Large Language Models and Book Summarization: Reading or Remembering, Which Is Better?

Tairan Fu[1], Javier Conde[2], Pedro Reviriego[2], Javier Coronado-Blázquez[3], Nina Melero[2,4] and Elena Merino-Gómez[5]

[1]Politecnico di Milano, Italy
[2]Information Processing and Telecommunications Center, ETSI Telecomunicación, Universidad Politécnica de Madrid, Spain
[3]Banco de España, Spain
[4]New York University, Madrid Campus, Spain
[5]Escuela de Ingenierías Industriales, Universidad de Valladolid, Spain

Abstract

Summarization is a core task in Natural Language Processing (NLP). Recent advances in Large Language Models (LLMs) and the introduction of large context windows reaching millions of tokens make it possible to process entire books in a single prompt. At the same time, for well-known books, LLMs can generate summaries based only on internal knowledge acquired during training. This raises several important questions: How do summaries generated from internal memory compare to those derived from the full text? Does prior knowledge influence summaries even when the model is given the book as input? In this work, we conduct an experimental evaluation of book summarization with state-of-the-art LLMs. We compare summaries of well-known books produced using (i) only the internal knowledge of the model and (ii) the full text of the book. The results show that having the full text provides more detailed summaries in general, but some books have better scores for the internal knowledge summaries. This puts into question the capabilities of models to perform summarization of long texts, as information learned during training can outperform summarization of the full text in some cases.

Keywords

LLMs, Evaluation, Summarization

1. Introduction

Summarization is one of the most common tasks in Natural Language Processing (NLP) [1]. Over the years, many techniques and models have been used for summarization, typically trained on pairs of documents, summaries. For example, early methods relied on statistical features such as term frequency, sentence position, and cue phrases to select relevant sentences [2]. Later, neural network–based approaches, particularly sequence-to-sequence models, performed summarization by generating new sentences that capture the meaning of the source document [3]. More recently, transformer-based pre-trained encoders like BERT have been used to improve summarization performance by using rich contextual representations [4].

The development of Large Language Models (LLMs) opened new directions for summarization, since the models can be given the text and asked to summarize it in a single prompt [5]. The use of LLMs can be more sophisticated by using prompt engineering, model fine-tuning, or knowledge distillation to achieve better performance or use smaller models [2]. For well-known texts, another option is to rely on the model's internal knowledge to provide a summary, for example, of a book without providing the text [6].

In fact, LLMs are commonly seen to have two types of memories: internal (or parametric) and external [7]. The internal memory is encoded in the model parameters, representing the knowledge acquired during training. This parametric memory has been shown to store facts, linguistic patterns, and in some cases even verbatim passages from training data [8]. In contrast, the external memory is provided through the context window at inference time. In the case of summarization, it allows one to provide the model with the text to summarize. The size of the context window limits the amount of external memory, with the first generation of models having windows of just a few thousand tokens, while newer models such as GPT-4.1 and Gemini 2.5 reach the million-token range. Since most books have at most a few hundred thousand words, one million tokens is sufficient to pass the entire book in the input prompt for summarization. These two forms of memory can interact with parametric knowledge overriding contextual input, or the two may reinforce each other [9]. However, the interaction between memorization in weights and processing of contextual input remains an open research question.

In this context, the study of book summarization based on internal and external memory is an interesting topic that can improve our understanding of the inner workings of LLMs and also of their limitations when performing summarization of large texts. This paper takes an initial step in this direction by performing a comparison of book summaries generated by LLMs based on internal and external knowledge. In particular, the following contributions are made:

1. Conduct to the best of our knowledge the first evaluation comparing LLM generated book summaries with internal and external memory using state-of-the-art models with large context windows.
2. Create an open repository with the code, books, and summaries of the experiments to facilitate further research on the topic and reproducibility of our results.
3. Show that for most books, the use of the full text for summarization improves performance compared to summaries based only on parametric memory.
4. Show that for some books, internal knowledge summaries get better scores than summaries done with access to the full text of the book.
5. Discuss the potential reasons for the better perfor-

SEPLN 2026: 42nd International Conference of the Spanish Society for Natural Language Processing
tairan.fu@polimi.it (T. Fu); javier.conde@upm.es (J. Conde); pedro.reviriego@upm.es (P. Reviriego); javier.coronado@telefonica.com (J. Coronado-Blázquez); nina.melero@nyu.edu (N. Melero)

mance of internal summaries.

The rest of the paper is organized as follows. Section 2 summarizes related works on text summarization focusing on the use of LLMs and on the different types of memory in LLMs and their interactions. The methodology used for the evaluation is described in Section 3 and the results are presented and discussed in Section 4. The paper ends with the conclusion in Section 5.

