterance. In a case study with the conversation: "Did you have a caramel macchiato today?</s>Yes!</s>What kind of coffee did you have today?</s>," the model successfully identifies the key information, responding with, "I had a delicious caramel macchiato this morning."

We replicate this experiment across 10 different prompt formats, each containing 20 samples. The results indicate that 68% of the model's responses accurately include essential information. Further details on these formats are available in Appendix G.

4.13 Hyper-parameter sensitives

We investigate the impact of two hyper-parameters in our method: the number of utterances in SMR samples (s) and the number of query-response pairs in LMR samples (l), both ranging from $\{8, 12, 16, 20, 24, 28, 32\}$. We maintain a constant total number of utterances in the training set, with examples like $s = 8 \times$ samples $= 24,000$ and $s = 12 \times$ samples $= 16,000$. Results shown in Figure 9 reveal that StreamingDialogue performs best with higher values of s and l, optimally at $s \in \{28, 32\}$ and $l \in \{20, 24, 28\}$. This suggests that longer texts enhance the model's learning.
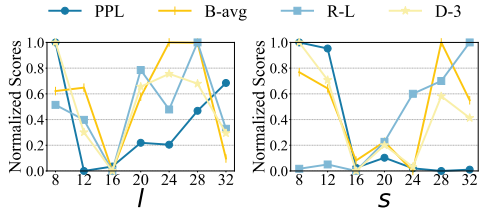
Figure 9: Normalized performance scores (PPL, B-avg, R-L, and D-3) on MSC for various l with s fixed at 28 and various s with l fixed at 24.

5 Conclusion

Generating high-quality open-domain dialogues with prolonged contexts is quite challenging. Existing solutions, like dense attention, have efficiency issues. While StreamingLLM supports efficient language modeling, it struggles to preserve historical information, leading to low-quality generation in prolonged conversations. In this paper, we introduce StreamingDialogue, a framework capable of facilitating efficient and prolonged dialogue. By identifying separator tokens EoU as "conv-attn sinks" and compressing dialogue information into them with minimal losses, StreamingDialogue conserves memory, enhances efficiency, and augments long-term memory capabilities. Additionally, we propose two learning strategies to enhance conv-attn sink aggregation and memory reactivation. Our method shows better performance compared to strong baselines. In the future, we will explore extending StreamingDialogue towards never-ending dialogue in the context of lifelong learning.

10

Acknowledgments and Disclosure of Funding

We sincerely thank Songhao Wu and Ang Lv for their valuable discussion and feedback on the manuscript.

This work was supported by the National Natural Science Foundation of China (NSFC Grant No. 62122089, U21B2027), Beijing Outstanding Young Scientist Program NO. BJJWZYJH012019100020098, and Intelligent Social Governance Platform, Major Innovation & Planning Interdisciplinary Platform for the "Double-First Class" Initiative, Renmin University of China, the Fundamental Research Funds for the Central Universities, and the Research Funds of Renmin University of China.

References

[1] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.

[2] Aiyuan Yang, Bin Xiao, Bingning Wang, Borong Zhang, Ce Bian, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, Dong Yan, et al. Baichuan 2: Open large-scale language models. *arXiv preprint arXiv:2309.10305*, 2023.

[3] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[4] Yuhan Liu, Xiuying Chen, Xiaoqing Zhang, Xing Gao, Ji Zhang, and Rui Yan. From skepticism to acceptance: Simulating the attitude dynamics toward fake news. *arXiv preprint arXiv:2403.09498*, 2024.

[5] Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023.

[6] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[8] Liu Yuhan, Chen Xiuying, and Yan Rui. Unleashing the power of large models: Exploring human-machine conversations. In *Proceedings of the 22nd Chinese National Conference on Computational Linguistics (Volume 2: Frontier Forum)*, pages 16–29, 2023.

[9] Chia-Hsuan Lee, Hao Cheng, and Mari Ostendorf. Dialogue state tracking with a language model using schema-driven prompting. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4937–4949, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021. emnlp-main.404. URL `https://aclanthology.org/2021.emnlp-main.404`.

