Do Language Models Understand the Cognitive Tasks Given to Them? Investigations with the *N*-Back Paradigm

Xiaoyang Hu
Brown University
xiaoyang_hu@brown.edu

Richard L. Lewis
University of Michigan
rickl@umich.edu

Abstract

Cognitive tasks originally developed for humans are now increasingly used to study language models. While applying these tasks is often straightforward, interpreting their results can be challenging. In particular, when a model underperforms, it is often unclear whether this results from a limitation in the cognitive ability being tested or a failure to understand the task itself. A recent study argues that GPT 3.5's declining performance on 2-back and 3-back tasks reflects a working memory capacity limit similar to humans (Gong et al., 2024). By analyzing a range of open-source language models of varying performance levels on these tasks, we show that the poor performance is due at least in part to a limitation in task comprehension and task set maintenance. We challenge the best-performing model with progressively harder versions of the task (up to 10-back) and experiment with alternative prompting strategies, before analyzing model attentions. Our larger aim is to contribute to the ongoing conversation around refining methodologies for the cognitive evaluation of language models.[1]

1 Introduction

Psychologists rely on behavioral experiments to test hypotheses about cognitive constructs and processes. For these experiments to be valid, participants have to understand exactly what they are being asked to do. To that end, human study protocols often include detailed task instructions, demonstrations, and practice runs. When adapting these experiments for language models, ensuring task comprehension can be more challenging, given that these models are often more hesitant than humans to express uncertainty (Zhou et al., 2024).

A recent study applies the n-back task (Figure 1) to GPT 3.5 and concludes from the model's poor 2-back and 3-back performance that it has a working memory capacity limit (WMCL) of approximately 3, apparently similar to humans (Gong et al., 2024). This interpretation raises two concerns. First, while WMCL is well established in human cognition, we cannot assume these same constraints exist or can be meaningfully measured in language models. Second, these results may reflect the model's failure to understand the task requirements rather than any inherent memory limitation.

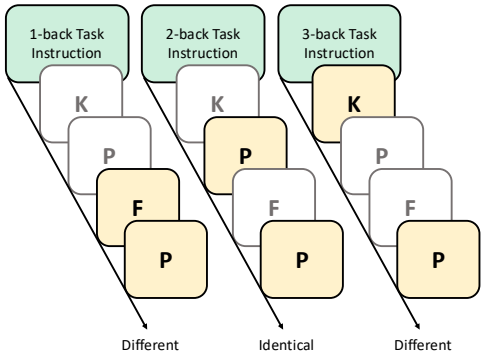

Figure 1: The n-back task is a common working memory task in which subjects are presented with a sequence of stimuli. At each step, they must decide whether the current item matches the one appearing n step(s) earlier. This requires them to continuously update a list of n most recent stimuli in the working memory.

In this paper, we show that low-performing language models, even when provided with detailed n-back task instructions and demonstrations, commit errors that are consistent with a different m-back task ($m \neq n$). Notice that, if a human subject committed such systematic errors, we would conclude that they had misunderstood the task. In comparison, intermediate models, including GPT 3.5, tend to start with the correct task but drift toward a different one as errors accumulate, resulting in poor average 2-back and 3-back performance, consistent with Gong et al. 2024. High-performing models, on the other hand, consistently execute the correct

[1]Code available at https://github.com/hxiaoyang/lm-nback.

Findings of the Association for Computational Linguistics: ACL 2025, pages 2665–2677

task, even for larger n's, achieving task accuracies of 90.08%, 90.08%, and 84.75% for $n = 8, 9, 10$.

The remainder of this paper is organized as follows. Section 2 covers relevant background and related work. Section 3 introduces the dataset, models, prompting approach, and evaluation metrics. Section 4.1 benchmarks each model on 1-back, 2-back, and 3-back tasks, identifying three distinct performance tiers. Section 4.2 investigates whether these performance disparities are explained by differences in task comprehension. Section 4.3 examines the models' ability to consistently apply the correct task set throughout each trial (task set maintenance). In Section 4.4, we challenge the best model to perform 1-back through 10-back tasks and notice a signature of task comprehension. In Sections 4.5 and 4.6, we discuss additional experiments with alternative prompting strategies for comparison. In Section 4.7, we identify an attention pattern whose prevalence predicts 2-back task performance.

2 Background and Related Work

There has been a growing body of work that evaluates pre-trained language models using cognitive tasks originally developed for humans. These efforts often aim to identify whether the models exhibit cognitive constructs or capabilities that are present in humans. Subjects of study include theory of mind (Strachan et al., 2024; Gandhi et al., 2024), analogical reasoning (Hu et al., 2023; Webb et al., 2023), cognitive biases (Binz and Schulz, 2023; Lampinen et al., 2024), and WMCL (Gong et al., 2024), among many others. Such evaluations are susceptible to both overclaiming and underclaiming. On the one hand, false positives can result from training data contamination (Sainz et al., 2023), potentially compromising the validity of vignette-based assessments where models may produce memorized responses. On the other hand, underestimation of model capabilities can happen when we erroneously assume task comprehension, especially for smaller models (Hu and Frank, 2024). Prior studies have also investigated how well language models adhere to prompt instructions, especially compared to humans (Webson and Pavlick, 2022; Webson et al., 2023). In light of other methodological challenges in the cognitive evaluation of language models, such as prompt sensitivity and cultural biases, Ivanova 2023 outlines recommendations for best practices.

Virtually any task, from routine text comprehension to complex problem solving, involves the creation of intermediate or partial results. Successful task completion requires that these results be maintained in a way that facilitates later access. In humans, this mechanism is known as *working memory*, one of the most studied constructs in psychology for over half a century (Miyake and Shah, 1999). This concept can be extended to transformer-based language models designed to process interdependent, serial information. In fact, the transformer architecture, particularly its attention mechanism where key-query matching drives retrieval (Vaswani et al., 2017), bears striking resemblance to cue-based parsing and retrieval models proposed in psycholinguistics (Lewis et al., 2006), making it a promising candidate for modeling human sentence processing. One of the most salient and mysterious aspects of human working memory is its severely constrained capacity (Miller, 1956; Cowan, 2012). One prominent task used to measure working memory capacity is the n-back task (Kirchner, 1958).

