The Distraction Penalty: Cost-Efficient Long-Document Summarization

Anonymous ACL submission

Abstract

Large Language Models (LLMs) enable long-context summarization with larger input windows. However, increased length comes with proportionally increasing costs for inference. In this paper, we present an upper bound analysis of how much input context can be reduced without degrading summary quality. We use reference based salience scoring as an oracle and vary token budgets across three domains to characterize compression-cost relationships, validated using both ROUGE and BERTScore. Under oracle-guided selection using reference summaries as an upper bound, selective truncation reduces inference costs by 47–62%, and in verbose domains actively improves ROUGE-1 by nearly 10% over full-context baselines. We demonstrate that the optimal truncation strategy is domain-dependent, providing empirically-grounded domain-sensitive routing heuristics derived from quantitative positional analysis. Code is available at https://anonymous.4open.science/r/SAT.

1 Introduction

Large language models (LLMs) have demonstrated strong performance on abstractive summarization of long-form documents across multiple domains (Beltagy et al., 2020; Shaham et al., 2022). Due to the increasing usage of LLMs for such tasks, the costs incurred from each API call increase significantly. This brings us to the need for principled decisions about how much input text should be provided to the model.

Existing production agentic frameworks tend to utilize the best model or multiple models to guarantee the accuracy of the results. This comes at a detriment to both the cost and the speed of responses. Prior work has also shown that larger context windows do not necessarily lead to better summarization (Ravaut et al., 2024). This raises a fundamental question: how much input context is actually necessary to achieve high quality summaries?

There is also an implicit assumption that more tokens provide better output. Long-form documents tend to contain redundancy, background information, or off-topic content. Such characteristics may cause the model to hallucinate or affect the summary quality. It is also noted that models often focus more on the initial and final segments of the input text (Liu et al., 2024). Understanding whether summarization quality improves with increasing context or saturates beyond a certain point has important implications for both system design and deployment costs.

To address this, we perform an upper-bound analysis using salience-guided input truncation. We manipulate the input document to better fit our summarization architecture. We employ multiple salience scoring heuristics to rank document chunks by estimated relevance, retaining only the most salient portions. We compare this oracle extraction against simple baselines (first-k, random-k, lead-n) under fixed token budgets, following standard practices (Nallapati et al., 2017). By holding the summarization model, decoding strategy, and evaluation metrics fixed, we successfully isolate the effect of input selection from other confounding variables to determine when additional context stops helping.

Our experiments cover multiple domains to characterize how the quality-compression trade-off differs across document types. In summary, our primary contributions are:

- **Cost-Reduction via Selective Truncation:** Under oracle-guided selection, we demonstrate that selective truncation can reduce token volume by 46–61% with minimal quality loss. Using model-specific pricing proxies, this lowers inference costs by 47–62% depending on the domain and truncation strategy.
- **Mitigation of the Distraction Penalty:** In

noisy, verbose domains, we show that oracle-based truncation not only saves costs but actively improves ROUGE-1 scores (Lin, 2004) relative to full-context processing, with a relative improvement of 9.6% for GovReport under hybrid salience truncation.

- **Domain-Sensitive Routing Heuristic:** We provide empirical evidence that the optimal truncation strategy is domain-dependent. For journalistic text, naive positional truncation achieves near zero quality loss. For scientific literature, oracle salience selection achieves quality statistically indistinguishable from full context while reducing costs by 55%. For verbose government documents, any truncation improves quality over the full-context baseline.

2 Literature Review

2.1 Salience and Context selection in Summarization

Prior work has explored the use of salience to guide model behavior. Salient Information Prompting (Xu et al., 2024) proposes salience aware prompting to "steer" abstractive summarization by explicitly prompting the model with key information. This work largely operates within the paradigm of prompt augmentation, i.e., adding markers to guide the model to improve summarization. Our approach functions as the inverse: prompt reduction where salience scores are used as heuristic proxies to rank chunks within documents.

2.2 Context Utilization in Long Document Summarization

Recent work has challenged the assumption that longer input contexts improve summarization. Ravaut et al. (2024) demonstrate that most LLMs often fail to effectively utilize information from very long inputs. Our work extends this literature by providing domain-stratified quantitative analysis linking positional content biases to measurable inference cost reductions. This leads to the *MiddleSum* (Ravaut et al., 2024) phenomenon, where context from the middle of the documents is underrepresented. These findings suggest that summarization quality may degrade beyond a certain input length, especially for long-form documents. Additionally, indiscriminately increasing context length can incur higher costs without any gain in output quality. We provide a granular analysis of how this utilization varies across domains (e.g., news vs. scientific literature). We also explicitly link this behavioral phenomenon to financial inference costs.

