Testing Memory Capabilities in Large Language Models with the Sequence Order Recall Task

Mathis Pink
Max Planck Institute for Software Systems
Saarbrücken, Germany
mpink@mpi-sws.org

Vy A. Vo
Intel Labs
Hillsboro, OR 97124

Qinyuan Wu
Max Planck Institute for Software Systems
Saarbrücken, Germany
qwu@mpi-sws.org

Jianing Mu
University of Texas at Austin
Austin, TX
jmu@utexas.edu

Javier S. Turek
Intel Labs
Hillsboro, OR 97124
javierturek@gmail.com

Uri Hasson
Princeton Neuroscience Institute
Princeton, NJ 08544
hasson@princeton.edu

Kenneth A. Norman
Princeton Neuroscience Institute
Princeton, NJ 08544
knorman@princeton.edu

Sebastian Michelmann
New York University
New York, NY 10003
s.michelmann@nyu.edu

Alexander Huth
University of Texas at Austin
Austin, TX
huth@cs.utexas.edu

Mariya Toneva
Max Planck Institute for Software Systems
Saarbrücken, Germany
mtoneva@mpi-sws.org

Abstract

Many benchmarks focus on evaluating Large Language Models (LLMs) on facts and semantic relations, primarily assessing their semantic memory. However, some memories in language are linked to their contexts, like time and place, following Human episodic memory. To address the gap in evaluating memory in LLMs, we introduce the Sequence Order Recall Task (SORT). SORT requires LLMs to recall the correct order of text segments from a text excerpt. We present an initial evaluation dataset, Book-SORT, comprising 36000 samples extracted from 9 books recently added to the public domain. When the text is given to models in-context, we find that instruction-tuned LLMs can perform this task. However, when models need to rely memory stored in their weights or not presented with the text excerpts, their accuracies drop below 60%, near or at chance levels. We hope that SORT will drive the development of memory-augmented LLMs.

1 Introduction

In recent years, large language models (LLMs) have shown an impressive performance on many benchmarks that test long-term knowledge factual or semantic knowledge [1, 2, 3, 4, 5, 6, 7]. In

Latinx in AI @ NeurIPS-24 (LXAI 2024).

contrast to semantic memories, episodic memory links memories to their contexts, such as the time and place they occurred. This ability to organize memory based on spatial and temporal details enables humans to reconstruct events that occurred in the possibly distant past, predict the future. Existing benchmarks focus on evaluating specific capabilities of LLMs related to memory. Needle In A Haystack [8] and FLenQA [9] evaluate a model's ability to reason or search over in-context access to all relevant text. Long Range Arena [10], SCROLLS [11], and MULD [12] evaluate performance over long context lengths. Other benchmarks evaluate LLM's semantic memory via temporal reasoning [13, 14, 15] (e.g. lunch happens before dinner), causal reasoning [16] (e.g. she is eating, therefore she is hungry), or commonsense knowledge (e.g. food is edible) [17] acquired during pre-training. However, these benchmarks fail to evaluate links between memories, and do not assess episodic memory capabilities in LLMs.

To address the gap in evaluating memory in LLMs, we propose the Sequence Order Recall Task (SORT). SORT requires models to recall the correct order of two segments of text from a entire text excerpt. Notably, this task is extendable to other types of ordered input sequences, such as audio and video, and is self-supervised so it does not require additional human or synthetic annotations. We provide a specific text implementation of this task on story books developed as an evaluation dataset, i.e., Book-SORT. Book-SORT contains over 36k pairs of text segments from 9 books. Using this task and dataset, we evaluate a range of instruction-tuned LLMs on their ability to recall the correct order of segments in two conditions: i) when the text excerpt is provided in-context, ii) a memory-less condition without presenting the text excerpt, and iii) providing the text excerpt during fine-tuning as information stored in their parametric weights. Models perform well in in-context conditions, achieving up to 95% accuracy. Model performance in memory-less condition drops below 60%, which is unsurprising given the assumption of lack of exposure to Book-SORT. Finally, we further fine-tune a model to encourage improved parametric memory of the text. However, this does not improve the model's performance.