2. Related work

There are many recent works on summarization and LLM memory. The following subsections provide an overview of current research activities on these topics as well as some historical context.

2.1. Text summarization

As discussed in the Introduction, text summarization has been a central task in NLP for decades [1]. The techniques used to produce summaries have evolved from early statistical-based methods that rely on word frequencies [10] to the use of complex neural models such as LLMs. There are two main types of automatically generated summaries: extractive and abstractive [2]. In the first case, the summary is built by extracting fragments from the original text, while in the latter, the process generates new text to summarize the input. Initially, summarization approaches tried to find the most relevant parts of the text by computing metrics on words and sentences and extracting them to build the summary. The use of classical machine learning algorithms was also explored for extractive summarization [11].

The development of powerful deep learning models led to the use of more advanced neural architectures such as recurrent neural networks (RNN), convolutional neural networks (CNN), and long-short-term memory networks (LSTM) for summarization [3],[12]. The introduction of the Transformer architecture revolutionized NLP in general and text summarization in particular. Bidirectional Encoder Representations from Transformers (BERT), with its bidirectional encoder, has been widely used in extractive summarization, where embeddings are leveraged to select the most salient content from a document [4]. Instead, the Text-to-Text Transfer Transformer (T5) has been used for abstractive summarization, as autoregressive decoders enable the generation of fluent and coherent summaries beyond simple sentence extraction [13]. The use of Transformers set a new state-of-the-art and laid the foundations for the emergence of large language models designed to handle summarization as part of a broader range of complex natural language tasks.

Large Language Models (LLMs) have shown good performance for text summarization, leveraging their extensive training on massive amounts of text to generate coherent, contextually rich, and human-like summaries [14]. Unlike earlier task-specific architectures, these models demonstrate strong zero-shot and few-shot capabilities, enabling effective summarization without the need for extensive fine-tuning. Recent benchmarks consistently highlight their superiority over traditional encoder–decoder models, although challenges remain in ensuring factual consistency, mitigating hallucinations, and aligning outputs with user needs [15],[16]. Early approaches to book summarization were constrained by narrow context windows, necessitating "divide-and-conquer" strategies or hierarchical merging of chapter-level summaries [17]. However, a few recent models have introduced context windows spanning millions of tokens, enabling the processing of entire literary works in a single prompt.

2.2. Large Language Model Memory

The memory of LLMs has been widely studied to try to understand how models learn and memorize information in their parameters during training [7]. In some cases, models reproduce the content of their training data verbatim, raising concerns about data leakage and potential privacy violations [18]. The amount of information stored by the models appears to be related to the number of parameters, with some works indicating that the information that can be memorized is the number of parameters in the model multiplied by a small number of bits [8].

In addition to this parametric memory, which is filled during model training, LLMs have a dynamic memory that can be defined at inference time and is given by their context window. This window enables the user to provide information, such as text to summarize and the instructions to do it, at inference time. This information is upper-bounded by the maximum input length supported by the model architecture, which constrains the amount of information that can be processed at once. The size of the context window has increased from a few thousand tokens in early models to over one million tokens in state-of-the-art models [19]. However, having a large context window does not guaranty that the model can always use this information correctly [20]. Several issues have been identified, with models often overemphasizing recent tokens while neglecting information from earlier parts of the input, a phenomenon known as "Lost in the Middle" [21]. Also, as the input length grows, models also struggle with context fragmentation making it difficult to integrate relevant information consistently across distant segments of the input text [22].

Another interesting topic is the interaction between both types, parametric and dynamic, of memory, with recent studies showing that models tend to rely on the information provided in the input text over their internal knowledge [9]. This is relevant for book summarization, as for well-known books models will have internal knowledge that can interfere with the book text provided as input.

3. Methodology

This section describes the methodology and experimental setup used in our evaluation[1]. First, the overall approach is presented, followed by a discussion of the books, LLMs and prompts considered.

3.1. Overall approach

The evaluation follows the procedure used in [6] to generate summaries based only on the internal knowledge of the LLM. First, an LLM is used to generate the book summaries. In the case of internal knowledge summaries, we use a prompt only with the instructions, while for the external knowledge summaries, the input includes the full text of the book in addition to the instructions.

[1]The code and data including the books and summaries are available at https://github.com/aMa2210/llm-summary-internal-vs-external.

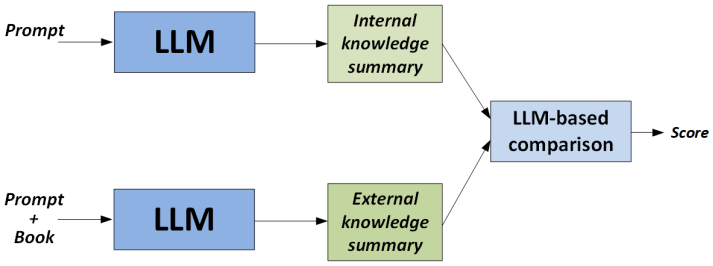


Figure 1: Overall approach to generate and evaluate the summaries.