[10] Songhao Wu, Quan Tu, Hong Liu, Jia Xu, Zhongyi Liu, Guannan Zhang, Ran Wang, Xiuying Chen, and Rui Yan. Unify graph learning with text: Unleashing llm potentials for session search. In *Proceedings of the ACM on Web Conference 2024*, pages 1509–1518, 2024.

[11] Quan Tu, Yanran Li, Jianwei Cui, Bin Wang, Ji-Rong Wen, and Rui Yan. MISC: A mixed strategy-aware model integrating COMET for emotional support conversation. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 308–319, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10. 18653/v1/2022.acl-long.25. URL `https://aclanthology.org/2022.acl-long.25`.

11

[12] Björn Bebensee and Haejun Lee. Span-selective linear attention transformers for effective and robust schema-guided dialogue state tracking. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 78–91, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.6. URL `https://aclanthology.org/2023.acl-long.6`.

[13] Ang Lv, Jinpeng Li, Yuhan Chen, Gao Xing, Ji Zhang, and Rui Yan. DialoGPS: Dialogue path sampling in continuous semantic space for data augmentation in multi-turn conversations. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1267–1280, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.70. URL `https://aclanthology.org/2023.acl-long.70`.

[14] Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

[15] Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. *arXiv preprint arXiv:2309.17453*, 2023.

[16] Zihao Ye, Qipeng Guo, Quan Gan, Xipeng Qiu, and Zheng Zhang. Bp-transformer: Modelling long-range context via binary partitioning. *arXiv preprint arXiv:1911.04070*, 2019.

[17] Sinong Wang, Belinda Z Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020.

[18] Yukang Chen, Shengju Qian, Haotian Tang, Xin Lai, Zhijian Liu, Song Han, and Jiaya Jia. Longlora: Efficient fine-tuning of long-context large language models. *arXiv preprint arXiv:2309.12307*, 2023.

[19] Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in Neural Information Processing Systems*, 35:16344–16359, 2022.

[20] Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*, 2023.

[21] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G Carbonell, Quoc Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988, 2019.

[22] Pedro Henrique Martins, Zita Marinho, and Andre Martins. ∞-former: Infinite memory transformer. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5468–5485, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.375. URL `https://aclanthology.org/2022.acl-long.375`.

[23] Yunpeng Huang, Jingwei Xu, Zixu Jiang, Junyu Lai, Zenan Li, Yuan Yao, Taolue Chen, Lijuan Yang, Zhou Xin, and Xiaoxing Ma. Advancing transformer architecture in long-context large language models: A comprehensive survey. *arXiv preprint arXiv:2311.12351*, 2023.

[24] Charles Packer, Vivian Fang, Shishir G Patil, Kevin Lin, Sarah Wooders, and Joseph E Gonzalez. Memgpt: Towards llms as operating systems. *arXiv preprint arXiv:2310.08560*, 2023.

[25] Baptiste Roziere, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, et al. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*, 2023.

[26] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

12

[27] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[28] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

[29] Yuhan Chen, Ang Lv, Ting-En Lin, Changyu Chen, Yuchuan Wu, Fei Huang, Yongbin Li, and Rui Yan. Fortify the shortest stave in attention: Enhancing context awareness of large language models for effective tool use. *arXiv preprint arXiv:2312.04455*, 2023.

[30] Xiaoran Liu, Hang Yan, Shuo Zhang, Chenxin An, Xipeng Qiu, and Dahua Lin. Scaling laws of rope-based extrapolation. *arXiv preprint arXiv:2310.05209*, 2023.

[31] Hongzhan Lin, Ang Lv, Yuhan Chen, Chen Zhu, Yang Song, Hengshu Zhu, and Rui Yan. Mixture of in-context experts enhance llms' long context awareness. *arXiv preprint arXiv:2406.19598*, 2024.

[32] Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. Extending context window of large language models via positional interpolation. *arXiv preprint arXiv:2306.15595*, 2023.

[33] Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. Yarn: Efficient context window extension of large language models. *arXiv preprint arXiv:2309.00071*, 2023.

[34] Anian Ruoss, Grégoire Delétang, Tim Genewein, Jordi Grau-Moya, Róbert Csordás, Mehdi Bennani, Shane Legg, and Joel Veness. Randomized positional encodings boost length generalization of transformers. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1889–1903, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-short.161. URL `https://aclanthology.org/2023.acl-short.161`.