2 Sequence Order Recall Task and Evaluation Framework

We introduce a novel model evaluation task: recalling the order of parts of a sequence the Sequence Order Recall Task (SORT). SORT is adapted from recency judgment tasks used in cognitive psychology to evaluate episodic memory in humans and animals [18, 19]. In this task, a sequence is presented to a participant, and, after some delay, the participant is asked to judge the order in which two segments of the sequence appeared.

The general form of the task can be described as follows. We have sequential data $\mathbf{X} \in \mathbb{R}^{T \times F}$ where T is the number of time-steps (e.g. tokens in a text) and F is the number of features (e.g. vocabulary size). We define start indices $\mathbf{t_j}$ and $\mathbf{t_k}$ for pairs of segments of length L in $\mathbf{X}$, such that both $\mathbf{t_j} < \mathbf{t_k}$ and $\mathbf{t_j} + L \leq \mathbf{t_k}$. Using these, we extract non-overlapping segments from the original sequence $\mathbf{X}$ as $\mathbf{X_i} = \mathbf{X}[\mathbf{t_i} : \mathbf{t_i}{+}L - 1, :]$. The order of segments $\mathbf{X_j}$ and $\mathbf{X_k}$ is randomized, yielding $[\mathbf{X_A}\ \mathbf{X_B}]$, which is then given as part of a model's input. The objective of a model M_θ is to infer whether $\mathbf{t_A} < \mathbf{t_B}$, i.e. in SORT, the task of a model is to predict which of two non-overlapping subsequences $\mathbf{X_A}$ and $\mathbf{X_B}$ has the lower starting index in $\mathbf{X}$.

2.1 Evaluating LLMs on SORT

To evaluate LLMs on SORT, we rely on the models' ability to follow task instructions, which is why we focus on instruction-tuned models. Using a single prompt formulation across all models may favor a particular model. To prevent this, we compiled a set of 12 prompts that vary in the user prompt. To select a prompt that works well for each model in our experiments, we use a held-out validation set consisting of 400 samples and used the best performing prompt for each model.

To generate an answer from a model, we greedily sample an answer token $\mathbf{a} = \mathbf{argmax}(\mathbf{M}_\theta(I))$ from the model $\mathbf{M}_\theta$, which is parameterized by θ, and decode the sampled answer token $\mathbf{a}$ as either "A" or "B". The answer is evaluated as correct if it corresponds to the segment that truly appears first in $\mathbf{X}$. For proprietary (OpenAI) models that do not allow completing assistant responses with prepended text, we omit P_{answer} below. In this case we resort to generating a sequence of 25 tokens, which we then parse as A or B responses.

In-context evaluation. To evaluate LLMs on the in-context condition, we use a prompt schema in which the model input I is given by

$$I = [P_{system}\ P_{context}\ \mathbf{X}\ P_{task}\ P_{label_A}\ \mathbf{X_A}\ P_{label_B}\ \mathbf{X_B}\ P_{question}\ P_{answer}],$$

where P_{system} is a system prompt; $P_{context}$ contains a contextualization (e.g. book title) as well as the instruction to "read" the text $\mathbf{X}$; P_{task} instructs the model for the positional order recall task to read two segments and describes the objective: answering which of the two labeled segments appears first in $\mathbf{X}$; P_{label_A} and P_{label_B} are the labels for the first and second shown segment $\mathbf{X_A}$ and $\mathbf{X_B}$ (e.g. the characters "A" and "B"); $P_{question}$ repeats the task objective as a question; finally, P_{answer} provides the beginning of the answer string as "Answer: Segment".