To the best of our knowledge, Gong et al. 2024 are the first to apply the n-back task to a language model, specifically the GPT 3.5 TURBO variant of ChatGPT. They experiment with different prompting strategies, including those incorporating feedback and reasoning. As n increases from 1 to 3, they observe a sharp decline in model performance and conclude that the model has a WMCL of approximately 3. Zhang et al. 2024 also examine working memory in language models using a task described as n-back, but with all stimuli presented simultaneously. This departs from the standard paradigm and imposes different working memory demands. Moreover, while they acknowledge that the poor performance in smaller models may stem from limited understanding of the "intent of the input", they do not control for task comprehension as a confounding variable.

Multi-hop question answering is another working memory task paradigm, in which two or more reasoning steps must be performed sequentially to resolve complex queries. This task is interestingly different from n-back in that it places *implicit* demands on working memory in a single forward pass. For instance, answering "*The spouse of the performer of Imagine was...*" requires first identifying John Lennon as the performer of the song and then determining that Yoko Ono was his spouse. Biran et al. 2024 find that, after early model lay-

ers resolve the initial step, later layers often prove deficient in completing the second.

3 Methods

3.1 Data and Prompts

We use the dataset from Gong et al. 2024 (MIT License). For each n-back task, there are 50 trials in total. Each trial consists of a sequence of 24 letters. In exactly 8 random positions within each sequence, the letters are the same as those appearing n step(s) earlier. After each letter prompt, the models are instructed to answer "{*current letter*} *and* {*letter* n *back*} *are* {*different / identical*}" This is designed to facilitate chain-of-thought reasoning (Wei et al., 2022) and to make explicit the specific letter retrieved by the model for comparison with the current one.

```
SYS: [TASK INSTRUCTIONS]
USR: k
LLM: k and none are different.
USR: k
LLM: k and k are identical.        } DEMO
USR: a
LLM: a and k are different.
     ⋮

SYS: [TASK INSTRUCTIONS]
USR: e
LLM: e and none are different.
USR: f
LLM: f and e are different.        } TEST
USR: f
LLM: f and f are identical.
     ⋮
```

To teach the models the correct answer format and maximize their chances of correctly inferring the tasks, each trial begins with a demonstration, which includes a sequence of 24 letters and the correct responses. The "without demo" trials in Section 4.2 are the only exception. Following the demonstration, a new sequence of 24 letters is presented, one at a time, and the models are prompted to respond after each letter. An example 1-back trial is shown above; actual model responses are underlined.

3.2 Models

We use GPT 3.5 TURBO and open-source instruction-tuned models from the QWEN (Bai et al., 2023), LLAMA (Dubey et al., 2024), and GEMMA (Team et al., 2024) families. Each model is prompted recursively to complete the trials. For the open-source models, we analyze the token log probabilities and attention patterns in addition to the generated responses.

Tier	Model	1bk	2bk	3bk
T3	QWEN 1.5 14B CHAT	1.00	0.09	0.08
	LLAMA 3.1 8B INSTR.	1.00	0.14	0.17
	GEMMA 2 9B INSTR.	1.00	0.15	0.20
	QWEN 1.5 32B CHAT	1.00	0.14	0.22
T2	GEMMA 2 27B INSTR.	1.00	0.57	0.36
	GPT 3.5 TURBO	1.00	0.51	0.43
T1	QWEN 2 72B INSTR.	1.00	0.81	0.84
	LLAMA 3.1 70B INSTR.	1.00	0.99	0.93

Table 1: Average retrieval accuracies on 1-, 2-, and 3-back tasks, organized by performance tier.

3.3 Metrics

The n-back task requires continuously matching the current letter and the letter from n steps back to determine the correct label. Compared to binary labels, the retrieved letters offer better insight into the models' understanding of the task. And since the correct label is almost always assigned given the correct retrieval, our analyses focus on the retrieval accuracies and the log probabilities of the retrieved letters. One possibility for low performance is that, despite being prompted to do the n-back task, a model might be following m-back instructions instead. To investigate this, we adopt *counterfactual* measures by providing n-back instructions and evaluating the accuracies and log probabilities of retrievals consistent with the m-back task. We also apply variants of these measures, which we detail in later sections.

4 Experimental Results

4.1 Task Performance

We begin by comparing retrieval accuracies across models for all three tasks (Figure 2) and categorizing them into three performance tiers (Table 1).

T3: $\leq 20\%$ on 2- and 3-back.

T2: $\sim 50\%$ on 2-back; $\sim 40\%$ on 3-back.

T1: $> 80\%$ on 2- and 3-back.

For subsequent analyses, we select the best-performing model, LLAMA 3.1 70B INSTRUCT (T1), the worst-performing model, QWEN 1.5 14B CHAT (T3), and GEMMA 2 27B INSTRUCT (T2) to represent each performance tier.

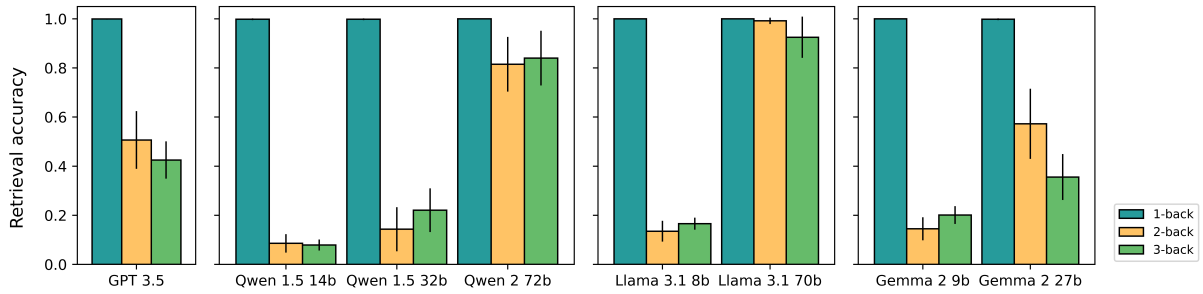

Figure 2: Average retrieval accuracies on 1-, 2-, and 3-back tasks, grouped by model family.