2.3 Token Pruning and Efficiency in Large Language Models

As the context windows grow, the computational costs of the attention-mechanism incurred during the prefill stage increase drastically. Methods like **Saliency-driven Dynamic Token Pruning (SDTP)** (Tao et al., 2025) reduce overhead by selectively removing redundant tokens during model execution. Such methods primarily target architectural or inference time optimization, reducing FLOPs and memory usage. Other system level approaches like **LLMLingua** (Gu et al., 2023) focus on compressing the input prompt before it reaches the model. We however treat token reduction as a **data selection problem**. We study input level truncation as an external variable, holding the summarization backend fixed and analyzing how reducing input context impacts downstream quality and inference costs. We also explicitly quantify the overhead cost of salience computation itself, demonstrating that scoring overhead is economically negligible relative to inference savings. We do not compare against token-level compression methods such as LLMLingua, which operate at subword granularity rather than sentence-chunk level and represent a complementary rather than competing approach.

3 Methodology

This section describes the experimental pipeline used to study the effect of input truncation on summarization quality and inference cost. By fixing the summarization backend, decoding parameters, and evaluation metrics, we vary only the retained input text to study context-utilization efficiency.

3.1 Objective Formulation

Our primary objective is to evaluate the trade-off between inference cost (which scales linearly with $\hat{L}$) and the empirical quality of the resulting summary $\hat{S}$. To isolate the effect of context length from other confounding variables, we treat the input selection process as an oracle-bounded optimization problem. By constraining $\hat{L}$ to a strict token budget B such that $\hat{L} \leq B$, we seek to understand the upper-bound limits of summary quality achievable under severe contextual compression.

3.2 Document Chunking

Each document D is segmented into non-overlapping, sentence-aligned chunks $\{c_i\}$ using a deterministic greedy strategy:

- Sentences are extracted using the NLTK Punkt sentence tokenizer (Bird et al., 2009).
- Sentences are greedily accumulated until adding the next sentence would exceed a *dataset-dependent token limit* (512 tokens for CNN/DM; 1,024 tokens for GovReport and ArXiv).
- A new chunk begins only at sentence boundaries, ensuring no semantic content is split across chunks.

This approach ensures that truncation operates over coherent text units, prevents overlap redundancy across chunks during salience estimation, and enhances reproducibility. Token counts per chunk are computed using the dataset-appropriate tokenizer.

3.3 Salience Scoring Strategies

We rank chunks by estimated importance using three heuristic proxies. All scores are computed using reference summaries as an optimistic oracle signal and normalized to $[0,\ 1]$ per dataset.

- **TF-IDF:** Lexical overlap with reference summaries, computed using unigram and bigram features limited to the 6,000 highest-frequency terms.
- **Cosine:** Semantic embedding similarity with reference summaries, computed using `multi-qa-mpnet-base-dot-v1` (Reimers and Gurevych, 2019) with a maximum sequence length of 512 tokens. Reference summary embeddings are encoded once per document and reused across all chunks.
- **Hybrid:** A weighted combination of TF-IDF and Cosine scores:

$$s_{\text{hybrid}}(c_i) = \alpha \cdot s_{\text{cosine}}(c_i) + (1-\alpha) \cdot s_{\text{tfidf}}(c_i) \tag{1}$$

  where $\alpha = 0.7$, weighting semantic similarity more heavily than lexical overlap.

Table 3 reports the wall-clock overhead of each scoring method across all three datasets.

3.4 Truncation Strategies

Budgeted Oracle Truncation. Given a token budget B, chunks are ranked by *salience-efficiency*. The salience score is normalized by chunk token count, $s(c_i)\ /\ |c_i|$ and processed in descending order. Each chunk is greedily selected if it fits within the remaining budget capacity. Chunks that would exceed the remaining budget are skipped rather than halting selection, allowing smaller high salience chunks later in the ranking to fill residual budget space. This formulation maximizes information density within the budget constraint rather than raw salience, analogous to a greedy knapsack approximation. Selected chunks are restored to their original document order before concatenation. The resulting truncated document $\hat{D} \subseteq D$ satisfies $\hat{L} \leq B$.