Fine-tuning context evaluation. For incorporating $\mathbf{X}$ in the model, in this condition we fine-tune the model with $\mathbf{X}$. During evaluation the model input I is given by:

$$I = [P_{system}\ P_{task}\ P_{label_A}\ \mathbf{X_A}\ P_{label_B}\ \mathbf{X_B}\ P_{question} P_{answer}].$$

The difference to the above condition is that that the source text $\mathbf{X}$ and the preceding $P_{context}$ are omitted, only keeping the presentation of two segments and the instruction to recall their order.

Memory-less evaluation. For the memory-less condition, the model is **not** presented with the text excerpt $\mathbf{X}$. Thus, the input I is the same as for the fine-tuning context evaluation but without the fine-tuning step with $\mathbf{X}$.

2.2 Book-SORT Dataset

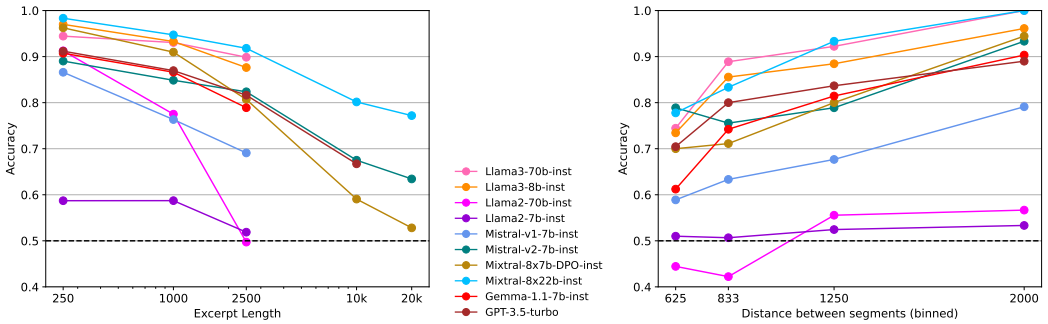
We created an English language dataset to evaluate LLMs named Book-SORT . The selected data considered several factors: (1) we chose long texts (mean length 72,700 words) that exceed the context windows of most LLMs; (2) we selected books from *Project Gutenberg* that recently entered the U.S. public domain to avoid ethical and copyright issues, and minimize pre-training contamination in LLMs. Within these constraints, we aimed to maximize content diversity, including narrative fiction novels, a physics text, and an extended essay. The dataset is available at https://huggingface.co/datasets/memari/booksort.

We constructed the dataset such that varies across factors that can affect model performance on SORT. We first varied (1) L_E, the length of the text excerpt presented in context. We set $L_E = \{250, 1000, 2500, 10000, 20000\}$ words. The largest two values excluded one book that was too short, and are meant to test models with extended context windows. Another factors that may affect task performance: (2) L_S, the length of the segments from the text, and (3) D_S, the distance between the segments in the original text. We set $L_S = \{20, 50\}$ words. We then created 4 different distance bins $D_S = \{d_0, d_1, d_2, d_3\}$, whose values were bounded by the excerpt length L_E. Within each unique combination of L_E and L_S, we randomly sampled 110 excerpts (100 for model evaluation, and 10 for prompt selection) from each of the 9 books. All excerpts and segments began at a sentence boundary. Within each combination of L_E, L_S, we randomly sampled 4 different segment pairs, one from each distance bin D_S. Finally, for all 110 trials within each of these 3 factors, we counterbalanced the correct answer. This yielded a well-controlled and easily extendable dataset of about $36K$ text segment pairs for SORT evaluation.

3 Experimental Setup and Results

We evaluate a selection of open models ranging from 7B to 8x22B parameter transformer models. Initial experiments with non-instruction-tuned models resulted in chance performance on Book-SORT, which we attribute to the lack of instruction tuning, and thus focus on evaluating instruction-tuned models in this work. We have selected models from different model families including Llama3 [20], Llama2 [21], Mistral [22], Mixtral [23], Gemma [24] and OpenAI GPTs [25]. The SORT code is available at https://github.com/bridge-ai-neuro/SORT.