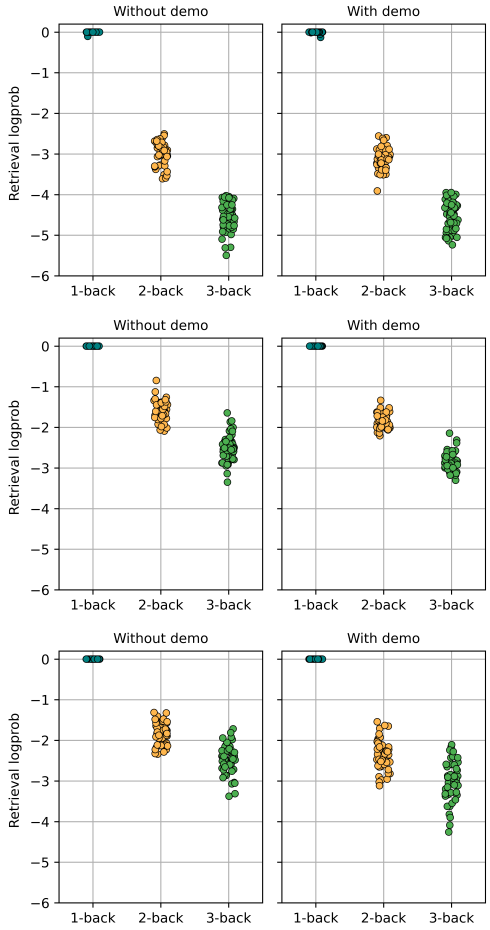

Figure 3: Retrieval log probabilities for 1-back task continuations, with and without demonstrations. From top to bottom are results for QWEN 1.5 14B CHAT (T3), GEMMA 2 27B INSTRUCT (T2), and LLAMA 3.1 70B INSTRUCT (T1). Each point corresponds to the average retrieval log probability of one trial.

4.2 Task Comprehension

To better understand the source of these performance disparities, we ask: Can models infer the task from the instructions and demonstrations—and if so, which serves as a more effective cue? To address these questions, we 1) provide n-back instructions with and without demonstrations, 2) present three continuations, each consistent with a different m-back task, and 3) measure the average retrieval log probabilities for each trial.

Let $\mathsf{P}^{-}_{n,m}$ be the average m-back retrieval log probability given n-back instructions only. Let $\mathsf{P}_{n,m}$ be the average m-back retrieval log probability given n-back instructions and demonstrations.

1-back. Under 1-back instructions, $\mathsf{P}_{1,1} > \mathsf{P}_{1,2} > \mathsf{P}_{1,3}$ across all models. The same is true when no task demonstrations are provided, with no significant difference between $\mathsf{P}_{1,m}$ and $\mathsf{P}^{-}_{1,m}$ for $m = 1, 2, 3$, as shown in Figure 3. Overall, this is unsurprising, given the near-perfect performances of all models on 1-back trials.

2-back. We analyze the representative model from each tier (Figure 4).

T3: Under 2-back instructions, including with demonstrations, 1-back continuations are assigned to be the most plausible, with both $\mathsf{P}^{-}_{2,1} > \mathsf{P}^{-}_{2,2} > \mathsf{P}^{-}_{2,3}$ and $\mathsf{P}_{2,1} > \mathsf{P}_{2,2} > \mathsf{P}_{2,3}$. The task demonstrations do bring $\mathsf{P}_{2,2}$ and $\mathsf{P}_{2,3}$ closer to $\mathsf{P}_{2,1}$, although this is not enough to offset the strong 1-back priors.

T2: Under 2-back instructions only, the ordering of $\mathsf{P}^{-}_{2,m}$ remains the same, albeit with $\mathsf{P}^{-}_{2,2}$ and $\mathsf{P}^{-}_{2,3}$ noticeably closer to $\mathsf{P}^{-}_{2,1}$ than for T3. However, with additional task demonstrations, 2-back continuations are assigned to be the most likely, with $\mathsf{P}_{2,2} > \mathsf{P}_{2,1} > \mathsf{P}_{2,3}$.

T1: Somewhat surprisingly, we notice that $\mathsf{P}^{-}_{2,2} > \mathsf{P}^{-}_{2,1} > \mathsf{P}^{-}_{2,3}$, showing that the model is able to infer the task from the instructions alone. However, the demonstrations do help further consolidate the mapping.

3-back. As shown in Figure 5, the 3-back patterns are largely analogous to the 2-back case.

Summary. Through analyzing models from different performance tiers, we identify three distinct levels of task comprehension capabilities. The T3

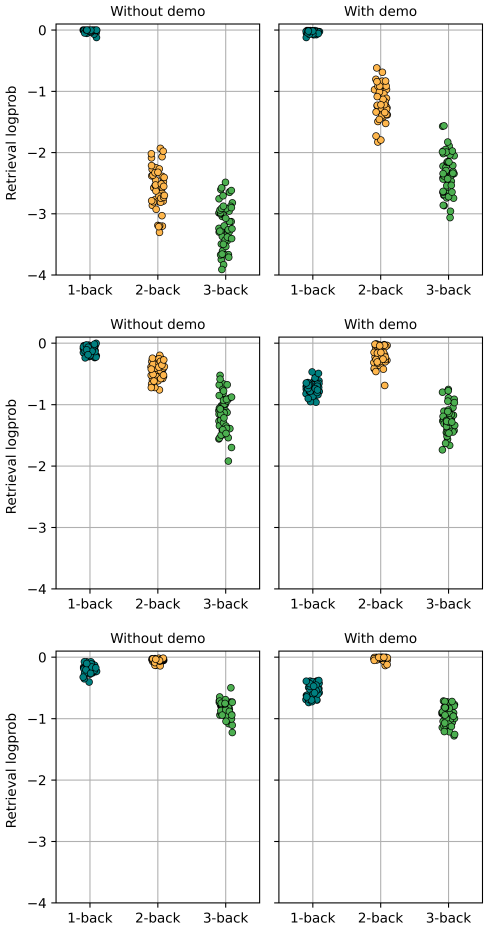

Figure 4: Retrieval log probabilities for 2-back task continuations, with and without demonstrations. From top to bottom are results for QWEN 1.5 14B CHAT (T3), GEMMA 2 27B INSTRUCT (T2), and LLAMA 3.1 70B INSTRUCT (T1). Each point corresponds to the average retrieval log probability of one trial.

