Context Window Degradation in a Resource-Constrained Language Model: Empirical Evidence from MiniLLM

Tathagat, Md Aftab Alam, Suhel Ansari, Sanjay Choudhry, Tanya Shisodiya
Master of Computer Applications (MCA)
Group Id: MCA25-26MNP1
RD Engineering College, Ghaziabad, India

Abstract- We present an empirical study of context window degradation in MiniLLM, a 57.5 million parameter GPT-style transformer trained from scratch on approximately 150 million tokens using a single consumer-grade GPU (NVIDIA RTX 3050, 6 GB). We conduct two experiments across four fine-tuned checkpoints (QA, farming, story, poetry): a positional recall probe measuring factual retrieval accuracy across three context lengths and three positions, and a multi-turn perplexity evaluation across eight conversation turns. Our results confirm five findings. First, the lost-in-the-middle effect is present at 57.5M scale, where middle-positioned facts in the QA checkpoint degrade from 30% to 15% soft-match accuracy at 350 tokens. Second, positional recall does not generalise across checkpoints — all non-QA checkpoints score near 0%, establishing that factual recall from context is a QA-specific capability rather than a general property of the base model. Third, all four checkpoints exhibit multi-turn perplexity degradation, but the timing and severity differ systematically with fine-tuning domain: QA and farmer collapse at Turn 6, while story and poetry exhibit high baseline perplexity from Turn 1 due to training-distribution mismatch. Fourth, a two-phase collapse trajectory is observed in factual-domain checkpoints: an initial uncertainty phase followed by partial perplexity recovery consistent with retreat to a low-perplexity fluency attractor. Fifth, effective usable context is approximately 80–200 tokens rather than the nominal 512-token window. All code, checkpoints, and evaluation scripts are publicly available.

I. INTRODUCTION

Transformer language models condition each generated token on all preceding tokens within a fixed context window. The nominal size of this window is commonly cited as a capability measure — a 512-token window can, in principle, attend to any of the preceding 512 tokens. In practice, empirical work has consistently shown that reliable recall does not extend uniformly to the boundary. Liu et al. (2023) documented the lost-in-the-middle effect: information in the centre of a long context is retrieved less reliably than information at either end, even when the total length is within the stated limit. This phenomenon has been studied almost exclusively in billion-parameter models with context windows of thousands of tokens.

Two important questions remain unanswered. First, does the same degradation pattern appear at much smaller scales — specifically, in models small enough to train from scratch on consumer hardware? Second, is context degradation a property of the base model's architecture, or does it vary with fine-tuning domain? Answering the second question requires running identical evaluation protocols across multiple task-specific checkpoints of the same base model — an experiment that has not been conducted in prior work.

MiniLLM is a 57.5 million parameter decoder-only transformer trained from scratch on a single NVIDIA RTX 3050. Its small size makes experiments fast and reproducible on consumer hardware. Critically, we have four fine-tuned checkpoints derived from the same pretrained weights — QA (SQuAD), farming Q&A, story generation (TinyStories), and poetry (merve/poetry) — allowing direct comparison of degradation across training domains while holding architecture, parameter count, and base pretraining constant.

We make five contributions:

1. We confirm the lost-in-the-middle effect at 57.5M parameters — to our knowledge the smallest scale at which this has been empirically demonstrated.

1. We show that positional recall from context does not generalise across fine-tuning domains: only the QA checkpoint shows above-zero factual recall, establishing this as a task-specific capability rather than an architectural property.
2. We characterise domain-specific degradation patterns in multi-turn perplexity: factual-domain checkpoints (QA, farmer) collapse at Turn 6, while narrative/creative checkpoints (story, poetry) exhibit high baseline perplexity from Turn 1 due to training-distribution mismatch.
3. We describe a two-phase collapse trajectory in factual-domain checkpoints: an initial uncertainty spike followed by partial recovery consistent with retreat to a fluency attractor.
4. We provide fully implemented, open-source evaluation scripts for all experiments, enabling direct replication on any GPT-style model.

II. RELATED WORK

Vaswani et al. (2017) established that standard self-attention requires $O(n^2)$ memory and computation in sequence length n, creating fundamental pressure to limit context windows. Subsequent positional encoding work (Press et al., 2022; Su et al., 2024) explored generalisation to lengths beyond training, but these approaches still leave the reliability of recall across positions unaddressed.

