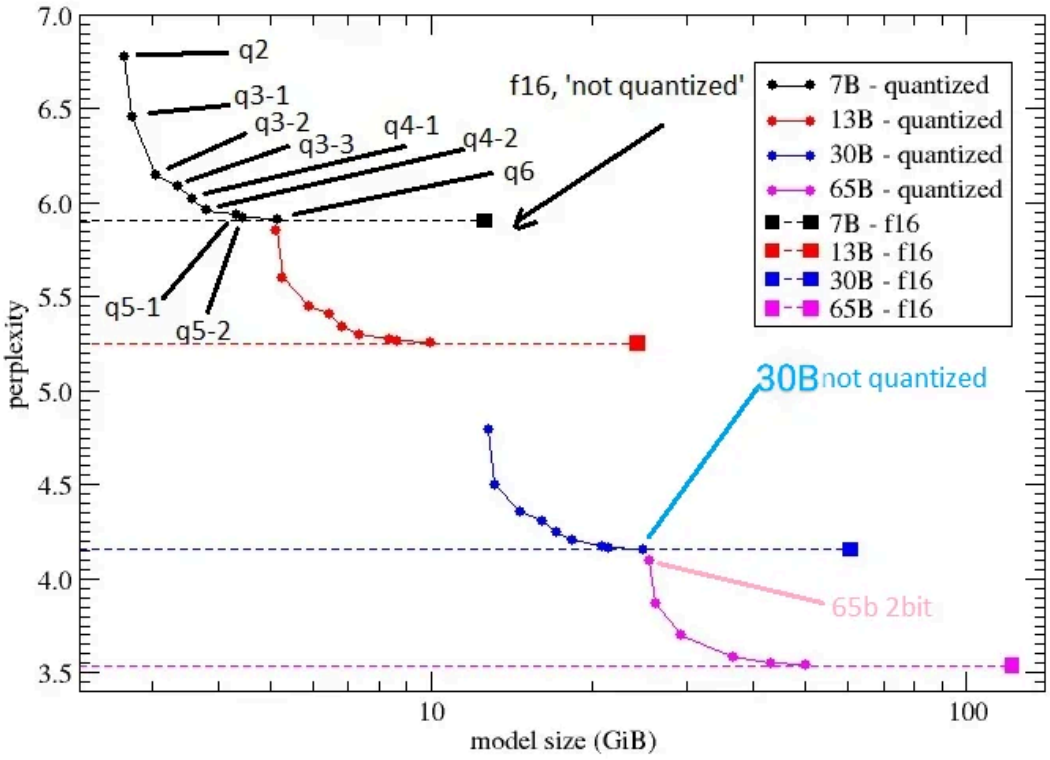
# Experiment Details

This is a supplementary document to the lttlabs.com article titled, “LLM Quantization Part 3: Honey, I Shrank The Numbers!”. It serves to detail the experiment parameters and results that were too long to include in the article.

This whole experiment was inspired by the following community-made "perplexity vs. model size" plot.




There is also a helpful public resource for deciding which GGUF format best fits a given use case in case you want to figure out which works best for your system.

## Methodology

All quantization, measurement, and iMatrix generation was done with llama.cpp (`llama-quantize`, `llama-perplexity`, `llama-imatrix`), producing GGUF V3 files. Every quant for a given model was derived from the same BF16 base GGUF.

**Perplexity** was measured on the WikiText-2-raw test set (`wiki.test.raw`), the full file, using `llama-perplexity` at its default context length and stride. Chunk counts were reported by the tool rather than set manually, and varied by model (for example, 584 chunks for Qwen2.5-32B and 564 for Llama-3.3-70B). The same dataset and settings were used for every model and every quantization level.

**HellaSwag** was measured with llama.cpp's built-in evaluator (`llama-perplexity --hellaswag`) on 1000 randomly selected tasks from `hellaswag_val_full.txt` (`--hellaswag-tasks 1000`), zero-shot, using a fixed seed of 42 so every model and quant saw the identical task sample. Reported scores are length-normalized accuracy (`acc_norm`), and the error bars on the graphs are 95% confidence intervals over those 1000 tasks.

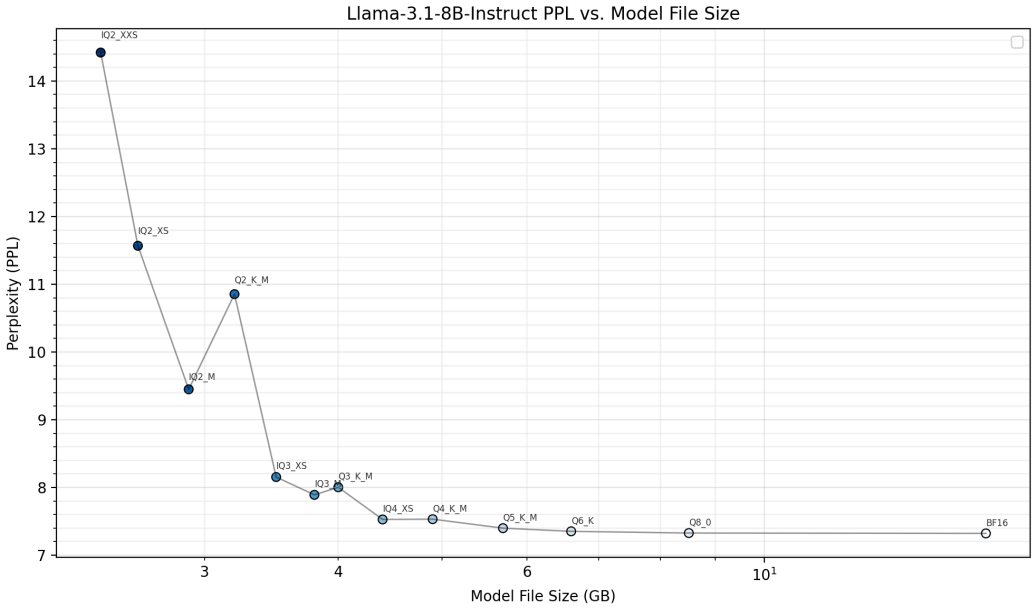
**The iMatrix** for each model was generated with `llama-imatrix` from that model's BF16 copy, calibrated on WikiText-2-raw (`wiki.train.raw`).

# Results

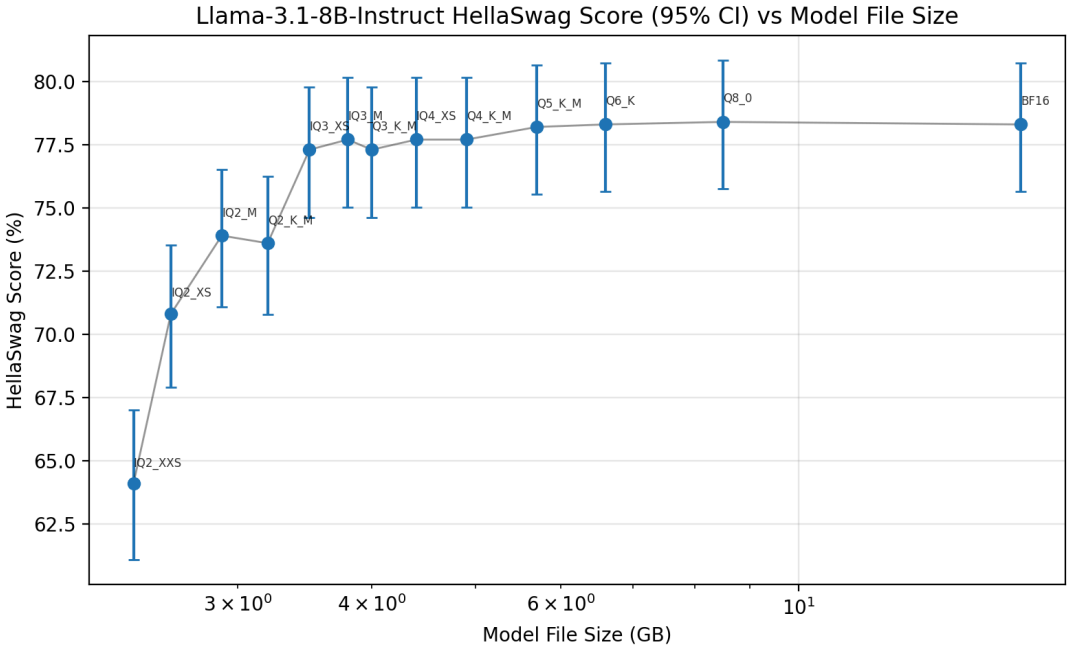
## Llama-3.1-8B-Instruct

(Darker points indicate heavier quantization (smaller model size), while lighter points indicate less quantization or non-quantized models)

Perplexity vs. model size:




HellaSwag score vs. model size:



## Model Quantization with llama-quantize

Original model: Huggingface meta-llama/Llama-3.1-8B-Instruct .safetensors model weights

For consistency and stability, we choose the most-downloaded base models rather than the latest releases.

(llama-quantize cmd: llama-quantize <input_model> <output_model> <quantize_type>)

| Task | Input | Output | Output size | (Input) model size and quant size | Quantize time |
|---|---|---|---|---|---|
| Convert original model to gguf format | Llama_3.1_8b_instruct Folder<br><br>(downloaded directly from Hugging face) | Llama_3.1_8B_Instruct-8.0B-**BF16**.gguf | 16.1GB | | |
| K-Quantize model from BF16 to Q8_0 | Llama_3.1_8B_Instruct-8.0B-BF16.gguf | Llama_3.1_8B_Instruct-**Q8_0**.gguf | 8.5 GB | model size = 15317.02 MiB<br>quant size = 8137.64<br>juliana broochMiB | 46201.43 ms |
| K-Quant BF16 to Q6_K | Llama_3.1_8B_Instruct-8.0B-BF16.gguf | Llama_3.1_8B_Instruct-**Q6_K**.gguf | 6.6 GB | model size = 15317.02 MiB<br>quant size = 6282.97 MiB | 47847.38 ms |
| K-Quant BF16 to Q5_K_M | Llama_3.1_8B_Instruct-8.0B-BF16.gguf | Llama_3.1_8B_Instruct-**Q5_K_M**.gguf | 5.7 GB | model size = 15317.02 MiB<br>quant size = 5459.93 MiB | 48616.60 ms |
| K-Quant BF16 to Q4_K_M | Llama_3.1_8B_Instruct-8.0B-BF16.gguf | Llama_3.1_8B_Instruct-**Q4_K_M**.gguf | 4.9 GB | model size = 15317.02 MiB<br>quant size = 4685.30 MiB | 49762.91 ms |
| K-Quant BF16 to Q3_K_M | Llama_3.1_8B_Instruct-8.0B-BF16.gguf | Llama_3.1_8B_Instruct-**Q3_K_M**.gguf | 4.0 GB | model size = 15317.02 MiB<br>quant size = 3825.27 MiB | 36701.46 ms |
| K-Quant BF16 to Q2_K | Llama_3.1_8B_Instruct-8.0B-BF16.gguf | Llama_3.1_8B_Instruct-**Q2_K**.gguf | 3.2 GB | model size = 15317.02 MiB<br>quant size = 3024.38 MiB | 36935.37 ms |
| Generate imatrix | | | | | |
| I-Quant BF16 to IQ4_XS | Llama_3.1_8B_Instruct-8.0B-BF16.gguf | Llama_3.1_8B_Instruct-**IQ4_XS**.gguf | 4.4 GB | model size = 15317.02 MiB<br>quant size = 4234.15 MiB | 79695.93 ms |

| | | | | | |
|---|---|---|---|---|---|
| I-Quant BF16 to IQ3_M | Llama_3.1_8B_Instruct-8.0B-BF16.gguf | Llama_3.1_8B_Instruct-**IQ3_M**.gguf | 3.8 GB | model size = 15317.02 MiB<br>quant size = 3602.02 MiB | 97428.45 ms |
| I-Quant BF16 to IQ3_XS | Llama_3.1_8B_Instruct-8.0B-BF16.gguf | Llama_3.1_8B_Instruct-**IQ3_XS**.gguf | 3.5 GB | model size = 15317.02 MiB<br>quant size = 3348.27 MiB | 115108.41 ms |
| I-Quant BF16 to IQ2_M | Llama_3.1_8B_Instruct-8.0B-BF16.gguf | Llama_3.1_8B_Instruct-**IQ2_M**.gguf | 2.9 GB | model size = 15317.02 MiB<br>quant size = 2804.23 MiB | 84160.80 ms |
| I-Quant BF16 to IQ2_XS | Llama_3.1_8B_Instruct-8.0B-BF16.gguf | Llama_3.1_8B_Instruct-**IQ2_XS**.gguf | 2.6 GB | model size = 15317.02 MiB<br>quant size = 2477.59 MiB | **228033.10 ms** |
| I-Quant BF16 to IQ2_XXS | Llama_3.1_8B_Instruct-8.0B-BF16.gguf | Llama_3.1_8B_Instruct-**IQ2_XXS**.gguf | 2.4 GB | model size = 15317.02 MiB<br>quant size = 2280.59 MiB | 122067.34 ms |

Perplexity Result

**System_info**
Device 0: NVIDIA RTX A6000, compute capability 8.6, VMM: yes
Build: 7714 (8e649571c) with GNU 11.4.0 for Linux x86_64
llama-perplexity CPU threads: N_threads = 16

**Model_info**
Model_name: Llama_3.1_8B_Instruct
Model_params: 8.03B
Baseline format: GGUF (BF16)
GGUF version: version GGUF V3 (latest)
Load_tensors: offloaded 33/33 layers to GPU (means all layers were offloaded to GPU (ngl = full)
llama_kv_cache:  size =  256.00 MiB,  CUDA0 KV buffer size = 256.00 MiB

**Dataset_info**
Dataset: WikiText-2 test set
Context length: 2048
Evaluation context sequence length: 512
Batch size: 2048
Ubatch size: 512
Number of sequences: 4
Tokenizer: same as model (BPE)
Calculating perplexity over 564 chunks (A chunk is a fixed-length evaluation window used by llama.cpp to compute perplexity)

(llama-perplexity cmd: llama-perplexity -m <model> -f <eval_dataset>:

e.g.: CUDA_VISIBLE_DEVICES=0 llama-perplexity -m /media/labs/0480953A809532E2/Users/Administrator/models/llama_3.1_8b_instruct/Llama_3.1_8B_Instruct-8.0B-BF16.gguf -f wikitext-2-raw/wiki.test.raw)