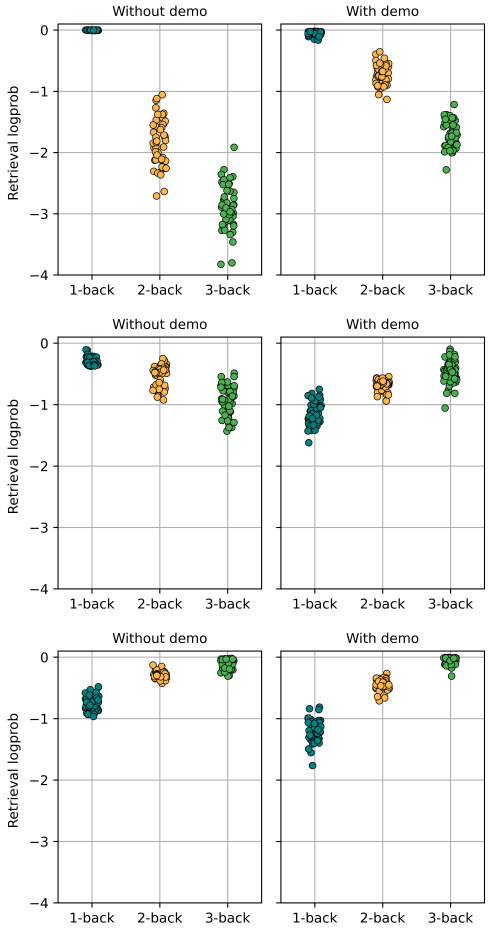

Figure 5: Retrieval log probabilities for 3-back task continuations, with and without demonstrations. From top to bottom are results for QWEN 1.5 14B CHAT (T3), GEMMA 2 27B INSTRUCT (T2), and LLAMA 3.1 70B INSTRUCT (T1). Each point corresponds to the average retrieval log probability of one trial.

model fails to map 2-back and 3-back instructions to the correct responses, given either the instructions or demonstrations, suggesting it completely misunderstands the task; the T2 model fails to map 2-back and 3-back instructions to the correct responses, given the instructions, but can do so if demonstrations are also provided; the T1 model can map 2-back and 3-back instructions to the correct responses based on the instructions alone, suggesting a robust understanding of the tasks. The T3 model's failure to understand the task is corroborated by analyses in Section 4.6. Even when provided with short demo sequences and immediate corrective feedback, it still fails to get 2 consecutive correct responses. This suggests that its poor performance stems from an inability to understand the task, rather than any memory limitation.

4.3 Task Set Maintenance

Each n-back trial consists of a sequence of 24 letters. Successful task completion requires consistent adherence to the task instructions as more stimuli are presented. Here, we investigate whether language models show a progressive decline in their ability to produce n-back consistent responses over time. Previously, performance metrics were averaged across time steps for each trial. Now, we average across trials for each time step. At each time step i in the n-back task, we measure the average accuracy of m-back consistent retrievals for each $m \leq n$, given the model's own responses up to time step $i-1$. Denote this as $\mathsf{A}_{n,\cdot}(m, i)$.

1-back. Unsurprisingly, $\mathsf{A}_{1,\cdot}(1, i)$ stays close to 1 for each model as i increases (not shown).

2-back. As shown in Figure 6:

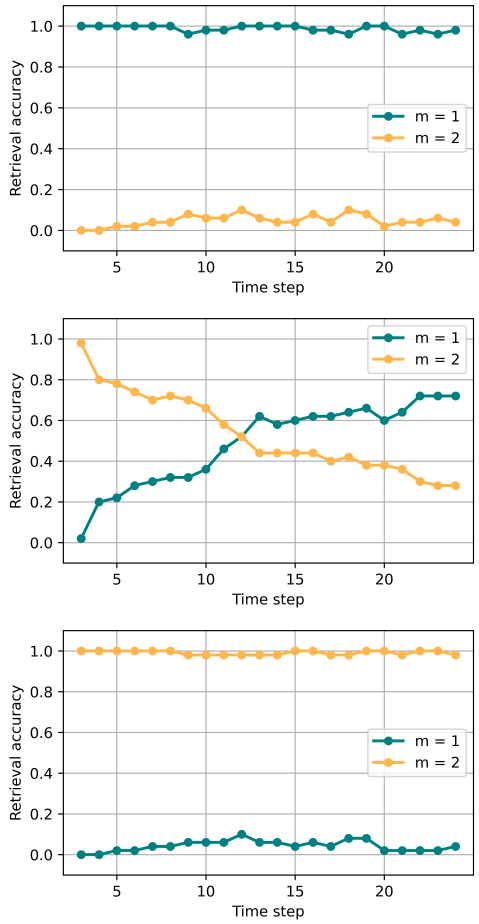

Figure 6: $A_{2,\cdot}(m, i)$ for $m = 1, 2$ and $3 \leq i \leq 24$. From top to bottom are results for QWEN 1.5 14B CHAT (T3), GEMMA 2 27B INSTRUCT (T2), and LLAMA 3.1 70B INSTRUCT (T1).

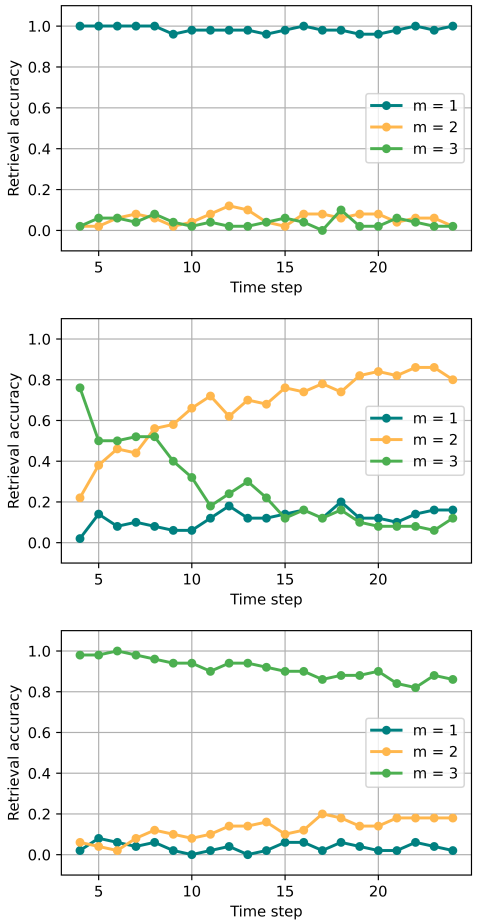

Figure 7: $A_{3,\cdot}(m, i)$ for $m = 1, 2, 3$ and $4 \leq i \leq 24$. From top to bottom are results for QWEN 1.5 14B CHAT (T3), GEMMA 2 27B INSTRUCT (T2), and LLAMA 3.1 70B INSTRUCT (T1).

T3: Throughout the task, $A_{2,\cdot}(1, i)$ and $A_{2,\cdot}(2, i)$ stay close to 1 and 0, respectively, consistent with findings from Section 4.2.