[35] Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. Personalizing dialogue agents: I have a dog, do you have pets too? In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2204–2213, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1205. URL `https://aclanthology.org/P18-1205`.

[36] Jing Xu, Arthur Szlam, and Jason Weston. Beyond goldfish memory: Long-term open-domain conversation. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5180–5197, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.356. URL `https://aclanthology.org/2022.acl-long.356`.

[37] Karthik Gopalakrishnan, Behnam Hedayatnia, Qinlang Chen, Anna Gottardi, Sanjeev Kwatra, Anu Venkatesh, Raefer Gabriel, and Dilek Hakkani-Tür. Topical-Chat: Towards Knowledge-Grounded Open-Domain Conversations. In *Proc. Interspeech 2019*, pages 1891–1895, 2019. doi: 10.21437/Interspeech.2019-3079. URL `http://dx.doi.org/10.21437/Interspeech.2019-3079`.

[38] Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Inigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. Multiwoz–a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. *arXiv preprint arXiv:1810.00278*, 2018.

[39] Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33: 17283–17297, 2020.

13

[40] Qingyang Wu and Zhou Yu. Stateful memory-augmented transformers for dialogue modeling. *arXiv preprint arXiv:2209.07634*, 2022.

[41] Alessandro Sordoni, Yoshua Bengio, Hossein Vahabi, Christina Lioma, Jakob Grue Simonsen, and Jian-Yun Nie. A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, page 553–562, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450337946. doi: 10.1145/2806416.2806493. URL https://doi.org/10.1145/2806416.2806493.

[42] Iulian Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron Courville, and Yoshua Bengio. A hierarchical latent variable encoder-decoder model for generating dialogues. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.

[43] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In Pierre Isabelle, Eugene Charniak, and Dekang Lin, editors, *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL https://aclanthology.org/P02-1040.

[44] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://aclanthology.org/W04-1013.

[45] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. In Kevin Knight, Ani Nenkova, and Owen Rambow, editors, *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1014. URL https://aclanthology.org/N16-1014.

[46] Vitou Phy, Yang Zhao, and Akiko Aizawa. Deconstruct to reconstruct a configurable evaluation metric for open-domain dialogue systems. *arXiv preprint arXiv:2011.00483*, 2020.

[47] Suvodip Dey and Maunendra Sankar Desarkar. Dial-m: A masking-based framework for dialogue evaluation. In *Proceedings of the 24th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 77–84, 2023.

[48] Andrea Madotto, Zhaojiang Lin, Chien-Sheng Wu, and Pascale Fung. Personalizing dialogue agents via meta-learning. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5459, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1542. URL https://aclanthology.org/P19-1542.

[49] Joseph L Fleiss. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378, 1971.

[50] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[51] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

[52] Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. GLM: General language model pretraining with autoregressive blank infilling. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.26. URL https://aclanthology.org/2022.acl-long.26.

14

[53] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL `https://aclanthology.org/N19-1423`.

15
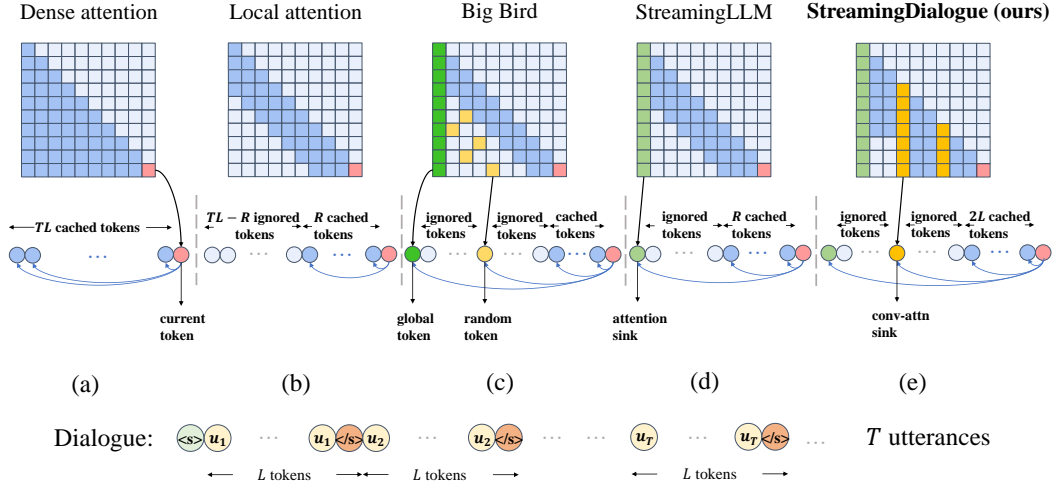
Figure 10: Attention maps' visualization of StreamingDialogue and various other methods. In a dialogue with T utterances, each averaging L tokens, dense attention caches TL tokens, local attention caches R tokens (where R is the window size), Big Bird caches global size + random size + R tokens, StreamingLLM caches $R + 1$ tokens, and StreamingDialogue requires caching up to $1 + T + 2L$ tokens.

