Semantic Masking in a Needle-in-a-haystack Test for Evaluating Large Language Model Long-Text Capabilities

Ken Shi and Gerald Penn
Dept. of Computer Science
University of Toronto
CANADA
{kenshi,gpenn}@cs.toronto.edu

Abstract

In this paper, we introduce the concept of Semantic Masking, where semantically coherent surrounding text (the haystack) interferes with the retrieval and comprehension of specific information (the needle) embedded within it. We propose the Needle-in-a-Haystack-QA Test, an evaluation pipeline that assesses LLMs' long-text capabilities through question answering, explicitly accounting for the Semantic Masking effect. We conduct experiments to demonstrate that Semantic Masking significantly impacts LLM performance more than text length does. By accounting for Semantic Masking, we provide a more accurate assessment of LLMs' true proficiency in utilizing extended contexts, paving the way for future research to develop models that are not only capable of handling longer inputs but are also adept at navigating complex semantic landscapes.

1 Introduction

Many state-of-the-art Large Language Models (LLMs) have recently claimed to have extended the input context window to 128K or above. (e.g., GPT-4 (OpenAI et al., 2024), LLaMA-3.1 (Dubey et al., 2024), etc.) Such extensions significantly boost these models' abilities to take on a wider range of tasks as they enable them to take longer documents such as story outlines or even full stories as their input. Specifically, they can aid authors in creative writing. In the Flower and Hayes model (Andriessen et al., 1996), writing is viewed as a network of three main cognitive processes: Planning, Translating and Reviewing. As we extend the context window, LLMs can not only aid authors in the Planning stage through brief writing prompts, but also help them in the Translating stage by taking in and expanding on the story outline; or in the Reviewing stage by taking in and refining the full story. However, the effectiveness of the extended context window remains questionable, as the evaluation metric those modifications are based on are mostly about language modeling ability, which does not necessarily capture how well the models utilize context in various downstream tasks — tasks that require understanding and interpretation of the context, especially in creative writing.

In addition to those language-modeling-oriented metrics such as perplexity (Brown et al., 1992), many recent works on long text processing have turned to the Needle-in-a-haystack Pressure Test (Chandrayan et al., 2024), which is a more retrieval-oriented evaluation that inserts a statement (the needle) into a larger piece of text (the haystack) and asks LLMs or LLM-based Retrieval Augmented Generation (RAG) models to retrieve it. Alternatively, one can generalize information retrieval to free-form question answering in order to test how well the long input context is understood.

However, one important factor that has been more or less ignored is the **Semantic Masking** effect the haystack may have on the needle or the question. In the original work, the haystacks are chosen solely by the length of the document in a random process. This process, although easily adaptable to different context window sizes, does not represent the practical usage of long context window in downstream tasks well. In practice, the long context provided, such as stories or books, is often semantically coherent, meaning that each sentence or paragraph should be more semantically related to its neighbours compared to the needle and the haystack chosen randomly. Semantic Masking in this case denotes the interference the surrounding text may impose on the needle, which effectively acts like a mask that hides the information in the needle.

In this work, we will demonstrate how Semantic Masking effect might be a more important factor that impacts LLM's long text capabilities than text length. Based on the findings, we also propose

Proceedings of the First Workshop on Writing Aids at the Crossroads of AI, Cognitive Science and NLP (WRAICOGS 2025), pages 16–23
January 20, 2025.

an evaluation pipeline that assesses LLM's long text capabilities through question-answering with the Needle-in-a-haystack approach while taking account for the Semantic Masking effect. We select a subset of questions and their corresponding stories from NarrativeQA (Kočiský et al., 2018). We name this pipeline as *Needle-in-a-haystack-QA Test*.

The main contributions of our work are: 1) We propose the Needle-in-a-haystack-QA Test, a pipeline based on the Needle-in-a-haystack Test that assesses LLM's long text capabilities; 2) We define and demonstrate **Semantic Masking** effect in the Needle-in-a-haystack-QA Test through a few experiments; 3) We suggest a novel difficulty assessment for the Needle-in-a-haystack-QA Test that can generalize to any QA dataset when used in any Needle-in-a-haystack setting.

2 Related Work

2.1 Long Text Capability Metrics

In many early days effort in extending context window for transformer-based language models such as Transformer-XL (Dai et al., 2019) and Longformer (Beltagy et al., 2020), perplexity has been the dominant metric to evaluate how well the model adapts to the extended context window, and has carried onto many recent work for measuring long text capabilities (Chen et al., 2023) (Jin et al., 2024) (Wu et al., 2024). While perplexity does measure the language modeling ability nicely, it does not necessarily capture its ability to utilize the input context.