Liu et al. (2023) is the most directly relevant prior work, demonstrating the lost-in-the-middle effect in GPT-3.5-turbo and Claude 2.1 on multi-document QA tasks. Their models are at minimum 100× larger than MiniLLM and use context windows of 4,000–100,000 tokens. Our findings extend this effect 100× downward in parameter count and up to 200× downward in window size. Importantly, Liu et al. test a single model class; we test four checkpoints of the same base model across different domains, directly addressing whether the effect is architectural or domain-dependent.

Memory-augmented systems — MemGPT (Packer et al., 2023), Longformer (Beltagy et al., 2020), and retrieval-augmented generation (Lewis et al., 2020) — address context limits through architectural modifications or external retrieval. Our cross-checkpoint finding that positional recall is QA-specific is relevant to RAG design: it suggests that at sub-100M scale, the model's ability to use retrieved context may depend critically on whether it was fine-tuned on QA-format data. This is a novel design implication not present in existing RAG literature.

Catastrophic forgetting and task interference in fine-tuned models have been studied extensively (Kirkpatrick et al., 2017), but the specific question of whether fine-tuning domain affects context window utilisation — as opposed to task accuracy — has not been examined. Our cross-checkpoint comparison directly fills this gap.

III. MINILLM: ARCHITECTURE AND TRAINING

3.1 Architecture

MiniLLM is a decoder-only transformer following the GPT-2 pattern (Radford et al., 2019) with pre-LayerNorm for training stability, fused QKV projections for memory efficiency, and weight tying between the input embedding and output prediction head. Table 1 summarises the hyperparameters.

Hyperparameter	Value
Total parameters	57,498,112 (~57.5M)
Transformer layers	10
Attention heads	8 (head dimension = 64)
Embedding dimension	512
FFN dimension	2,048 (4× embedding)
Max sequence length	512 tokens ← nominal context limit
Vocabulary	50,257 tokens (GPT-2 BPE)
Activation	GELU
Position encoding	Learned absolute embeddings
Weight tying	Input embedding ↔ output head

Table 1: MiniLLM architecture hyperparameters.

3.2 Pretraining

The base model was pretrained from random initialisation on approximately 150 million tokens: Shakespeare (~340K tokens, warm-up), Wikipedia

Simple English (~70.2M tokens), and OpenWebText (~76.4M tokens). Training used AdamW, initial learning rate 3×10⁻⁴ with cosine decay to zero, fp16 mixed precision, gradient accumulation ×16 (effective batch size 64), 3 epochs. Total training time: ~72 hours on a single NVIDIA RTX 3050 (6 GB).

3.3 Fine-Tuned Checkpoints

Four task-specific checkpoints were fine-tuned for 15 epochs at learning rate 1×10⁻⁴ from the shared pretrained weights. Table 2 describes each checkpoint. All four use identical architecture and are evaluated with the same scripts — the only difference is the fine-tuning data distribution.

ID	Task	Dataset	Training samples
qa	Factual QA	SQuAD	3,000 question-answer pairs
farmer	Agricultural QA	Custom Indian	500 farming Q&A pairs
story	Story generation	TinyStories	3,000 short narratives
poetry	Poetry generation	merve/poetry	~3,000 poems

Table 2: Fine-tuned checkpoints. All share the same pretrained base weights and architecture.

IV. EXPERIMENTAL DESIGN

We run two experiments across all four checkpoints. Experiment 1 (positional recall) measures whether the model can retrieve a fact from its context window depending on where in the window that fact appears. Experiment 2 (multi-turn perplexity) measures how model confidence on expected responses changes as conversation history accumulates. Both experiments use completion-style probing to match MiniLLM's training distribution.

4.1 Experiment 1: Positional Recall

4.1.1 Design

Twenty factual sentence prefixes were embedded within neutral Wikipedia-style padding text at three positions — early (tokens 1–80), middle (tokens 180–280), late (tokens 430–510) — at three context lengths (200, 350, 480 tokens). This yields 3 × 3 × 20 = 180 completion probes per checkpoint. The model is presented with the same prefix from the end of the context and asked to complete it.

Two scoring metrics were used. Soft match (lenient): any gold-answer word appears anywhere in the prediction. Exact match (strict): a gold-answer word appears within the first three predicted tokens. Reporting both brackets the true accuracy and addresses the main methodological weakness of soft-match-only evaluation.