T2: $A_{2,\cdot}(2, i)$ crosses below $A_{2,\cdot}(1, i)$ halfway through the task, suggesting a gradual shift from 2-back to 1-back behavior.

T1: Throughout the task, $A_{2,\cdot}(2, i)$ and $A_{2,\cdot}(1, i)$ stay close to 1 and 0, respectively, contrary to T3.

3-back. As shown in Figure 7:

T3: Throughout the task, $A_{3,\cdot}(1, i)$ stays close to 1 while both $A_{3,\cdot}(2, i)$ and $A_{3,\cdot}(3, i)$ stay close to 0, consistent with Section 4.2.

T2: After a transient initial lead, $A_{3,\cdot}(3, i)$ is quickly surpassed by $A_{3,\cdot}(2, i)$, suggesting yet greater difficulty with task set maintenance.

T1: Throughout the task, $A_{3,\cdot}(3, i)$ remains close to 1, though it shows a gradual decline over time. Meanwhile, $A_{3,\cdot}(1, i)$ and $A_{3,\cdot}(2, i)$ remain low.

Effect of error accumulation. Despite 2-back instructions and demonstrations, the T2 model gradually drifts toward 1-back consistent responses over time, suggesting that the accumulation of 1-back consistent errors may have significantly biased subsequent responses. To test this hypothesis, we manipulate the model's response history by providing m-back consistent responses for i steps following n-back instructions and demonstrations. We then compute the average m-back accuracy for time steps $i + 1$ through 24, denoted as $A_{n,m}(m, i + 1 : 24)$. Figure 8 shows that, as 1-back errors accumulate, 1-back responses are increasingly favored by the T2 model for subsequent steps, despite 2- or 3-back instructions and demonstrations. In comparison, both $A_{2,2}(2, i + 1 : 24)$ and $A_{3,3}(3, i + 1 : 24)$ remain relatively low, showing that correct responses do not bias subsequent answers to the same degree.

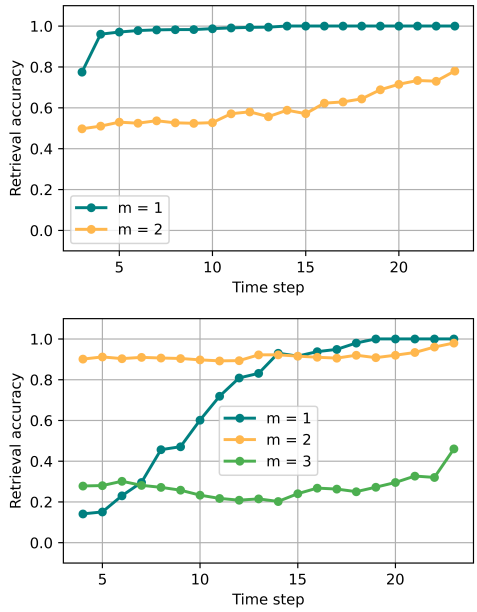

Figure 8: Top: $\mathsf{A}_{2,m}(m, i+1:24)$ for $m = 1, 2$ and $3 \leq i \leq 23$, using GEMMA 2 27B INSTRUCT (T2). Bottom: $\mathsf{A}_{3,m}(m, i+1:24)$ for $m = 1, 2, 3$ and $4 \leq i \leq 23$, using the same model.

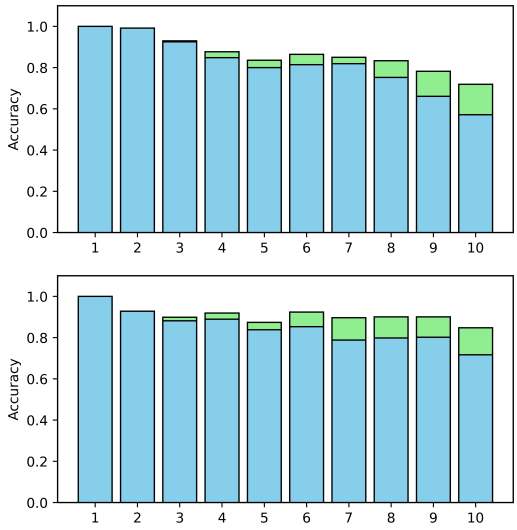

Figure 9: 1-back to 10-back accuracies for LLAMA 3.1 70B INSTRUCT with (bottom) and without (top) curriculum learning. Each full bar corresponds to task (identical/different categorization) accuracy. The blue portion corresponds to retrieval accuracy.

Model	2bk	3bk
LLAMA 3.1 70B INSTR.	0.99 (–.00)	0.62 (–.31)
GEMMA 2 27B INSTR.	0.61 (+.04)	0.31 (–.05)
QWEN 1.5 14B CHAT	N/A	N/A

Table 2: Retrieval accuracies on 2-back and 3-back tasks, for representative models, with interactive demos.

4.4 T1 Model Performance as *N* Increases

Given that the best model, LLAMA-3.1-70B-INSTRUCT, performs well for 1 through 3-back tasks, we would like to know how its performance might change for larger n's. Figure 9 shows that the retrieval accuracy gradually declines as n increases; although, even at $n = 8, 9, 10$, the model is still able to exactly retrieve the correct letters 75.25%, 66.08%, and 57.1% of the time, which translates to task accuracies of 83.33%, 78.25%, and 71.92%. In addition, we measure $\mathsf{P}_{n,m}$ for each $n, m \in \{1, 2, 3, ..., 10\}$, as shown in Figure 10. We notice that $\max_m \mathsf{P}_{n,m} = \mathsf{P}_{n,n}$ for $1 \leq n < 10$. Moreover, $\mathsf{P}_{n,m}$ tends to decrease symmetrically as m deviates from n. We argue that this pattern points to true n-back task understanding.

4.5 Curriculum Learning

The practice of training models on examples of increasing difficulty is known in machine learning as *curriculum learning* (Bengio et al., 2009). Here, we repeat the experiments from Section 4.4 with in-context curriculum learning to gradually familiarize the model with the task. Specifically, before prompting LLAMA 3.1 70B INSTRUCT to perform an n-back task, we provide instructions and demonstrations that include letter sequences and corresponding correct responses for tasks ranging from 1-back to n-back. For example, to prepare the model for the 4-back task, we prepend 4 complete demonstration sequences (1-back to 4-back) to the context, before starting the test sequence. As shown in Figure 9, this approach leads to significant improvements in performance for larger n values. The model achieves retrieval accuracies of 79.83%, 80.17%, and 71.67% and task accuracies of 90.08%, 90.08%, and 84.75% for $n = 8, 9, 10$.

