RNNs are not Transformers (Yet): The Key Bottleneck on In-Context Retrieval

Kaiyue Wen*[1], Xingyu Dang*[2], Kaifeng Lyu[3]

[1] Stanford University [2]Tsinghua University [3]Simons Institute, UC Berkeley → Tsinghua University *Equal Contribution

Transformers v.s. RNNs: Inference Cost

(L: input length)

Transformers
- **Memory Cost:** $O(L)$
- **Time Cost:** $O(L^2)$

RNNs
- **Memory Cost:** $O(1)$
- **Time Cost:** $O(L)$

Recent Trend: RNN-based LLMs
- **RWKV, RetNet**
- **State-Space Models (SSMs): S4, Mamba, Mamba 2**

Mind the Representation Gaps!

Problems that ∃Transformer can solve but no RNN can
- **Sparse averaging of inputs given a mask** (Sanford-Hsu-Telgarsky, 2023)
- **"k-hop" induction heads** (Sanford-Hsu-Telgarsky, 2023)
- **Copying the input itself** (Jelassi-Brandfonbrener-Kakade-Malach, 2024)

Transformers are not omnipotent, though

Problems that ∃RNN can solve but no Transformer can
- **Dyck** (Hahn, 2020) (infinite precision, hard attention)
- **Simulating a (non-solvable) semi-automaton** (Liu et al., 2023)

Maybe we just didn't use them correctly?

Chain-of-Thought (CoT) can improve the representation power!

Transformer:
- **{Problems solvable by Transformers directly}** $\subset \mathbf{TC^0}$ (Merrill & Sabharwel, 2023)
- $\boldsymbol{P} \subset$ **{Problems solvable by Transformers + CoT}** (Feng et al., 2023; Li et al., 2024)

Question: How does CoT improve RNNs?

(Setting: assuming O(log L)-bit precision by default; # params does not grow with L)

Main Results

Message 1: Even with CoT, RNNs struggle with many simple tasks that require retrieving information from the context ("in-context retrieval").

Associative Recall: Given a sequence of tokens and a query token q, output the next token of q in the sequence.

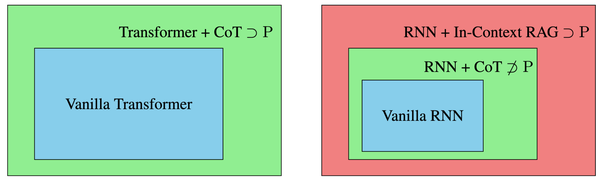
Index: Given a sequence of tokens and a query token i, output the i-th token in the sequence.

c-gram Retrieval: Query a c-gram in the sequence.

Counting: Count the number of a query token q.

Theorem: For each problem, ∃small & shallow Transformer that can solve it, but RNNs cannot, even with CoT.

Curse of Memory Efficiency: Even with CoT, RNNs are still using too little memory to store the entire context.



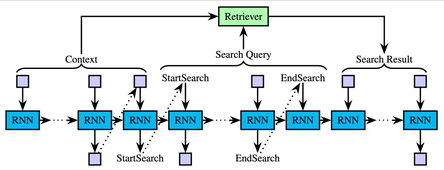
Transformer RNN (This work)

Message 2: Enhancing the in-context retrieval capability of RNNs can close the representation gap.

Solution 1: Explicitly Adding "In-Context" RAG

In-Context Retrieval Augmented Generation (RAG): Allow LLMs to call a function to retrieve information from the context via regular expression.

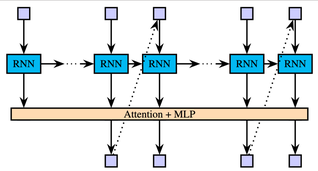
Theorem: With CoT and In-Context RAG, (linear) RNNs have enough representation power to solve all the problems in P.



Solution 2: Mixing with some Transformer layers

Of course, adding 1,000 Transformer layers may solve the issue. But what's the minimal possible change?

Theorem: With CoT, (linear) RNNs + just one Transformer layer at the end have enough representation power to solve all the problems in P.



Future Direction: Design better hybrid architecture based on representation/optimization theory?

When is In-Context Retrieval Needed?

Associative Recall (Long history in AI: Willshaw et al., 1969; Hopfield, 1982; Hinton & Anderson, 2014; Arora et al., 2023)

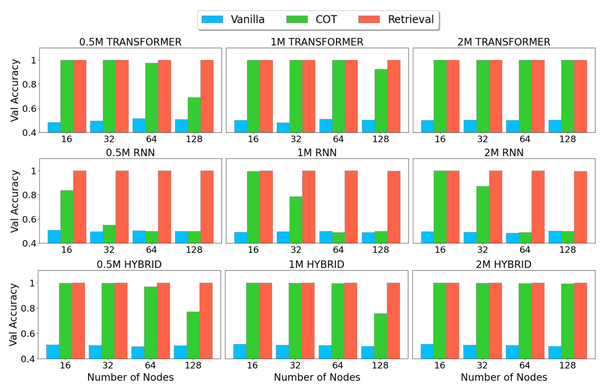
Example Text from Wikipedia:
- Singapore, officially the Republic of Singapore, is an island country and city-state in Southeast Asia ..... (very long text) ..... Singapore was not greatly affected by the First World War (1914–18), as the conflict did not spread to Southeast _____

Many reasoning tasks implicitly require retrieval (not always obvious)

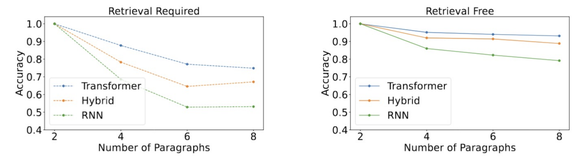
Example Task: IsTree (from the literature of streaming complexity)
- Given the description of an n-node graph, determine whether it is a tree.

Theorem: Transformers can solve IsTree with CoT of length $O(n)$, but RNNs cannot, even with CoT.

Experiments



Comparison of Different Architectures on isTree
- Transformer: Llama 2; RNN: Mamba; Hybrid: Mamba + 1 Transformer layer



Same Qualitative Behavior in natural language tasks

RNN's performance degrades faster when model needs to retrieve documents in the context to answer question compared to only looking at the most recent documents. (Data: Hotpot-QA, Model: Phi 1.5B vs Phi Mamba)