Recently, many works have shifted their primary metric to the Needle-in-a-haystack Pressure Test (Chandrayan et al., 2024) to test LLM's long text capability (Ivgi et al., 2023) (Zhao et al., 2024) (Li et al., 2024). However, the current design of the test favours heavily on RAG systems as the goal is simply to retrieve the needle from the haystack. Turning the retrieval task to free-form question answering would significantly boost the difficulty of the test as it requires the model to understand the input context and query to fetch an answer.

2.2 Needle-in-a-haystack in Cognitive Science

Our use of the term, "Needle-in-a-haystack," relates to an earlier thread of research in cognitive science (Zock, 2006), which governs a lexical access problem, in which a person fails to retrieve a known word from memory at the moment, despite having a strong feeling that the word is on the "tip of their tongue" (Brown and Mcneill, 1966). In this case, "Needle-in-a-haystack" is a metaphor for searching for this word, where the needle is the precise target, and the haystack is the person's mental lexicon.

In the case of lexical access, the difficulty has been shown to arise from two kinds of masking: semantic and phonological, which correspond to potential overlap in meaning and form, respectively. While the phonological component is less of a concern for LLMs since the models only indirectly and incompletely represent pronunciation, the impact from semantic associations between words is definitely observable. Neverthless, we are also interested in semantic masking effects at the phrasal or sentential level.

2.3 Question Answering with Long Text

In question answering, early works such as QuALITY (Pang et al., 2022) concern questions that have context at around 5K tokens; on the other hand, ELI5 (Fan et al., 2019) and LLeQA (Louis et al., 2024) concern Long Form Question Answering (LFQA), which focuses on generating longer answers. None of the above works are suitable for testing the extended context window for state-of-the-art LLM that has 128K or larger context window.

Context of such an enormous size demands the model's ability of reading comprehension. NarrativeQA (Kočiský et al., 2018) is a dataset designed for testing reading comprehension with 2 tasks: answering questions based on summary or full story. The former task was much more popular, as early models are only capable of handling context of size closer to the summary. The latter task is often underlooked.

NarrativeQA contains 1567 stories evenly split between books and movie scripts. For the purpose of this work, we only kept the book portion of stories as the candidate input and will mostly operate with stories under 50K tokens for the sake of computing.

3 Method

In this section, we will first discuss the setup of the Needle-in-a-haystack-QA Test. Based on the test, we will list a few experiments that utilize this test to demonstrate what role the Semantic Masking effect and text length play in demonstrating LLM's long text capabilities.

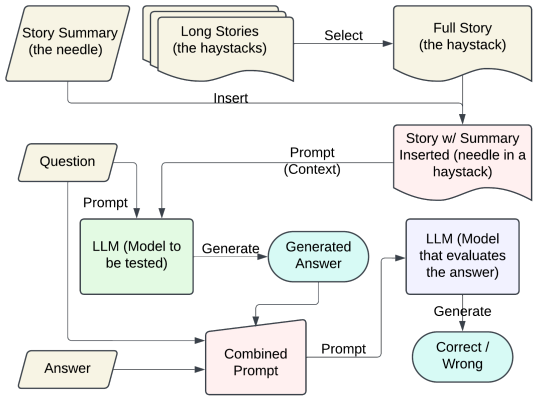

Figure 1: A simplified overview of the Needle-in-a-haystack-QA Test pipeline. All the yellow components (Question, Answer, Long Stories, Full Story and Story Summary) are immediate data from selected dataset (in this work, NarrativeQA).

3.1 Needle-in-a-haystack-QA Test

Figure 1 shows an overview of the Needle-in-a-haystack-QA pipeline on a single query. For a given question, we first identify the needle (the summary of the story which the question is based on) and a haystack (the full text of a story); then we combined the two by inserting the needle into a random paragraph break in the haystack. This combined text is fed to the tested model (the LLM to be tested, the model in green) as context and the question as user prompt.