4.6 Interactive Demo

We explore an alternative prompting strategy that more closely mirrors human study paradigms. After receiving task instructions, human participants typically go through brief demo sequences with an experimenter to confirm their understanding. For 2-back trials, we interleave short example sequences of four letters in the forms A-B-A-C and A-B-C-B. Feedback is given for each model response. If a model provides two consecutive correct answers (retrieval and label) within 10 attempts, we proceed with the test sequence. A similar procedure is applied for 3-back trials.

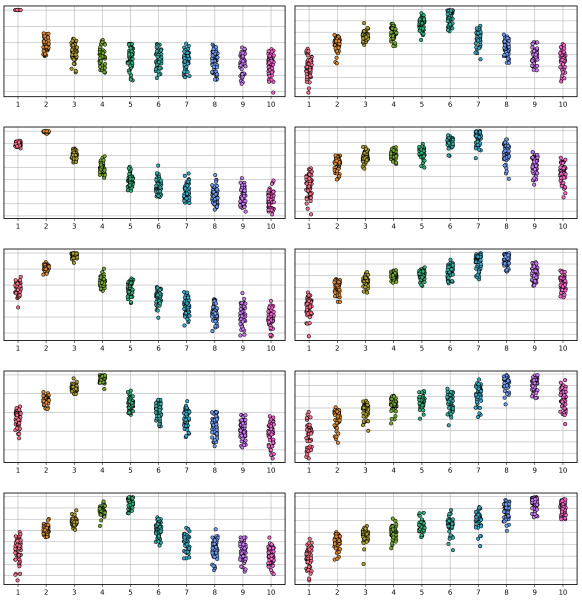

Figure 10: Retrieval log probabilities for 1-back to 10-back task continuations under 1-back to 10-back task instructions for LLAMA 3.1 70B INSTRUCT (T1).

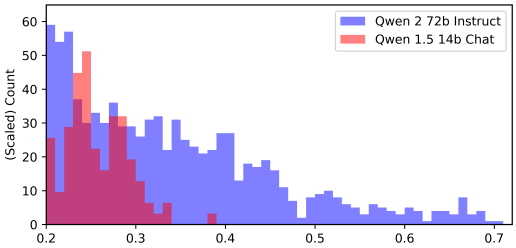

Figure 11: 2-back MRAT counts between 0.2 and 1 for QWEN 1.5 14B CHAT (T3) and QWEN 2 72B INSTRUCT (T1), aggregated over all layers, heads, and trials. QWEN 1.5 14B CHAT counts are scaled by a factor of $\frac{\text{QWEN 2 72B Attention Count}}{\text{QWEN 1.5 14B Attention Count}} = 3.2$.

For both 2-back and 3-back tasks, QWEN 1.5 14B CHAT (T3) fails to achieve two consecutive correct answers after 10 demo sequences, further confirming the model's difficulty with task comprehension. The complete 2-back dialogue is included in Appendix A. Interestingly, GEMMA 2 27B INSTRUCT (T2) performs better on 2-back trials compared to the original experiments but does worse on 3-back trials, as shown in Table 2. LLAMA 3.1 70B INSTRUCT (T1) maintains high performance at 99% on 2-back trials but shows a significant drop in performance on 3-back.

4.7 Attention Analysis

Attentions in transformer-based language models reveal how much each generated token attends to every preceding token. We hypothesize that, for each retrieval, a more performant model should attend more to the source token from n steps back. This is precisely what we observe in the QWEN models. For each (trial, layer, head), we obtain the *mean retrieval attention* (MRAT) by averaging the attention each retrieved token gives to the correct source token. Compared to the 14B model, QWEN 2 72B INSTRUCT (T1) contains a much larger proportion of high-MRAT attentions (Figure 11), with its highest scoring attention (71.98%) closely matching our hypothesized pattern (Figure 12). However, LLAMA models do not exhibit this pattern to the same degree. Attentions in LLAMA 3.1 models are much more diffuse. The maximum MRATs for LLAMA 3.1 8B INSTRUCT and LLAMA 3.1 70B INSTRUCT are 4.86% and 8.52%, respectively.

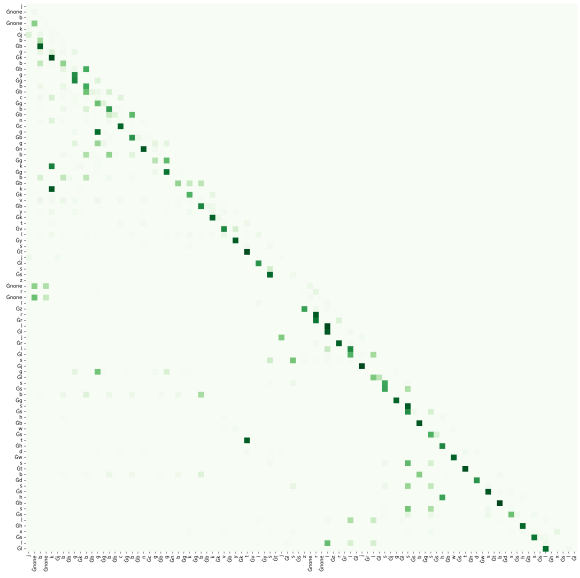

Figure 12: QWEN 2 72B INSTRUCT (T1) attention pattern with the highest MRAT (71.98%) at trial 48, layer 79, and head 63. The top left and bottom right sections correspond to the demo and test sequences, respectively.

5 Conclusion

In this work, we apply the n-back task, a common working memory test, to a range of language models, identifying three distinct performance tiers. We find that these tiers differ not only in retrieval accuracy but also in our measure of task understanding and task set maintenance, suggesting that the performance gap is due at least in part to these differences. We challenge the best model to perform 1 through 10-back tasks, noticing a signature of task comprehension and the benefit of in-context curriculum learning for larger n's. We find that

interactive demos, though closer to human study paradigms, are less effective at conveying the task. Finally, we notice that more performant models tend to have higher retrieval attentions.

6 Limitations

Prompt selection. Despite our careful selection of prompts and experimentation with various prompting strategies, the potential for more effective prompts or techniques to enhance task understanding remains.