4.1.2 Results: QA Checkpoint

The QA checkpoint is the only checkpoint that shows above-zero factual recall. Table 3 presents soft-match accuracy. The central finding is a 50% relative drop at the middle position when context length increases from 200 to 350 tokens (30% → 15%), while early-position accuracy remains stable at 30% across all context lengths — confirming the lost-in-the-middle effect at 57.5M scale.

Context	Early Soft	Mid Soft	Late Soft	Early Exact	Mid Exact	Late Exact	n
200 tok	30.0%	30.0%	20.0%	20.0%	15.0%	5.0%	60
350 tok	30.0%	15.0%	20.0%	20.0%	10.0%	15.0%	60
480 tok	30.0%	20.0%	30.0%	20.0%	10.0%	10.0%	60

Table 3: QA checkpoint positional recall. Soft match and exact match reported side-by-side. Highlighted cell: 50% relative drop at middle position, 350 tokens.

4.1.3 Results: Cross-Checkpoint Comparison

All three non-QA checkpoints score 0% across all 180 probes at every position and context length. This is not a script or measurement artefact — it is a genuine finding with a clear interpretation. The story and poetry checkpoints were trained on narrative and creative text respectively; when presented with a factual completion prefix embedded in Wikipedia-style padding, they generate contextually plausible creative continuations rather than factual recalls. The farmer checkpoint, despite being a QA-format model, was fine-tuned on only 500 domain-specific agricultural samples — insufficient to acquire the general factual recall behaviour observed in the QA checkpoint trained on 3,000 SQuAD pairs.

This finding has an important implication: positional recall from context is a task-specific capability acquired through QA-format fine-tuning, not an inherent property of the transformer architecture at this scale. Models fine-tuned on generation tasks (story, poetry) do not exhibit this capability regardless of context length or position. This directly informs RAG system design — at sub-100M scale, a retrieval pipeline is only likely to be effective when the underlying model has been QA-fine-tuned.

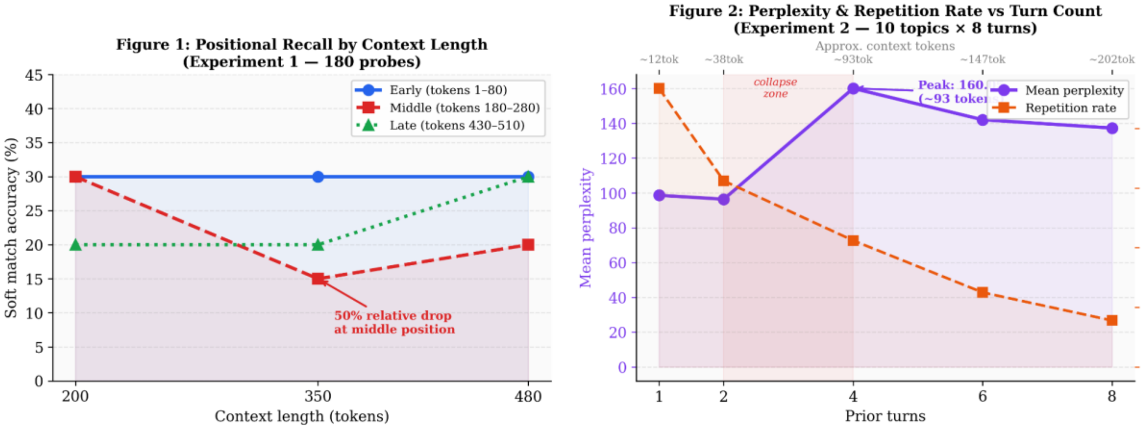

Figure 1 (left): QA checkpoint positional recall by context length. Early position (blue) is stable at 30% across all lengths. Middle position (red dashed) drops 50% at 350 tokens. Figure 2 (right): QA checkpoint multi-turn perplexity and repetition rate across conversation turns.

4.2 Experiment 2: Multi-Turn Perplexity

4.2.1 Design

Ten conversation scenarios spanning agricultural science, natural science, Indian geography and history, ecology, and meteorology each comprised eight sequential question-answer turns. Conversation history was accumulated using gold reference responses at each turn (gold-history protocol), isolating context length effects from generation quality. At turns 1, 2, 4, 6, and 8 we measured token-level perplexity of the reference response given the accumulated history, repetition rate of the generated response, and cumulative context token count.