A   Dataset details

Refer to Table 5 for specific details regarding the PersonaChat, MSC, Topical-Chat and MultiWOZ datasets.

Table 5: Details of dialogue datasets. We present the number of utterances (Utts.) and the average length per utterance (Avg. L) for each session in the training and test sets.

Data	Data Type	Train		Test	
		Utts.	Avg. L	Utts.	Avg. L
PersonaChat	Total	122499	13.59	14602	13.85
MSC	Session 1	59894	14.16	6572	15.47
	Session 2	46420	31.44	5939	30.86
	Session 3	47259	32.90	5924	32.94
	Session 4	11870	32.25	5940	34.67
	Session 5	-	-	5945	36.43
	Total	165443	25.66	30320	29.77
Topical-Chat	Total	188378	26.76	11760	26.98
MultiWOZ	Total	113552	18.92	14744	19.23

B   Attention patterns & implementation details

We visualize the attention maps of StreamingDialogue and several other baselines focusing on different attention patterns, as shown in Figure 10. Below are the details of the experiments for each method.

During the SMR & LMR phase, we construct a short-memory reconstruction dataset comprising $n_s = 6857$ samples, each containing a random selection of $s = 28$ utterances from dialogue datasets. Simultaneously, for the long-memory reactivation dataset, we generate $n_l = 8000$ samples, each consisting of a random selection of $l = 24$ query-response pairs from the full set of query-response pairs in dialogue datasets, with one additional pair randomly chosen from the l pairs appended to the end of each sample. The SMR and LMR datasets are merged and shuffled for co-training Llama-2-7B.

16

We only train the attention layer for 1 epoch, with the learning rate set to 5e-5, utilizing cosine annealing for adjusting the learning rate, and setting the warm-up step to 0. The SMR & LMR phase requires about 2 hours on two A100-40G GPUs.

In the supervised learning phase, StreamingDialogue undergoes fine-tuning based on the model trained with SMR & LMR, while baselines focusing on different attention patterns are fine-tuned on Llama-2-7B. All models fine-tune only the attention layer for 2 epochs, with the learning rate set to 5e-5, utilizing cosine annealing for adjusting the learning rate, and setting the warm-up step to 0. This phase demands only approximately 1 hour on two A100-40G GPUs. MemBART, HRED and VHRED can be fine-tuned directly on dialogue datasets.

In inference, we set the maximum generation length to 120 and report the average results of all episodes. Dialogue generation takes only about 15 minutes on a single A100-40G GPU.

C   Additional results for main evaluation

Table 6 presents the experimental results incorporating baselines HRED and VHRED. Our method achieves favorable results compared to any baseline across all metrics.

Table 6: Results on the MSC dataset. ↓ indicates lower values are better, while ↑ indicates the opposite. The highest-performing result for each metric is highlighted in bold, and the second-highest is underlined.

Method	B-avg ↑	R-1 ↑	R-2 ↑	D-1 ↑	D-2 ↑	USL-H ↑	Dial-M ↓
StreamingLLM	16.76	<u>15.25</u>	<u>2.44</u>	<u>9.18</u>	<u>26.93</u>	<u>86.91</u>	**1.71**
HRED	15.72	14.75	1.85	7.37	20.91	58.70	2.13
VHRED	<u>17.02</u>	15.16	1.48	5.28	14.72	59.31	2.35
Ours	**19.33**	**17.18**	**2.77**	**11.54**	**32.58**	**90.48**	<u>1.76</u>

Table 7 reports the results of the C score on PersonaChat. Our StreamingDialogue still achieves the best results among the baselines, except for dense attention. As an efficient algorithm, our method can significantly improve the speed compared to dense attention while maintaining the contextual and character consistency of long conversations.