Mechanistic understanding. Another limitation is that we do not examine the internal model circuits that may be responsible for inferring and maintaining task sets. However, our experiments with the n-back paradigm provide a good starting point for future research. Causal interventions on smaller models may yield insights into the underlying mechanisms.

LLAMA 3.1 model attentions. As mentioned in Section 4.7, LLAMA models seem to have much more diffuse attentions. Whereas QWEN 1.5 14B CHAT and QWEN 2 72B INSTRUCT reach maximum MRATs of 38.95% and 71.98%, respectively, the same values for both LLAMA models are under 10%, even though QWEN (Bai et al., 2023) and LLAMA (Dubey et al., 2024) models both use *grouped query attention* (Ainslie et al., 2023). We are puzzled by this difference and call for closer examination in future work.

7 Acknowledgments

We thank the anonymous reviewers for their helpful feedback. XH would also like to acknowledge the participants in the Brown MICAAD study, who partially inspired this project.

References

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebron, and Sumit Sanghai. 2023. GQA: Training generalized multi-query transformer models from multi-head checkpoints. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4895–4901, Singapore. Association for Computational Linguistics.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609v1*.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48.

Marcel Binz and Eric Schulz. 2023. Using cognitive psychology to understand gpt-3. *Proceedings of the National Academy of Sciences*, 120(6):e2218523120.

Eden Biran, Daniela Gottesman, Sohee Yang, Mor Geva, and Amir Globerson. 2024. Hopping too late: Exploring the limitations of large language models on multi-hop queries. *arXiv preprint arXiv:2406.12775*.

Nelson Cowan. 2012. *Working memory capacity*. Psychology press.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783v2*.

Kanishk Gandhi, J.-Philipp Fränken, Tobias Gerstenberg, and Noah D. Goodman. 2024. Understanding social reasoning in language models with language models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Dongyu Gong, Xingchen Wan, and Dingmin Wang. 2024. Working memory capacity of chatgpt: An empirical study. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(9):10048–10056.

Jennifer Hu and Michael Frank. 2024. Auxiliary task demands mask the capabilities of smaller language models. In *First Conference on Language Modeling*.

Xiaoyang Hu, Shane Storks, Richard Lewis, and Joyce Chai. 2023. In-context analogical reasoning with pre-trained language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1953–1969, Toronto, Canada. Association for Computational Linguistics.

Anna A Ivanova. 2023. Running cognitive evaluations on large language models: The do's and the don'ts. *arXiv preprint arXiv:2312.01276v2*.

Wayne K Kirchner. 1958. Age differences in short-term retention of rapidly changing information. *Journal of experimental psychology*, 55(4):352.

Andrew K Lampinen, Ishita Dasgupta, Stephanie CY Chan, Hannah R Sheahan, Antonia Creswell, Dharshan Kumaran, James L McClelland, and Felix Hill. 2024. Language models, like humans, show content effects on reasoning tasks. *PNAS nexus*, 3(7).

Richard L Lewis, Shravan Vasishth, and Julie A Van Dyke. 2006. Computational principles of working memory in sentence comprehension. *Trends in cognitive sciences*, 10(10):447–454.

George A Miller. 1956. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review*, 63(2):81.

Akira Miyake and Priti Shah. 1999. *Models of Working Memory: Mechanisms of Active Maintenance and Executive Control*. Cambridge University Press.

Oscar Sainz, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. 2023. NLP evaluation in trouble: On the need to measure LLM data contamination for each benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10776–10787, Singapore. Association for Computational Linguistics.

James WA Strachan, Dalila Albergo, Giulia Borghini, Oriana Pansardi, Eugenio Scaliti, Saurabh Gupta, Krati Saxena, Alessandro Rufo, Stefano Panzeri, Guido Manzi, et al. 2024. Testing theory of mind in large language models and humans. *Nature Human Behaviour*, pages 1–11.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295v4*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Taylor Webb, Keith J Holyoak, and Hongjing Lu. 2023. Emergent analogical reasoning in large language models. *Nature Human Behaviour*, 7(9):1526–1541.

Albert Webson, Alyssa Loo, Qinan Yu, and Ellie Pavlick. 2023. Are language models worse than humans at following prompts? it's complicated. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7662–7686, Singapore. Association for Computational Linguistics.

Albert Webson and Ellie Pavlick. 2022. Do prompt-based models really understand the meaning of their prompts? In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2300–2344, Seattle, United States. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Chunhui Zhang, Yiren Jian, Zhongyu Ouyang, and Soroush Vosoughi. 2024. Working memory identifies reasoning limits in language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 16896–16922.

Kaitlyn Zhou, Jena Hwang, Xiang Ren, and Maarten Sap. 2024. Relying on the unreliable: The impact of language models' reluctance to express uncertainty. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3623–3643, Bangkok, Thailand. Association for Computational Linguistics.

A Interactive Demo Dialogue

QWEN 1.5 14B CHAT (T3) 2-back interactive demo dialogue.

```
<|im_start|>system
You are a participant in a cognitive task. You will
be shown a sequence of letters, one at a time. For
each letter, determine if it is the same as the
letter 2 steps before it. Answer in the following
format: '[current letter] and [letter 2 back] are
[identical/different].' Note that, for the first 2
letters, there won't be any letter 2 steps back;
write 'none' for '[letter 2 back]' in this case.
Let's think step by step.
<|im_end|>

<|im_start|>user
For example, given the sequence t, z, h, z,
the answers should be:
t and none are different.
z and none are different.
h and t are different.
z and z are identical.
Now, given the sequence t, z, t, h,
what should the answers be?
<|im_end|>

<|im_start|>assistant
t and none are different.
z and t are different.
t and z are different.
h and t are different.
<|im_end|>

<|im_start|>user
This is incorrect.
The answers should be:
t and none are different.
(There was no letter 2 steps ago.)
z and none are different.
(There was no letter 2 steps ago.)
t and t are identical.
(The letter 2 steps ago was t.)
h and z are different.
(The letter 2 steps ago was z.)
Now, given the sequence v, n, c, n,
what should the answers be?
<|im_end|>

<|im_start|>assistant
v and none are different.
n and v are different.
c and n are different.
n and n are identical.
<|im_end|>

<|im_start|>user
This is incorrect.
The answers should be:
v and none are different.
(There was no letter 2 steps ago.)
n and none are different.
(There was no letter 2 steps ago.)
c and v are different.
(The letter 2 steps ago was v.)
n and n are identical.
(The letter 2 steps ago was n.)
Now, given the sequence v, n, v, c,
what should the answers be?
<|im_end|>

<|im_start|>assistant
v and none are different.
n and v are different.
v and n are different.
c and v are different.
(The letter 2 steps ago was v.)
<|im_end|>

<|im_start|>user
This is incorrect.
The answers should be:
v and none are different.
(There was no letter 2 steps ago.)
n and none are different.
(There was no letter 2 steps ago.)
v and v are identical.
(The letter 2 steps ago was v.)
c and n are different.
(The letter 2 steps ago was n.)
Now, given the sequence k, l, b, l,
what should the answers be?
<|im_end|>

<|im_start|>assistant
k and none are different.
l and k are different.
b and l are different.
l and l are identical.
<|im_end|>

<|im_start|>user
This is incorrect.
The answers should be:
k and none are different.
(There was no letter 2 steps ago.)
l and none are different.
(There was no letter 2 steps ago.)
b and k are different.
(The letter 2 steps ago was k.)
l and l are identical.
(The letter 2 steps ago was l.)
Now, given the sequence k, l, k, b,
what should the answers be?
<|im_end|>

<|im_start|>assistant
k and none are different.
l and k are different.
k and l are different.
b and k are different.
(The letter 2 steps ago was k.)
<|im_end|>

<|im_start|>user
This is incorrect.
The answers should be:
k and none are different.
(There was no letter 2 steps ago.)
l and none are different.
(There was no letter 2 steps ago.)
k and k are identical.
(The letter 2 steps ago was k.)
b and l are different.
(The letter 2 steps ago was l.)
Now, given the sequence r, f, m, f,
what should the answers be?
<|im_end|>
```

```
<|im_start|>assistant
r and none are different.
f and r are different.
m and f are different.
f and f are identical.
<|im_end|>

<|im_start|>user
This is incorrect.
The answers should be:
r and none are different.
(There was no letter 2 steps ago.)
f and none are different.
(There was no letter 2 steps ago.)
m and r are different.
(The letter 2 steps ago was r.)
f and f are identical.
(The letter 2 steps ago was f.)
Now, given the sequence r, f, r, m,
what should the answers be?
<|im_end|>

<|im_start|>assistant
r and none are different.
f and r are different.
r and f are different.
m and r are different.
(The letter 2 steps ago was r.)
<|im_end|>

<|im_start|>user
This is incorrect.
The answers should be:
r and none are different.
(There was no letter 2 steps ago.)
f and none are different.
(There was no letter 2 steps ago.)
r and r are identical.
(The letter 2 steps ago was r.)
m and f are different.
(The letter 2 steps ago was f.)
Now, given the sequence y, p, w, p,
what should the answers be?
<|im_end|>

<|im_start|>assistant
y and none are different.
p and y are different.
w and p are different.
p and p are identical.
<|im_end|>

<|im_start|>user
This is incorrect.
The answers should be:
y and none are different.
(There was no letter 2 steps ago.)
p and none are different.
(There was no letter 2 steps ago.)
w and y are different.
(The letter 2 steps ago was y.)
p and p are identical.
(The letter 2 steps ago was p.)
Now, given the sequence y, p, y, w,
what should the answers be?
<|im_end|>

<|im_start|>assistant
y and none are different.
p and y are different.
y and p are different.
w and y are different.
(The letter 2 steps ago was y.)
<|im_end|>

<|im_start|>user
This is incorrect.
The answers should be:
y and none are different.
(There was no letter 2 steps ago.)
p and none are different.
(There was no letter 2 steps ago.)
y and y are identical.
(The letter 2 steps ago was y.)
w and p are different.
(The letter 2 steps ago was p.)
Now, given the sequence s, j, g, j,
what should the answers be?
<|im_end|>

<|im_start|>assistant
s and none are different.
j and s are different.
g and j are different.
j and j are identical.
<|im_end|>

<|im_start|>user
This is incorrect.
The answers should be:
s and none are different.
(There was no letter 2 steps ago.)
j and none are different.
(There was no letter 2 steps ago.)
g and s are different.
(The letter 2 steps ago was s.)
j and j are identical.
(The letter 2 steps ago was j.)
Now, given the sequence s, j, s, g,
what should the answers be?
<|im_end|>

<|im_start|>assistant
s and none are different.
j and s are different.
s and j are different.
g and s are different.
(The letter 2 steps ago was s.)
<|im_end|>
```

B Reciting *N* Most Recent Stimuli

We experiment with an alternative answer format that encodes task requirements in greater detail. For 2-back trials, models are instructed to answer "*current:* {*current letter*}*, 1 back:* {*letter 1 back*}*, 2 back:* {*letter 2 back*}*; current letter* {*current letter*} *and letter 2 back* {*letter 2 back*} *are* {*different / identical*}." The 3-back format is analogous.

Retrieval accuracies show significant improvements, including for T3 models, though their performances still lag slightly on 3-back trials (Figure 13). We include these results only for comparison, given that this format changes the original task into one that allows covert verbal rehearsal. In human exper-

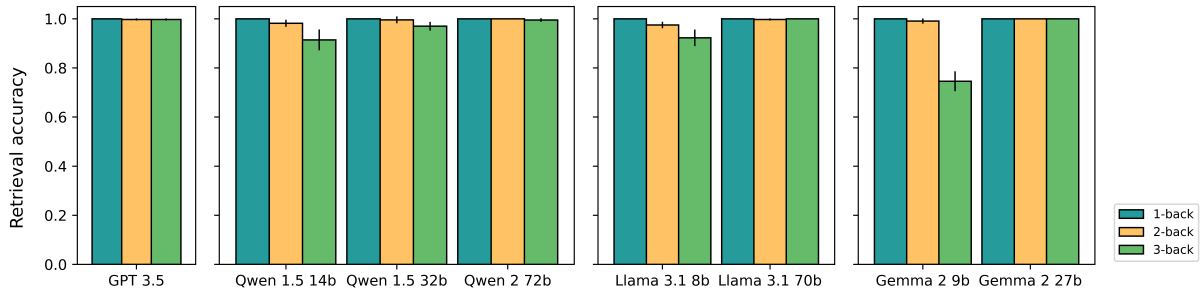

Figure 13: Average retrieval accuracies on 1-, 2-, and 3-back tasks, grouped by model family.

iments, participants would not have enough time to recite all n most recent letters upon presentation of each new letter. However, these results do highlight the malleability of language models' performance on working memory tasks.