



International Journal of Neural Networks and Artificial Intelligence

Journal homepage: <https://trpub.org/journal.php?id=bR2aXTjoAZ>

TRPub

IJNNAI

International Journal of
Neural Networks and
Artificial Intelligence

Vol. 1 Issue 1 2026

Review Article

Open Access

Sparse Attention Mechanisms for Long Document Summarisation on Commodity Hardware: A Review of the Literature

Aurelie Peralta^{1*}, Carmen Mahlangu², Jelena Tanaka³

¹Department of Electrical Electronics, Douro Academy of Electrical Electronics, Guyana

²Department of Electrical Electronics, Niger University Centre for Electrical Electronics, Italy

³Department of Electrical Electronics, Han River Graduate Institute of Electrical Electronics, Kenya

Article Info

Volume 1, Issue 1, 2026

doi: [10.62715/ijnnai.2026.v1.i1.pp1-7](https://doi.org/10.62715/ijnnai.2026.v1.i1.pp1-7)

Received: 02 March 2026

Accepted: 26 April 2026

Published: 15 June 2026

Abstract

Transformer summarisers read their input through self-attention, whose time and memory cost grows with the square of the sequence length. A scientific paper, a legal judgment or a government report is many times longer than the input window of the standard pretrained models, so in practice the document is truncated and the summary is written from its opening pages. Sparse attention was developed to remove that limit, and a sizeable literature now describes windowed, global, random, hashed and low-rank variants together with summarisation models built on them. This narrative review examines that literature from the position of a user who has one consumer graphics card or only a processor. It sets out why full attention does not scale, groups the proposed mechanisms into four families, describes how they have been carried into encoder-decoder summarisers, and considers what changes when memory, and not asymptotic complexity, is the binding constraint. It also discusses how long-document summaries are evaluated and why overlap metrics alone are a weak guide. The review finds that simple local attention with a small number of global tokens remains the most practical choice on modest hardware, that implementation details often matter more than the attention pattern, and that faithfulness on long inputs is still poorly measured.

Keywords: *Sparse attention, Long document summarisation, Efficient transformers, Abstractive summarisation, Memory-constrained inference, Self-attention*

© 2026 Aurelie Peralta et al. This is an open access article under the CC BY license (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

1. Introduction

Abstractive summarisation changed character when sequence-to-sequence networks with attention were shown to produce fluent summaries of news articles (See et al., 2017), and it changed again when pretrained Transformers arrived. Models such as BART and PEGASUS are pretrained on large text collections with denoising or gap-sentence objectives and then fine-tuned on document and summary pairs (Lewis et al., 2020; Zhang et al., 2020). On short inputs they are hard to improve on. Their weakness is the length of what they can read.

* Corresponding author: Aurelie Peralta, Department of Electrical Electronics, Douro Academy of Electrical Electronics, Guyana. E-mail: aurelie.peralta@douro.edu

© 2026 Aurelie Peralta et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The Transformer encoder relates every token to every other token (Vaswani et al., 2017). That design is the source of its strength, and it also fixes a practical ceiling: BERT accepts 512 word-piece tokens and BART 1,024 (Devlin et al., 2019; Lewis et al., 2020). A research article or a court decision runs to many thousands. The usual workaround is to truncate, which was tolerable for news, where the lead paragraph carries most of the content, and is much less so for documents whose conclusions sit at the end.

Sparse attention is the family of techniques that lets each token attend to a chosen subset of positions, so that cost grows roughly linearly with length. Most of the published work reports experiments on data-centre accelerators. Many of the people who would benefit from summarising long documents, in universities, small firms, newsrooms and public offices, have a single consumer graphics card or a laptop. This review reads the literature with that reader in mind and asks which ideas survive the move to ordinary equipment.

2. Scope and Approach

This is a narrative review of the published literature on efficient attention as it bears on summarising long documents. It draws on the original papers that introduced the main mechanisms, on the summarisation models built from them, and on surveys and benchmark studies that compare them. It is not a systematic review and makes no claim to completeness; the selection favours work that is widely cited and whose methods are publicly implemented, since those are the methods a practitioner can use. No new experiments are reported. Where the article speaks of commodity hardware it means a machine with one consumer graphics card, or none, and the memory that such machines normally carry.