Prompt eval time is the time it takes to process the initial text input (the prompt) and build the necessary context (KV cache) before the model starts generating new tokens. (prompt forward pass)

| Model | quant_type | file_size | PPL | Load time | Prompt eval time |
|---|---|---|---|---|---|
| Llama_3.1_8B_Instruct-8.0B-BF16.gguf | BF16 | 14.96 GiB (16.00 BPW) | 7.3224 +/- 0.04674 | 4422.21 ms (corresponds to model initialization | 0.19 ms per token, 5149.81 tokens per second |

| | | | | and weight loading overhead) | |
|---|---|---|---|---|---|
| Llama_3.1_8B_Instruct-Q8_0.gguf | Q8_0 | 7.95 GiB (8.50 BPW) | 7.3278 +/- 0.04677 | 4853.90 ms | 0.20 ms per token, 5048.36 tokens per second |
| Llama_3.1_8B_Instruct-Q6_K.gguf | Q6_K | 6.14 GiB (6.56 BPW) | 7.3533 +/- 0.04699 | 3792.99 ms | 0.23 ms per token, 4330.16 tokens per second |
| Llama_3.1_8B_Instruct-Q5_K_M.gguf | Q5_K - Medium | 5.33 GiB (5.70 BPW) | 7.3999 +/- 0.04731 | 3035.54 ms | 0.22 ms per token, 4587.74 tokens per second |
| Llama_3.1_8B_Instruct-Q4_K_M.gguf | Q4_K - Medium | 4.58 GiB (4.89 BPW) | 7.5327 +/- 0.04815 | 1566.28 ms | 0.21 ms per token, 4733.26 tokens per second |
| Llama_3.1_8B_Instruct-Q3_K_M.gguf | Q3_K - Medium | 3.74 GiB (4.00 BPW) | 8.0034 +/- 0.05133 | 1320.88 ms | 0.22 ms per token, 4510.52 tokens per second |
| Llama_3.1_8B_Instruct-Q2_K.gguf | Q2_K - Medium | 2.95 GiB (3.16 BPW) | 10.8538 +/- 0.07080 | 1108.03 ms | 0.27 ms per token, 3729.54 tokens per second |
| Llama_3.1_8B_Instruct-IQ4_XS.gguf | IQ4_XS - 4.25 bpw | 4.13 GiB (4.42 BPW) | 7.5284 +/- 0.04832 | 1431.12 ms | 0.19 ms per token, 5234.77 tokens per second |
| Llama_3.1_8B_Instruct-IQ3_M.gguf | IQ3_S mix - 3.66 bpw | 3.52 GiB (3.76 BPW) | 7.8913 +/- 0.04925 | 1284.87 ms | 0.21 ms per token, 4819.87 tokens per second |
| Llama_3.1_8B_Instruct-IQ3_XS.gguf | IQ3_XS - 3.3 bpw | 3.27 GiB (3.50 BPW) | 8.1542 +/- 0.05141 | 1214.14 ms | 0.21 ms per token, 4800.03 tokens per second |
| Llama_3.1_8B_Instruct-IQ2_M.gguf | IQ2_M - 2.7 bpw | 2.74 GiB (2.93 BPW) | 9.4489 +/- 0.06138 | 1093.81 ms | 0.23 ms per token, 4284.64 tokens per second |
| Llama_3.1_8B_Instruct-IQ2_XS.gguf | IQ2_XS - 2.3125 bpw | 2.42 GiB (2.59 BPW) | 11.5678 +/- 0.07609 | 977.22 ms | 0.24 ms per token, 4145.61 tokens per second |
| Llama_3.1_8B_Instruct-IQ2_XXS.gguf | IQ2_XXS - 2.0625 bpw | 2.23 GiB (2.38 BPW) | 14.4200 +/- 0.09705 | 950.47 ms | 0.21 ms per token, 4874.86 tokens per second |

## HellaSwag Score

**HellaSwag** (an acronym for Harder Endings Longer contexts and Low-shot Activities Situations With Adversarial Generations) measures a model's language understanding by asking it to choose the most reasonable ending of sentences among several options that share the same beginning. More info can be found: https://github.com/ggml-org/llama.cpp/discussions/2321 , https://github.com/EleutherAI/lm-evaluation-harness/tree/main/lm_eval/tasks/hellaswag

This test runs on 1000 random tasks in the hellaswag_val_full.txt datafile.

(HellaSwag test cmd: llama-perplexity --hellaswag -f hellaswag_val_full.txt --hellaswag-tasks 1000 --seed 42 -kvu -m <model.gguf>)

| Model | quant_type | acc_norm | 95% confidence interval |
|---|---|---|---|
| Llama_3.1_8B_Instruct-8.0B-BF16.gguf | BF16 | 78.30000000% | [75.6395%, 80.7439%] |
| Llama_3.1_8B_Instruct-Q8_0.gguf | Q8_0 | 78.40000000% | [75.7433%, 80.8393%] |
| Llama_3.1_8B_Instruct-Q6_K.gguf | Q6_K | 78.30000000% | [75.6395%, 80.7439%] |
| Llama_3.1_8B_Instruct-Q5_K_M.gguf | Q5_K - Medium | 78.20000000% | [75.5357%, 80.6485%] |
| Llama_3.1_8B_Instruct-Q4_K_M.gguf | Q4_K - Medium | 77.70000000% | [75.0168%, 80.1712%] |
| Llama_3.1_8B_Instruct-Q3_K_M.gguf | Q3_K - Medium | 77.30000000% | [74.6021%, 79.7889%] |
| Llama_3.1_8B_Instruct-Q2_K.gguf | Q2_K - Medium | 73.60000000% | [70.7814%, 76.2380%] |
| Llama_3.1_8B_Instruct-IQ4_XS.gguf | IQ4_XS - 4.25 bpw | 77.70000000% | [75.0168%, 80.1712%] |
| Llama_3.1_8B_Instruct-IQ3_M.gguf | IQ3_S mix - 3.66 bpw | 77.70000000% | [75.0168%, 80.1712%] |
| Llama_3.1_8B_Instruct-IQ3_XS.gguf | IQ3_XS - 3.3 bpw | 77.30000000% | [74.6021%, 79.7889%] |
| Llama_3.1_8B_Instruct-IQ2_M.gguf | IQ2_M - 2.7 bpw | 73.90000000% | [71.0902%, 76.5269%] |
| Llama_3.1_8B_Instruct-IQ2_XS.gguf | IQ2_XS - 2.3125 bpw | 70.80000000% | [67.9066%, 73.5342%] |
| Llama_3.1_8B_Instruct-IQ2_XXS.gguf | IQ2_XXS - 2.0625 bpw | 64.10000000% | [61.0780%, 67.0140%] |

## Inference Test

This test uses the same prompt and a fixed seed number across all models to compare instruction-following behavior and output stability at different quantization levels

Cmd used:

```
CUDA_VISIBLE_DEVICES=0  llama-cli \
  -m <model_path>\
  --seed 42 \
  --temp 0 \
  --ctx-size 2048 \
  -n 256 \
  -p "Return a JSON object with exactly the following keys:
- \"summary\"
- \"advantages\"
- \"disadvantages\"

Rules:
- \"summary\" must be exactly 3 sentences.
- \"advantages\" must be an array of exactly 3 strings.
- \"disadvantages\" must be an array of exactly 3 strings.
- Do not include any extra text outside the JSON.

Topic: model quantization in large language models."
```

| Model | Output | Prompt processing | Text generation |
|---|---|---|---|
| Llama_3.1_8B_Instruct-8.0B-**BF16**.gguf | { "summary": "Model quantization is a technique used to reduce the size and computational requirements of large language models. This is achieved by representing model weights and activations with lower precision data types, such as integers or half-precision floats. By doing so, model quantization can significantly improve the inference speed and reduce the memory footprint of large language models.", "advantages": ["Improved inference speed", "Reduced memory footprint", "Increased model deployment efficiency"], "disadvantages": ["Potential loss of accuracy", "Increased complexity in training and deployment", "Limited support for certain deep learning frameworks"] } | 1049.4 t/s | 47.4 t/s |

| | | | |
|---|---|---|---|
| Llama_3.1_8B_Instruct-**Q8_0**.gguf | {<br>"summary": "Model quantization is a technique used to reduce the size and computational requirements of large language models. This is achieved by representing model weights and activations with lower precision data types, such as integers or **half-precision** floats. By doing so, model quantization can significantly improve the inference speed and reduce the memory footprint of large language models.",<br>"advantages": ["Improved inference speed", "Reduced memory footprint", "Increased model deployment efficiency"],<br>"disadvantages": ["Potential loss of accuracy", "Increased complexity in training and deployment", "Limited support for certain deep learning frameworks"]<br>} | 3697.6 t/s | 82.9 t/s |
| Llama_3.1_8B_Instruct-**Q6_K**.gguf | {<br>"summary": "Model quantization is a technique used to reduce the size and computational requirements of large language models. This is achieved by representing model weights and activations using lower precision data types, such as integers or **ternary** values. By doing so, model quantization can significantly improve the inference speed and reduce the memory footprint of large language models.",<br>"advantages": ["Improved inference speed", "Reduced memory footprint", "Increased model deployment efficiency"],<br>"disadvantages": ["Potential loss of accuracy", "Increased complexity in training and deployment", "Limited support for certain deep learning frameworks"]<br>} | 3262.8 t/s | 99.6 t/s |

| | | | |
|---|---|---|---|
| Llama_3.1_8B_Instruct-**Q5_K_M**.gguf | {<br>"summary": "Model quantization is a technique used to reduce the size and computational requirements of large language models. This is achieved by representing model weights and activations with lower precision data types, such as integers or **fixed-point** numbers. By doing so, model quantization can significantly improve the inference speed and reduce the memory footprint of large language models.",<br>"advantages": ["Improved inference speed", "Reduced memory footprint", "Increased model deployment efficiency"],<br>"disadvantages": ["Potential loss of model accuracy", "Increased complexity in model training and optimization", "Limited support for certain deep learning frameworks"]<br>} | 3217.8 t/s | 113.7 t/s |
| Llama_3.1_8B_Instruct-**Q4_K_M**.gguf | {<br>"summary": "Model quantization is a technique used to reduce the computational requirements and memory usage of large language models. It involves representing model weights and activations with lower precision data types, such as integers or **floating-point numbers with fewer bits**. This can significantly improve the inference speed and reduce the memory footprint of the model.",<br>"advantages": ["Reduced memory usage", "Improved inference speed", "Increased model deployment on **low-resource devices**"],<br>"disadvantages": ["Potential loss of model accuracy", "Increased computational complexity for training", "Limited support for certain deep learning frameworks"]<br>} | 3323.1 t/s | 128.0 t/s |
| Llama_3.1_8B_Instruct-**Q3_K_M**.gguf | {<br>"summary": "Model quantization is a technique used to reduce the precision of model weights and activations in large language models, resulting in significant memory and computational cost savings. This technique is particularly useful for deploying models on edge devices or in resource-constrained environments. However, quantization can also lead to a loss in model accuracy.",<br>"advantages": ["Reduced memory requirements", "Improved computational efficiency", "Faster deployment on **edge devices**"],<br>"disadvantages": ["Potential loss in model accuracy", "Increased training complexity", "**Difficulty in selecting optimal quantization parameters**"] | 2981.7 t/s | 118.1 t/s |

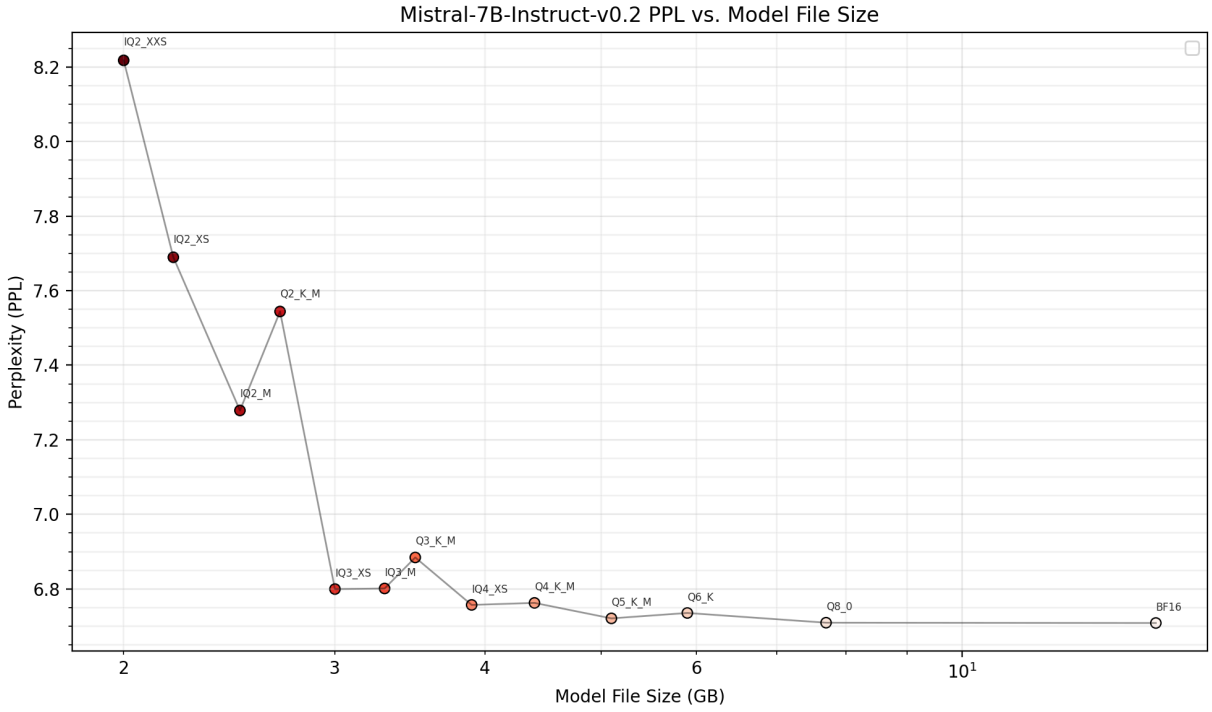
| | | | |
|---|---|---|---|
| | } | | |
| Llama_3.1_8B_Instruct-**Q2_K**.gguf | {<br>"summary": "Model quantization in large language models is a technique to reduce model size while maintaining performance. This is achieved by approximating complex operations with simpler ones. This results in reduced computational cost and memory usage.",<br>"advantages": ["Reduced model size", "Faster inference speed", "**Lower memory usage**"],<br>"disadvantages": ["Potential loss of accuracy", "Increased training time", "**Difficulty in debugging**"]<br>} | 2426.0 t/s | 141.7 t/s |
| Llama_3.1_8B_Instruct-**IQ4_XS**.gguf | {<br>"summary": "Model quantization is a technique used to reduce the size and computational requirements of large language models. This is achieved by approximating the model's weights and activations to lower precision data types, such as integers or lower-precision floating-point numbers. By doing so, model quantization can significantly reduce the memory footprint and computational complexity of large language models.",<br>"advantages": [<br>"Reduced memory requirements, allowing for larger models to be deployed on devices with limited memory",<br>"Faster inference times due to reduced computational complexity",<br>"Improved model deployment on edge devices and mobile platforms"<br>],<br>"disadvantages": [<br>"Potential loss of model accuracy due to weight and activation quantization",<br>"Increased complexity in model training and optimization",<br>"Difficulty in choosing the optimal quantization strategy for a given model"<br>]<br>} | 3622.8 t/s | 141.1 t/s |
| Llama_3.1_8B_Instruct-**IQ3_M**.gguf | {<br>"summary": "Model quantization is a technique used to reduce the size and computational requirements of large language models. It involves approximating the model's weights and activations to reduce precision, which can be done using various methods such as | 3163.6 t/s | 144.5 t/s |