Table 7: Results of the C score on the PersonaChat dataset. ↑ indicates higher values are better.

Method	Dense	Local	Big Bird	StreamingLLM	MemBART	Ours
C (%) ↑	3.10	-3.40	-4.00	-4.70	0.77	2.70

Table 8 presents the results for the Topical-Chat and MultiWOZ datasets, where our method outperforms all strong baselines by retaining more complete historical information.

D   Human evaluation details

D.1   Information about evaluators

We engage four crowdsource evaluators to assess the performance of our method. These evaluators exhibit outstanding English proficiency, enabling them to accurately discern subtle nuances and meanings in language expressions. Moreover, they possess a comprehensive understanding of the distinctions among fluency, coherence, and consistency, and are adept at determining which response is superior. In terms of human evaluation, we pass the review by relevant institutions, and we anonymize all evaluators' responses. After a reasonable assessment of the workload, we pay each evaluator $1.70 per 10 samples.

D.2   Task description

Table 9 presents the detailed task description provided to evaluators, where responses generated by different models are randomized in each sample.

17

Table 8: Results on the Topical-Chat and MultiWOZ datasets. ↓ indicates lower values are better, while ↑ indicates the opposite. The highest-performing result for each metric is highlighted in bold, and the second-highest is underlined.

Data	Method	PPL ↓	ROUGE-1 ↑	ROUGE-2 ↑	ROUGE-L ↑	Dial-M ↓
Topical-Chat	Dense	**9.49**	**15.70**	<u>3.65</u>	**14.88**	3.09
	Local	27.55	12.60	2.09	10.37	7.02
	Big Bird	10.36	14.21	3.55	11.79	3.01
	StreamingLLM	10.34	14.25	3.55	11.84	3.05
	MemBART	12.54	13.86	2.98	13.18	<u>2.83</u>
	Ours	<u>9.80</u>	<u>15.46</u>	**3.99**	<u>14.37</u>	**2.66**
MultiWOZ	Dense	<u>4.51</u>	<u>24.79</u>	<u>13.93</u>	<u>24.67</u>	<u>2.27</u>
	Local	5.38	24.26	13.47	24.15	2.45
	Big Bird	4.79	24.38	13.26	24.30	2.51
	StreamingLLM	4.76	23.66	13.09	23.41	2.47
	MemBART	5.36	20.05	12.41	19.94	2.37
	Ours	**4.34**	**25.26**	**14.27**	**25.20**	**2.25**

Table 9: Task description provided to evaluators in human evaluation.

Task description

We aim to evaluate the quality of dialogues generated by various models, specifically focusing on fluency, coherence, and consistency. You will be presented with a dialogue history followed by two responses generated by different models for the latest utterance. For each evaluation metric—fluency, coherence, and consistency—please identify the superior response. Assign a "1" if the first response is better, a "2" if the second response is better, and "0" if both responses are of equal quality. Ensure that your selections clearly reflect which response better meets each metric.

The specific descriptions of the three metrics are as follows:

(1) **Fluency** assesses whether the response itself is well-written and grammatically correct.
(2) **Coherence** refers to the response being relevant to the content of the historical dialogue.
(3) **Consistency** requires the response to remain consistent with the persona information and objective facts from the historical dialogue.

E   Details on the generality of conv-attn sinks

To demonstrate the generality of the conv-attn sink phenomenon, i.e., the model aggregates more attention on separators than on other words and tokens, we provide both qualitative and quantitative analysis.

For **qualitative analysis**, we visualize the attentions using various training methods, attention mechanisms, and dataset constructions to observe the behavior of attention patterns.

(1) Regarding the independency on the training methods, we compare Llama2 with GLM [52]. Llama2 is pretrained by the next-token prediction objective, while GLM is pretrained on an autoregressive blank infilling objective.