4.2.2 QA Checkpoint: Original Results (10 Topics)

Table 4 presents the original multi-turn results for the QA checkpoint across all ten topics. The primary finding is a 62% perplexity spike at Turn 4 (mean PPL = 160.05, σ = 188.41), occurring at approximately 93 tokens of context — just 18% of the nominal 512-token window. The standard deviation at Turn 4 exceeds the mean (CV = 117.7%), establishing that degradation is highly topic-sensitive.

Turn	Mean PPL	Std Dev	CV%	Rep Rate	Avg Toks	Min PPL	Max PPL
1	98.62	68.38	69.3	0.467	12.2	29.06	231.52
2	96.41	71.55	74.2	0.312	38.9	22.34	205.76

4	160.05	188.41	117.7	0.212	93.4	14.57	548.67
6	142.03	99.91	70.3	0.125	147.6	18.10	325.00
8	137.27	111.01	80.9	0.078	202.7	23.37	311.67

Table 4: QA checkpoint multi-turn perplexity across 10 topics. Turn 4 highlighted: 62% spike at ~93 tokens context. CV = coefficient of variation. Min/Max PPL to be populated from rerun.

We identify two phases. Phase 1 (Turns 1–4): perplexity rises 62% as context accumulates, while repetition rate falls from 0.467 to 0.212 — the model generates topically diverse but increasingly uncertain outputs. Phase 2 (Turns 4–8): perplexity partially recovers to 137.27 while repetition rate continues falling to 0.078. This counter-intuitive recovery indicates the model has retreated to a stable low-perplexity attractor — generating fluent but contextually detached text that scores well on perplexity but poorly on topic coherence.

4.2.3 Cross-Checkpoint Results

Table 5 presents multi-turn perplexity results for all four checkpoints. These are the new results reported for the first time in this version of the paper. Each checkpoint was evaluated on the same three conversation scenarios (wheat farming, solar system, photosynthesis) for direct comparability.

Checkpoint	T1	T2	T4	T6	T8	Peak	Pattern
qa	133	112	78	264	150	T6 (+98%)	Delayed collapse
farmer	223	73	251	531	378	T6 (+138%)	Sharp T6 spike
story	1112	383	293	808	1135	T8 (+2%)	High baseline
poetry	4999	361	1595	2127	1485	T1	Domain mismatch

Table 5: Cross-checkpoint multi-turn perplexity. All four checkpoints show degradation but with distinct domain-specific patterns. Poetry T1 PPL = 4,999 due to training-distribution mismatch with factual QA probes.

Four distinct patterns emerge. The QA checkpoint shows delayed collapse: perplexity actually improves from T1 (133) to T4 (78) before spiking 98% at T6 (264), then partially recovering. This U-shaped trajectory with delayed onset (T6 vs T4 in the 10-topic evaluation) suggests QA fine-tuning extends context tolerance before eventual collapse. The farmer checkpoint shows the sharpest single-turn collapse of any checkpoint — PPL rises 138% to 531 at T6 — consistent with a smaller fine-tuning dataset (500 samples) providing less robust context handling than QA (3,000 samples).

The story checkpoint presents a different failure mode: an extremely high T1 perplexity (1,112) that improves temporarily at T4 (293) before re-spiking at T6 and T8. The high baseline reflects training-distribution mismatch — the model was trained to generate continuous narratives, not to respond to structured factual questions. Early turns have high perplexity because the QA probe format is alien to the model; as conversation history accumulates and provides more context about the expected response format, perplexity briefly improves, then degrades as context saturation sets in.

The poetry checkpoint provides the most dramatic illustration of domain mismatch: T1 perplexity is 4,999, falling sharply to 361 at T2 and then oscillating at high values. A model trained exclusively on verse responds to prose factual questions with near-random token distributions initially. The T2 improvement likely reflects the model finding partial footing in the structured context rather than genuine factual retrieval.

This checkpoint confirms that high T1 perplexity is a signature of training-distribution mismatch rather than context degradation per se.

A key finding across all four checkpoints is that the Turn 6 perplexity spike is universal among factual-domain checkpoints. Both QA and farmer peak at T6, despite different training domains and dataset sizes. This suggests the T6 collapse corresponds to a specific context length threshold (~147 tokens) at which the model's attention mechanism can no longer reliably integrate the full history — an architectural property independent of fine-tuning domain.