| | | | |
|---|---|---|---|
| | binarization, **ternarization**, and weight pruning. This can lead to significant improvements in inference speed and memory usage.",<br>"advantages": ["Improved inference speed", "Reduced memory usage", "Increased model deployment on edge devices"],<br>"disadvantages": ["Potential loss of accuracy", "Increased training time for quantized models", "Difficulty in choosing optimal quantization method"]<br>} | | |
| Llama_3.1_8B_Instruct-**IQ3_XS**.gguf | {<br>"summary": "Model quantization is a technique used to reduce the computational cost and memory requirements of large language models. It involves approximating the model's weights and activations to reduce precision while maintaining accuracy. This is particularly useful for deployment on low-power devices or in real-time applications.",<br>"advantages": ["Reduced memory requirements", "Improved inference speed", "Lower computational cost"],<br>"disadvantages": ["Potential loss of accuracy", "Increased complexity in implementation", "Dependence on specific hardware or software"]<br>} | 3148.2 t/s | 148.0 t/s |
| Llama_3.1_8B_Instruct-**IQ2_M**.gguf | {"summary": ["Model quantization is a technique used to reduce the size of large language models.", "It is achieved by reducing the precision of the model's weights and activations.", "This process can significantly reduce the memory and computational requirements of the model."], "advantages": ["Reduced memory requirements", "Faster inference times", "Improved model deployment on edge devices"], "disadvantages": ["Loss of accuracy", "Increased computational requirements for training", "Potential loss of model performance"]} | 2882.0 t/s | 157.1 t/s |
| Llama_3.1_8B_Instruct-**IQ2_XS**.gguf | {<br>"summary": [<br>"Model quantization is a technique used to reduce the computational resources required by large language models.",<br>"It involves representing model weights and parameters in a more compact format, often using techniques such as quantization or **Huffman coding**.",<br>"This can lead to significant memory and computational savings, making large language models more deployable in various applications." | 2935.3 t/s | 159.9 t/s |

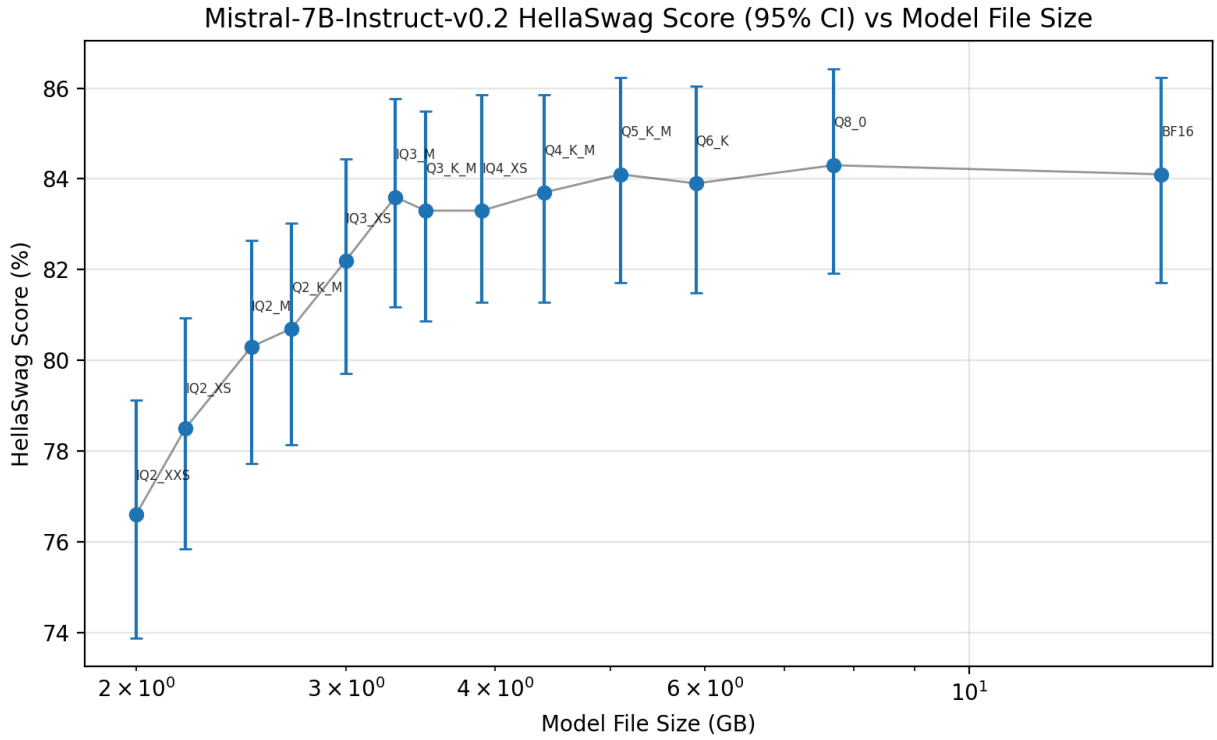
| | | | |
|---|---|---|---|
| | ],<br>"advantages": [<br>"Improved model efficiency",<br>"Reduced memory usage",<br>"Increased deployability"<br>],<br>"disadvantages": [<br>"Potential loss of accuracy due to quantization",<br>"Increased complexity of quantization techniques",<br>"Potential need for additional computational resources for quantization"<br>]<br>} | | |
| Llama_3.1_8B_Instruct-**IQ2_XXS**.gguf | ## JSON Object<br>### Model Quantization in Large Language Models<br>### Summary<br>This is a summary of the topic. The model quantization is a technique used in large language models. It is used to reduce the size of the model.<br><br>### Advantages<br>- The model quantization has several advantages. It reduces the size of the model. It is used in large language models.<br>- It is used in large language models. It reduces the size of the model. It is used in large language models.<br><br>### Disadvantages<br>- The model quantization has several disadvantages. It is used in large language models. It reduces the size of the model.<br>- It is used in large language models. It reduces the size of the model. It is used in large language models.<br><br>### Example<br>```json<br>{<br>"summary": [<br>"The model quantization is a technique used in large language models. It is used to reduce the size of the model.",<br>"The model quantization is a technique used in large language models. It is used to reduce the size of the model.",<br>"The model quantization is a technique used in large language models. It is used to reduce the size of the model."<br>],<br>"advantages": [ | 3184.5 t/s | 160.0 t/s |

## Mistral-7B-Instruct-v0.2

Perplexity vs. model size:




HellaSwag score vs. model size:



Model Quantization with llama-quantize

Original model: Huggingface mistralai/Mistral-7B-Instruct-v0.2

| Task | Input | Output | Output size | (Input) model size and quant size | Quantize time |
|---|---|---|---|---|---|
| Convert original model to gguf format | Mistral_7b_instruct_v0.2 folder | Mistral_7B_Instruct_V0.2-7.2B-**BF16**.gguf | 14.5 GB | | |
| K-Quantize model from BF16 to Q8_0 | Mistral_7B_Instruct_V0.2-7.2B-BF16.gguf | Mistral_7B_Instruct_V0.2-**Q8_0**.gguf | 7.7 GB | model size = 13813.02 MiB<br>quant size = 7338.64 MiB | 40504.70 ms |
| K-Quant BF16 to Q6_K | Mistral_7B_Instruct_V0.2-7.2B-BF16.gguf | Mistral_7B_Instruct_V0.2-**Q6_K**.gguf | 5.9 GB | model size = 13813.02 MiB<br>quant size = 5666.09 MiB | 44933.23 ms |
| K-Quant BF16 to Q5_K_M | Mistral_7B_Instruct_V0.2-7.2B-BF16.gguf | Mistral_7B_Instruct_V0.2-**Q5_K_M**.gguf | 5.1 GB | model size = 13813.02 MiB<br>quant size = 4892.99 MiB | 49496.76 ms |
| K-Quant BF16 to Q4_K_M | Mistral_7B_Instruct_V0.2-7.2B-BF16.gguf | Mistral_7B_Instruct_V0.2-**Q4_K_M**.gguf | 4.4 GB | model size = 13813.02 MiB<br>quant size = 4165.37 MiB | 49529.61 ms |
| K-Quant BF16 to Q3_K_M | Mistral_7B_Instruct_V0.2-7.2B-BF16.gguf | Mistral_7B_Instruct_V0.2-**Q3_K_M**.gguf | 3.5 GB | model size = 13813.02 MiB<br>quant size = 3355.27 MiB | 37553.50 ms |
| K-Quant BF16 to Q2_K | Mistral_7B_Instruct_V0.2-7.2B-BF16.gguf | Mistral_7B_Instruct_V0.2-**Q2_K**.gguf | 2.7 GB | model size = 13813.02 MiB<br>quant size = 2592.57 MiB | 37740.50 ms |
| Generate imatrix | | | | | |
| I-Quant BF16 to IQ4_XS | Mistral_7B_Instruct_V0.2-7.2B-BF16.gguf | Mistral_7B_Instruct_V0.2-**IQ4_XS**.gguf | 3.9 GB | model size = 13813.02 MiB<br>quant size = 3725.96 MiB | 74614.82 ms |

| | | | | | |
|---|---|---|---|---|---|
| I-Quant BF16 to IQ3_M | Mistral_7B_Instruct_V0.2-7.2B-BF16.gguf | Mistral_7B_Instruct_V0.2-**IQ3_M**.gguf | 3.3 GB | model size = 13813.02 MiB<br>quant size = 3132.02 MiB | 89246.94 ms |
| I-Quant BF16 to IQ3_XS | Mistral_7B_Instruct_V0.2-7.2B-BF16.gguf | Mistral_7B_Instruct_V0.2-**IQ3_XS**.gguf | 3.0 GB | model size = 13813.02 MiB<br>quant size = 2878.27 MiB | 106647.94 ms |
| I-Quant BF16 to IQ2_M | Mistral_7B_Instruct_V0.2-7.2B-BF16.gguf | Mistral_7B_Instruct_V0.2-**IQ2_M**.gguf | 2.5 GB | model size = 13813.02 MiB<br>quant size = 2384.16 MiB | 75593.93 ms |
| I-Quant BF16 to IQ2_XS | Mistral_7B_Instruct_V0.2-7.2B-BF16.gguf | Mistral_7B_Instruct_V0.2-**IQ2_XS**.gguf | 2.2 GB | model size = 13813.02 MiB<br>quant size = 2095.72 MiB | **223104.20 ms** |
| I-Quant BF16 to IQ2_XXS | Mistral_7B_Instruct_V0.2-7.2B-BF16.gguf | Mistral_7B_Instruct_V0.2-**IQ2_XXS**.gguf | 2.0 GB | model size = 13813.02 MiB<br>quant size = 1898.72 MiB | 117298.01 ms |

Perplexity Result

**System_info**
Device 0: NVIDIA RTX A6000, compute capability 8.6, VMM: yes
Build: 7714 (8e649571c) with GNU 11.4.0 for Linux x86_64
N_threads = 16

**Model_info**
Model_name: Mistral_7B_Instruct_V0.2
Model_params: 7.24B
Baseline format: GGUF (BF16)
GGUF version: version GGUF V3 (latest)
Load_tensors: offloaded 33/33 layers to GPU
llama_kv_cache: size = 256.00 MiB, CUDA0 KV buffer size = 256.00 MiB

**Dataset_info**
(Same as previous)
calculating perplexity over 642 chunks

| Model | quant_type | file_size | PPL | Load time | Prompt eval time |
|---|---|---|---|---|---|

| | | | | | |
|---|---|---|---|---|---|
| Mistral_7B_Instruct_V0.2-7.2B-BF16.gguf | BF16 | 13.49 GiB (16.00 BPW) | 6.7086 +/- 0.04371 | 3704.67 ms | 0.18 ms per token, 5438.29 tokens per second |
| Mistral_7B_Instruct_V0.2-Q8_0.gguf | Q8_0 | 7.17 GiB (8.50 BPW) | 6.7094 +/- 0.04373 | 5832.66 ms | 0.19 ms per token, 5257.94 tokens per second |
| Mistral_7B_Instruct_V0.2-Q6_K.gguf | Q6_K | 5.53 GiB (6.56 BPW) | 6.7355 +/- 0.04402 | 3179.51 ms | 0.23 ms per token, 4443.28 tokens per second |
| Mistral_7B_Instruct_V0.2-Q5_K_M.gguf | Q5_K - Medium | 4.78 GiB (5.67 BPW) | 6.7210 +/- 0.04374 | 2846.93 ms | 0.21 ms per token, 4770.44 tokens per second |
| Mistral_7B_Instruct_V0.2-Q4_K_M.gguf | Q4_K - Medium | 4.07 GiB (4.83 BPW) | 6.7625 +/- 0.04400 | 2475.27 ms | 0.20 ms per token, 4946.22 tokens per second |
| Mistral_7B_Instruct_V0.2-Q3_K_M.gguf | Q3_K - Medium | 3.28 GiB (3.89 BPW) | 6.8844 +/- 0.04464 | 2109.27 ms | 0.21 ms per token, 4692.43 tokens per second |
| Mistral_7B_Instruct_V0.2-Q2_K.gguf | Q2_K - Medium | 2.53 GiB (3.00 BPW) | 7.5440 +/- 0.04908 | 1606.76 ms | 0.26 ms per token, 3822.10 tokens per second |
| Mistral_7B_Instruct_V0.2-IQ4_XS.gguf | IQ4_XS - 4.25 bpw | 3.64 GiB (4.32 BPW) | 6.7571 +/- 0.04409 | 2277.90 ms | 0.18 ms per token, 5457.80 tokens per second |
| Mistral_7B_Instruct_V0.2-IQ3_M.gguf | IQ3_S mix - 3.66 bpw | 3.06 GiB (3.63 BPW) | 6.8012 +/- 0.04334 | 1981.17 ms | 0.20 ms per token, 4988.86 tokens per second |
| Mistral_7B_Instruct_V0.2-IQ3_XS.gguf | IQ3_XS - 3.3 bpw | 2.81 GiB (3.33 BPW) | 6.7996 +/- 0.04305 | 1827.22 ms | 0.20 ms per token, 5068.11 tokens per second |
| Mistral_7B_Instruct_V0.2-IQ2_M.gguf | IQ2_M - 2.7 bpw | 2.33 GiB (2.76 BPW) | 7.2786 +/- 0.04645 | 1560.76 ms | 0.23 ms per token, 4394.06 tokens per second |
| Mistral_7B_Instruct_V0.2-IQ2_XS.gguf | IQ2_XS - 2.3125 bpw | 2.05 GiB (2.43 BPW) | 7.6894 +/- 0.04896 | 1440.15 ms | 0.23 ms per token, 4298.92 tokens per second |
| Mistral_7B_Instruct_V0.2-IQ2_XXS.gguf | IQ2_XXS - 2.0625 bpw | 1.85 GiB (2.20 BPW) | 8.2175 +/- 0.05243 | 1363.62 ms | 0.19 ms per token, 5134.28 tokens per second |