Since LLMs have the tendency to answer a question in long answer form, instead of instructing the model to answer in a specific format, we keep the generated answer as it is and introduce an evaluator model (the model in purple) to assess the answer. The answer generated by the tested model is combined with the question and the groundtruth answer into a combined prompt. This combined prompt identifies each of these three data and asks whether the generated answer is correct. The combined prompt is then used in the evaluator model to generate a Boolean judgement for the generated answer. (Note: From the test conducted, allowing the model to provide an explanation to justify its judgement helps the model make more reliable decisions. Therefore in the implementation, it is suggested to use prompt that encourages the model to provide an explanation of its decision and strip the decision afterwards)

In this pipeline, one can vary the selection, insertion, or prompt construction process to perform controlled variable experiments. The experiments described below will focus on testing the effect of different haystack selections with fixed insertion process and prompt templates.

3.2 Experiment 0: Validating the evaluator model

In principle, the evaluator model and the tested model should be different to avoid bias in the evaluation. Even then, automatic evaluation of free-form answer remains to be in a doubtful position. It is important to understand the evaluator model's capability of evaluating a generated answer before putting it in the hot seat.

Conveniently, in NarrativeQA, each question q_i has two groundtruth answers, $a_i^{(1)}$ and $a_i^{(2)}$, written independently by two different experts. This makes it possible to skip the tested model generation stage and testify the evaluator model by using one of the ground truth answers as the groundtruth and the other as the "generated" answer. We will also test the evaluator model's stability by using the answers the other way around to see if the judgement aligns with each other, and the same setting multiple times to test if the model's judgement over the same query is stable.

Ideally, the two groundtruth answers, although may vary in the exact wording, should both represent the same answer. Achieving a high accuracy in this test will prove the model's capability of evaluating answers given the question and the correct answer.

We will also get rid of questions that a verified evaluator model fail to consistently answer when swapping the groundtruth and the "generated" answer, as it may indicate the outlier question that the two expert answers potentially disagreed.

3.3 Experiment 1: Examining Semantic Masking Effect

We define Semantic Masking as the interference that the surrounding haystack text imposes on the needle. To measure it quantitatively, we use the most common metric for measuring semantic relatedness between text, namely the cosine similarity between the semantic vector representations of the needle and the summary of the haystack. We chose cosine similarity because, while embedding models are not always explicitly optimized with a direct cosine objective, their training paradigms strongly incentivize the network parameters to arrange semantically akin texts closer together in the

embedding space, which makes cosine similarity a fitting semantic relatedness metric. For the purpose of this work, we will use MPNet (Song et al., 2020; Reimers and Gurevych, 2019) vectors as the semantic representations. We chose MPNet vectors because, in our experience, MPNet is one of the most robust sentence embedding models in various semantic similarity and downstream sentence-level tasks.

To demonstrate the effect of Semantic Masking, we need to place the needle in haystacks that could impose enough semantic interference, which in this case refers to haystacks that have high similarity score to the needle. For NarrativeQA, the best matching haystack is the full story that corresponds to the selected needle, which according to our measurement, has a cosine similarity score of 1 because the needle is the summary of the haystack.

In this experiment, for each question q_i, we will insert the summary of associated story s_i to the story itself, denoted as d_i. By comparing the performance of having s_i in d_i as context with the performance of only providing s_i as context, we can see how Semantic Masking can significantly impact the difficulty of Needle-in-a-haystack Test.

We will test the significance of the result by running the McNemar Test (McNemar, 1947) on all queries that are determinant. Queries with inconsistent or disagreed answers will not participate in the test.

We are also interested in how the result differs before and after introducing the haystacks. For this we define flip rate, which is calculated by

$$r_f = \frac{\#\ \text{CASES ANSWER CHANGED}}{\#\ \text{CASES}} \quad (1)$$

3.4 Experiment 2: Question Difficulty Assessment

In addition to Semantic Masking, there are many other factors that may significantly impact the result of the test. One of which is question difficulty. Assessing the difficulty of a question in QA tasks has been a challenge, yet it is essential for our proposed test to identify questions that are of reasonable difficulty in order to draw meaningful conclusions. For example, if a question can be answered without any context, or if a question cannot be answered with any form of provided context, neither of the questions would produce meaningful statistics in the Needle-in-a-haystack-QA Test. For this reason, we propose a difficulty assessment scheme for each question based on their performance with the tested LLM.

For each question, we perform three tests of different context level: no context, summary only, and full story only. Each test contains 5 runs of the exact same setting and another 5 that use the second groundtruth instead of the first. The collective result can be denoted as correct, wrong, inconsistent and disagreed. Correct / wrong indicates that all 10 runs yield the correct / wrong answer; Inconsistent means that there is one or more runs out of the 10 that yield a different decision; Disagreed means that the result of the first 5 runs does not align with the last 5 runs, meaning that the decision differs when swapping to the other groundtruth.