(2) Regarding the independency on the attention mechanisms, we compare Llama2 with BERT [53]. Llama2 uses a unidirectional causal attention mechanism, while BERT uses a bidirectional attention mechanism.

(3) Reagarding the independency on the dataset constructions, we compare two-person dialogues with multi-party dialogues. Otherwise, we also have validated that the different separators for orgnizing the dialogue also exhibit the same phenomenon, as illustrated in Figure 1.

The visualization results under different settings are shown in Figure 11, which demonstrates the consistent conclusion that separators will attract more attention than other tokens in the dialogue. For the two-person dialogue, the data we use is "<BOS>Hey!<EOS>Morning<EOS>Wanna grab coffee later?<EOS>Totally! Starbucks?<EOS>," and for the multi-party dialogue, the data we use is "<BOS>Alice: How was your weekend<EOS>Bob: Good, yours?<EOS>Charlie: Great,

18
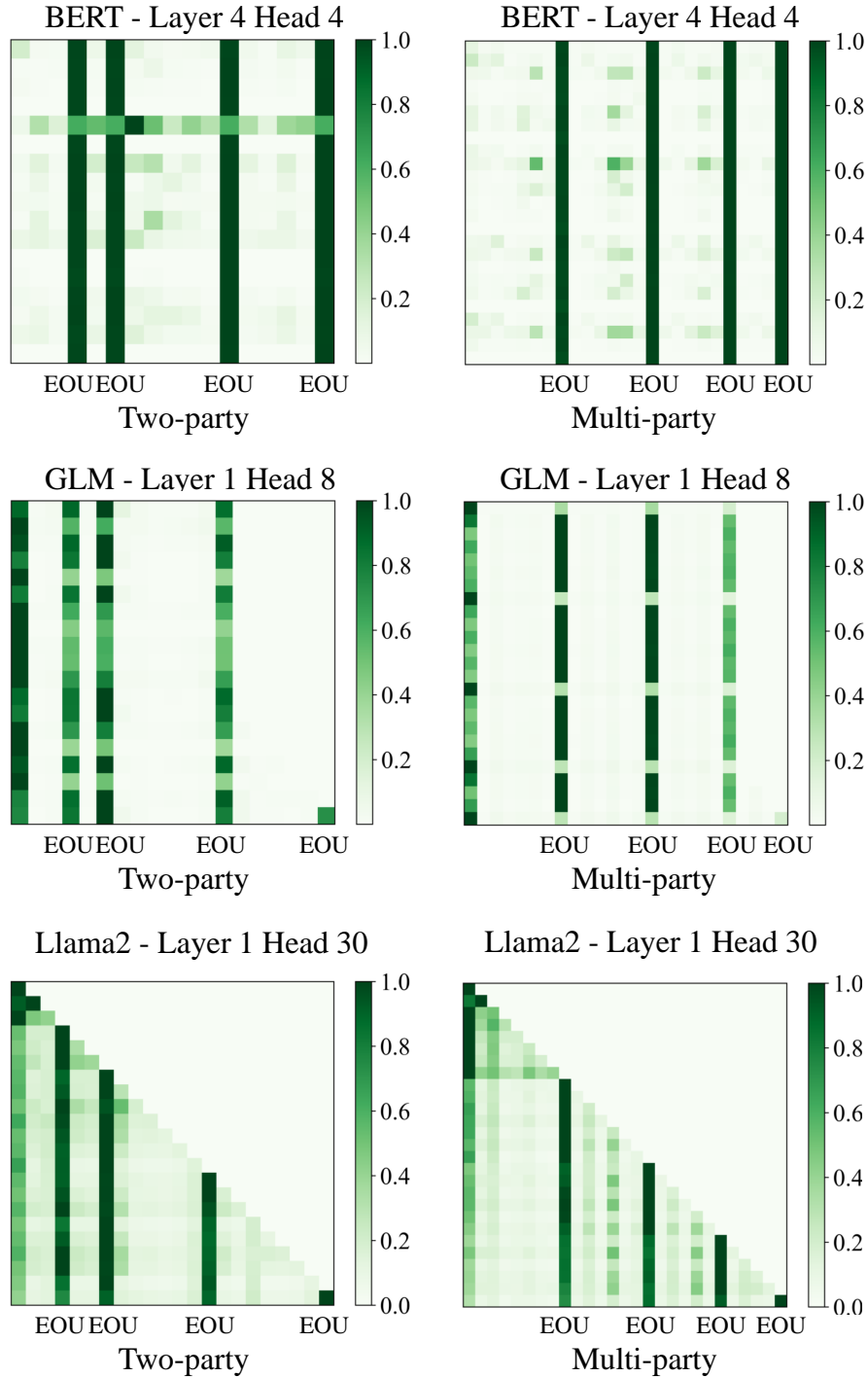
Figure 11: Attention maps under different settings.

