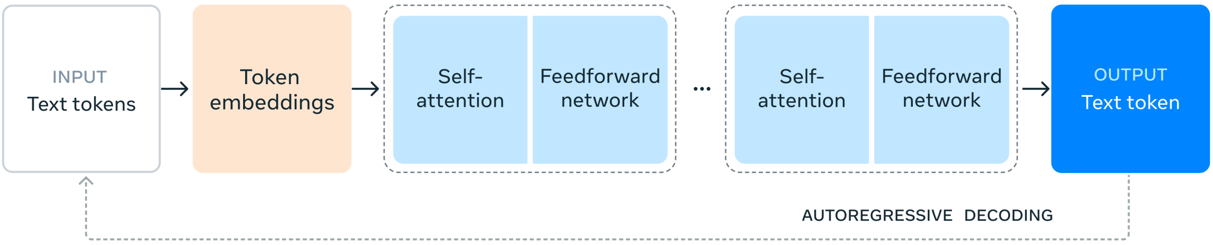
As our largest model yet, training Llama 3.1 405B on over 15 trillion tokens was a major challenge. To enable training runs at this scale and achieve the results we have in a reasonable amount of time, we significantly optimized our full training stack and pushed our model training to over 16 thousand H100 GPUs, making the 405B the first Llama model trained at this scale.



To address this, we made design choices that focus on keeping the model development process scalable and straightforward.

- We opted for a standard decoder-only transformer model architecture with minor adaptations rather than a mixture-of-experts model to maximize training stability.
- We adopted an iterative post-training procedure, where each round uses supervised fine-tuning and direct preference optimization. This enabled us to create the highest quality synthetic data for each round and improve each capability's performance.

Compared to previous versions of Llama, we improved both the quantity and quality of the data we use for pre- and post-training. These improvements include the development of more careful pre-processing and curation pipelines for pre-training data, the development of more rigorous quality assurance, and filtering approaches for post-training data.

As expected per scaling laws for language models, our new flagship model outperforms smaller models trained using the same procedure. We also used the 405B parameter model to improve the post-training quality of our smaller models.

To support large-scale production inference for a model at the scale of the 405B, we quantized our models from 16-bit (BF16) to 8-bit (FP8) numerics, effectively lowering the compute requirements needed and allowing the model to run within a single server node.

Instruction and chat fine-tuning

With Llama 3.1 405B, we strove to improve the helpfulness, quality, and detailed instruction-following capability of the model in response to user instructions while