Based on the result of the three tests, we can assign each question a difficulty level. Table 1 shows all possible difficulty level along with description of their categorization scheme in plain English, where "occasionally" denotes inconsistent output. Questions that have any disagreed decision are considered invalid and will not participate in any further evaluation process.

Among the 10 categories, easy, standard, puzzling, mildly challenging and challenging are considered as reasonable difficulty, and they roughly span 3/4 of all questions. Commonsense and confusing questions are questions that could be answered without context, meaning that either the question is factoid or the model has been trained on the story; Incapable questions are questions that could not be answered with any level of context, which would not make a difference no matter what haystack selection process is chosen; Nonsense questions are in counterintuitive scenarios that yield the answer on the full story but not on summary, which their corresponding full stories are not suitable to serve as haystacks for themselves. In experiment that selects question based on the question difficulty, questions in the 5 reasonable difficulty categories are prioritized.

We will demonstrate how question difficulty also plays an important role in setting up the tests. We will do so by performing post hoc experiments in experiment 1 with the proposed difficulty assessment. We will also conduct the McNemar test and compute flip rates to compare with results from experiment 1.

Difficulty	Description
commonsense	can answer even without context
easy	can answer when given summary or full story
standard	can answer when given summary, occasionally when given full story
puzzling	can occasionally answer when given summary or full story
mildly-challenging	can answer when given summary, but not full story
challenging	can occasionally answer when given summary, but not full story
incapable	cannot answer with any level of context
confusing	can occasionally answer even without context
nonsense	cannot answer with summary but can with full story
invalid	if there is a disagreement between assessment when using the two groundtruth

Table 1: Difficulty Assessment for Questions and description

3.5 Experiment 3: Controlling Haystack Properties

As mentioned earlier, one can test how different haystacks impact the difficulty of the test by controlling variables during the haystack selection process. In this work, we examine how the semantic relatedness of the haystack to the needle and the length of the haystack can impact the test performance of a fixed tested model.

We pick a few questions Q and their corresponding stories D. For each question q_i and its associated story d_i, we pick a set of haystack stories $D^{(i)}$ that are of similar length but a wide range of semantic similarity with respect to the reference or vice versa when controlling the other variable. We will pair every question q_i along with its associated summary s_i with haystack stories $d_j^{(i)}$ from the set $D^{(i)}$ to form queries, where s_i is inserted into $d_j^{(i)}$ and serves as the context.

To ensure the experiment results are comparable across the board, stories that are of similar length are all within 25K$\pm$2.5K tokens, and stories that are of similar semantic similarity have a cosine score within 0.3$\pm$0.02 with respect to their reference story.

For a few of the post hoc studies, we will calculate the point-biserial correlation (PBC) score to test whether there exists any association between a continuous variable such as document length or cosine similarity to the difficulty of the question-answering task.

Model Name	Agreement Rate
LLaMA-3.1-8B-Instruct	77.18%
GPT-4	95.01%

Table 2: The agreement rate between using groundtruth 1 as groundtruth, groundtruth 2 as "generated" answer and vice versa. An ideal model should achieve 100% agreement rate.

Context for each q_i	Accuracy	Flip Rate
s_i (summary only)	92.05%	–
d_i (story only)	59.93%	–
s_i in d_i (inserted)	83.15%	17.65%
McNemar Test	**p-value:**	2.659e-07
	χ^2**:**	26.483

Table 3: The accuracy and flip rate when conducting Needle-in-a-haystack-QA Test on LLaMA-3.1-8B-Instruct. The flip rate is calculated from s_i to s_i in d_i. In this table, it is assumed that $q_i \in Q$, $s_i \in S$ and $d_i \in D$ unless specified otherwise.

4 Result

4.1 Experiment 0: Validating the evaluator model

In this experiment, we tested two LLMs as the potential evaluator model: LLaMA-3.1-8B-Instruct and GPT-4. The overall agreement rate is shown in Table 2. Since GPT-4 achieved a much higher agreement rate close, we will be using GPT-4 as the evaluator model for the rest of experiments and LLaMA-3.1-8B-Instruct as the tested model.