3. Why Full Attention Does Not Scale

In a self-attention layer each of the n input tokens produces a query, a key and a value, and the layer computes a score for every query and key pair. The score matrix therefore has $n \times n$ entries per head, and it must be held in memory during training so that gradients can pass back through it. Doubling the input quadruples the matrix. Parameters do not grow with length, so the model itself fits comfortably; it is the activations that exhaust the card.

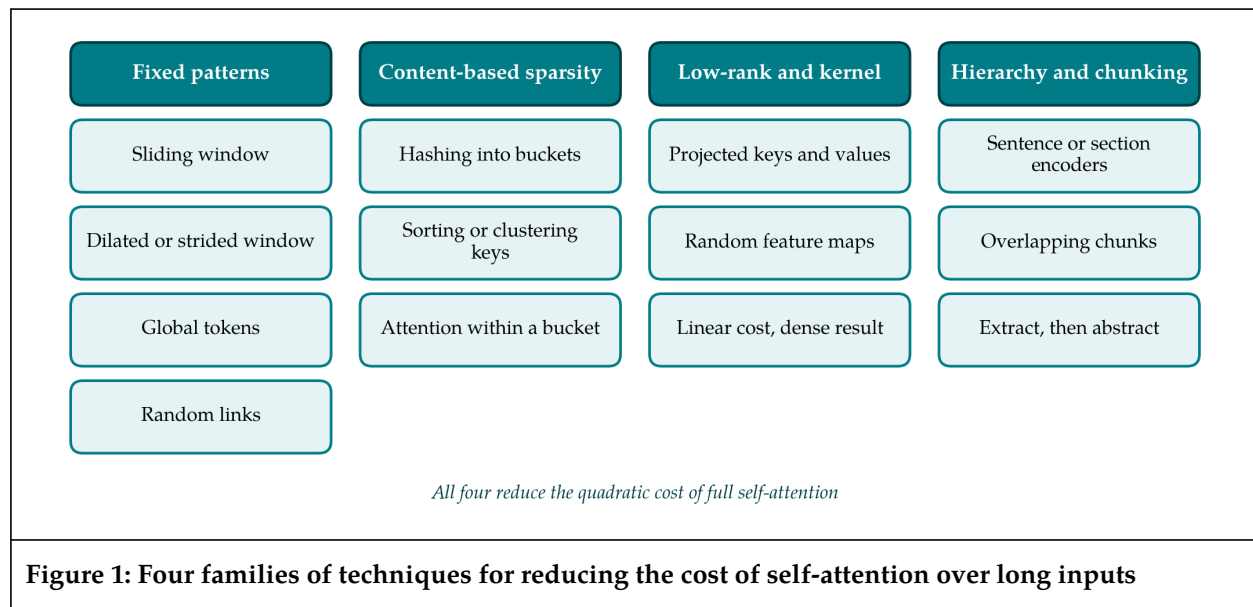
Two consequences follow for a user with limited memory. Training is harder than inference, because training stores activations for every layer while inference can discard them as it goes. And the encoder is the problem, not the decoder: a summary is short, so decoder self-attention is cheap, and the cross-attention from summary to document grows only linearly with document length. Most sparse designs accordingly leave the decoder alone and change only how the encoder reads.

4. Families of Efficient Attention

The survey by Tay et al. (2022) sorts efficient Transformers by the device each uses to avoid the full score matrix. For the purpose of summarisation four families are enough, and Figure 1 lays them side by side. The first two are sparse in the strict sense, since most pairs of positions are never scored. The third approximates the dense matrix instead of thinning it, and the fourth restructures the input so that no single attention call sees the whole document.

4.1. Fixed patterns

The earliest sparse designs fixed in advance which positions may attend to which. Child et al. (2019) factorised attention into strided and local parts for the generation of long sequences. Longformer gives each token a sliding window of neighbours, optionally dilated in upper layers, and adds a handful of global tokens that attend to, and are attended by, every position (Beltagy et al., 2020). BigBird combines the same window and global tokens with a set of random connections, and its authors prove that the resulting sparse graph keeps the universal approximation and Turing completeness properties of the full model (Zaheer et al., 2020). The appeal of these patterns is that they are simple to implement and their memory use is predictable before a run begins.