HellaSwag Score

| Model | quant_type | acc_norm | 95% confidence interval |
|---|---|---|---|
| Mistral_7B_Instruct_V0.2-7.2B-BF16.gguf | BF16 | 84.10000000% | [81.7036%, 86.2354%] |
| Mistral_7B_Instruct_V0.2-Q8_0.gguf | Q8_0 | 84.30000000% | [81.9144%, 86.4231%] |
| Mistral_7B_Instruct_V0.2-Q6_K.gguf | Q6_K | 83.90000000% | [81.4930%, 86.0475%] |
| Mistral_7B_Instruct_V0.2-Q5_K_M.gguf | Q5_K - Medium | 84.10000000% | [81.7036%, 86.2354%] |
| Mistral_7B_Instruct_V0.2-Q4_K_M.gguf | Q4_K - Medium | 83.70000000% | [81.2825%, 85.8596%] |
| Mistral_7B_Instruct_V0.2-Q3_K_M.gguf | Q3_K - Medium | 83.30000000% | [80.8618%, 85.4833%] |
| Mistral_7B_Instruct_V0.2-Q2_K.gguf | Q2_K - Medium | 80.70000000% | [78.1383%, 83.0267%] |
| Mistral_7B_Instruct_V0.2-IQ4_XS.gguf | IQ4_XS - 4.25 bpw | 83.70000000% | [81.2825%, 85.8596%] |
| Mistral_7B_Instruct_V0.2-IQ3_M.gguf | IQ3_S mix - 3.66 bpw | 83.60000000% | [81.1773%, 85.7656%] |
| Mistral_7B_Instruct_V0.2-IQ3_XS.gguf | IQ3_XS - 3.3 bpw | 82.20000000% | [79.7073%, 84.4462%] |
| Mistral_7B_Instruct_V0.2-IQ2_M.gguf | IQ2_M - 2.7 bpw | 80.30000000% | [77.7209%, 82.6472%] |
| Mistral_7B_Instruct_V0.2-IQ2_XS.gguf | IQ2_XS - 2.3125 bpw | 78.50000000% | [75.8472%, 80.9347%] |
| Mistral_7B_Instruct_V0.2-IQ2_XXS.gguf | IQ2_XXS - 2.0625 bpw | 76.60000000% | [73.8772%, 79.1192%] |

Inference Test

| Model | Output | Prompt processing | Text generation |
|---|---|---|---|

| | | | |
|---|---|---|---|
| Mistral_7B_Instruct_V0.2-7.2B-**BF16**.gguf | {<br>"summary": [<br>"Model quantization is a technique used to reduce the computational requirements of large language models by representing model weights with fewer bits.",<br>"This process can significantly reduce inference time and model size, making it possible to deploy models on resource-constrained devices.",<br>"Quantization methods include dynamic and static quantization, with the former allowing for greater accuracy at the cost of increased complexity."<br>],<br>"advantages": [<br>"Reduces computational requirements and inference time",<br>"Decreases model size for deployment on resource-constrained devices",<br>"Improves model efficiency and enables wider adoption"<br>],<br>"disadvantages": [<br>"May result in loss of model accuracy due to quantization",<br>"Requires additional computational resources during training",<br>"May not be suitable for models with very high precision requirements"<br>]<br>} | 892.9 t/s | 49.7 t/s |

| | | | |
|---|---|---|---|
| Mistral_7B_Instruct_V0.2-**Q8_0**.gguf | {<br>"summary": [<br>"Model quantization is a technique used to reduce the computational requirements of large language models by representing model weights with fewer bits.",<br>"This process can significantly reduce inference time and model size, making it **possible to** deploy models on resource-constrained devices.",<br>"Quantization **methods** include dynamic and static quantization, with the former allowing for greater accuracy at the cost of increased complexity."<br>],<br>"advantages": [<br>"Reduces computational requirements and inference time",<br>"Decreases model size for deployment on **resource-constrained devices**",<br>"Improves model efficiency and enables wider adoption"<br>],<br>"disadvantages": [<br>"May result in loss of model accuracy",<br>"Requires additional computational resources during quantization process",<br>"May not be suitable for models with very **high precision requirements**"<br>]<br>} | 3445.0 t/s | 86.7 t/s |

| | | | |
|---|---|---|---|
| Mistral_7B_Instruct_V0.2-**Q6_K**.gguf | {<br>"summary": [<br>"Model quantization is a technique used to reduce the computational requirements of large language models by representing model weights with fewer bits.",<br>"This process can significantly decrease inference time and model size, making it more **feasible** for deployment on resource-constrained devices.",<br>"Quantization **strategies** include dynamic and static quantization, with the former allowing for greater accuracy at the cost of increased complexity."<br>],<br>"advantages": [<br>"Reduces computational requirements and inference time",<br>"Decreases model size for **easier deployment**",<br>"Improves model efficiency and resource utilization"<br>],<br>"disadvantages": [<br>"May result in loss of model accuracy",<br>"Requires additional computational resources for quantization",<br>"May not be suitable for models with **complex architectures**"<br>]<br>} | 3027.4 t/s | 104.2 t/s |

| | | | |
|---|---|---|---|
| Mistral_7B_Instruct_V0.2-**Q5_K_M**.gguf | {<br>"summary": [<br>"Model quantization is a technique used to reduce the computational requirements of large language models by representing model weights with fewer bits.",<br>"This process can significantly decrease inference time and model size, making it more efficient for deployment on resource-constrained devices.",<br>"Quantization strategies include dynamic and static quantization, with the former allowing for **adaptive** quantization during inference."<br>],<br>"advantages": [<br>"Reduces computational requirements and inference time",<br>"Decreases model size for efficient deployment",<br>"Adaptive quantization possible with dynamic strategies"<br>],<br>"disadvantages": [<br>"**Potential** loss of model accuracy due to quantization",<br>"Increased **complexity** in implementing quantization strategies",<br>"Requires **careful tuning** to maintain acceptable performance"<br>]<br>} | 3041.5 t/s | 120.2 t/s |

| | | | |
|---|---|---|---|
| Mistral_7B_Instruct_V0.2-**Q4_K_M**.gguf | {<br>"summary": [<br>"Model quantization is a technique used to reduce the computational requirements of large language models by representing model weights with lower precision. This can lead to significant reductions in inference time and memory usage.",<br>"**Quantization methods** like pruning, binarization, and dynamic quantization are commonly used to achieve these savings without sacrificing model accuracy.",<br>"Implementing model quantization in large language models can enable deployment on resource-constrained devices and make these models more accessible to a wider audience."<br>],<br>"advantages": [<br>"Significant reduction in inference time and memory usage",<br>"**Preservation** of model accuracy with various quantization methods",<br>"Deployment on resource-constrained devices and wider accessibility"<br>],<br>"disadvantages": [<br>"Potential loss of model accuracy with some quantization methods",<br>"Increased complexity in implementing and fine-tuning quantization techniques",<br>"**Trade-off** between model size, accuracy, and computational efficiency"<br>]<br>} | 3108.0 t/s | 135.9 t/s |

| | | | |
|---|---|---|---|
| Mistral_7B_Instruct_V0.2-**Q3_K_M**.gguf | {<br>"summary": "[1] Model quantization is a technique used to reduce the computational requirements and memory footprint of large language models by representing model weights with fewer bits. [2] This process can significantly speed up inference time and reduce model size, making it more accessible for deployment on resource-constrained devices. [3] However, quantization can potentially impact model accuracy and may require retraining to maintain performance.",<br>"advantages": ["Reduces computational requirements and memory footprint", "Speeds up inference time", "Makes large language models more accessible for deployment"],<br>"disadvantages": ["Potential impact on model accuracy", "May require retraining to maintain performance", "Limited support for some model architectures"]<br>} | 2743.7 t/s | 125.8 t/s |
| Mistral_7B_Instruct_V0.2-**Q2_K**.gguf | {<br>"summary": "Model quantization is an essential technique for reducing the computational requirements of large language models. It involves converting model weights from high-precision floating-point format to low-precision formats, such as int8 or float16. This process improves inference speed and reduces memory usage, making large language models more accessible for resource-constrained environments. However, quantization may lead to a loss of model accuracy, which can be mitigated through techniques like k-fold pruning and quantization-aware training.",<br>"advantages": ["Improves inference speed", "Reduces memory usage", "Makes large language models more accessible"],<br>"disadvantages": ["Loss of model accuracy", "Precision drop may affect performance", "Requires careful calibration to maintain acceptable trade-offs"]<br>} | 2175.5 t/s | 150.6 t/s |

| | | | |
|---|---|---|---|
| Mistral_7B_Instruct_V0.2-**IQ4_XS**.gguf | {<br>"summary": [<br>"Model quantization is a technique used to reduce the computational requirements of large language models by representing model weights with fewer bits.",<br>"This process can significantly reduce inference time and model size, making it possible to deploy models on resource-constrained devices.",<br>"Recent advances in quantization algorithms and hardware support have made it a practical solution for deploying large language models in production."<br>],<br>"advantages": [<br>"Reduces computational requirements and inference time",<br>"Decreases model size for deployment on resource-constrained devices",<br>"Makes large language models deployable in production environments"<br>],<br>"disadvantages": [<br>"Quantization can lead to a loss of model accuracy",<br>"Requires careful calibration to maintain model performance",<br>"May not be suitable for models with very high precision requirements"<br>]<br>} | 3426.4 t/s | 151.9 t/s |

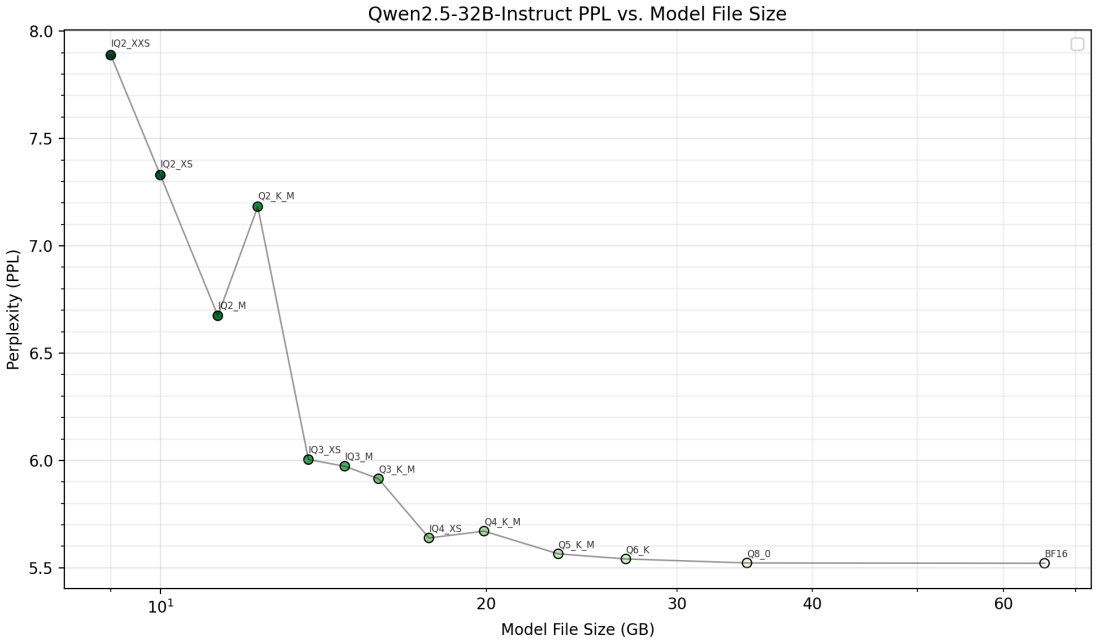
| | | | |
|---|---|---|---|
| Mistral_7B_Instruct_V0.2-**IQ3_M**.gguf | {<br>"summary": ["Model quantization is a technique used to reduce the computational requirements of large language models by representing model weights with fewer bits.",<br>"This process can significantly reduce inference time and model size, making it more efficient to deploy and use in real-world applications.",<br>"However, quantization can also lead to a loss of model accuracy and may require retraining to regain performance."],<br>"advantages": ["Reduces computational requirements and inference time",<br>"Decreases model size for easier deployment",<br>"Improves efficiency for real-world applications"],<br>"disadvantages": ["May result in a loss of model accuracy",<br>"Retraining may be required to regain performance",<br>"Potential trade-off between model size and accuracy"]<br>} | 3082.3 t/s | 154.6 t/s |
| Mistral_7B_Instruct_V0.2-**IQ3_XS**.gguf | {<br>"summary": ["Model quantization is a technique used to reduce the size and complexity of large language models while maintaining their performance.",<br>"It achieves this by converting model weights to lower bit representations, which significantly reduces storage requirements and inference time.",<br>"Quantization is essential for deploying large language models in resource-constrained environments, such as edge devices and mobile applications."],<br>"advantages": ["Reduces storage requirements and inference time",<br>"Preserves model performance",<br>"Enables deployment on resource-constrained devices"],<br>"disadvantages": ["May introduce quantization errors that can affect model accuracy",<br>"Requires careful calibration to maintain model performance",<br>"Can increase training time due to the need for quantization during model development"]<br>} | 2946.2 t/s | 157.7 t/s |

| | | | |
|---|---|---|---|
| Mistral_7B_Instruct_V0.2-**IQ2_M**.gguf | {<br>"summary": ["Model quantization is a technique used to reduce the computational requirements of large language models.", "It involves converting model weights from floating-point to integer values.", "This process can significantly reduce the model's size and improve inference speed."],<br>"advantages": ["Model quantization reduces computational requirements.", "It improves inference speed.", "Quantized models require less memory."],<br>"disadvantages": ["Quantization can lead to loss of model accuracy.", "It may require significant time and resources to implement.", "The process can be complex and error-prone."]<br>} | 2651.8 t/s | 168.6 t/s |
| Mistral_7B_Instruct_V0.2-**IQ2_XS**.gguf | {<br>"summary": [<br>"Model quantization is a technique used in large language models to reduce the computational requirements and memory usage of the model. By quantizing the weights and activations of the model, the computational cost can be significantly reduced, making it more efficient for deployment in various applications.",<br>"For instance, **Google's T5-XL model**, which has over 11 billion parameters, was made deployable by quantizing it to just 1% of its original size. This not only reduces the inference time but also makes it more accessible for edge devices.",<br>"The process of quantization involves converting the floating-point weights and activations to fixed-point representations, which can be stored and processed more efficiently."<br>],<br>"advantages": [<br>"Reduces computational requirements",<br>"Makes large models deployable on edge devices",<br>"Improves inference time"<br>],<br>"disadvantages": [<br>"Quantization may lead to loss of model accuracy",<br>"Requires significant computational effort during the quantization process",<br>"May result in increased latency during inference | 2664.7 t/s | 170.3 t/s |