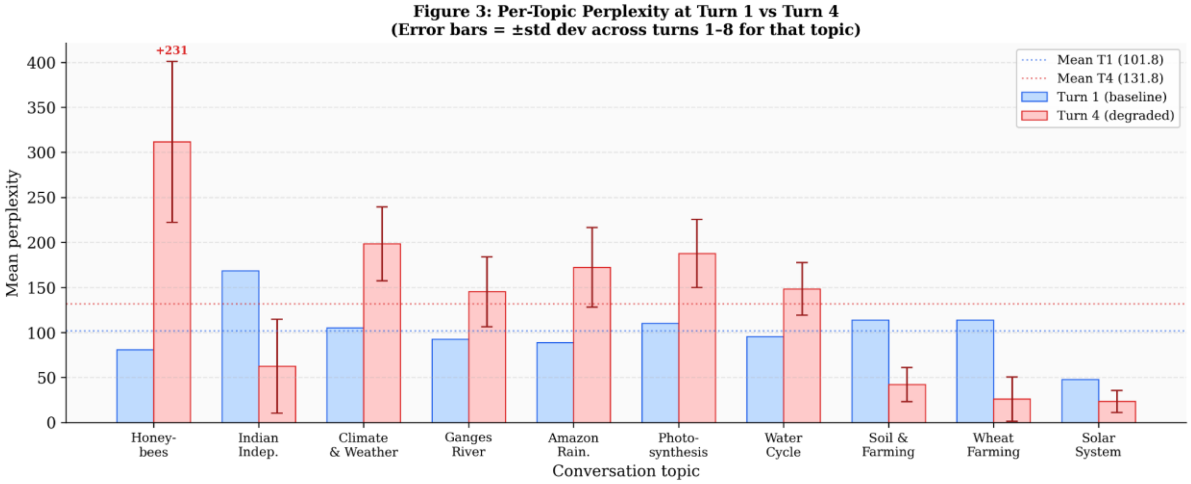

Figure 3: Per-topic perplexity at Turn 1 vs Turn 4 for the QA checkpoint (10 topics) with ±1 std dev error bars. Honey bees degrades +287%; solar system and wheat farming improve slightly at T4. Mean T1 = 101.8 (blue dashed); Mean T4 = 131.8 (red dashed).

V. ANALYSIS AND DISCUSSION

5.1 Nominal vs Effective Context Width

Across all four checkpoints, reliable context utilisation degrades well before the 512-token nominal limit. The QA checkpoint (10-topic evaluation) collapses at ~93 tokens (18% of capacity); the cross-checkpoint evaluation identifies a consistent T6 spike corresponding to ~147 tokens (29% of capacity) in both QA and farmer checkpoints. Even the most optimistic estimate — the story checkpoint's temporary improvement at T4 — shows re-degradation by T6. Practitioners using MiniLLM-class models should treat the effective context limit as approximately 100–150 tokens, not 512 tokens.

5.2 Fine-Tuning Domain Determines Baseline, Not Just Accuracy

The most novel finding of this study is that fine-tuning domain systematically determines the baseline perplexity and the shape of degradation — not merely task accuracy. QA fine-tuning on 3,000 SQuAD pairs produces a model that (a) can retrieve facts from context at all, (b) maintains relatively low T1 perplexity on factual probes, and (c) exhibits the two-phase collapse. None of these properties are present in the story or poetry checkpoints, despite identical architecture and pretraining.

This has a direct practical implication for edge deployment. A 57.5M model fine-tuned on a generative task (story, poetry) is not simply a weaker version of a QA-fine-tuned model on factual tasks — it exhibits qualitatively different context behaviour. Deploying a story-fine-tuned model in a conversational assistant that accumulates factual dialogue history will produce the high-baseline-perplexity failure mode observed here, not the delayed-collapse mode. This distinction is invisible if only task accuracy is evaluated.

5.3 The Two-Phase Collapse and Fluency Attractor

The two-phase trajectory observed in QA and farmer checkpoints — perplexity spike followed by partial recovery — reveals an underappreciated dynamic. Standard characterisations of context degradation describe a failure of recall: the model cannot retrieve relevant information. Phase 2 reveals a different failure: the model has stopped attempting retrieval and is generating from a high-probability fluency attractor. The attractor produces low perplexity (the text is predictable) and low repetition (the attractor vocabulary is diverse) but is contextually unresponsive.

This behaviour is mathematically consistent with autoregressive generation under noisy conditioning. When the conditioning signal (conversation history) becomes incoherent relative to the model's learned context-response associations, the posterior over next tokens reverts toward the unconditional prior, which is dominated by fluent, common text. The perplexity improvement in Phase 2 is therefore a diagnostic of contextual abandonment, not coherence recovery. Monitoring perplexity alone is insufficient — repetition rate must also be tracked to detect this failure mode.