4.2. Content-based sparsity

A fixed window assumes that the relevant context is nearby. Content-based methods try to find the relevant positions wherever they are. Reformer hashes queries and keys with locality-sensitive hashing so that similar vectors fall into the same bucket, and attention is computed only within buckets; it also uses reversible layers so that activations need not be stored for backpropagation (Kitaev et al., 2020). The idea is attractive for long documents, where a term defined in the introduction may matter fifty paragraphs later. Its cost is engineering complexity: sorting, chunking and multiple hash rounds add overhead that only pays off on very long sequences.

4.3. Low-rank and kernel approximations

A third line of work keeps attention dense in effect while never forming the full matrix. Linformer projects keys and values to a much shorter length on the assumption that the attention matrix is approximately low rank (Wang et al., 2020). Performer rewrites the softmax kernel as an inner product of random feature maps, which allows the products to be reordered so that cost is linear in length (Choromanski et al., 2021). These methods are not sparse, but they compete for the same use and are compared in the same benchmarks. One practical drawback for summarisation is that they cannot easily reuse the weights of a model pretrained with ordinary attention.

4.4. Hierarchy and chunking

The oldest answer to long inputs is to respect the structure of the document. Cohan et al. (2018), who introduced the arXiv and PubMed summarisation datasets, encoded each section separately and let the decoder attend over sections as well as words. Extractive models built on BERT score sentences and can be run over a document in pieces (Liu and Lapata, 2019), and a common pipeline selects salient passages first and passes only those to an abstractive model. Chunking needs no new attention mechanism at all, which makes it the cheapest route on small hardware, though information that spans chunks is lost unless the chunks overlap or a second stage merges them.

5. Sparse Attention in Summarisation Models

Three encoder-decoder models carried sparse attention into summarisation proper. The Longformer Encoder-Decoder replaces the encoder attention of BART with the windowed and global pattern and is initialised from BART's weights, with position embeddings copied to cover the longer input (Beltagy et al., 2020). BigBird was paired with a PEGASUS decoder in the same spirit (Zaheer et al., 2020). LongT5 adds to the T5 architecture a transient global mechanism, in which blocks of tokens are summed into

temporary global tokens at each layer, and adopts the gap-sentence pretraining objective of PEGASUS (Guo et al., 2022). Table 1 summarises how these and the other approaches discussed here behave from the standpoint of a memory-limited user.

Table 1: Approaches to long-input attention and their implications for a memory-limited user			
Approach	How cost is reduced	Reuses pretrained weights	Main caution
Window plus global tokens (Beltagy et al., 2020)	Each token scores only neighbours and a few global positions	Yes, from BART	Window size sets both memory and reach
Window, global and random (Zaheer et al., 2020)	Adds random links to shorten paths across the document	Yes, with PEGASUS	Block implementation adds overhead on short inputs
Transient global (Guo et al., 2022)	Block summaries act as temporary global tokens	Own pretraining	Larger variants exceed small cards in training
Hashed buckets (Kitaev et al., 2020)	Attention only among similar vectors	No	Complex; gains appear only at great length
Low-rank or kernel (Wang et al., 2020; Choromanski et al., 2021)	Approximates the dense matrix at linear cost	Not directly	Approximation error; fewer summarisation checkpoints
Chunk or extract first (Liu and Lapata, 2019)	Document split so each call is short	Yes, any model	Cross-chunk content can be lost

The most direct comparison for summarisation is that of Huang et al. (2021), who introduced a corpus of long government reports and examined several efficient encoder attentions under a fixed memory budget. Their argument is worth stating plainly, because it frames the whole subject: an efficient attention is valuable chiefly because it lets the model read more of the document in the same memory, and reading more of the document is what improves the summary. They also point out that encoder-decoder attention becomes a cost of its own once inputs are very long, and propose a strided variant to contain it.

A second message from the comparative literature is one of modesty. The Long Range Arena benchmark evaluated many efficient Transformers on a common suite of long-sequence tasks and found no single mechanism that led on all of them, with differences in quality often smaller than differences in speed and memory (Tay et al., 2021). For a practitioner this shifts attention from the elegance of the pattern to the availability of a pretrained checkpoint and a stable implementation.