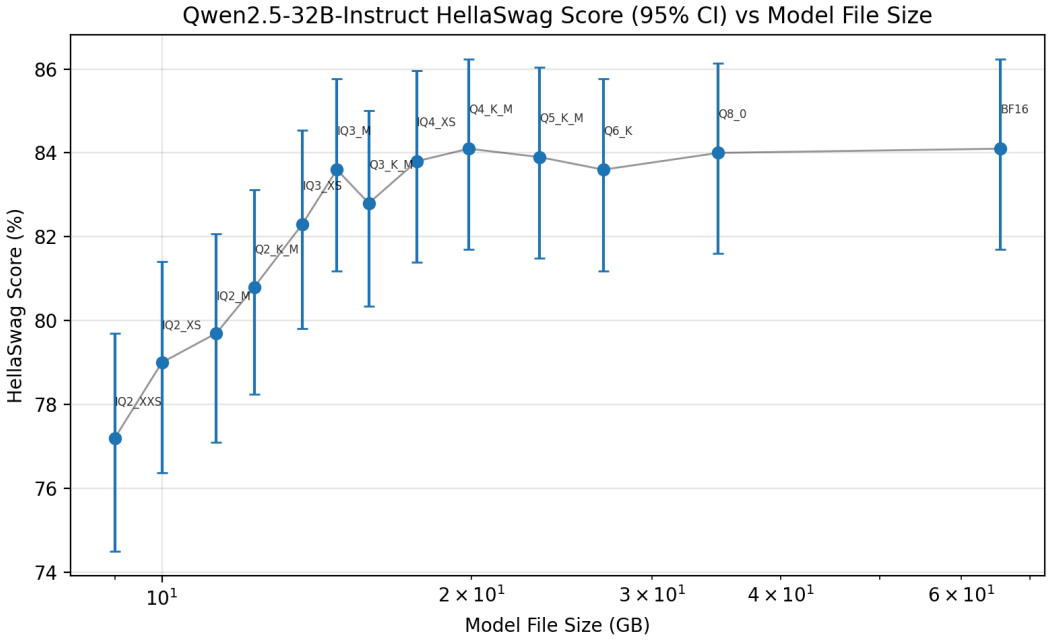
| | | | |
|---|---|---|---|
| Mistral_7B_Instruct_V0.2-**IQ2_XXS**.gguf | {<br>"summary": "Model quantization is a technique used to reduce the size and memory requirements of large language models. It involves compressing model parameters and improving efficiency without sacrificing performance. Summarization: Model quantization enables the deployment of large language models on resource-constrained devices.",<br>"advantages": [ "Reduces model size and memory requirements", "Improves efficiency and performance", "Enables deployment on resource-constrained devices" ],<br>"disadvantages": [ "May result in slight loss of model performance", "Requires careful optimization to ensure quality", "May require additional engineering effort" ]<br>} | 3050.0 t/s | 175.5 t/s |

# Qwen2.5-32B-Instruct

## Perplexity vs. model size:




## HellaSwag score vs. model size:



Model Quantization with llama-quantize

Original model: Huggingface Qwen/Qwen2.5-32B-Instruct

| Task | Input | Output | Output size | (Input) model size and quant size | Quantize time |
|---|---|---|---|---|---|
| Convert original model to gguf format | Qwen2.5-32B-Instruct folder | Qwen2.5-32B-Instruct-**BF16**.gguf | 65.5 GB | | |
| K-Quantize model from BF16 to Q8_0 | Qwen2.5-32B-Instruct-BF16.gguf | Qwen2.5-32B-Instruct-**Q8_0**.gguf | 34.8 GB | model size = 62494.27 MiB<br>quant size = 33202.08 MiB | 209549.34 ms |
| K-Quant BF16 to Q6_K | Qwen2.5-32B-Instruct-BF16.gguf | Qwen2.5-32B-Instruct-**Q6_K**.gguf | 26.9 GB | model size = 62494.27 MiB<br>quant size = 25634.93 MiB | 208442.07 ms |
| K-Quant BF16 to Q5_K_M | Qwen2.5-32B-Instruct-BF16.gguf | Qwen2.5-32B-Instruct-**Q5_K_M**.gguf | 23.3 GB | model size = 62494.27 MiB<br>quant size = 22178.82 MiB | 224249.54 ms |
| K-Quant BF16 to Q4_K_M | Qwen2.5-32B-Instruct-BF16.gguf | Qwen2.5-32B-Instruct-**Q4_K_M**.gguf | 19.9 GB | model size = 62494.27 MiB<br>quant size = 18926.01 MiB | 225187.77 ms |
| K-Quant BF16 to Q3_K_M | Qwen2.5-32B-Instruct-BF16.gguf | Qwen2.5-32B-Instruct-**Q3_K_M**.gguf | 15.9 GB | model size = 62494.27 MiB<br>quant size = 15191.14 MiB | 171442.16 ms |
| K-Quant BF16 to Q2_K | Qwen2.5-32B-Instruct-BF16.gguf | Qwen2.5-32B-Instruct-**Q2_K**.gguf | 12.3 GB | model size = 62494.27 MiB<br>quant size = 11736.98 MiB | 168734.28 ms |
| Generate imatrix | | | | | |
| I-Quant BF16 to IQ4_XS | Qwen2.5-32B-Instruct-BF16.gguf | Qwen2.5-32B-Instruct-I**Q4_XS**.gguf | 17.7 GB | model size = 62494.27 MiB<br>quant size = 16867.80 MiB | 351026.70 ms |
| I-Quant BF16 to IQ3_M | Qwen2.5-32B-Instruct-BF16.gguf | Qwen2.5-32B-Instruct-I**Q3_M**.gguf | 14.8 GB | model size = 62494.27 MiB<br>quant size = 14118.33 MiB | 426441.57 ms |
| I-Quant BF16 to IQ3_XS | Qwen2.5-32B-Instruct-BF16.gguf | Qwen2.5-32B-Instruct-I**Q3_XS**.gguf | 13.7 GB | model size = 62494.27 MiB<br>quant size = 13064.89 MiB | 503197.71 ms |
| I-Quant BF16 to | Qwen2.5-32B | Qwen2.5-32B | 11.3 GB | model size = 62494.27 | 352659.99 |

| | | | | | |
|---|---|---|---|---|---|
| IQ2_M | -Instruct-BF16.gguf | -Instruct-I**Q2_M**.gguf | | MiB<br>quant size = 10736.91 MiB | ms |
| I-Quant BF16 to IQ2_XS | Qwen2.5-32B-Instruct-BF16.gguf | Qwen2.5-32B-Instruct-I**Q2_XS**.gguf | 10.0 GB | model size = 62494.27 MiB<br>quant size = 9490.56 MiB | **1023714.55 ms** |
| I-Quant BF16 to IQ2_XXS | Qwen2.5-32B-Instruct-BF16.gguf | Qwen2.5-32B-Instruct-I**Q2_XXS**.gguf | 9.0 GB | model size = 62494.27 MiB<br>quant size = 8604.31 MiB | 548480.68 ms |

Perplexity Result

(using only one A6000)

**System_info**
Device 0: NVIDIA RTX A6000, compute capability 8.6, VMM: yes
Build: 7714 (8e649571c) with GNU 11.4.0 for Linux x86_64juliana brooch
N_threads = 16

**Model_info**
Model_name: Qwen2.5 32B Instruct
Model_params: 32.76B
Baseline format: GGUF (BF16)
GGUF version: version GGUF V3 (latest)

**Dataset_info**
(Same as previous)
calculating perplexity over 584 chunks

| Model | quant_type | file_size | load_tensor | llama_kv_cache | PPL | Load time | Prompt eval time |
|---|---|---|---|---|---|---|---|
| Qwen2.5-32B-Instruct-BF16.gguf<br><br>**(projected to use 61828 MiB of device memory vs. 48272 MiB of free device memory)** | BF16 | 61.03 GiB (16.00 BPW) | offloaded **49/65** layers to GPU, CUDA0 model buffer size = 46128.21 MiB, CUDA_Host model buffer size = 16366.06 MiB | size = 512.00 MiB, CUDA0 KV buffer size = 384.00 MiB, **CPU KV buffer** size = 128.00 MiB | 5.5208 +/- 0.03476 | 45567.06 ms | 1.86 ms per token, 538.01 tokens per second |
| Qwen2.5-32B-Instruct-Q8_0.gguf | Q8_0 | 32.42 GiB (8.50 BPW) | offloaded 65/65 layers to GPU, CUDA0 model buffer size = 32413.18 MiB, | size = 512.00 MiB, CUDA0 KV buffer size = 512.00 MiB | 5.5223 +/- 0.03479 | 18365.80 ms | 0.80 ms per token, 1249.43 tokens per second |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | CUDA_Host model buffer size = 788.91 MiB | | | | |
| Qwen2.5-32B-Instruct-Q6_K.gguf | Q6_K | 25.03 GiB (6.56 BPW) | offloaded 65/65 layers to GPU, CUDA0 model buffer size = 25025.85 MiB, CPU model buffer size = 609.08 MiB | size = 512.00 MiB, CUDA0 KV buffer size = 512.00 MiB | 5.5409 +/- 0.0349 3 | 21775.55 ms | 0.95 ms per token, 1051.14 tokens per second |
| Qwen2.5-32B-Instruct-Q5_K_M.gguf | Q5_K - Medium | 21.66 GiB (5.68 BPW) | offloaded 65/65 layers to GPU, CUDA0 model buffer size = 21668.35 MiB, CPU model buffer size = 510.47 MiB | size = 512.00 MiB, CUDA0 KV buffer size = 512.00 MiB | 5.5647 +/- 0.0350 6 | 12354.76 ms | 0.88 ms per token, 1133.66 tokens per second |
| Qwen2.5-32B-Instruct-Q4_K_M.gguf | Q4_K - Medium | 18.48 GiB (4.85 BPW) | offloaded 65/65 layers to GPU, CUDA0 model buffer size = 18508.35 MiB, CPU model buffer size = 417.66 MiB | size = 512.00 MiB, CUDA0 KV buffer size = 512.00 MiB | 5.6705 +/- 0.0359 8 | 10032.60 ms | 0.84 ms per token, 1183.93 tokens per second |
| Qwen2.5-32B-Instruct-Q3_K_M.gguf | Q3_K - Medium | 14.84 GiB (3.89 BPW) | offloaded 65/65 layers to GPU, | size = 512.00 MiB, CUDA0 KV | 5.9145 +/- 0.0373 3 | 8087.55 ms | 0.91 ms per token, 1102.06 tokens per |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | CUDA0 model buffer size = 14872.10 MiB, CPU model buffer size = 319.04 MiB | buffer size = 512.00 MiB | | | second |
| Qwen2.5-32B-Instruct-Q2_K.gguf | Q2_K - Medium | 11.46 GiB (3.01 BPW) | offloaded 65/65 layers to GPU, CUDA0 model buffer size = 11493.35 MiB, CPU model buffer size = 243.63 MiB | size = 512.00 MiB, CUDA0 KV buffer size = 512.00 MiB | 7.1824 +/- 0.0472 1 | 6330.19 ms | 1.08 ms per token, 927.84 tokens per second |
| Qwen2.5-32B-Instruct-IQ4_XS.gguf | IQ4_XS - 4.25 bpw | 16.47 GiB (4.32 BPW) | offloaded 65/65 layers to GPU, CUDA0 model buffer size = 16473.35 MiB, CPU model buffer size = 394.45 MiB | size = 512.00 MiB, CUDA0 KV buffer size = 512.00 MiB | 5.6388 +/- 0.0356 9 | 8647.49 ms | 0.77 ms per token, 1306.38 tokens per second |
| Qwen2.5-32B-Instruct-IQ3_M.gguf | IQ3_S mix - 3.66 bpw | 13.79 GiB (3.61 BPW) | offloaded 65/65 layers to GPU, CUDA0 model buffer size = 13799.29 MiB, CPU model buffer size = 319.04 MiB | size = 512.00 MiB, CUDA0 KV buffer size = 512.00 MiB | 5.9734 +/- 0.0376 9 | 7252.44 ms | 0.84 ms per token, 1190.54 tokens per second |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Qwen2.5-32B-Instruct-IQ3_XS.gguf | IQ3_XS - 3.3 bpw | 12.76 GiB (3.35 BPW) | offloaded 65/65 layers to GPU, CUDA0 model buffer size = 12745.85 MiB, CPU model buffer size = 319.04 MiB | size = 512.00 MiB, CUDA0 KV buffer size = 512.00 MiB | 6.0039 +/- 0.0380 2 | 6761.36 ms | 0.84 ms per token, 1193.42 tokens per second |
| Qwen2.5-32B-Instruct-IQ2_M.gguf | IQ2_M - 2.7 bpw | 10.49 GiB (2.75 BPW) | offloaded 65/65 layers to GPU, CUDA0 model buffer size = 10417.86 MiB, CPU model buffer size = 319.04 MiB | size = 512.00 MiB, CUDA0 KV buffer size = 512.00 MiB | 6.6740 +/- 0.0428 0 | 5723.02 ms | 0.96 ms per token, 1046.21 tokens per second |
| Qwen2.5-32B-Instruct-IQ2_XS.gguf | IQ2_XS - 2.3125 bpw | 9.27 GiB (2.43 BPW) | offloaded 65/65 layers to GPU, CUDA0 model buffer size = 9246.93 MiB, CPU model buffer size = 243.63 MiB | size = 512.00 MiB, CUDA0 KV buffer size = 512.00 MiB | 7.3297 +/- 0.0481 5 | 5051.07 ms | 0.98 ms per token, 1018.46 tokens per second |
| Qwen2.5-32B-Instruct-IQ2_XXS.gguf | IQ2_XXS - 2.0625 bpw | 8.40 GiB (2.20 BPW) | offloaded 65/65 layers to GPU, CUDA0 model buffer size = 8360.68 MiB, CPU | size = 512.00 MiB, CUDA0 KV buffer size = 512.00 MiB | 7.8884 +/- 0.0525 0 | 4812.11 ms | 0.82 ms per token, 1220.29 tokens per second |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | model buffer size = 243.63 MiB | | | | |