Baseline Truncation Methods. To contextualize oracle performance, we compare against three naive truncation baselines:

- **First-k:** Greedily retains chunks in document order, skipping any chunk that would exceed the remaining budget, to pack as close to the budget limit as possible. Unlike Lead-n, First-k may produce non-contiguous selections when an oversized chunk is encountered mid-document.
- **Random-k:** Randomly shuffles all document chunks and greedily selects within the token budget, using a fixed seed ($seed = 42$) for reproducibility. Selected chunks are restored to original document order before summarization.
- **Lead-n:** Retains leading sentence-aligned chunks sequentially, halting at the first chunk that would exceed the budget. Unlike First-k, Lead-n never skips chunks, preserving strict document contiguity.

All truncation strategies are constrained to the same token budget B per dataset, enabling direct comparison of selection effectiveness. We note that for CNN/DailyMail, where the chunk token limit equals the token budget (512 tokens), First-k and Lead-n produce near-identical selections. Since no individual chunk can exceed the remaining budget, First-k's skip logic never triggers. Meaningful differences between these two baselines emerge for GovReport and ArXiv, where 1,024-token chunks can exceed the residual budget space.

3.5 Evaluation Metrics & Cost Estimation

Quality. We report ROUGE-1, ROUGE-2, and ROUGE-L (Lin, 2004), focusing on *relative changes* under compression rather than absolute scores. To capture semantic similarity beyond lexical overlap, we additionally report BERTScore F1 (Zhang et al., 2020) for all experimental conditions. Statistical significance of quality differences is assessed via 95% Bootstrap Confidence Intervals (BCIs) over 1,000 resamples computed for both ROUGE-1 and ROUGE-L; Figure 2 visualizes the ROUGE-1 BCIs across all methods and datasets.

Cost Estimation. Inference cost is estimated using model-specific per-token pricing proxies reflecting the relative computational cost of each architecture:

$$C_{\text{inference}} = \left(\frac{\hat{L}}{1000}\right) \times \text{price_per_1K_tokens} \tag{2}$$

where `price_per_1K_tokens` is \$0.0005 for `facebook/bart-large-cnn`, \$0.001 for `allenai/led-base-16384`, and \$0.002 for `allenai/led-large-16384-arxiv`. These serve as conservative proxy estimates; absolute cost values will vary by provider and over time. But the relative cost-quality trade-offs between truncation methods are invariant to the specific pricing used. Note that BART and LED are open-source models that run locally. The pricing values above represent hypothetical cloud deployment cost proxies to enable relative cost comparisons across methods, not actual measured expenditure.

4 Experimental Setup

4.1 Datasets

We evaluate on three established abstractive summarization benchmarks spanning distinct linguistic domains:

- **GovReport** (Huang et al., 2021): Long-form government reports (973 test documents).
- **CNN/DailyMail** (Hermann et al., 2015): News articles with multi-sentence highlights (11,490 test documents).
- **ArXiv** (Cohan et al., 2018): Scientific papers with author-written abstracts (6,440 test documents).

Average document lengths before truncation are 9,208 tokens for GovReport, 856 tokens for CNN/DailyMail, and 7,972 tokens for ArXiv, yielding compression ratios of approximately 44%, 59%, and 51% respectively under the oracle truncation condition. These three domains represent a deliberate spectrum of document structure: journalistic text with front-loaded information, verbose administrative prose with distributed relevance, and technically dense scientific writing. This diversity allows us to characterize how context-utilization efficiency varies across structurally distinct document types.

4.2 Token Budget

For each dataset, we define a fixed input token budget reflecting practical deployment constraints. We use a budget of 512 tokens for CNN/DailyMail, matching the chunk size and consistent with BART's 1,024-token context window and 4,096 tokens for GovReport and ArXiv. We select 4,096 as it represents approximately half the maximum LED context window (16,384 tokens), reflecting a practically constrained deployment scenario while still capturing meaningful compression trade-offs. These budgets are held constant across all salience strategies and experimental conditions, inducing substantially different compression ratios across datasets and allowing us to study how summarization quality varies as a function of retained context under realistic constraints.

$$\text{Compression Ratio} = \frac{\text{Tokens Retained}}{\text{Total Document Tokens}} \tag{3}$$

4.3 Summarization Models

To align with established baselines for each domain, we deploy dataset-specific pre-trained models via the Hugging Face Transformers Library (Wolf et al., 2020). For CNN/DailyMail, we deploy `facebook/bart-large-cnn`, based on the BART architecture (Lewis et al., 2020). For GovReport, we use the Longformer Encoder-Decoder (LED) (Beltagy et al., 2020) variant `allenai/led-base-16384`, and for ArXiv, we use the larger domain-adapted variant `allenai/led-large-16384-arxiv`.