6. What Changes on Commodity Hardware

Asymptotic complexity describes how cost grows, not where it starts. A linear method with a large constant can be slower than exact attention at the lengths a small card can hold in any case. The clearest demonstration is FlashAttention, which computes exact attention but organises the work in tiles so that the score matrix is never written out in full, bringing memory use down to linear in sequence length and reducing traffic between slow and fast memory on the accelerator (Dao et al., 2022). The lesson is that part of the problem sparse attention was invented to solve is an artefact of how attention was implemented.

Other levers are independent of the attention pattern and combine with it. Half-precision arithmetic and gradient checkpointing reduce training memory at some cost in time. Accumulating gradients over several small batches stands in for a large one. For inference, weights can be quantised to integers (Jacob et al., 2018), and a smaller student can be trained to imitate a large summariser by knowledge distillation (Hinton et al., 2015). None of these is specific to long documents, but on a small machine they often decide

whether a long-input model can be fine-tuned at all.

There is also a case for restraint that goes beyond convenience. [Strubell et al. \(2019\)](#) drew attention to the energy and financial cost of training large language models and to the way those costs restrict who can take part in research. Methods that bring long-document summarisation within reach of ordinary equipment widen participation, and reports that state memory use and running time alongside quality scores make it possible for others to judge whether a method is within their means.

7. How Summaries of Long Documents Are Judged

Nearly every paper cited here reports ROUGE, which counts the overlap of word sequences between a system summary and a reference ([Lin, 2004](#)). ROUGE is cheap and reproducible, and it rewards content selection reasonably well. It does not check whether a statement in the summary is supported by the source. [Maynez et al. \(2020\)](#) showed that abstractive systems frequently produce content that cannot be inferred from the input, and that overlap metrics correlate poorly with human judgements of faithfulness.

Long inputs sharpen the difficulty. A human judge must read the whole document to verify a summary, which is slow and expensive, so human evaluation of long-document systems is rare and usually small. The reference summaries themselves are often abstracts written by the authors for a different purpose. A reader comparing attention mechanisms should therefore treat small differences in ROUGE with caution, especially when the systems compared also differ in pretraining, input length or decoding settings.

8. Gaps and Directions

Several questions are left open by the literature as it stands. Comparisons at equal memory, of the kind [Huang et al. \(2021\)](#) carried out, remain uncommon, and comparisons at equal memory on consumer cards are rarer still. It is not well established how much of the benefit of a long-input model comes from the attention pattern and how much from simply seeing more text, nor how sparse models compare with a careful extract-then-abstract pipeline when both are given the same budget. Faithfulness metrics that work on inputs of many thousands of tokens are needed before claims about quality can be settled.

Languages other than English are thinly served: the long-input checkpoints that a practitioner can download were mostly pretrained on English text, and the summarisation corpora of long documents are concentrated in scientific, legal and governmental English. Work on efficient attention for morphologically rich or low-resource languages, and on documents with tables and figures, would extend the reach of the methods reviewed here. Finally, as exact attention becomes cheaper through better implementations, the boundary at which sparsity is worth its approximation should be measured again.

9. Conclusion

The literature on sparse attention offers a user of modest hardware more than one workable route to summarising long documents. Windowed attention with a few global tokens stands out for practical reasons: it is simple, its memory use is predictable, and it can inherit the weights of an existing summariser. More elaborate mechanisms have not shown a consistent advantage that would justify their complexity at the lengths small machines can handle. Implementation choices, precision, and the decision to chunk or extract first weigh as heavily as the attention pattern. The weakest part of the evidence concerns evaluation, and progress there would benefit every method discussed.

Conflict of interest

To be declared by the author(s) before publication.

Funding

To be declared by the author(s) before publication.