For Llama3-8b-Instruct, we evaluate whether inducing memory of the books' texts via fine-tuning increases performance on the task. As an additional baseline, we separately fine-tune the same model with summaries and/or reviews of the books instead of the actual book texts as part of the fine-tuning data that also includes $3,500$ unrelated instruction tuning samples from OpenHermes2.5 [26]. The inclusion of instruction data helps avoid catastrophic forgetting of prior instruction tuning.



(a) By excerpt lengths

(b) By segment distances

Figure 1: Factors affecting performance on in-context memory condition. (a) Average accuracy by excerpt length. (b) Average accuracy by distance between segments ($L_E = 2500$).

3.1 In-Context Memory Condition

The results with all the models are presented in Fig. 1a. Nearly all models achieve above 77% accuracy on SORT using in-context memory, reaching up to 95% for excerpts of 2500 words or shorter. However, when we plot performance at each unique excerpt length, we see a consistent monotonic decrease in average accuracy. Surprisingly, the performance degrades below 80% for the models. Moreover, we tested the difference between 20 and 50 words for segment lengths. Models handle longer segments (50 words) slightly more effectively than shorter segments (20 words), with a measured improvement of up to 4%.

We further evaluate the effect of the factor of distance between the text segments in the excerpt and how it may influence the model performance. Figure 1b shows an increasing trend in average accuracy as the distance between segments increases. The improvement in accuracy is observed across all models.

3.2 Fine-tuning Memory Condition

We find that despite the Llama3-8b model fine-tuned on data including book-chunks having some memory of the book's text, the epoch-matched performance between the model fine-tuned without the book-chunks does not differ statistically for any epoch, shown in Fig. 2 (Right).

3.3 Memory-less Condition

The results of this experiment are depicted in Fig. 2 (Left) for 4 of the best performing models. Average performance of models on the memory-less condition is mostly around chance–much lower than performance on the in-context condition. This is not surprising as we specifically chose books recently released to public domain with the intent that they would not have been part of the pretraining datasets of these models.

Further, we observe that some models have above-chance performances (up to 60% accuracy) for some segment distances. While these performances are low, it is surprising these models are able to do the task above chance levels, if the assumption that they have not been trained on the books in Book-SORT texts is correct.

4 Conclusions

We provide a new evaluation task, SORT, for assessing memory in large language models. SORT can be used with any text data and without the need for annotation. We created the Book-SORT dataset based on this task with public domain books. Our evaluation results for a range of LLMs on Book-SORT in the in-context condition highlight the strength of current models dealing with the task in-context. In line with previous findings [27, 9], we see that performance degrades with an increasing number of tokens in the context. We also observe that increasing the distance between segments

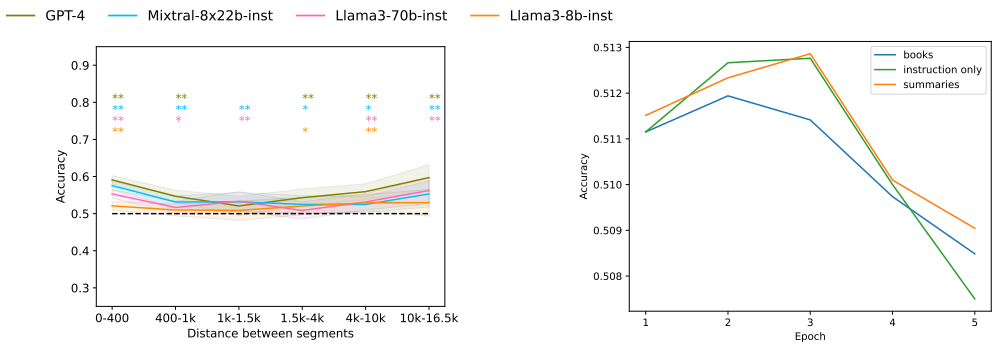


Figure 2: (Left) Model performance for memory-less condition by segment distance (95% bootstrapped confidence interval, $L_S = 50$ words). Significant difference from chance is marked with $*$ (*p-value$<$0.05,**p-value$<$0.01). (Right): Performance of fine-tuned memory condition model over books, book summaries, and unrelated instructions.

improves models' performance. On the memory-less condition and the fine-tuned memory condition, we showed that LLMs perform at or close to chance. However, a few achieve slight above-chance performance for certain segment distances. This could be due to prior knowledge about the books or common sense reasoning.

References

[1] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2020.

[2] Kamoi Ryo, Goyal Tanya, and Rodriguez Juan Diego. Wice: Real-world entailment for claims in wikipedia. *arXiv preprint arXiv: 2303.01432 v1*, 2023.

[3] Robert L Logan IV, Nelson F Liu, Matthew E Peters, Matt Gardner, and Sameer Singh. Barack's wife hillary: Using knowledge-graphs for fact-aware language modeling. *arXiv preprint arXiv:1906.07241*, 2019.

[4] Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*, 2019.

[5] Jifan Yu, Xiaozhi Wang, Shangqing Tu, Shulin Cao, Daniel Zhang-Li, Xin Lv, Hao Peng, Zijun Yao, Xiaohan Zhang, Hanming Li, et al. Kola: Carefully benchmarking world knowledge of large language models. *arXiv preprint arXiv:2306.09296*, 2023.

[6] Kai Sun, Yifan Ethan Xu, Hanwen Zha, Yue Liu, and Xin Luna Dong. Head-to-tail: How knowledgeable are large language models (llm)? aka will llms replace knowledge graphs? *arXiv preprint arXiv:2308.10168*, 2023.

[7] Hady Elsahar, Pavlos Vougiouklis, Arslen Remaci, Christophe Gravier, Jonathon Hare, Frederique Laforest, and Elena Simperl. T-rex: A large scale alignment of natural language with knowledge base triples. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.

[8] Greg Kamradt. Llmtest_needleinahaystack, 2023. Accessed: 2024-06-03.

[9] Mosh Levy, Alon Jacoby, and Yoav Goldberg. Same task, more tokens: the impact of input length on the reasoning performance of large language models. *arXiv preprint arXiv:2402.14848*, 2024.

[10] Yi Tay, Mostafa Dehghani, Samira Abnar, Yikang Shen, Dara Bahri, Philip Pham, Jinfeng Rao, Liu Yang, Sebastian Ruder, and Donald Metzler. Long range arena : A benchmark for efficient transformers. In *International Conference on Learning Representations*, 2021.

[11] Uri Shaham, Elad Segal, Maor Ivgi, Avia Efrat, Ori Yoran, Adi Haviv, Ankit Gupta, Wenhan Xiong, Mor Geva, Jonathan Berant, et al. Scrolls: Standardized comparison over long language

sequences. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 12007–12021, 2022.

[12] George Hudson and Noura Al Moubayed. Muld: The multitask long document benchmark. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3675–3685, 2022.

[13] Qiang Ning, Hao Wu, Rujun Han, Nanyun Peng, Matt Gardner, and Dan Roth. Torque: A reading comprehension dataset of temporal ordering questions. In *Conference on Empirical Methods in Natural Language Processing*, 2020.

[14] Ben Zhou, Kyle Richardson, Qiang Ning, Tushar Khot, Ashish Sabharwal, and Dan Roth. Temporal reasoning on implicit events from distant supervision. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1361–1371, 2021.

[15] Yu Feng, Ben Zhou, Haoyu Wang, Helen Jin, and Dan Roth. Generic temporal reasoning with differential analysis and explanation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12013–12029, 2023.

[16] Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*, 2023.

[17] Mete Ismayilzada, Debjit Paul, Syrielle Montariol, Mor Geva, and Antoine Bosselut. Crow: Benchmarking commonsense reasoning in real-world tasks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9785–9821, 2023.

[18] Howard Eichenbaum. Memory on time. *Trends in cognitive sciences*, 17(2):81–88, 2013.

[19] Lila Davachi and Sarah DuBrow. How the hippocampus preserves order: the role of prediction and context. *Trends in cognitive sciences*, 19(2):92–99, 2015.

[20] AI@Meta. Llama 3 model card. 2024.

[21] 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.

[22] 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.

[23] Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

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

[25] 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.

[26] Teknium. Openhermes 2.5: An open dataset of synthetic data for generalist llm assistants, 2023.

[27] Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024.

A Additional details on Book-SORT dataset

The list of books used to create Book-SORT is presented in Table 1. To preprocess the books, we wrote custom Python code to only retain the book text that formed a continuous narrative. We stripped the front and back matter of the book, and extracted chapter titles if they existed. 8 of the 9 books contained individual section or chapter breaks. For these 8 books, we parsed the text corresponding to each chapter. Chapter titles or section headings (e.g. 'VI' to indicate section six) were removed, and all remaining text was concatenated. This string was split into words (assuming simple whitespace separators with Python `string.split()`) to produce a final text array for each book. This text array was sampled for the Book-SORT dataset.

The segment distance L_S for Book-SORT is sampled from one of four distance bins. The right edge for sampling from each bin for excerpt length $L_E \leq 2500$ is given by $d_0 = L_E/4$, $d_1 = L_E/3$, $d_2 = L_E/2$, and $d_3 = L_E/0.8$. For longer excerpt length of $L_E > 2500$ is given by $d_0 = 1000$, $d_1 = L_E/4$, $d_2 = L_E/2$, and $d_3 = L_E/0.8$. Distance is computed between the beginning of the first segment and the beginning of the second segment. A minimum distance L_S is considered, therefore producing adjacent, non-overlapping segments.

License. We will make our code and data openly available under a permissive BSD-3 license for code. Data including Book-SORT will be available under a CC0 license in anonymous.

Table 1: Project Gutenberg metadata on Book-SORT books.

ID	Title	Author	Words	Release	Pub	LoCC[1]	Subjects
69087	The Murder of Roger Ackroyd	Christie, Agatha	69,720	10/2/2022	1926	PR	Detective and mystery stories; Fiction: Private investigators - England, Murder - Investigation, Belgians - England
72578	Tom Swift and His Talking Pictures	Appleton, Victor	43,853	1/1/2024	1928	PZ	Adventure stories; Motion pictures
72600	The Trumpeter of Krakow	Kelly, Eric Philbrook	59,081	1/2/2024	1928	PZ	Juvenile fiction: Middle Ages, Poland - History - Casimir IV, 1447-1492
72869	Meet the Tiger	Charteris, Leslie	79,946	2/4/2024	1928	PR	Fiction: Private investigators - England; Detective and mystery stories
72958	Hunting for Hidden Gold	Dixon, Franklin W.	42,354	2/14/2024	1928	PZ	Juvenile fiction: Brothers, Gold mines and mining, Montana, Robbers and outlaws; Mystery and detective stories
72963	The Nature of the Physical World	Eddington, Arthur Stanley, Sir	104,530	2/15/2024	1928	Q	Physics - Philosophy; Science - Philosophy
72972	Money for Nothing	Wodehouse, P.G. (Pelham Grenville)	82,331	2/16/2024	1928	PR	Humorous stories; Fiction: Swindlers and swindling, Greed
73017	Pomona; or, the Future of English	De Selincourt, Basil	9,273	2/22/2024	1928	PE	English language
73042	The Well of Loneliness	Hall, Radclyffe	163,217	2/26/2024	1928	PR	Fiction: Lesbians - England - Social conditions

B Model details

All open models we used in this work can be downloaded from HuggingFace: Llama3-70b-inst, Llama3-8b-inst, Mixtral-8x22b-inst, Mixtral-8x7b-DPO-inst, Mistral-v1-7b-inst, Mistral-v2-7b-inst, Llama2-70b-inst, Llama2-7b-inst, Gemma-1.1-7b-inst. For the OpenAI models, we used the gpt-3.5-turbo-0125 version of GPT-3.5, and gpt-4-turbo-2024-04-09 for GPT-4.