HellaSwag Score

| Model | quant_type | acc_norm | 95% confidence interval |
|---|---|---|---|
| Qwen2.5-32B-Instruct-BF16.gguf | BF16 | 84.10000000% | [81.7036%, 86.2354%] |
| Qwen2.5-32B-Instruct-Q8_0.gguf | Q8_0 | 84.00000000% | [81.5983%, 86.1415%] |
| Qwen2.5-32B-Instruct-Q6_K.gguf | Q6_K | 83.60000000% | [81.1773%, 85.7656%] |
| Qwen2.5-32B-Instruct-Q5_K_M.gguf | Q5_K - Medium | 83.90000000% | [81.4930%, 86.0475%] |
| Qwen2.5-32B-Instruct-Q4_K_M.gguf | Q4_K - Medium | 84.10000000% | [81.7036%, 86.2354%] |
| Qwen2.5-32B-Instruct-Q3_K_M.gguf | Q3_K - Medium | 82.80000000% | [80.3366%, 85.0124%] |
| Qwen2.5-32B-Instruct-Q2_K.gguf | Q2_K - Medium | 80.80000000% | [78.2428%, 83.1215%] |
| Qwen2.5-32B-Instruct-IQ4_XS.gguf | IQ4_XS - 4.25 bpw | 83.80000000% | [81.3877%, 85.9536%] |
| Qwen2.5-32B-Instruct-IQ3_M.gguf | IQ3_S mix - 3.66 bpw | 83.60000000% | [81.1773%, 85.7656%] |
| Qwen2.5-32B-Instruct-IQ3_XS.gguf | IQ3_XS - 3.3 bpw | 82.30000000% | [79.8121%, 84.5407%] |
| Qwen2.5-32B-Instruct-IQ2_M.gguf | IQ2_M - 2.7 bpw | 79.70000000% | [77.0955%, 82.0772%] |
| Qwen2.5-32B-Instruct-IQ2_XS.gguf | IQ2_XS - 2.3125 bpw | 79.00000000% | [76.3669%, 81.4111%] |
| Qwen2.5-32B-Instruct-IQ2_XXS.gguf | IQ2_XXS - 2.0625 bpw | 77.20000000% | [74.4985%, 79.6933%] |

## Inference Test

(using only one A6000)

| Model | Output | Prompt processing | Text generation |
|---|---|---|---|
| Qwen2.5-32B-Instruct-BF16.gguf | {<br>"summary": "Model quantization in large language models involves reducing the precision of the weights and activations to decrease the model size and improve inference speed without significantly compromising accuracy. This technique is crucial for deploying large models on devices with limited computational resources, such as smartphones or embedded systems. Quantization can be applied to various levels of precision, from 8-bit to even lower, depending on the trade-off between performance and accuracy.",<br>"advantages": [<br>"Reduces memory usage and storage requirements, making models more accessible on a variety of devices.",<br>"Improves inference speed due to reduced computational complexity and more efficient use of hardware.",<br>"Enables deployment of large models on edge devices, expanding the reach and applicability of AI technologies."<br>],<br>"disadvantages": [<br>"May lead to a decrease in model accuracy, especially with lower precision quantization levels.",<br>"Requires careful calibration and tuning to maintain performance, which can be time-consuming and resource-intensive.",<br>"Not all operations and layers in a model may support quantization, limiting its applicability in certain architectures."<br>]<br>} | 117.9 t/s | 2.9 t/s |

| | | | |
|---|---|---|---|
| Qwen2.5-32B-Instruct-Q8_0.gguf | {<br>"summary": "Model quantization in large language models involves reducing the precision of the weights and activations to decrease model size and improve inference speed without significantly compromising accuracy. This technique is crucial for deploying large models on devices with limited computational resources, such as smartphones or embedded systems. Quantization can be applied to various levels of precision, from 8-bit to even lower, depending on the trade-off between performance and accuracy.",<br>"advantages": [<br>"Reduces memory usage and storage requirements, making models more accessible on a variety of devices.",<br>"Improves inference speed due to reduced computational complexity and more efficient use of hardware.",<br>"Enables deployment of large models on edge devices, expanding the reach and applicability of AI technologies."<br>],<br>"disadvantages": [<br>"May lead to a decrease in model accuracy, especially with lower precision quantization levels.",<br>"Requires careful calibration and tuning to maintain performance, which can be time-consuming and resource-intensive.",<br>"Not all operations and layers in a model may support quantization, limiting its applicability in certain architectures."<br>]<br>} | 947.2 t/s | 20.8 t/s |

| | | | |
|---|---|---|---|
| Qwen2.5-32B-Instruct-Q6_K.gguf | {<br>"summary": "Model quantization in large language models involves reducing the precision of the weights **from 32-bit** floating point to lower bit-widths, which decreases model size and computational cost without significantly affecting performance. This technique is crucial for deploying large models on devices with limited computational resources, such as smartphones or embedded systems. Quantization can be applied to both **training and inference** phases, with varying degrees of complexity and impact on model accuracy.",<br>"advantages": ["Reduces memory usage and computational requirements, making large models more accessible on resource-constrained devices.", "Decreases latency and increases throughput, which is beneficial for real-time applications.", "Lowers energy consumption, which is particularly important for mobile and IoT devices."],<br>"disadvantages": ["May lead to a decrease in model accuracy, especially when quantizing to very low bit-widths.", "Requires careful calibration to maintain performance, which can be time-consuming and resource-intensive.", "Not all operations and layers in a model can be effectively quantized, limiting the scope of its application."]<br>} | 806.3 t/s | 25.5 t/s |

| | | | |
|---|---|---|---|
| Qwen2.5-32B-Instruct-Q5_K_M.gguf | {<br>"summary": "Model quantization in large language models involves reducing the precision of the weights from 32-bit floating point to lower bit-widths, which can significantly decrease the model size and computational requirements without substantial loss in performance. This technique is crucial for deploying large models on devices with limited computational resources, such as smartphones or embedded systems. Quantization can also speed up inference times, making real-time applications more feasible.",<br>"advantages": ["Reduces memory usage and storage requirements, making large models more accessible on a variety of devices.", "Decreases computational load, which can lead to faster inference times and lower power consumption.", "Facilitates easier deployment of large models in edge computing environments where resources are constrained."],<br>"disadvantages": ["May lead to a slight decrease in model accuracy, depending on the level of quantization applied.", "Requires careful calibration to ensure that the quantized model maintains acceptable performance levels.", "Not all operations and layers in a model can be effectively quantized, which can complicate the quantization process."]<br>} | 878.2 t/s | 29.6 t/s |

| | | | |
|---|---|---|---|
| Qwen2.5-32B-Instruct-Q4_K_M.gguf | {<br>"summary": "Model quantization in large language models involves reducing the precision of the model's weights to decrease its **memory footprint** and improve inference speed without significantly compromising accuracy. This technique is crucial for deploying large models on devices with limited computational resources, such as smartphones or embedded systems. Quantization can be applied to various levels of precision, from 8-bit to even lower, depending on the trade-off between performance and accuracy that is acceptable for the application.",<br>"advantages": [<br>"Reduces memory usage, making large models more accessible on devices with limited storage.",<br>"Improves inference speed due to smaller model size and lower precision arithmetic operations.",<br>"Enables deployment of large models on edge devices, expanding the reach of AI applications."<br>],<br>"disadvantages": [<br>"May lead to a decrease in model accuracy, especially with lower precision quantization.",<br>"Requires careful calibration to maintain performance, which can be a complex process.",<br>"Not all operations or layers in a model may support quantization, limiting its applicability."<br>]<br>} | 911.1 t/s | 33.8 t/s |

| | | | |
|---|---|---|---|
| Qwen2.5-32B-Instruct-Q3_K_M.gguf | {<br>"summary": "Model quantization in large language models involves reducing the precision of the model's weights to smaller bit-widths, which can significantly decrease the model size and computational requirements without a substantial loss in performance. This technique is crucial for deploying large models on devices with limited computational resources, such as smartphones or embedded systems. However, it requires careful calibration to maintain the model's accuracy and efficiency.",<br>"advantages": [<br>"Reduces the memory footprint and computational requirements of large language models, making them more accessible for deployment on a variety of devices.",<br>"Improves the **energy efficiency** of model inference, which is particularly beneficial for mobile and IoT applications.",<br>"Facilitates faster model loading and inference times, enhancing user experience in real-time applications."<br>],<br>"disadvantages": [<br>"May lead to a slight decrease in model accuracy if not properly optimized, as the reduced precision can affect the model's ability to learn complex patterns.",<br>"The process of quantization can be complex and requires specialized knowledge to ensure that the model's performance is not compromised.",<br>"Not all models benefit equally from quantization; some may require extensive tuning to achieve both efficiency and acceptable performance levels."<br>]<br>} | 814.4 t/s | 30.3 t/s |

| | | | |
|---|---|---|---|
| Qwen2.5-32B-Instruct-Q2_K.gguf | {<br>"summary": "Model quantization in large language models reduces model size and speeds up inference by using fewer bits for model weights. This technique is crucial for deploying large models on resource-constrained devices or environments. Despite its benefits, quantization can sometimes lead to a slight decrease in model performance or accuracy.",<br>"advantages": ["Reduces memory usage, making large models more accessible on various devices.", "Speeds up inference time, which is beneficial for real-time applications.", "Decreases the **carbon footprint** of model deployment by reducing computational requirements."],<br>"disadvantages": ["May lead to a decrease in model accuracy or performance after quantization.", "Requires careful tuning to balance between model size reduction and performance retention.", "Not all models benefit equally from quantization, and some may experience more **significant performance drops.**"]<br>} | 701.0 t/s | 38.6 t/s |
| Qwen2.5-32B-Instruct-IQ4_XS.gguf | {<br>"summary": "Model quantization in large language models refers to the process of reducing the precision of the weights from 32-bit floating point to lower bit depths, which significantly decreases the model size and computational requirements, making deployment on resource-constrained devices feasible.",<br>"advantages": ["Reduces memory usage and storage requirements", "Improves inference speed and energy efficiency", "Facilitates deployment on devices with limited computational resources"],<br>"disadvantages": ["May lead to a decrease in model accuracy", "Quantization process can be complex and requires careful tuning", "Not all model architectures benefit equally from quantization"]<br>} | 1010.5 t/s | 38.3 t/s |

| | | | |
|---|---|---|---|
| Qwen2.5-32B-Instruct-IQ3_M.gguf | {<br>"summary": ["Model quantization in large language models reduces the model's size and computational requirements by using **lower precision arithmetic**, which can significantly speed up inference and reduce memory usage.", "This technique allows for more efficient deployment of large models on devices with limited resources, such as mobile phones or embedded systems.", "However, it requires careful calibration to ensure that the reduction in precision does not significantly degrade the model's performance."],<br>"advantages": ["Reduces the memory footprint and accelerates inference speed, making it easier to deploy on resource-constrained devices.", "Enables cost-effective scaling of large models, as less computational power is needed for operations.", "Improves energy efficiency, which is crucial for applications running on battery-powered devices."],<br>"disadvantages": ["May lead to a decrease in model accuracy if not properly calibrated, as lower precision can introduce rounding errors.", "The process of quantization can be complex and requires expertise to balance between model size and performance.", "Not all models benefit equally from quantization; some may show more significant performance drops than others."]<br>} | 880.7 t/s | 38.7 t/s |

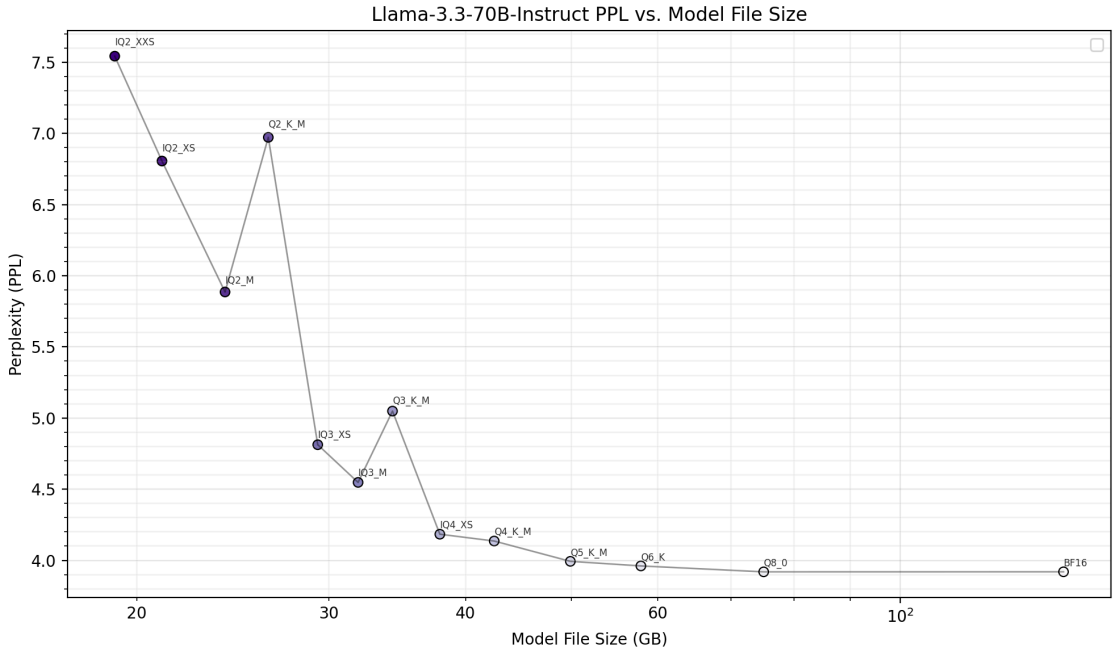
| | | | |
|---|---|---|---|
| Qwen2.5-32B-Instruct-IQ3_XS.gguf | {<br>"summary": "Model quantization in large language models involves reducing the precision of the weights and **activations** to decrease the model size and improve inference speed. This technique allows for more efficient deployment on devices with limited computational resources, such as mobile phones or embedded systems. However, it requires careful tuning to maintain the model's performance after quantization.",<br>"advantages": [<br>"Reduces memory usage and storage requirements, making models more accessible on a variety of devices.",<br>"Improves inference speed, which is crucial for real-time applications and interactive services.",<br>"Enables deployment on edge devices with limited computational power, expanding the reach of AI applications."<br>],<br>"disadvantages": [<br>"May lead to a decrease in model accuracy if not properly optimized.",<br>"Requires additional effort to fine-tune the quantized model to restore performance.",<br>"Not all models benefit equally from quantization, and some may suffer more significant performance drops."<br>]<br>} | 865.7 t/s | 39.8 t/s |

| | | | |
|---|---|---|---|
| Qwen2.5-32B-Instruct-IQ2_M.gguf | {<br>"summary": "Model quantization in large language models involves reducing the precision of the model's weights to decrease computational costs and memory usage without significantly affecting performance. This technique allows for more efficient deployment of models on devices with limited resources, such as smartphones or embedded systems. Quantization can be applied to various types of neural network models, including transformers, which are commonly used in large language models.",<br>"advantages": [<br>"Reduces memory usage, making it easier to deploy models on devices with limited storage.",<br>"Decreases computational costs, allowing for faster inference times.",<br>"Improves energy efficiency, which is particularly beneficial for mobile and embedded devices."<br>],<br>"disadvantages": [<br>"May lead to a slight decrease in model accuracy due to the loss of precision in weight representation.",<br>"The quantization process can be complex and may require careful tuning to maintain performance.",<br>"Not all hardware supports lower precision operations, which can limit the effectiveness of quantization."<br>]<br>} | 785.2 t/s | 42.7 t/s |