References

- Beltagy, I., Peters, M.E. and Cohan, A. (2020). [Longformer: The long-document transformer](#). arXiv preprint arXiv:2004.05150.
- Child, R., Gray, S., Radford, A. and Sutskever, I. (2019). [Generating long sequences with sparse transformers](#). arXiv preprint arXiv:1904.10509.
- Choromanski, K., Likhoshesterov, V., Dohan, D., Song, X., Gane, A., Sarlos, T., Hawkins, P., Davis, J., Mohiuddin, A., Kaiser, L., Belanger, D., Colwell, L. and Weller, A. (2021). [Rethinking attention with Performers](#). *Proceedings of the International Conference on Learning Representations*.
- Cohan, A., Dernoncourt, F., Kim, D.S., Bui, T., Kim, S., Chang, W. and Goharian, N. (2018). [A discourse-aware attention model for abstractive summarization of long documents](#). *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 615-621. <https://doi.org/10.18653/v1/N18-2097>
- Dao, T., Fu, D.Y., Ermon, S., Rudra, A. and Ré, C. (2022). [FlashAttention: Fast and memory-efficient exact attention with IO-awareness](#). *Advances in Neural Information Processing Systems*, 35.
- Devlin, J., Chang, M.-W., Lee, K. and Toutanova, K. (2019). [BERT: Pre-training of deep bidirectional transformers for language understanding](#). *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 4171-4186. <https://doi.org/10.18653/v1/N19-1423>
- Guo, M., Ainslie, J., Uthus, D., Ontanon, S., Ni, J., Sung, Y.-H. and Yang, Y. (2022). [LongT5: Efficient text-to-text transformer for long sequences](#). *Findings of the Association for Computational Linguistics: NAACL 2022*.
- Hinton, G., Vinyals, O. and Dean, J. (2015). [Distilling the knowledge in a neural network](#). arXiv preprint arXiv:1503.02531.
- Huang, L., Cao, S., Parulian, N., Ji, H. and Wang, L. (2021). [Efficient attentions for long document summarization](#). *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.
- Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., Adam, H. and Kalenichenko, D. (2018). [Quantization and training of neural networks for efficient integer-arithmetic-only inference](#). *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2704-2713.
- Kitaev, N., Kaiser, L. and Levine, A. (2020). [Reformer: The efficient transformer](#). *Proceedings of the International Conference on Learning Representations*.
- Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V. and Zettlemoyer, L. (2020). [BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension](#). *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7871-7880. <https://doi.org/10.18653/v1/2020.acl-main.703>
- Lin, C.-Y. (2004). [ROUGE: A package for automatic evaluation of summaries](#). *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, 74-81.
- Liu, Y. and Lapata, M. (2019). [Text summarization with pretrained encoders](#). *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 3730-3740. <https://doi.org/10.18653/v1/D19-1387>
- Maynez, J., Narayan, S., Bohnet, B. and McDonald, R. (2020). [On faithfulness and factuality in abstractive summarization](#). *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 1906-1919. <https://doi.org/10.18653/v1/2020.acl-main.173>
- See, A., Liu, P.J. and Manning, C.D. (2017). [Get to the point: Summarization with pointer-generator networks](#). *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, 1073-1083. <https://doi.org/10.18653/v1/P17-1099>
- Strubell, E., Ganesh, A. and McCallum, A. (2019). [Energy and policy considerations for deep learning in](#)

- NLP. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3645-3650. <https://doi.org/10.18653/v1/P19-1355>
- Tay, Y., Dehghani, M., Abnar, S., Shen, Y., Bahri, D., Pham, P., Rao, J., Yang, L., Ruder, S. and Metzler, D. (2021). *Long Range Arena: A benchmark for efficient transformers. Proceedings of the International Conference on Learning Representations*.
- Tay, Y., Dehghani, M., Bahri, D. and Metzler, D. (2022). *Efficient transformers: A survey. ACM Computing Surveys*, 55(6): Article 109. <https://doi.org/10.1145/3530811>
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L. and Polosukhin, I. (2017). *Attention is all you need. Advances in Neural Information Processing Systems*, 30: 5998-6008.
- Wang, S., Li, B.Z., Khabsa, M., Fang, H. and Ma, H. (2020). *Linformer: Self-attention with linear complexity*. arXiv preprint arXiv:2006.04768.
- Zaheer, M., Guruganesh, G., Dubey, K.A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L. and Ahmed, A. (2020). *Big Bird: Transformers for longer sequences. Advances in Neural Information Processing Systems*, 33.
- Zhang, J., Zhao, Y., Saleh, M. and Liu, P.J. (2020). *PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. Proceedings of the 37th International Conference on Machine Learning*, PMLR 119: 11328-11339.