Model selection per dataset is determined by the token budget rather than dynamic per-document routing: CNN/DailyMail truncated inputs average 284-454 tokens and fall entirely within BART-large-cnn's context window, so BART is used

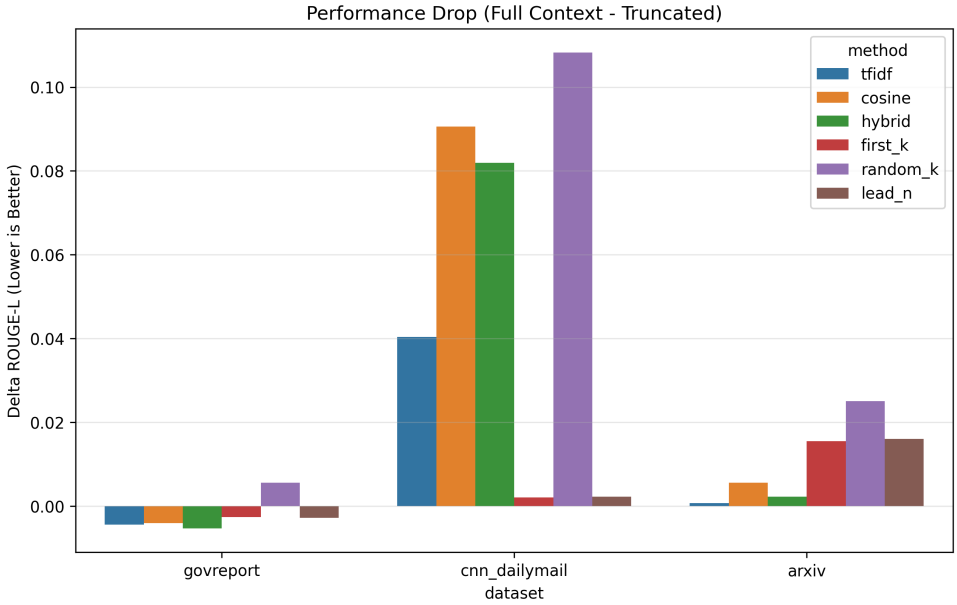


Figure 1: Absolute ROUGE-L performance change when full-context inputs are replaced with truncated inputs. Negative values indicate improvement over the full-context baseline.

throughout. GovReport truncated inputs average 3,500-3,800 tokens and require LED's extended context window. ArXiv always uses the domain-adapted LED-large variant regardless of truncation method.

Decoding hyperparameters are held constant across all truncation conditions *within* each dataset. For CNN/DailyMail, we use beam search with 4 beams, length penalty 2.0, no-repeat n-gram size of 3, and a length constraint of 64 to 256 tokens. For GovReport and ArXiv, we use the same beam search configuration with a length constraint of 150 to 512 tokens, appropriate for long-form abstractive summaries.

4.4 Full Context Baselines

We compute full-context summarization baselines for all three datasets. By utilizing models capable of handling extended context windows (LED-base and LED-large for long-form documents), we process the full source texts without truncation. This establishes the maximum achievable baseline performance and the baseline token expenditure against which all truncation strategies are evaluated.

Experiments were conducted across three compute environments. The GovReport and CNN/DailyMail experiments ran on a single NVIDIA Tesla P100 GPU via Kaggle notebooks. The full-context ArXiv baseline required an NVIDIA A100 (40GB VRAM) via Vast.ai cloud compute due to the extended input lengths. The complete experimental pipeline required approximately 130 hours of total GPU runtime.

4.5 Evaluation Protocol

To ensure a fair and interpretable evaluation, we compare full-context summaries against truncated inputs strictly constrained to dataset-specific token budgets. This comparison is performed systematically across all baseline and oracle salience strategies. Statistical significance of quality differences between full-context and truncated conditions is assessed via 95% Bootstrap Confidence Intervals over 1,000 resamples, computed independently for both ROUGE-1 and ROUGE-L. Figure 1 reports the absolute ROUGE-L performance change across all methods and datasets, where negative values indicate improvement over the full-context baseline.

5 Results and Discussion

We present the empirical results of our truncation analysis, evaluating the impact of context reduction across three dimensions: absolute summarization quality, the economic cost-quality trade-off, and positional content utilization.

5.1 Context Utilization and the "Distraction Penalty"

Figure 1 presents the ROUGE-L performance change of all truncated conditions relative to their

Domain	Strategy	Rationale
Journalistic	Lead-n	Inverted pyramid; zero overhead.
Scientific	TF-IDF	Distributed info; 55% savings.
Verbose/Gov	Any	Distraction penalty dominates.