thanks<EOS>Alice: Awesome<EOS>." "<BOS>" denotes the start symbol, and "<EOS>" denotes the end symbol.

For **quantitative analysis**, we set a threshold indicating the occurrence of the conv-attn sink phenomenon in an attention head. This threshold is defined as the average attention aggregated by conv-attn sinks being three times or more than that aggregated by normal tokens. We report the

19

Table 10: Proportion of attention heads exhibiting the conv-attn sink phenomenon across models.

Model	Llama2	GLM	BERT
Proportion (%)	14.36	16.32	79.86

proportion of these conv-attn heads among all attention heads using three heterogeneous models, as depicted in the Table 10. This illustrates the universality of the conv-attn sink. Furthermore, we observe a stronger tendency of the conv-attn sink in the BERT model, warranting further investigation in our future work.

F   Impact of additional knowledge

The original datasets used for both training and testing contain some additional knowledge: Persona-Chat includes personas, Topical-Chat includes grounding knowledge, and MultiWOZ includes belief states. Our main experiments utilize only the dialogue portions of these datasets, without leveraging the additional knowledge. To make our results more compelling, we explore the impact of incorporating this extra knowledge into our experiments.

For MultiWOZ, we add the belief states before each corresponding utterance. The results are shown in the Table 11.

Table 11: The results of integrating the belief states on the MultiWOZ dataset.

Method	PPL	BLEU-avg	BLEU-1	BLEU-2	Distinct-1	Distinct-2	Distinct-3
Dense	1.92	25.56	48.33	29.14	3.74	6.86	8.89
StreamingLLM	2.19	25.70	47.53	29.21	4.48	9.09	12.60
Ours	1.98	25.77	48.58	29.38	5.30	10.03	13.60

Since our method retains historical information by compressing each utterance's information into conv-attn sinks, only the conv-attn sinks from the previous utterances will be attended to in subsequent utterances. Therefore, for Topical-Chat and Persona-Chat, we consider two settings:

(1) We treat each sentence of the grounding knowledge/persona profiles as an utterance, and the subsequent utterances can only attend to their conv-attn sinks. The results are shown in the Table 12.

Table 12: Results on the Topical-Chat and Persona-Chat datasets under the setting of treating each sentence of the grounding knowledge/persona profiles as an utterance.

Data	Method	PPL	Distinct-2	Distinct-3	Dial-M
PersonaChat	Dense	7.19	43.56	66.27	2.53
	StreamingLLM	8.36	33.17	53.58	2.47
	Ours	7.60	39.16	61.06	2.36
Topical-Chat	Dense	3.24	39.07	57.64	4.32
	StreamingLLM	8.31	16.87	23.56	3.72
	Ours	3.20	31.47	49.10	2.57

(2) We use the grounding knowledge/persona profiles as a prompt: "The conversation will be based on the following knowledge: <knowledge> detailed knowledge <conversation>" in Topical-Chat and "The conversation will be based on the following persona profile: <persona> detailed persona profiles <conversation>" in Persona-Chat, allowing the subsequent utterances to fully attend to it. The results are shown in the Table 13.

In a setting that includes grounding knowledge, our method consistently retains the memory of both grounding knowledge and historical dialogue. As a result, our method still outperforms the baseline, except for dense attention. As an efficient algorithm, our method can significantly improve speed compared to dense attention while maintaining the contextual and character consistency of long conversations.

20

Table 13: Results on the Topical-Chat and Persona-Chat datasets under the setting of treating the grounding knowledge/persona profiles as a prompt.