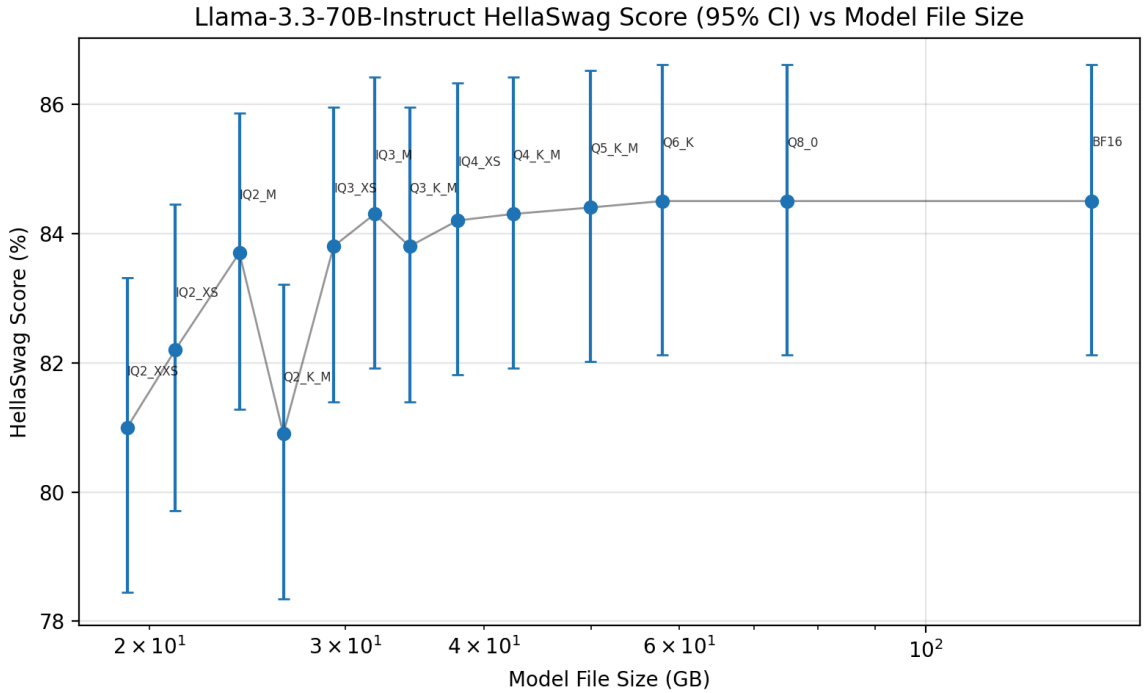
| | | | |
|---|---|---|---|
| Qwen2.5-32B-Instruct-IQ2_XS.gguf | {<br>"summary": "Model quantization is a technique that reduces the precision of the numerical values in a large language model, thereby decreasing its size and improving inference speed. By using fewer bits to represent the weights, quantization enables the model to run more efficiently on hardware with limited computational resources. However, this process requires careful consideration to ensure that the model's performance is not significantly compromised.",<br>"advantages": ["Reduced memory usage, making models more accessible on various devices.", "Faster inference times due to smaller model size and lower precision computations.", "Lower power consumption, which is beneficial for deployment on energy-sensitive devices."],<br>"disadvantages": ["Potential degradation in model accuracy due to loss of precision.", "Requires additional tuning to maintain performance after quantization.", "Not all models or tasks benefit equally from quantization, requiring careful selection."]<br>} | 774.2 t/s | 43.1 t/s |
| Qwen2.5-32B-Instruct-IQ2_XXS.gguf | {<br>"summary": "Model quantization in large language models reduces the model size by using lower precision arithmetic, which can significantly speed up inference and reduce memory usage. However, this technique can introduce accuracy trade-offs, as the reduced precision may lead to loss of information. Despite these challenges, quantization remains a crucial technique for deploying large models on resource-constrained devices.",<br>"advantages": [<br>"Reduces model size, making it easier to deploy on devices with limited resources.",<br>"Improves inference speed by using lower precision arithmetic.",<br>"Decreases memory usage, allowing for more efficient use of computational resources."<br>],<br>"disadvantages": [<br>"May lead to a loss of accuracy due to reduced precision.",<br>"Quantization process can be complex and may require fine-tuning to achieve optimal results.",<br>"Potential trade-offs between model performance and quantization benefits."<br>]<br>} | 920.8 t/s | 44.8 t/s |

# Llama-3.3-70B-Instruct

## Perplexity vs. model size:




## HellaSwag score vs. model size:



Model Quantization with llama-quantize

Original model: Huggingface meta-llama/Llama-3.3-70B-Instruct

| Task | Output | Output size | (Input) model size and quant size | Quantize time |
|---|---|---|---|---|
| Convert original model to gguf format | llama-3.3-70B-instruct-**BF16**.gguf | 141.1 GB | | |
| K-Quantize model from BF16 to Q8_0 | llama-3.3-70B-instruct-**Q8_0**.gguf | 75.0 GB | model size = 134573.03 MiB<br>quant size = 71494.28 MiB | 142465.58 ms |
| K-Quant BF16 to Q6_K | llama-3.3-70B-instruct-**Q6_K**.gguf | 57.9 GB | model size = 134573.03 MiB<br>quant size = 55198.94 MiB | 185686.14 ms |
| K-Quant BF16 to Q5_K_M | llama-3.3-70B-instruct-**Q5_K_M**.gguf | 49.9 GB | model size = 134573.03 MiB<br>quant size = 55198.94 MiB | 266676.71 ms |
| K-Quant BF16 to Q4_K_M | llama-3.3-70B-instruct-**Q4_K_M**.gguf | 42.5 GB | model size = 134573.03 MiB<br>quant size = 40543.11 MiB | 288066.29 ms |
| K-Quant BF16 to Q3_K_M | llama-3.3-70B-instruct-**Q3_K_M**.gguf | 34.3 GB | model size = 134573.03 MiB<br>quant size = 32672.53 MiB | 199973.38 ms |
| K-Quant BF16 to Q2_K | llama-3.3-70B-instruct-**Q2_K**.gguf | 26.4 GB | model size = 134573.03 MiB<br>quant size = 25145.77 MiB | 218083.42 ms |
| Generate imatrix<br><br>(llama-imatrix -m /media/labs/ModelDrive/llama-3.3-70b-instruct/llama-3.3-70B-instruct-Q8_0.gguf -ngl auto --no-ppl -f ./wikitext-2-raw/wiki.train.raw -o llama-3.3-70b-instruct_imatrix.gguf) | | | | ~4.7 hours |
| I-Quant BF16 to IQ4_XS | llama-3.3-70B-instruct-**IQ4_** | 37.9 GB | model size = 134573.03 MiB | 575664.26 ms |

| | | | | |
|---|---|---|---|---|
| | **XS**.gguf | | quant size = 36139.30 MiB | |
| I-Quant BF16 to IQ3_M | llama-3.3-70B-instruct-**IQ3_M**.gguf | 31.9 GB | model size = 134573.03 MiB<br>quant size = 30450.03 MiB | 754522.16 ms |
| I-Quant BF16 to IQ3_XS | llama-3.3-70B-instruct-**IQ3_XS**.gguf | 29.3 GB | model size = 134573.03 MiB<br>quant size = 27942.53 MiB | 933494.29 ms |
| I-Quant BF16 to IQ2_M | llama-3.3-70B-instruct-**IQ2_M**.gguf | 24.1 GB | model size = 134573.03 MiB<br>quant size = 22994.45 MiB | 628186.53 ms |
| I-Quant BF16 to IQ2_XS | llama-3.3-70B-instruct-**IQ2_XS**.gguf | 21.1 GB | model size = 134573.03 MiB<br>quant size = 20155.19 MiB | **2125798.23 ms** |
| I-Quant BF16 to IQ2_XXS | llama-3.3-70B-instruct-**IQ2_XXS**.gguf | 19.1 GB | model size = 134573.03 MiB<br>quant size = 18205.19 MiB | 1073457.38 ms |

**iMatrix** generation memory usage:

```
projected memory use with initial parameters [MiB]:
  - CUDA0 (NVIDIA RTX A6000):  48549 total,  36162 used,  12109 free vs. target of   1024
  - CUDA1 (NVIDIA RTX A6000):  48549 total,  35588 used,  12683 free vs. target of   1024
projected to use 71750 MiB of device memory vs. 96544 MiB of free device memory
```

```
load_tensors: offloaded 81/81 layers to GPU
load_tensors:        CUDA0 model buffer size = 35549.56 MiB
load_tensors:        CUDA1 model buffer size = 34880.09 MiB
load_tensors:    CUDA_Host model buffer size =  1064.62 MiB
```

## Perplexity Result

**System_info**
Device 0: NVIDIA RTX A6000, compute capability 8.6, VMM: yes
Device 1: NVIDIA RTX A6000, compute capability 8.6, VMM: yes
Build: 7714 (8e649571c) with GNU 11.4.0 for Linux x86_64juliana brooch
N_threads = 16

**Model_info**
Model_name: Llama 3.3 70b Instruct
Model_params: 70.55 B
Baseline format: GGUF (BF16)
GGUF version: version GGUF V3 (latest)

**Dataset_info**
(Same as previous)

calculating perplexity over 564 chunks

| Model | quant_type | file_size | load_tensor | llama_kv_cache | PPL | Load time | Prompt eval time |
|---|---|---|---|---|---|---|---|
| llama-3.3-70B-instruct-BF16.gguf<br><br>**(using two A6000, projected to use 133890 MiB of device memory vs. 96544 MiB of free device memory)** | BF16 | 131.42 GiB (16.00 BPW) | offloaded **56/81** layers to GPU, CUDA0 model buffer size = 45697.75 MiB, CUDA1 model buffer size = 46069.72 MiB, CUDA_Host model buffer size = 42805.56 MiB | size = 640.00 MiB, CUDA0 KV buffer size = 224.00 MiB, CUDA1 KV buffer size = 216.00 MiB, CPU KV buffer size = 200.00 MiB | 3.9196 +/- 0.02254 | 67707.61 ms | 4.40 ms per token, 227.33 tokens per second |
| llama-3.3-70B-instruct-Q8_0.gguf<br><br>(using one A6000) | Q8_0 | 69.82 GiB (8.50 BPW) | offloaded **53/81** layers to GPU, offloaded 53/81 layers to GPU, CUDA0 model buffer size = 46151.91 MiB, CUDA_Host model buffer size = 25342.38 MiB | size = 640.00 MiB, CUDA0 KV buffer size = 416.00 MiB, CPU KV buffer size = 224.00 MiB | 3.9193 +/- 0.02256 | 36215.73 ms | 3.45 ms per token, 290.01 tokens per second |
| llama-3.3-70B-instruct-Q6_K.gguf | Q6_K | 53.91 GiB (6.56 BPW) | offloaded **69/81** layers to GPU, CPU model buffer | size = 640.00 MiB, CUDA0 KV buffer size = | 3.9609 +/- 0.02282 | 22856.64 ms | 2.57 ms per token, 389.73 tokens per second |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | size = 821.95 MiB, CUDA0 model buffer size = 46343.73 MiB, CUDA_Host model buffer size = 8033.25 MiB | 544.00 MiB, CPU KV buffer size = 96.00 MiB | | | |
| llama-3.3-70B-instruct-Q5_K_M.gguf | Q5_K - Medium | 46.51 GiB (5.66 BPW) | offloaded **79/81** layers to GPU, CPU model buffer size = 688.88 MiB, CUDA0 model buffer size = 45755.73 MiB, CUDA_Host model buffer size = 1183.75 MiB | size = 640.00 MiB, CUDA0 KV buffer size = 624.00 MiB, CPU KV buffer size = 16.00 MiB | 3.9931 +/- 0.0231 1 | 16368.89 ms | 1.91 ms per token, 522.30 tokens per second |
| llama-3.3-70B-instruct-Q4_K_M.gguf | Q4_K - Medium | 39.59 GiB (4.82 BPW) | offloaded 81/81 layers to GPU, CPU model buffer size = 563.62 MiB, CUDA0 model buffer size = 39979.48 MiB | size = 640.00 MiB, CUDA0 KV buffer size = 640.00 MiB | 4.1358 +/- 0.0241 5 | 13310.68 ms | 1.75 ms per token, 570.52 tokens per second |
| llama-3.3-70B-instruct-Q3_K_M.gguf | Q3_K - Medium | 31.91 GiB (3.88 BPW) | offloaded 81/81 layers to GPU, CPU model buffer | size = 640.00 MiB, CUDA0 KV | 5.0488 +/- 0.0310 6 | 10901.88 ms | 1.89 ms per token, 530.25 tokens per second |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | size = 430.55 MiB, CUDA0 model buffer size = 32241.98 MiB | buffer size = 640.00 MiB | | | |
| llama-3.3-70B-instruct-Q2_K.gguf | Q2_K - Medium | 24.56 GiB (2.99 BPW) | offloaded 81/81 layers to GPU, CPU model buffer size = 328.78 MiB, CUDA0 model buffer size = 24816.98 MiB | size = 640.00 MiB, CUDA0 KV buffer size = 640.00 MiB | 6.9721 +/- 0.0458 1 | 8525.11 ms | 2.28 ms per token, 438.26 tokens per second |
| llama-3.3-70B-instruct-IQ4_XS.gguf | IQ4_XS - 4.25 bpw | 35.29 GiB (4.30 BPW) | offloaded 81/81 layers to GPU, CPU model buffer size = 532.31 MiB, CUDA0 model buffer size = 35606.98 MiB | size = 640.00 MiB, CUDA0 KV buffer size = 640.00 MiB | 4.1837 +/- 0.0244 2 | 11952.78 ms | 1.59 ms per token, 629.80 tokens per second |
| llama-3.3-70B-instruct-IQ3_M.gguf | IQ3_S mix - 3.66 bpw | 29.74 GiB (3.62 BPW) | offloaded 81/81 layers to GPU, CPU model buffer size = 430.55 MiB, CUDA0 model buffer size = 30019.48 MiB | size = 640.00 MiB, CUDA0 KV buffer size = 640.00 MiB | 4.5477 +/- 0.0264 4 | 10196.69 ms | 1.74 ms per token, 574.35 tokens per second |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| llama-3.3-70B-instruct-IQ3_XS.gguf | IQ3_XS - 3.3 bpw | 27.29 GiB (3.32 BPW) | offloaded 81/81 layers to GPU, CPU model buffer size = 430.55 MiB, CUDA0 model buffer size = 27511.98 MiB | size = 640.00 MiB, CUDA0 KV buffer size = 640.00 MiB | 4.8124 +/- 0.0283 1 | 9400.55 ms | 1.72 ms per token, 580.11 tokens per second |
| llama-3.3-70B-instruct-IQ2_M.gguf | IQ2_M - 2.7 bpw | 22.46 GiB (2.73 BPW) | offloaded 81/81 layers to GPU, CPU model buffer size = 430.55 MiB, CUDA0 model buffer size = 22563.91 MiB | size = 640.00 MiB, CUDA0 KV buffer size = 640.00 MiB | 5.8856 +/- 0.0370 7 | 7882.26 ms | 1.99 ms per token, 503.74 tokens per second |
| llama-3.3-70B-instruct-IQ2_XS.gguf | IQ2_XS - 2.3125 bpw | 19.68 GiB (2.40 BPW) | offloaded 81/81 layers to GPU, CPU model buffer size = 328.78 MiB, CUDA0 model buffer size = 19826.41 MiB | size = 640.00 MiB, CUDA0 KV buffer size = 640.00 MiB | 6.8056 +/- 0.0441 7 | 7000.36 ms | 2.02 ms per token, 494.49 tokens per second |
| llama-3.3-70B-instruct-IQ2_XXS.gguf | IQ2_XXS - 2.0625 bpw | 17.78 GiB (2.16 BPW) | offloaded 81/81 layers to GPU, CPU model buffer size = 328.78 MiB, CUDA0 model buffer | size = 640.00 MiB, CUDA0 KV buffer size = 640.00 MiB | 7.5425 +/- 0.0499 2 | 6174.44 ms | 1.68 ms per token, 596.83 tokens per second |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | size = 17876.41 MiB | | | | |