Once the summaries are generated, they are compared. Beyond traditional overlap-based metrics, recent literature has popularized the LLM-as-a-judge framework, in which LLMs are used to evaluate quality across dimensions such as faithfulness, relevance, and coherence, demonstrating high correlation with human judgment [23]. We adopt this scheme, and comparisons are made using the generation model itself or a second LLM. As a result, we have a quality score for each book. The general scheme is illustrated in Figure 1.

3.2. Books

The selection of books chosen for our evaluation was made to ensure that they are: 1) available in the public domain, 2) widely known, and 3) cover a wide range of authors, genres, and styles. In particular, a subset of the books considered in [6] that are in the public domain was used. The selection was made by two experts in literature. The list of books evaluated is shown in Table 1. It can be seen that all are widely known books that are for sure included in the training datasets of state-of-the-art LLMs and cover different periods and styles. English translations of the original text are used when the book was written in another language.

There are a number of features that can influence the difficulty of writing a summary for an LLM. For example, intuitively, the longer the book, the harder it is for the LLM to write a summary, as it needs to analyze and relate more tokens. In our set, the longest book, “The Count of Monte Cristo”, has close to half a million words and uses most of the LLM context window. On the other hand, the shortest one, “A Doll’s House” has less than thirty thousand words. However, length is not the only feature that affects the difficulty of summarizing. The complexity of the plot, the writing style, or the type of writing: prose, verse, or drama can also make summarizing difficult. For instance, capturing the plot of James Joyce’s Ulysses is intuitively more demanding than that of a conventional novel, and summarizing verse is likewise harder than summarizing prose. In fact, for some books summarizing is challenging even for humans as illustrated in Monty Python’s "Summarizing Proust Competition" sketch, where contestants are given fifteen seconds to condense Marcel Proust’s seven-volume masterpiece. The sketch highlights a fundamental truth: some works defy traditional summarization because their essence lies in psychological depth and style rather than a linear plot. For both humans and language models, stripping away these structural nuances to force a brief synopsis often destroys the work’s actual meaning.

3.3. LLMs

The models must be able to store the entire book in the context window to support external knowledge summarization. The longest book considered in the experiments is "The Count of Monte Cristo", which is about half a million words. Since typically up to 1.3-1.4 tokens per word are needed on average, placing the book in the context window can take close to 600,000 tokens[2]. At the time of writing this paper, very few models had context windows capable of storing this amount of information, and most of them were previews or not accessible to all users. There were, however, two widely used models with context windows of one million tokens and state-of-the-art performance: GPT-4.1 and Gemini-2.5, that we selected for our experiments. Summaries were generated with the two models and also evaluated with the two models. This enables us to detect evaluation biases where, for example, a model tends to favor its own summaries [24].

3.4. Instructions

The prompts used to generate the summaries follow the instructions used in [6] and focus on providing a detailed account of the plot, events, and characters. For the internal knowledge summaries, the prompt used is as follows:

“Provide a very detailed summary of the plot for the book "{title}" by {author}. The summary must be of the original book, NOT any adaptation like a film or TV show. Include all main events and the complete storyline, detailing every key development, situations, events with characters, and the conclusion. Do NOT include any historical context, literary analysis, or philosophical discussion, only the plot.”

This prompt focuses on a factual summary which, as discussed in the previous subsection, may not be meaningful or appropriate for all literary works.

To generate the summaries using the full text of the books, we used a similar prompt, adding that the model should write the summary based on the text provided and not use its internal knowledge:

“Provide a very detailed summary of the plot for the following book. The summary must be of the text provided, do NOT use any internal or previous knowledge that you may have on that book to create the summary. Include all main events and the complete storyline, detailing every key development, situations, events with characters, and the conclusion. Do NOT include any historical context, literary analysis, or philosophical discussion, only the plot. Full text of the book: "{fulltext of the book}".

[2] https://platform.openai.com/docs/concepts/tokens

Table 1
Books considered in the evaluation

No.	Author	Book
1	Jane Austen	*Pride and Prejudice*
2	Giovanni Boccaccio	*The Decameron*
3	Dante Alighieri	*The Divine Comedy*
4	Charlotte Brontë	*Jane Eyre*
5	Lewis Carroll	*Alice's Adventures in Wonderland*
6	Miguel de Cervantes	*Don Quixote*
7	Charles Dickens	*Great Expectations*
8	Daniel Defoe	*Robinson Crusoe*
9	Fyodor Dostoyevsky	*Crime and Punishment*
10	Alexandre Dumas	*The Count of Monte Cristo*
11	F. Scott Fitzgerald	*The Great Gatsby*
12	Gustave Flaubert	*Madame Bovary*
13	Nathaniel Hawthorne	*The Scarlet Letter*
14	Homer	*The Iliad*
15	Henrik Ibsen	*A Doll's House*
16	James Joyce	*Ulysses*
17	Luigi Pirandello	*Six Characters in Search of an Author*
18	William Shakespeare	*Hamlet*
19	Mary Shelley	*Frankenstein*
20	Bram Stoker	*Dracula*
21	Leo Tolstoy	*Anna Karenina*
22	Mark Twain	*The Adventures of Huckleberry Finn*
23	Jules Verne	*20,000 Leagues Under the Sea*
24	Virginia Woolf	*Mrs. Dalloway*
25	Thomas Mann	*Death in Venice*

It can be seen that no information is given on the author or title or the book and the models are instructed not to use any internal knowledge they may have. However, explicitly instructing the model to disregard its internal knowledge does not guarantee compliance, as LLMs frequently struggle with negative constraints and can still inadvertently rely on pre-trained data.

Finally, to compare the LLM-generated summaries, the following prompt was used:

"Please compare the following two summaries and tell me which is more complete and contains information on the complete storyline, key developments, events or characters missing in the other. If the first contains more information at the end, return a score of 1, if both summaries are the same of 0 and if the second is more complete of -1. {LLM generated internal summary} {LLM generated external summary}"

Here, we rely on the evaluative capabilities of the LLM, assuming that the order in which the summaries are presented does not bias the final assessment, while utilizing a simple three-point discrete metric. As discussed, for this last prompt, use the two models ensures that GPT-4.1 and Gemini-2.5 judge their own summaries and also those written by the other model. This enables us to detect any potential self-bias in the evaluation [24].

3.5. Procedure

As summarization must not introduce modifications to the information provided in the text, initially, we set the temperature to 0.4, a relatively low value. This provides a trade-off between avoiding models from deviating from the text and providing some freedom to write the summaries. To check the impact of the temperature on the summaries, the experiments were also run with a higher temperature value of 0.9 to assess the impact of enabling greater creativity in the models on the quality and variability of the summaries. For each model, we generate the summaries five times as this enables us to evaluate also the consistency of the models when writing summaries. Then comparisons are done pairwise between each of the internal and external summaries, so if the internal summaries are I_1, I_2, I_3, I_4, I_5 and the external E_1, E_2, E_3, E_4, E_5 we evaluate twenty-five pairs:

$$\{(I_1, E_1), (I_1, E_2), (I_1, E_3), (I_1, E_4), (I_1, E_5),$$
$$(I_2, E_1), (I_2, E_2), (I_2, E_3), (I_2, E_4), (I_2, E_5),$$
$$(I_3, E_1), (I_3, E_2), (I_3, E_3), (I_3, E_4), (I_3, E_5),$$
$$(I_4, E_1), (I_4, E_2), (I_4, E_3), (I_4, E_4), (I_4, E_5),$$
$$(I_5, E_1), (I_5, E_2), (I_5, E_3), (I_5, E_4), (I_5, E_5)\}$$

and add the results to obtain a score for each book in the range of -25 to 25.

4. Results and Analysis

To get an initial idea of the results, Figure 2 shows the average scores of the summaries generated at temperatures 0.4 and 0.9 for the 25 books. For each of them, the generator and judge models are shown horizontally and vertically, resulting in a 2-by-2 matrix for each temperature value. It can be seen that all values are negative, indicating that external summaries consistently outperform internal ones. Therefore, providing the full text improves the quality of the summarization. However, the degree of this improvement depends strongly on both the generator and the judge.

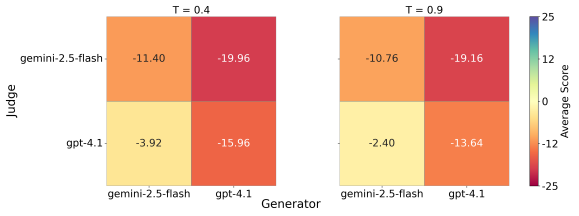


Figure 2: Comparison of the average scores across all the books.

The summaries generated by Gemini-2.5-Flash receive less negative scores regardless of who performs the evaluation. This suggests that Gemini's is less capable of summarizing from full text or that it maintains a more consistent performance when summarizing from prior knowledge. Regarding the evaluation, GPT-4.1 as a judge tends to produce scores that are significantly higher on average than those

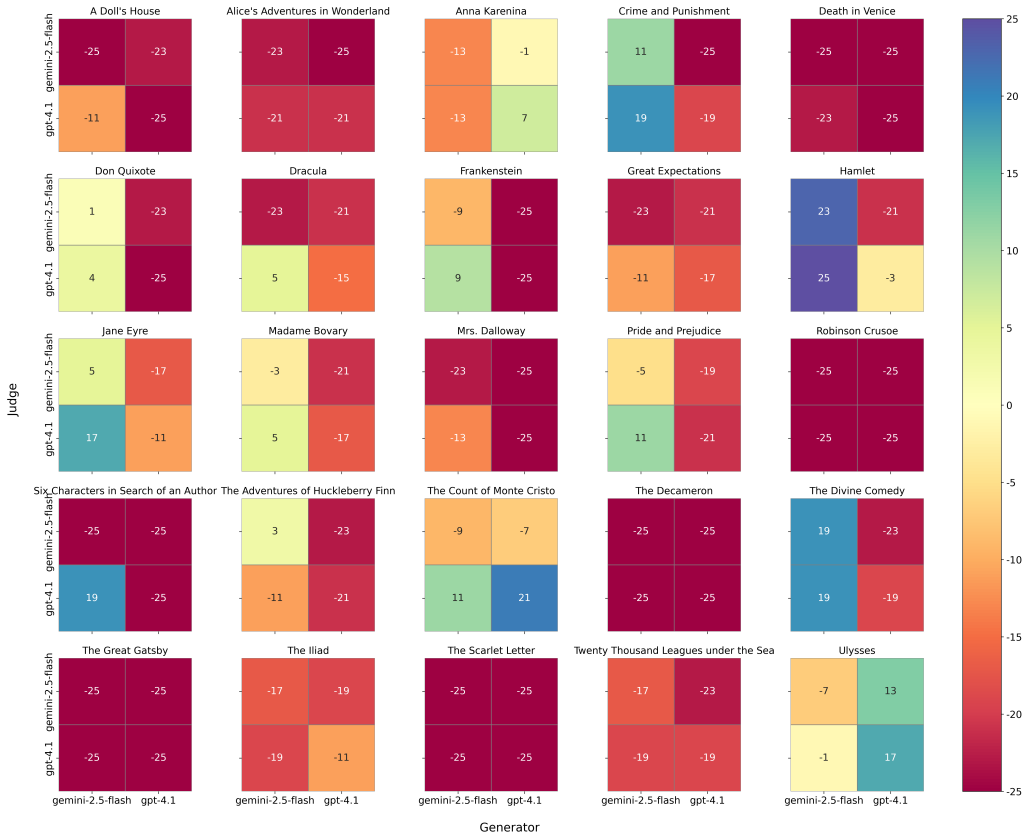


Figure 3: Comparison results obtained at temperature 0.4.

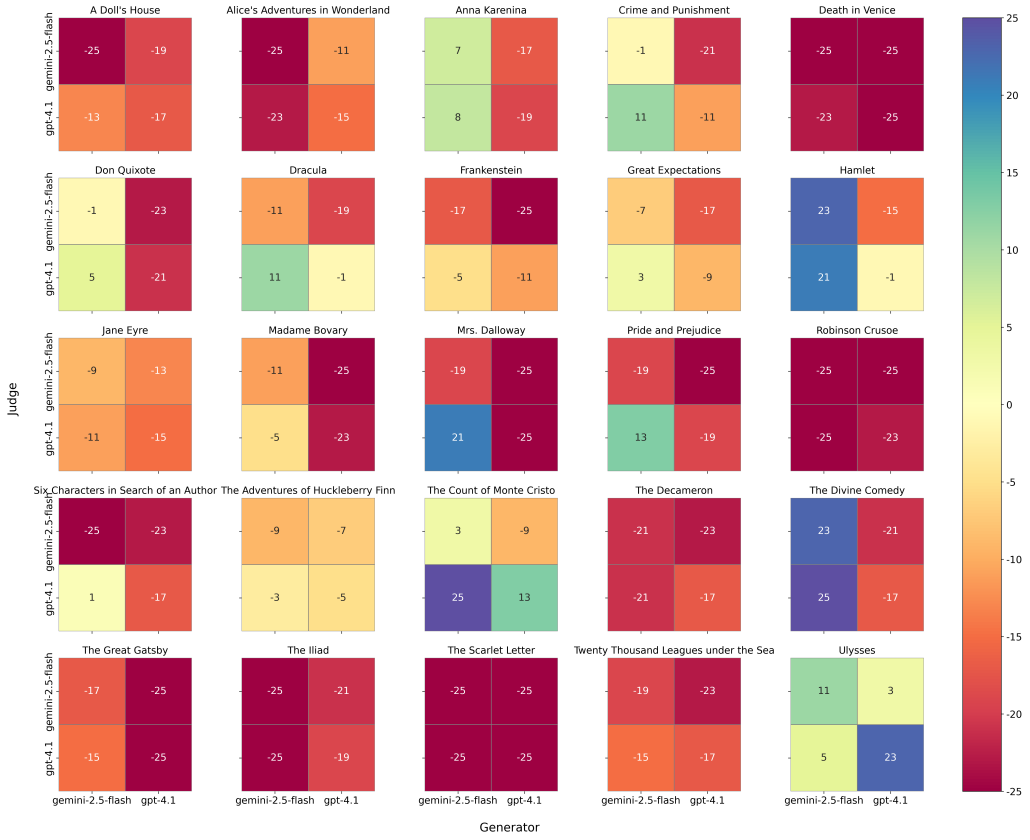
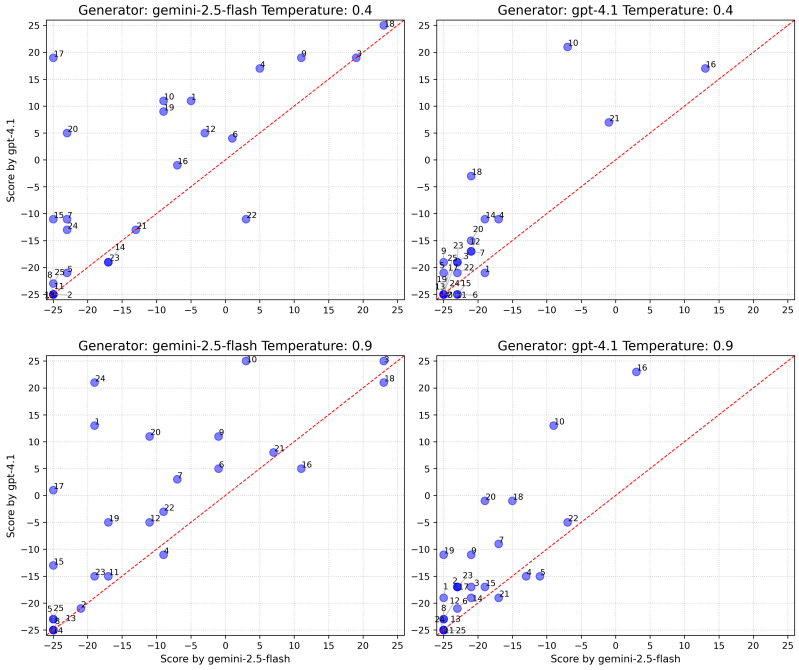


Figure 4: Comparison results obtained at temperature 0.9.



ID	Book Title
1	Pride and Prejudice
2	The Decameron
3	The Divine Comedy
4	Jane Eyre
5	Alice's Adventures in Wonderland
6	Don Quixote
7	Great Expectations
8	Robinson Crusoe
9	Crime and Punishment
10	The Count of Monte Cristo
11	The Great Gatsby
12	Madame Bovary
13	The Scarlet Letter
14	The Iliad
15	A Doll's House
16	Ulysses
17	Six Characters in Search of an Author
18	Hamlet
19	Frankenstein
20	Dracula
21	Anna Karenina
22	The Adventures of Huckleberry Finn
23	20,000 Leagues Under the Sea
24	Mrs. Dalloway
25	Death in Venice

Figure 5: Scatter plots of the summaries scores by evaluator model.

of Gemini for the same generator, so favoring internally generated summaries. The temperature has a small impact, increasing the temperature from 0.4 to 0.9 slightly increases the scores, suggesting that higher creativity narrows the gap between internal and external summaries. However, the relative ranking of model and judge combinations remains stable across temperatures. In summary, external context remains decisive, even with extensive prior knowledge, both LLMs produce notably better summaries when given access to the full text.

More detailed information is provided in Figures 3 and 4, which provide the same information but for each of the books. As expected, for most books, scores take negative values in many cases below -20, confirming that external summaries are generally preferred. However, several books deviate significantly from this pattern, with neutral or positive mean values in one or more generator/judge pairs. The most relevant ones were analyzed in more detail.

- *The Divine Comedy and Hamlet* had positive results when Gemini-2.5-Flash was the generator (19 to 25) and much lower when GPT4-1 generated the summaries (−3 to −25).
- *Ulysses* had positive results in most cases and large variations when increasing the temperature.
- *The Count of Monte Cristo* had very high scores, above 20 for some generator/evaluator pairs.

Some samples of the generated summaries for these four books were then manually analyzed. The human evaluation was generally in line with the LLM judgments. In the process, it was found that Gemini's summaries for the Divine Comedy were in some cases incomplete. This was not due to any limitation in the number of output tokens. It may be related to the fact that Gemini's external summaries tend to follow a chapter by chapter structure and the Divine Comedy has a large number of them. This failure illustrates the limitations of LLMs and the need for validation and cross checking of their outputs. These results suggest that for experimental narratives (Ulysses), high variance reflects the struggle to map non-linear structures while for massive plot-driven works (The Count of Monte Cristo), internal summaries avoid the "lost-in-the-middle" pitfalls of full-text processing.

This analysis of the outliers reveals that LLM summarization accuracy is text-dependent, not uniformly distributed across the literary corpus, even if all 25 books are classical and very well-known texts, to which models have certainly been extensively exposed during training. Factors such as genre, length, and plot complexity appear to influence performance, with poetry, theater, and long or complex books showing relatively better results for summaries generated from the model's internal knowledge than from the full text of the books.

To better compare the evaluator models, the scores for the summaries generated by GPT-4.1 and Gemini 2.5 Flash are shown as points in two dimensions, one corresponding to the score when Gemini 2.5 Flash is the judge and the other when GPT-4.1 is the evaluator. This visualizes the consistency in the evaluations. It can be seen that the evaluators generally agree on which summaries are better, with GPT-4.1 favoring internal summaries for the two generators and temperatures. Increasing the temperature slightly increases the dispersion of the results.

5. Conclusion

In this paper, we have compared LLM-generated summaries of well-known books based on prior knowledge with those generated from the full text of the books. The results show that summaries produced with the full text are consistently preferred over summaries generated purely from parametric knowledge. However, for some books (e.g., The Divine Comedy, Hamlet, Ulysses, The Count of Monte Cristo) internal summaries outperform the ones done with the full text in many configurations. Therefore, in some cases, re-

membering is better than reading. This suggests that the summarization of long texts remains a challenging task to LLMs when dealing with very long, poetic, or structurally complex works. Nevertheless, further research is required to confirm whether this 'reading over remembering' effect persists across a broader range of models and literary works.

Limitations

The findings of this work should be interpreted with caution, as several limitations may influence the results:

- Models: only two closed commercial LLMs (GPT-4.1, Gemini-2.5-Flash) were tested; the results may change with versions, other vendors, or open-weight models.
- Dataset: the 25 books evaluated are well-known, mostly Western books. The results may be different for other books.
- Prompting: single prompts focusing on the plot and characters were used for summary generation and evaluation; other prompt styles should be evaluated to assess their impact on the results.
- Judge: LLM-as-a-judge can be biased toward its own generations or produce inconsistent evaluations. Although cross-judging was implemented, additional validation from human experts would be beneficial.
- Metric Scope: The metric only measures plot completeness, meaning a detailed but inaccurate or hallucinated summary could still achieve a high score.
- Hallucination Analysis: Factual errors or fabrications were not manually checked, which is a risk for internal knowledge retrieval.
- Statistical Robustness: Results do not include standard deviations, confidence intervals, or significance testing to verify if observed differences are meaningful.
- Generalization: Findings may not transfer to other documents such as technical documentation or less known books.

6. Acknowledgements

This work was supported by the FUN4DATE (PID2022-136684OB-C22) and SMARTY (PCI2024-153434) projects funded by the Spanish Agencia Estatal de Investigación (AEI) 10.13039/501100011033, by TUCAN6-CM (TEC-2024/COM-460), funded by CM (ORDEN 5696/2024) and by the Chips Act Joint Undertaking project SMARTY (Grant no. 101140087). The evaluation was also done in part with equipment that was donated by NVIDIA to support our research.

References

[1] I. Mani, Automatic Summarization, John Benjamins Publishing Company, Amsterdam, The Netherlands, 2001.

[2] Y. Zhang, H. Jin, D. Meng, J. Wang, J. Tan, A comprehensive survey on process-oriented automatic text summarization with exploration of llm-based methods, arXiv preprint arXiv:2403.02901 (2024).

[3] R. Nallapati, B. Zhou, C. dos Santos, Ç. Gulçehre, B. Xiang, Abstractive text summarization using sequence-to-sequence RNNs and beyond, in: S. Riezler, Y. Goldberg (Eds.), Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, Association for Computational Linguistics, Berlin, Germany, 2016, pp. 280–290. URL: https://aclanthology.org/K16-1028/. doi:10.18653/v1/K16-1028.

[4] Y. Liu, M. Lapata, Text summarization with pretrained encoders, in: K. Inui, J. Jiang, V. Ng, X. Wan (Eds.), Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3730–3740. URL: https://aclanthology.org/D19-1387/. doi:10.18653/v1/D19-1387.

[5] L. Basyal, M. Sanghvi, Text summarization using large language models: a comparative study of mpt-7b-instruct, falcon-7b-instruct, and openai chat-gpt models, arXiv preprint arXiv:2310.10449 (2023).

[6] J. Coronado-Blázquez, Evaluating book summaries from internal knowledge in large language models: a cross-model and semantic consistency approach, arXiv preprint arXiv:2503.21613 (2025).

[7] N. Carlini, D. Ippolito, M. Jagielski, K. Lee, F. Tramer, C. Zhang, Quantifying memorization across neural language models, in: The Eleventh International Conference on Learning Representations, 2022.

[8] J. X. Morris, C. Sitawarin, C. Guo, N. Kokhlikyan, G. E. Suh, A. M. Rush, K. Chaudhuri, S. Mahloujifar, How much do language models memorize?, arXiv preprint arXiv:2505.24832 (2025).

[9] S. Cheng, L. Pan, X. Yin, X. Wang, W. Y. Wang, Understanding the interplay between parametric and contextual knowledge for large language models, arXiv preprint arXiv:2410.08414 (2024).

[10] H. P. Luhn, The automatic creation of literature abstracts, IBM Journal of research and development 2 (1958) 159–165.

[11] Y. Chali, S. A. Hasan, S. R. Joty, A svm-based ensemble approach to multi-document summarization, in: Canadian Conference on Artificial Intelligence, Springer, 2009, pp. 199–202.

[12] S. Song, H. Huang, T. Ruan, Abstractive text summarization using lstm-cnn based deep learning, Multimedia Tools and Applications 78 (2019) 857–875.

[13] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, Journal of Machine Learning Research 21 (2020) 1–67. URL: http://jmlr.org/papers/v21/20-074.html.

[14] T. Zhang, F. Ladhak, E. Durmus, P. Liang, K. McKeown, T. B. Hashimoto, Benchmarking large language models for news summarization, Transactions of the Association for Computational Linguistics 12 (2024) 39–57.

[15] X. Chen, Z. Chen, S. Cheng, Cothssum: Structured long-document summarization via chain-of-thought reasoning and hierarchical segmentation, Journal of King Saud University Computer and Information Sciences 37 (2025) 40.

[16] H. Askari, A. Chhabra, M. Chen, P. Mohapatra, Assessing llms for zero-shot abstractive summarization through the lens of relevance paraphrasing, arXiv preprint arXiv:2406.03993 (2024).

[17] J. Wu, L. Ouyang, D. M. Ziegler, N. Stiennon, R. Lowe,

J. Leike, P. Christiano, Recursively summarizing books with human feedback, arXiv preprint arXiv:2109.10862 (2021).

[18] N. Carlini, F. Tramer, E. Wallace, M. Jagielski, A. Herbert-Voss, K. Lee, A. Roberts, T. Brown, D. Song, U. Erlingsson, et al., Extracting training data from large language models, in: 30th USENIX security symposium (USENIX Security 21), 2021, pp. 2633–2650.

[19] G. Team, P. Georgiev, V. I. Lei, R. Burnell, L. Bai, A. Gulati, G. Tanzer, D. Vincent, Z. Pan, S. Wang, et al., Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context, arXiv preprint arXiv:2403.05530 (2024).

[20] H. Jin, X. Han, J. Yang, Z. Jiang, Z. Liu, C.-Y. Chang, H. Chen, X. Hu, Llm maybe longlm: Self-extend llm context window without tuning, arXiv preprint arXiv:2401.01325 (2024).

[21] N. F. Liu, K. Lin, J. Hewitt, A. Paranjape, M. Bevilacqua, F. Petroni, P. Liang, Lost in the middle: How language models use long contexts, arXiv preprint arXiv:2307.03172 (2023).

[22] Y. Bai, X. Lv, J. Zhang, H. Lyu, J. Tang, Z. Huang, Z. Du, X. Liu, A. Zeng, L. Hou, Y. Dong, J. Tang, J. Li, LongBench: A bilingual, multitask benchmark for long context understanding, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 3119–3137. URL: https://aclanthology.org/2024.acl-long.172/. doi:10.18653/v1/2024.acl-long.172.

[23] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. Xing, et al., Judging llm-as-a-judge with mt-bench and chatbot arena, Advances in neural information processing systems 36 (2023) 46595–46623.

[24] W. Xu, G. Zhu, X. Zhao, L. Pan, L. Li, W. Y. Wang, Pride and prejudice: Llm amplifies self-bias in self-refinement, arXiv preprint arXiv:2402.11436 (2024).