4.2 Experiment 1: Examining Semantic Masking Effect

To demonstrate the effect of Semantic Masking, we conduct Needle-in-a-haystack-QA Test on all question-document pairs where the summary s_i

Difficulty	Number	Difficulty	Number
easy	260	commonsense	44
standard	36	confusing	24
puzzling	4	nonsense	15
mildly-challenging	120	incapable	10
challenging	10	invalid	38

Table 4: Distribution of questions according to their difficulty. These difficulty categories are assigned by looking at the tested model's performance on the Needle-in-a-haystack-QA Test

will serve as the needle and the full story d_i will serve as the semantic masking haystack.

In Table 3, we can observe a significant accuracy drop when the supplied context is the full story instead of the summary. This indicates that the examined long text does provide sufficient challenges to the tested model. When we conduct the Needle-in-a-haystack-QA Test on the summary-story pairs, the accuracy also drops by a large margin, which suggests the influence the haystack have on the needle.

To see the influence numerically, we compute the flip rate (defined in 1) for s_i in d_i that uses s_i result as before and s_i in d_i result as after. The experiment obtained a p-value of 2.659e-07 from the McNemar Test, which suggests that using the full story as haystack does impose a statistically significant effect on the task.

Given the fact that more than half of the questions can be answered with full story as the context, we perform a post hoc study on the questions that cannot be answered with full story. With only questions that cannot be answered with full story context, the flip rate reached 31.54% with a p-value of 4.828e-08 under the McNemar test. This result shows how the Semantic Masking effect depends not only on the semantic relatedness, but also on questions themselves.

4.3 Experiment 2: Question Difficulty Assessment

The above experiment showed how question difficulty could impact task difficulty. It is only natural to perform another post-hoc study upon experiment 1 by further categorizing question difficulty using our proposed assessment.

We first need to understand the distribution of the questions based on our assessment. In Table 4, we can clearly see that the majority of the questions fall into the family of reasonable difficulties on the left. Although over half of them are considered as easy questions, there are still a decent number of standard, challenging and mildly challenging questions that ramp up the overall difficulty of the Test.

Difficulty	Flip Rate	p-value
easy	4.231%	9.765e-04
standard	25.00%	3.906e-03
puzzling	25.00%	1.0
mildly-challenging	28.33%	1.518e-08
challenging	70.00%	0.25

Table 5: The flip rate and p-value from McNemar Test for questions of the 5 reasonable difficulty

Controlled Variable	Flip Rate	PBC
Fixed Sem Relatedness	2.869%	-0.054
Fixed Haystack Length	7.524%	-0.084

Table 6: The flip rate and the PBC score when choosing haystack with certain controlled variables. The values are calculated on 10 questions, each inserted into 16-29 haystacks that meets the selection criteria, which makes a total of 160-290 round of tests.

In Table 5, we can see that the flip rate generally aligns with the assigned difficulty level and is mostly of statistical significance, except two, which is likely due to lack of data. This experiment further demonstrates the importance of difficulty assessment.

4.4 Experiment 3: Controlling Haystack Properties

We test two haystack properties for this experiment: Text Length and Semantic Relatedness. We randomly selected 10 questions that are of reasonable difficulty, and 16-29 haystacks per question within the range mentioned above (25K$\pm$2.5K tokens, 0.3$\pm$0.02 cosine score), which makes a total of 160-290 rounds of tests.

For each round of test, we conduct 5 runs of the exact same setting using the first groundtruth $a_i^{(1)}$ and another 5 using the second groundtruth $a_i^{(2)}$. This is to ensure the output of the model is consistently evaluated. Evaluations that have disagreement between the first groundtruth evaluation and the second groundtruth evaluation are excluded. Table 6 shows the flip rate of the haystack selection when controlling the semantic relatedness and length of the haystacks, as well as the PBC score.

The PBC score is a clear indication that neither of the two properties separates the model performance.

When choosing haystacks of similar semantic relatedness (relatively low) and varying length (in this case, chosen haystacks have length between 5K to 50K tokens), the flip rate is at 2.869%, which indicates that changing the length of the haystack barely affects the difficulty of the task.

In contrast, when choosing haystacks of similar length and varying semantic relatedness, although still on the low end, the flip rate increased by about 2.6 times. This indicates that varying the semantic relatedness of the haystack is far more effective than varying the length when adjusting the difficulty of the task. We suspect that the reason for the low flip rate is that chosen haystacks can only span 0 to 0.6 cosine similarity scores. It is difficult to find stories that are of high similarities for each document within the dataset.