Table 1: Domain-sensitive truncation routing heuristics.

Dataset	Region	Kept	Dropped	Ratio
CNN/DM	Begin (0–33%)	61.0%	32.0%	0.52×
	Middle (33–67%)	3.1%	31.9%	**10.3**×
	End (67–100%)	35.9%	36.1%	1.0×
GovReport	Begin (0–33%)	36.8%	30.8%	0.84×
	Middle (33–67%)	25.3%	39.1%	**1.55**×
	End (67–100%)	37.9%	30.1%	0.79×
ArXiv	Begin (0–33%)	39.8%	27.9%	0.70×
	Middle (33–67%)	24.9%	40.5%	**1.63**×
	End (67–100%)	35.3%	31.7%	0.90×

Table 2: Positional distribution of retained vs. discarded chunks under Hybrid oracle selection. *Ratio* = Dropped% / Kept% per region. Values above 1.0 indicate disproportionate middle-region discarding

full-context baselines. The results reveal strongly domain-specific patterns in context utilization.

- **News Articles (CNN/DailyMail):** Naive positional truncation results in near-zero degradation (ΔROUGE-1 ≤ 0.003, $p < 0.01$), while salience-based methods substantially penalize quality (TF-IDF ΔROUGE-1 $= -0.049$, Cosine $= -0.111$), empirically validating the *inverted pyramid* structure of journalistic writing.
- **Scientific Literature (ArXiv):** Naive positional truncation heavily penalizes quality (First-k ΔROUGE-1 $= -0.024$, $p = 0.000$), confirming that critical content is distributed throughout scientific papers. TF-IDF oracle achieves a ROUGE-L drop statistically indistinguishable from zero (ΔROUGE-L $= +0.001$, $p = 0.268$) while reducing inference cost by 55%.
- **Verbose Documents (GovReport):** All truncation methods including random subsampling outperform full context (Random-k ΔROUGE-1 $= +0.020$, $p = 0.000$), suggesting the distraction penalty is severe enough that even arbitrary input reduction provides benefit. Figure 2b confirms statistical significance across all methods.

The GovReport result provides strong evidence for the *Lost in the Middle* phenomenon (Liu et al., 2024). The Hybrid Salience method yields a 9.6% relative improvement in ROUGE-1 (0.4876 vs. 0.4449); complete scores are in Table 4 (Appendix A). BERTScore corroborates these findings: oracle methods achieve higher BERTScore for GovReport (Hybrid: 0.844 vs. 0.834) and comparable BERTScore for ArXiv, while CNN/DailyMail salience methods show lower BERTScore consistent with their ROUGE degradation.

Positional Analysis. Table 2 shows the positional distribution of retained versus dropped chunks. For CNN/DailyMail, 61% of retained chunks originate from the first third; the dropped-to-retained ratio in the middle third is 10.3×, quantifying the inverted pyramid structure. For GovReport, 39.1% of dropped chunks originate from the middle third versus 25.3% retained (ratio: 1.55×), consistent with Lost in the Middle. For ArXiv, the middle-third drop ratio of 1.63× explains why naive front-truncation fails.

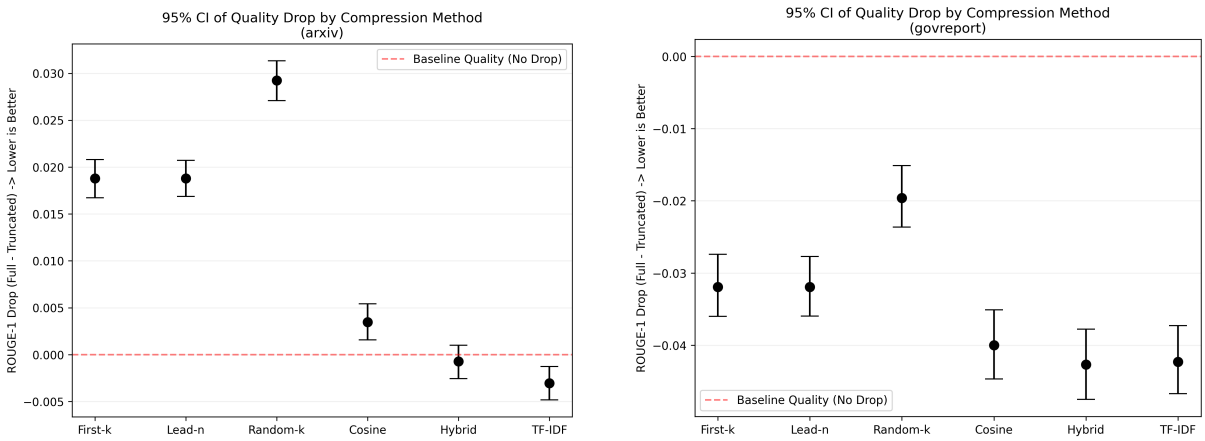
Failure Cases. Oracle truncation is not universally beneficial. For CNN/DailyMail, salience methods harm quality (TF-IDF ΔROUGE-1 $= -0.049$) by retaining high-scoring mid-document content at the expense of front-loaded key facts, suggesting domain structure should inform strategy selection in practice.

5.2 The Economics of Context Bloat

A primary objective of this study is to quantify the financial implications of extended context. Figure 3 maps the cost-quality frontier, comparing the estimated API inference costs of full-context processing (black stars) against all truncation methods.

Selective truncation reduces inference costs by 47%–62% across all datasets and methods without severe degradation in output quality. The minimum inference cost required to achieve 95% of the baseline ROUGE-L score is strictly domain-dependent (Figure 4).

For CNN/DailyMail, salience methods fail to meet the 95% quality threshold due to narrative disruption, while naive *Lead-n* truncation preserves quality at a fraction of the cost. For ArXiv, only oracle salience methods clear the 95% threshold, doing so while reducing input costs by over 50%. This confirms that investing in a salience-aware selection step is economically justified for technical and governmental domains, but unnecessary for



(a) ArXiv

(b) GovReport

Figure 2: 95% Bootstrap Confidence Intervals of ROUGE-1 performance drop (Full Context − Truncated) across 1,000 resamples. Negative values indicate improvement over the full-context baseline. CNN/DailyMail is omitted as positional baseline CIs overlap with zero and salience method degradation is visually apparent in Figure 1.

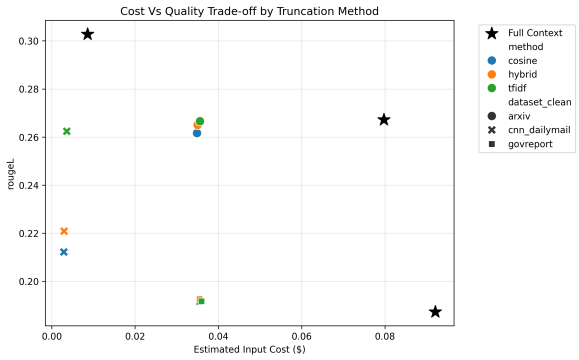


Figure 3: Estimated inference cost versus ROUGE-L quality. Selective truncation shifts the operating point significantly to the left (cheaper) and, in verbose domains, upward (higher quality).

Dataset	Method	Total (s)	Per Doc (ms)
CNN/DM	TF-IDF	27.2	2.4
	Cosine	31.0	2.7
	Hybrid	56.9	5.0
GovReport	TF-IDF	21.6	22.2
	Cosine	19.1	19.6
	Hybrid	40.5	41.6
ArXiv	TF-IDF	82.4	12.8
	Cosine	59.8	9.3
	Hybrid	139.5	21.7

Table 3: Wall-clock overhead of salience scoring. Hybrid requires both TF-IDF and Cosine passes. All computations run on CPU; no GPU is required for the scoring step.

journalistic text where positional structure already concentrates salient content.

Overhead of Salience Scoring. Table 3 reports salience scoring overhead. Per-document cost ranges from 2.4 ms (TF-IDF, CNN/DM) to 41.6 ms (Hybrid, GovReport) on CPU, yielding an estimated cost-benefit ratio exceeding 100:1 for GovReport under our proxy pricing assumptions. This scoring overhead is economically negligible relative to inference savings.

5.3 The Discretization Gap in Budgeted Truncation

Our analysis uncovers a secondary source of financial efficiency inherent to sentence-boundary-aware truncation. Figure 5 illustrates the relationship between predefined token limits and actual realized expenditure.

Because our truncation strategy preserves sentence boundaries by selecting only whole chunks, the selection process cannot partially fill residual budget space. A chunk is either included in full or skipped entirely. Consequently, the actual token count of the truncated input is consistently below the budget ceiling. For positional baselines (First-k, Lead-n) applied to long-form documents (GovReport and ArXiv), this gap is 8%–13%, yielding actual inference costs 8%–13% below the theoretical maximum. This structural gap ensures that chunk-level truncation inherently provides a degree of cost efficiency beyond what the token budget alone implies.

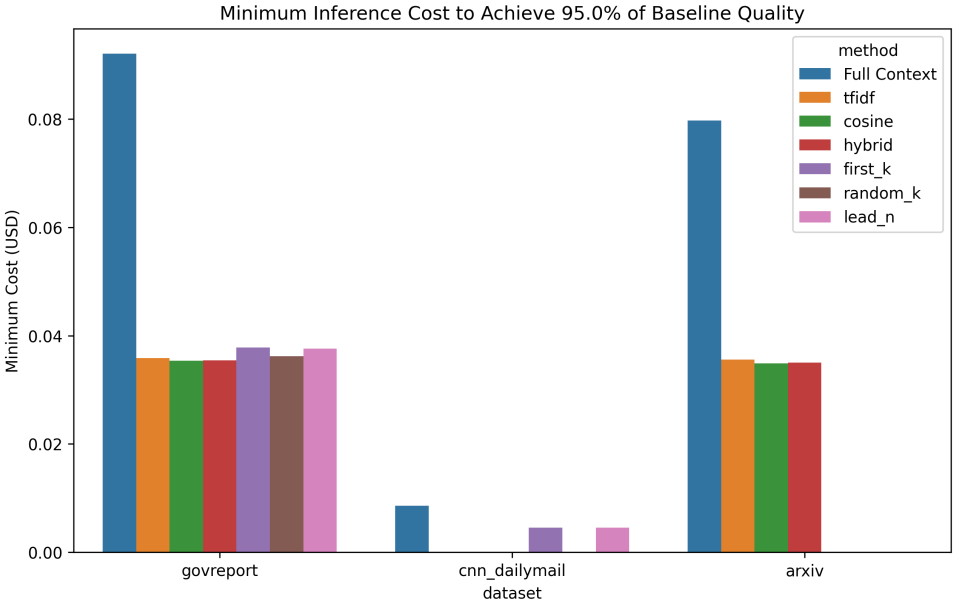


Figure 4: Minimum estimated inference cost required to preserve at least 95% of the full-context baseline ROUGE-L score.

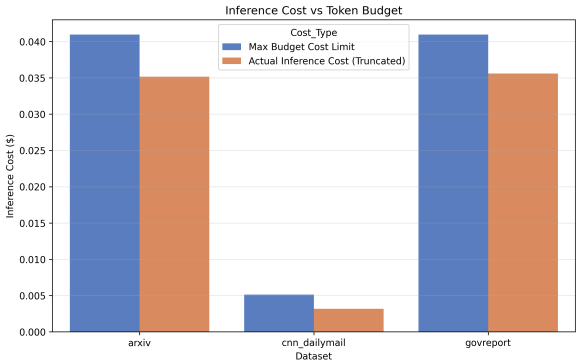


Figure 5: Comparison of the theoretical maximum cost dictated by the token budget versus the actual realized inference cost of truncated inputs.

6 Conclusion

We presented a controlled empirical study quantifying context-utilization efficiency in encoder-decoder summarization models, using oracle-guided salience truncation to answer a fundamental question: at what point does additional input context yield diminishing or negative returns?

Our analysis establishes that context-quality trade-offs are domain-dependent. For journalistic text, naive positional truncation achieves near-zero quality loss at zero overhead. For scientific literature, TF-IDF oracle selection achieves ROUGE-L statistically indistinguishable from full context ($p = 0.268$) while reducing costs by 55%. For verbose government documents, oracle truncation improves ROUGE-1 by up to 9.6% while cutting costs by 61%, as even random subsampling outperforms full-context processing.

Future work will explore replacing oracle salience heuristics with supervised proxy models trained on the oracle labels generated in this study. Such models could enable comparable cost reductions in deployment settings where reference summaries are unavailable, specifically targeting the distraction penalty in verbose domains. Extending this budgeted truncation analysis to modern decoder-only LLMs with sparse attention mechanisms remains an important open direction.

Limitations

While this study provides empirical insights into the cost-quality trade-offs of context utilization, we acknowledge several constraints in our experimental design.

First, our salience-guided truncation relies on reference summaries to compute TF-IDF, Cosine, and Hybrid scores. Since reference summaries are unavailable during real-world inference, the current approach functions as an upper-bound analysis to isolate the effect of context length on summarization quality. Translating these efficiency gains into production would require training a predictive proxy model to estimate chunk salience without access to reference text.

Second, our experiments are conducted exclu-

sively on encoder-decoder architectures (BART and LED). These models are well-established baselines for abstractive summarization; however, modern decoder-only LLMs with advanced attention mechanisms such as Rotary Positional Embedding (RoPE) (Su et al., 2024) and FlashAttention (Dao et al., 2022) may exhibit different context-utilization behaviors and positional biases. Future studies should replicate this budgeted truncation analysis across a wider variety of autoregressive architectures.

Lastly, while we evaluated across three distinct domains (journalistic, scientific, and governmental), our analysis is limited to English-language text. Additionally, while our evaluation employs both ROUGE and BERTScore, neither metric explicitly measures factual consistency, hallucination rates, or logical coherence. These are dimensions critical to long-document summarization quality that automated metrics do not fully capture.

Ethics Statement

Our research utilizes publicly available datasets (CNN/DailyMail, ArXiv, and GovReport) and adheres to their respective licensing agreements. The study does not involve human subjects or the collection of personally identifiable information (PII). A primary motivation of this work is to quantify and reduce the financial and environmental costs associated with LLM inference, contributing to the development of more sustainable and accessible NLP systems. All AI-assisted writing tools used in the preparation of this manuscript are disclosed in the LLM Usage Disclosure section.

LLM Usage Disclosure

In accordance with ACL guidelines, we acknowledge the use of large language models to assist in the writing process of this manuscript. Specifically, **Claude Sonnet 4.6** was utilized purely for editing, paraphrasing, and improving the grammatical flow of the text. All experimental designs, data analysis, scientific claims, and original ideas presented in this paper are strictly the work of the authors.

References

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

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, Inc.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 615–621.

Tri Dao, Daniel Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems*, volume 35, pages 16344–16359.

Huiqiang Gu, Shaohan Huang, Shantanu Tiwary, Chi-Min Hou, and Jinyao Jiang. 2023. LLMLingua: Compressing prompts for accelerated inference of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*.

Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1693–1701.

Luyang Huang, Shuyang Cao, Nikolaus Parulian, Heng Ji, and Lu Wang. 2021. Efficient attentions for long document summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1419–1436.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out, Proceedings of the ACL Workshop*, pages 74–81.

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

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. SummaRuNNer: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI-17)*, pages 3075–3081.

Mathieu Ravaut, Aixin Sun, Nancy Chen, and Shafiq Joty. 2024. On context utilization in summarization with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2764–2781, Bangkok. 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*.

Uri Shaham, Elad Segal, Maor Ivgi, Avia Efrat, Ori Yoran, Adi Haviv, Ankit Gupta, Wenhan Xiong, Mor Geva, Jonathan Berant, and Omer Levy. 2022. SCROLLS: Standardized comparison over long language sequences. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 12007–12021.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yexiang Liu. 2024. RoFormer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Y. Tao, Y. Tang, Y. Wang, M. Zhu, H. Hu, and Y. Wang. 2025. Saliency-driven dynamic token pruning for large language models. In *2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, Hyderabad. IEEE.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

L. Xu, M. A. Karim, S. Dingliwal, and A. Elangovan. 2024. Salient information prompting to steer content in prompt-based abstractive summarization. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 35–49, Miami. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTscore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

A Extended Summarization Metrics

Table 4 provides the complete ROUGE scores for the GovReport dataset. As discussed in Section 5.1, extracting the most salient chunks using oracle methods (TF-IDF, Cosine, and Hybrid) not only enforces the strict 4,096-token budget constraint, but actively improves the summarization quality. By filtering out distracting context, our Hybrid Salience method yields a 9.60% relative improvement in ROUGE-1 compared to processing the full, unedited document.

Truncation Method	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	Δ R-1 (%)
Full Context (Baseline)	0.4449	0.1356	0.1873	0.8337	–
Random-K	0.4645	0.1389	0.1817	0.8380	+4.41%
First-K	0.4847	0.1523	0.1899	0.8436	+8.94%
Lead-N	0.4858	0.1530	0.1901	0.8437	+9.20%
Cosine Salience	0.4849	0.1588	0.1914	0.8439	+8.99%
TF-IDF Salience	0.4872	0.1613	0.1917	0.8433	+9.51%
Hybrid Salience	**0.4876**	**0.1622**	**0.1927**	**0.8443**	**+9.60%**

Table 4: Absolute ROUGE and BERTScore F1 results for the GovReport dataset. All truncation methods outperform the full-context baseline; Hybrid Salience achieves the largest gain at +9.60% relative ROUGE-1 improvement.