



 **Hugging Face**


[Models](#) [Datasets](#) [Spaces](#) [Posts](#) [Docs](#) [Enterprise](#) [Pricing](#) [Log In](#) [Sign Up](#)


text-generation-inference 



MAIN 

EN 



 9,555

[TensorRT-LLM](#)

REFERENCE

[All TGI CLI options](#)

[Exported Metrics](#)

[API Reference](#)

CONCEPTUAL GUIDES

[V3 update, caching and chunking](#)

[Streaming](#)

[Quantization](#)

[Tensor Parallelism](#)

[PagedAttention](#)

[Safetensors](#)

[Flash Attention](#)

[Speculation \(Medusa, ngram\)](#)


[How Guidance Works \(via outlines\)](#)


[LoRA \(Low-Rank Adaptation\)](#)


[External Resources](#)

Join the Hugging Face community

and get access to the augmented documentation experience

 Collaborate on models, datasets and Spaces

 Faster examples with accelerated inference

 Switch between documentation themes

Sign Up

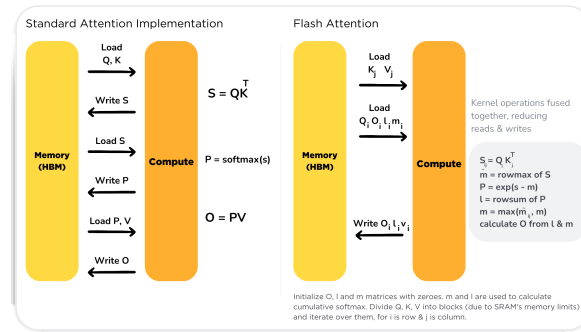
 to get started

Flash Attention

Scaling the transformer architecture is heavily bottlenecked by the self-attention mechanism, which has quadratic time and memory complexity. Recent developments in accelerator hardware mainly focus on enhancing compute capacities and not memory and transferring data between hardware. This results in attention operation having a memory bottleneck. **Flash Attention** is an attention algorithm used to reduce this problem and scale transformer-based models more efficiently, enabling faster training and inference.

Standard attention mechanism uses High Bandwidth Memory (HBM) to store, read and write keys, queries and values. HBM is large in memory, but slow in processing, meanwhile SRAM is smaller in memory, but faster in operations. In the standard attention implementation, the cost of loading and writing keys, queries, and values from HBM is high. It loads keys, queries, and values from HBM to GPU on-chip SRAM, performs a single step of the attention mechanism, writes it back to HBM, and repeats this for every single attention step. Instead, Flash Attention loads keys, queries, and values once, fuses the operations of the attention mechanism, and writes them back.

Flash Attention



It is implemented for supported models. You can check out the complete list of models that support Flash Attention [here](#), for models with flash prefix.

You can learn more about Flash Attention by reading the paper in this [link](#).

[Update on GitHub](#)

← Safetensors

Speculation (Medusa, ngram) →