Data	Method	PPL	Distinct-2	Distinct-3	Dial-M
PersonaChat	Dense	7.93	44.26	66.63	2.48
	StreamingLLM	7.99	36.40	57.44	2.91
	Ours	7.67	37.82	58.93	2.57
Topical-Chat	Dense	11.64	36.98	54.96	4.60
	StreamingLLM	30.37	26.07	34.26	3.61
	Ours	10.21	32.16	50.41	2.97

G Examples of prompt formats

Examples of prompt formats are as follows, where the "keywords" will be replaced with specific content.

1. "template": "A and B went to PLACE today.</s>They had a great time.</s>Who did A go to PLACE with today?</s>",
   "keywords": "A": "person", "B": "person", "PLACE": "place",
   "answer key": "B"

2. "template": "B made A's favorite food, FOOD, today.</s>A was delighted.</s>What food did B make for A today?</s>",
   "keywords": "A": "person", "B": "person", "FOOD": "food",
   "answer key": "FOOD"

3. "template": "A was doing ACTIVITY when B called.</s>A had to stop and answer the call.</s>What was A doing when B called?</s>",
   "keywords": "A": "person", "B": "person", "ACTIVITY": "activity",
   "answer key": "ACTIVITY"

4. "template": "A bought a new ITEM today.</s>B was impressed by A's purchase.</s>What item did A buy today?</s>",
   "keywords": "A": "person", "B": "person", "ITEM": "item",
   "answer key": "ITEM"

5. "template": "A participated in an EVENT today.</s>B cheered them on.</s>What event did A participate in?</s>",
   "keywords": "A": "person", "B": "person", "EVENT": "event",
   "answer key": "EVENT"

H Limitations

StreamingDialogue significantly reduces space and time complexity during the inference stage. Additionally, we can outperform the baseline under the non-training setting without additional cost. To optimize LLMs for the conv-attn sinks mode, we implement two learning strategies: short-memory reconstruction and long-memory reactivation. Consequently, this inevitably increases computational costs under the training setting, with the SMR and LMR phases requiring about two hours on two A100-40G GPUs.

While StreamingDialogue effectively enables prolonged conversations with long-term memory, there's merit in exploring selective caching of conv-attn sinks, focusing only on those aggregating key information. This will further enhance inference speed and reduce memory usage. Additionally, our utilization of dialogue structure is somewhat limited, and we aim to leverage conv-attn sinks to explore more intricate dialogue features in the future. Furthermore, evaluating our method across a wider range of structured texts will offer a more comprehensive assessment.

21

I   Broader impacts and safety issues

This paper identifies the conversational attention sink (conv-attn sink) phenomenon and proposes two learning strategies to naturally compress extended dialogue history into conv-attn sinks and effectively retrieve memories in subsequent conversations, thereby enabling prolonged streaming dialogue. Additionally, the datasets utilized, including PersonaChat and MSC, as well as the pre-trained models Llama-2-7B and Llama-2-7B-Chat, were sourced from their respective publishers through official open-source channels. Our enhancements are purely architectural and we will not release any new models or datasets. Nonetheless, the capability to efficiently compress and retrieve long dialogues may allow this technology to be used for monitoring conversations over extended periods, thus raising privacy concerns and the potential for surveillance without proper consent or transparency.

22

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In the abstract, we claim that StreamingDialogue efficiently compresses dialogue history into conversational attention sinks with minimal losses, enhancing the model's long-term memory and facilitating prolonged streaming conversations.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We create a separate "Limitations" section, i.e., Section H.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

23

Answer: [NA]

Justification: Our paper does not introduce theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We have shared a URL that contains executable code, enabling easy reproduction of our experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

24

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We have open-sourced our code and shared the link.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: See Section 4 and appendices.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We present the statistical significance of the experiments in Table 1.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

25

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: We provide detailed information on the computing resources required to reproduce the experiments in Section B.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

Answer: [Yes]

Justification: We have conformed the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: See Section I.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

26

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [Yes]

Justification: See Section I.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We have correctly cited all the data, scripts, and models we used.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

27

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: We have included a README document with our code.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [Yes]

Justification: See Section D.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [Yes]

Justification: See Section D.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

28