5 Conclusion

In this study, we proposed the Needle-in-a-haystack-QA Test to assess LLM's long text capabilities. Through the experiments we have drawn 2 major conclusions: 1) Length is not the primary factor that affects the difficulty of tests that follow the Needle-in-a-haystack approach; 2) Highly related haystack may impose Semantic Masking effect on the needle which exerts a more profound influence on LLM performance. Through these two conclusions, we wish to challenge the conventional emphasis on context length and suggest a more nuanced approach to evaluating LLM's long text capabilities.

We also propose a difficulty assessment framework that can be generalized to any question-answering dataset in assessing question difficulty. This framework is also essential in validating the meaningfulness of experiments designed from the Needle-in-a-haystack-QA Test.

There are also other factors we suspect may have an impact on the difficulty of the test, such as the position of needle insertion relative to the haystack. We will test these factors in subsequent experiments.

In conclusion, our work advocates for a more nuanced approach to evaluating and enhancing the long text capabilities of LLMs. By incorporating Semantic Masking considerations into evaluation metrics, we pave the way for the development of models that are not only proficient in handling extensive contexts but also adept at extracting and interpreting relevant information within them.

References

J. Andriessen, K. de Smedt, and M. Zock. 1996. Discourse planning: Empirical research and computer models. In T. Dijkstra and K. de Smedt, editors, *Computational Psycholinguistics: AI and Connectionist Models of Human Language Processing*, pages 247–278. Taylor & Francis, London.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *GitHub repository*, arXiv:2004.05150.

Peter F Brown, Stephen A Della Pietra, Vincent J Della Pietra, and Robert L Mercer. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

Roger Brown and David Mcneill. 1966. The "tip of the tongue" phenomenon. *Journal of Verbal Learning and Verbal Behavior*, 5:325–337.

Kedar Chandrayan, Lance Martin, Gregory Kamradt, Lazaro Hurtado, Arkady Arkhangorodsky, Ikko Eltociear Ashimine, Pavel Král, and Prabha Arivalagan. 2024. gkamradt/llmtest_needleinahaystack.

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

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

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

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. ELI5: Long form question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3558–3567. Association for Computational Linguistics.

Maor Ivgi, Uri Shaham, and Jonathan Berant. 2023. Efficient long-text understanding with short-text models. In *Transactions of the Association for Computational Linguistics*, volume 11, pages 284–299. MIT Press.

Hongye Jin, Xiaotian Han, Jingfeng Yang, Zhimeng Jiang, Zirui Liu, Chia-Yuan Chang, Huiyuan Chen, and Xia Hu. 2024. LLM maybe LongLM: SelfExtend LLM context window without tuning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 22099–22114. PMLR.

Tomáš Kočiský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The NarrativeQA reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.

Yuhong Li, Yingbing Huang, Bowen Yang, Bharat Venkitesh, Acyr Locatelli, Hanchen Ye, Tianle Cai, Patrick Lewis, and Deming Chen. 2024. SnapKV: LLM knows what you are looking for before generation. In *Proceedings of NeurIPS 2024*.

Antoine Louis, Gijs van Dijck, and Gerasimos Spanakis. 2024. Interpretable long-form legal question answering with retrieval-augmented large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 22266–22275.

Quinn McNemar. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and et al. 2024. GPT-4 technical report. *Preprint*, arXiv:2303.08774.

Richard Yuanzhe Pang, Alicia Parrish, Nitish Joshi, Nikita Nangia, Jason Phang, Angelica Chen, Vishakh Padmakumar, Johnny Ma, Jana Thompson, He He, and Samuel Bowman. 2022. QuALITY: Question answering with long input texts, yes! In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5336–5358, Seattle, United States. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. MPNet: Masked and permuted pre-training for language understanding. In *Proceedings of NeurIPS*.

Yingsheng Wu, Yuxuan Gu, Xiaocheng Feng, Weihong Zhong, Dongliang Xu, Qing Yang, Hongtao Liu, and Bing Qin. 2024. Extending context window of large language models from a distributional perspective. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7288–7301, Miami, Florida, USA. Association for Computational Linguistics.

Jun Zhao, Can Zu, Hao Xu, Yi Lu, Wei He, Yiwen Ding, Tao Gui, Qi Zhang, and Xuanjing Huang. 2024. LongAgent: Scaling language models to 128k context through multi-agent collaboration. *Preprint*, arXiv:2402.11550.

Michael Zock. 2006. Needles in a haystack and how to find them? the case of lexical access. *Linguistics in the Twenty First Century*, pages 155–162.