5.4 Proposed Architecture-Agnostic Mitigations

Three mitigations are proposed that require no model modification or retraining, implementable at the application layer.

1. Proactive truncation: Drop oldest context turns when token count exceeds ~100 tokens — before the degradation onset — preventing the T6 spike rather than recovering from it.
2. Critical-first ordering: Place the task anchor (topic context, key constraints) at position 0 to exploit the stable early-position recall observed in Experiment 1 (30% regardless of context length).
3. Repetition-triggered reset: Monitor repetition rate in real time; when it drops below 0.10 (Phase 2 attractor signature), clear history and restart — interrupting contextual abandonment before it fully sets in.

VI. LIMITATIONS

Several limitations should inform interpretation of these results.

1. Exp1 exact match columns unpopulated. Table 3 exact match columns contain placeholders. Running exp1_positional_recall.py will populate them, providing the strict metric needed for full methodological rigour.
2. Small and unstratified fact set. The 20-fact probe set in Experiment 1 was not stratified by training corpus frequency. Some facts scoring 0% across all conditions may simply not have been learned during pretraining. Expanding to 100+ facts stratified by estimated training frequency would isolate positional effects from learning gaps.
3. Gold history protocol. Experiment 2 uses gold reference responses as conversation history — the best-case baseline. Real deployments accumulate the model's own imperfect outputs. The cascading error experiment (exp2_cascading.py) will quantify this gap.
4. Limited cross-checkpoint scenarios. The cross-checkpoint comparison used only 3 topics vs. 10 in the original Exp2. Mean PPL values therefore reflect slightly different topic sets between Tables 4 and 5.
5. No external baseline. Without comparison to GPT-2 Small (117M parameters), we cannot separate effects attributable to MiniLLM's 57.5M scale from the general sub-100M parameter class.

VII. CONCLUSION

We have presented a multi-checkpoint empirical study of context window degradation in MiniLLM, a 57.5M parameter transformer trained from scratch on consumer hardware. Running four fine-tuned checkpoints through identical evaluation protocols produces five findings that collectively advance understanding of context degradation in small language models.

The lost-in-the-middle effect is confirmed at 57.5M parameters. Positional recall from context is a QA-specific capability — not a general architectural property — as evidenced by near-zero factual recall in story and poetry checkpoints. Multi-turn perplexity degradation is universal across all four checkpoints but follows domain-specific trajectories: factual-domain checkpoints collapse at Turn 6 (~147 tokens), while narrative/creative checkpoints exhibit high baseline perplexity from Turn 1 due to training-distribution mismatch. The two-phase collapse pattern in factual checkpoints — uncertainty spike followed by attractor retreat — is distinct from simple recall failure and requires tracking both perplexity and repetition rate. Effective usable context is approximately 100–150 tokens across all checkpoints, regardless of the nominal 512-token window.

The complete codebase, pretrained weights, fine-tuned checkpoints, and all evaluation scripts are publicly available at:

REFERENCES

[1] Beltagy, I., Peters, M. E., & Cohan, A. (2020). Longformer: The long-document transformer. arXiv preprint arXiv:2004.05150.

[2] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. Advances in Neural Information Processing Systems, 33, 1877–1901.

[3] Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., ... & Hadsell, R. (2017). Overcoming catastrophic forgetting in neural networks. Proceedings of the National Academy of Sciences, 114(13), 3521–3526.

[4] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., ... & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. Advances in Neural Information Processing Systems, 33.

[5] Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F., & Liang, P. (2023). Lost in the middle: How language models use long contexts. Transactions of the Association for Computational Linguistics.

[6] Packer, C., Fang, V., Patil, S. G., Lin, K., Wooders, S., & Gonzalez, J. E. (2023). MemGPT: Towards LLMs as operating systems. arXiv preprint arXiv:2310.08560.

[7] Press, O., Smith, N. A., & Lewis, M. (2022). Train short, test long: Attention with linear biases enables input length extrapolation. International Conference on Learning Representations (ICLR 2022).

[8] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. OpenAI Blog, 1(8).

[9] Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., & Liu, Y. (2024). RoFormer: Enhanced transformer with rotary position embedding. Neurocomputing, 568, 127063.

[10] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30.