HellaSwag Score

| Model | quant_type | acc_norm | 95% confidence interval |
|---|---|---|---|
| llama-3.3-70B-instruct-BF16.gguf | BF16 | 84.50000000% | [82.1253%, 86.6106%] |
| llama-3.3-70B-instruct-Q8_0.gguf | Q8_0 | 84.50000000% | [82.1253%, 86.6106%] |
| llama-3.3-70B-instruct-Q6_K.gguf | Q6_K | 84.50000000% | [82.1253%, 86.6106%] |
| llama-3.3-70B-instruct-Q5_K_M.gguf | Q5_K - Medium | 84.40000000% | [82.0199%, 86.5169%] |
| llama-3.3-70B-instruct-Q4_K_M.gguf | Q4_K - Medium | 84.30000000% | [81.9144%, 86.4231%] |
| llama-3.3-70B-instruct-Q3_K_M.gguf | Q3_K - Medium | 83.80000000% | [81.3877%, 85.9536%] |
| llama-3.3-70B-instruct-Q2_K.gguf | Q2_K - Medium | 80.90000000% | [78.3472%, 83.2163%] |
| llama-3.3-70B-instruct-IQ4_XS.gguf | IQ4_XS - 4.25 bpw | 84.20000000% | [81.8090%, 86.3292%] |
| llama-3.3-70B-instruct-IQ3_M.gguf | IQ3_S mix - 3.66 bpw | 84.30000000% | [81.9144%, 86.4231%] |
| llama-3.3-70B-instruct-IQ3_XS.gguf | IQ3_XS - 3.3 bpw | 83.80000000% | [81.3877%, 85.9536%] |
| llama-3.3-70B-instruct-IQ2_M.gguf | IQ2_M - 2.7 bpw | 83.70000000% | [81.2825%, 85.8596%] |
| llama-3.3-70B-instruct-IQ2_XS.gguf | IQ2_XS - 2.3125 bpw | 82.20000000% | [79.7073%, 84.4462%] |
| llama-3.3-70B-instruct-IQ2_XXS.gguf | IQ2_XXS - 2.0625 bpw | 81.00000000% | [78.4517%, 83.3111%] |

Inference Test

| Model | Output | Prompt processing | Text generation |
|---|---|---|---|

| | | | |
|---|---|---|---|
| llama-3.3-70B-instruct-BF16.gguf<br><br>**(using two A6000)** | {<br>"summary": "Model quantization is a technique used to reduce the precision of model weights in large language models, resulting in significant memory and computational savings. This method is particularly useful for deploying models on edge devices or in resource-constrained environments. By reducing the precision of model weights, quantization enables faster inference times and lower memory usage without sacrificing significant model accuracy.",<br>"advantages": [<br>"Reduced memory usage",<br>"Faster inference times",<br>"Improved model deployability"<br>],<br>"disadvantages": [<br>"Potential loss of model accuracy",<br>"Increased complexity in model training",<br>"Limited support for certain model architectures"<br>]<br>} | 51.1 t/s | 1.1 t/s |
| llama-3.3-70B-instruct-Q8_0.gguf<br><br>(using one A6000) | {"summary": "Model quantization is a technique used to reduce the precision of model weights in large language models, resulting in significant memory and computational savings. This method is particularly useful for deploying models on edge devices or in resource-constrained environments. By reducing the precision of model weights, quantization enables faster inference times and lower memory usage without sacrificing much accuracy.",<br>"advantages": ["Reduced memory usage", "Faster inference times", "Improved model deployability"],<br>"disadvantages": ["Potential loss of model accuracy", "Increased complexity in model training", "Limited support for certain model architectures"]} | 89.4 t/s | 1.9 t/s |

| | | | |
|---|---|---|---|
| llama-3.3-70B-instruct-Q6_K.gguf | {<br>"summary": "Model quantization is a technique used to reduce the precision of model weights in large language models, resulting in significant memory and computational savings. This reduction in precision can lead to a slight decrease in model accuracy, but the benefits often outweigh the costs. By applying model quantization, developers can deploy large language models on devices with limited resources, making them more accessible to a wider range of users.",<br>"advantages": [<br>"Reduced memory usage",<br>"Improved computational efficiency",<br>"Faster model deployment"<br>],<br>"disadvantages": [<br>"Potential loss of model accuracy",<br>"Increased complexity in model training",<br>"Limited support for certain model architectures"<br>]<br>} | 186.4 t/s | 4.5 t/s |
| llama-3.3-70B-instruct-Q5_K_M.gguf | {"summary": "Model quantization is a technique used to reduce the precision of model weights in large language models, resulting in significant memory and computational savings. This method is particularly useful for deploying models on edge devices or in resource-constrained environments. By reducing the precision of model weights, quantization enables faster inference times and lower memory usage without sacrificing much accuracy.",<br>"advantages": ["Reduced memory usage", "Faster inference times", "Improved model deployability"],<br>"disadvantages": ["Potential loss of model accuracy", "Increased complexity in model training", "Limited support for certain model architectures"]} | 388.0 t/s | 10.9 t/s |

| | | | |
|---|---|---|---|
| llama-3.3-70B-instruct-Q4_K_M.gguf | {<br>"summary": "Model quantization is a technique used to reduce the precision of model weights in large language models, resulting in significant memory and computational savings. This method is particularly useful for deploying models on edge devices or in resource-constrained environments. By reducing the model size, quantization enables faster inference times and lower latency, making it an attractive option for real-world applications.",<br>"advantages": [<br>"Reduced memory footprint",<br>"Faster inference times",<br>"Improved model deployability"<br>],<br>"disadvantages": [<br>"Potential loss of model accuracy",<br>"Increased complexity in model training",<br>"Limited support for certain model architectures"<br>]<br>} | 485.8 t/s | 16.5 t/s |
| llama-3.3-70B-instruct-Q3_K_M.gguf | {<br>"summary": "Model quantization in large language models is a technique used to reduce the memory and computational requirements of these models. This is achieved by representing the model's weights and activations with lower precision data types, such as integers instead of floating-point numbers. By doing so, model quantization enables the deployment of large language models on devices with limited resources, making them more accessible and widely available.",<br>"advantages": [<br>"Reduced memory usage",<br>"Faster inference times",<br>"Increased model accessibility"<br>],<br>"disadvantages": [<br>"Potential loss of model accuracy",<br>"Increased complexity in model training",<br>"Requires significant computational resources for quantization process"<br>]<br>} | 453.0 t/s | 15.3 t/s |

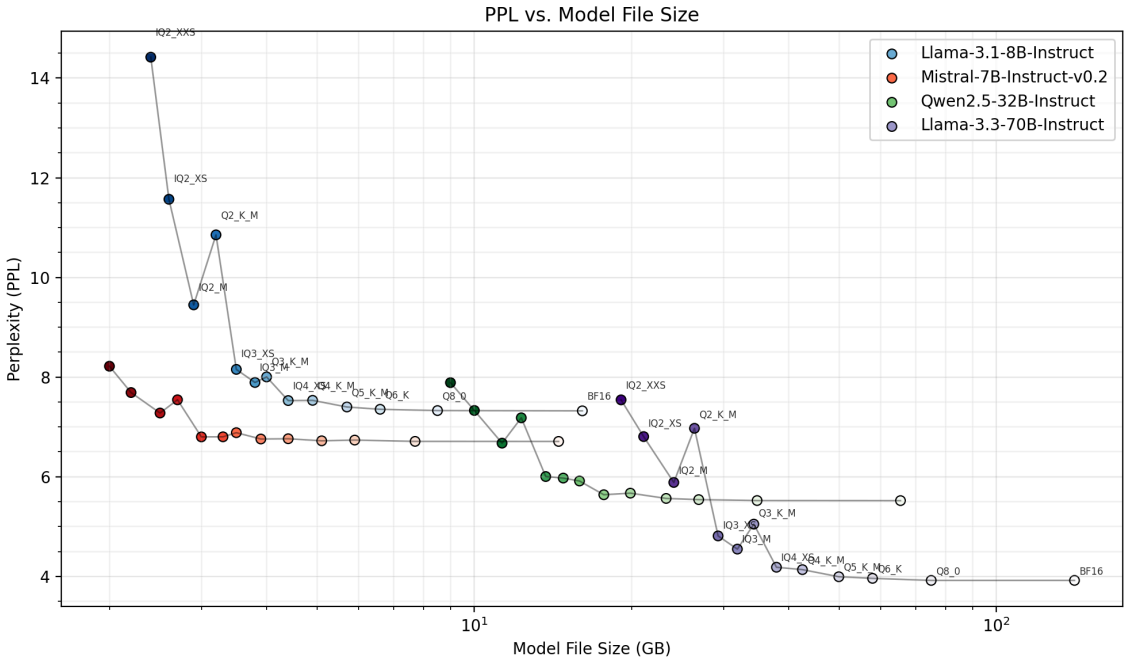
| | | | |
|---|---|---|---|
| llama-3.3-70B-instruct-Q2_K.gguf | {"summary": "Model quantization is a technique used to reduce the precision of model weights in large language models. This reduction in precision leads to a decrease in memory usage and computational cost. As a result, quantized models can be deployed on devices with limited resources, making them more accessible and efficient.", "advantages": ["Reduced memory usage", "Decreased computational cost", "Improved deployment flexibility"], "disadvantages": ["Potential loss of model accuracy", "Increased complexity in model training", "Limited support for certain hardware platforms"]} | 377.1 t/s | 18.9 t/s |
| llama-3.3-70B-instruct-IQ4_XS.gguf | {<br>"summary": "Model quantization is a technique used to reduce the precision of model weights in large language models, resulting in significant memory and computational savings. This reduction in precision can lead to a slight decrease in model accuracy, but the benefits often outweigh the costs. By applying model quantization, developers can deploy large language models on devices with limited resources, making them more accessible to a wider range of users.",<br>"advantages": [<br>"Reduced memory usage",<br>"Improved computational efficiency",<br>"Faster model deployment"<br>],<br>"disadvantages": [<br>"Potential loss of model accuracy",<br>"Increased complexity in model training",<br>"Limited compatibility with certain hardware"<br>]<br>} | 543.6 t/s | 18.6 t/s |
| llama-3.3-70B-instruct-IQ3_M.gguf | {"summary": "Model quantization is a technique used to reduce the precision of model weights in large language models, resulting in significant memory and computational savings. This technique is particularly useful for deploying models on edge devices or in environments with limited computational resources. By reducing the precision of model weights, quantization can help accelerate inference times and reduce the overall cost of model deployment.",<br>"advantages": ["Reduced memory usage", "Faster inference times", "Improved model deployability"],<br>"disadvantages": ["Potential loss of model accuracy", "Increased complexity in model training", "Limited support for certain model architectures"]} | 475.0 t/s | 19.6 t/s |

| | | | |
|---|---|---|---|
| llama-3.3-70B-instruct-IQ3_XS.gguf | {"summary": "Model quantization is a technique used to reduce the precision of model weights in large language models, resulting in significant memory and computational savings. This technique is particularly useful for deploying models on edge devices or in environments with limited computational resources. By reducing the precision of model weights, quantization enables faster inference times and lower memory usage without sacrificing significant model accuracy.", "advantages": ["Reduced memory usage", "Faster inference times", "Improved model deployability"], "disadvantages": ["Potential loss of model accuracy", "Increased complexity in model training", "Limited support for certain model architectures"]} | 479.2 t/s | 19.8 t/s |
| llama-3.3-70B-instruct-IQ2_M.gguf | { "summary": "Model quantization is a technique used to reduce the precision of model weights in large language models, which can lead to significant memory and computational savings. This is achieved by representing the weights using fewer bits, typically 8 or 16 bits, instead of the standard 32 bits. By doing so, model quantization enables the deployment of large language models on devices with limited resources, such as mobile phones or smart home devices.", "advantages": ["Reduced memory usage", "Faster inference times", "Lower power consumption"], "disadvantages": ["Potential loss of model accuracy", "Increased complexity in model training", "Limited support for certain model architectures"] } | 426.0 t/s | 21.6 t/s |
| llama-3.3-70B-instruct-IQ2_XS.gguf | {"summary": "Model quantization is a technique used to reduce the precision of model weights in large language models, which can lead to significant memory and computational savings. This is achieved by representing the model weights using lower-precision data types, such as integers or floats with fewer bits. By doing so, model quantization can help to deploy large language models on devices with limited memory and computational resources.", "advantages": ["Reduced memory usage", "Faster inference times", "Improved model deployment on edge devices"], "disadvantages": ["Potential loss of model accuracy", "Increased complexity in model training", "Limited support for certain model architectures"} | 426.9 t/s | 21.6 t/s |

| | | | |
|---|---|---|---|
| llama-3.3-70B-instruct-IQ2_XXS.gguf | {<br>"summary": "Model quantization is a technique used to reduce the precision of model weights and activations, which can lead to significant reductions in memory and compute requirements. This can be particularly useful for large language models, where the memory and compute requirements can be very high. By applying model quantization, developers can deploy these models on a wider range of devices, including those with limited resources.",<br>"advantages": [<br>"Reduced memory requirements",<br>"Improved inference speed",<br>"Simplified deployment on low-power devices"<br>],<br>"disadvantages": [<br>"Potential loss of model accuracy",<br>"Increased difficulty in achieving desired precision",<br>"Limited support from frameworks and libraries"<br>]<br>} | 505.2 t/s | 22.0 t/s |

# Consolidated Results Graphs

## Perplexity vs. model size:




## HellaSwag score vs. model size: