Host a Llama 2 API on GPU for Free

Llama 2 is a Gamechanger

Llama 2 is the latest text-generation model from Meta which currently outperforms every opensource alternative.

It beats out Falcon-40B (the previous best opensource foundation model) and is on par with GPT-3.5, only falling short of GPT-4 and PALM 2 (both closed source models, owned by OpenAI and Google respectively).

Model	Average
meta-llama/Llama-2-70b-hf	67.3
huggyllama/llama-65b	64.2
llama-65b	64.2
llama-30b	61.7
tiiuae/falcon-40b	61.5
meta-llama/Llama-2-13b-hf	58.7
TheBloke/Llama-2-13B-fp16	58.6
dvruette/llama-13b-pretrained-sft-do2	58.5
dvruette/llama-13b-pretrained	57.8
dvruette/llama-13b-pretrained-dropout	57.7
dvruette/llama-13b-pretrained-sft-epoch-1	56.8
mosaicml/mpt-30b	56.2
llama-13b	56.1
huggyllama/llama-13b	56
meta-llama/Llama-2-7b-hf	54.4
openlm-research/open_llama_13b	52.1
llama-7b	49.7
facebook/galactica-30b	48.6

Opensource Foundation Models Leaderboard on HuggingFace

On the same leaderboard as shown above, if you change the filters to include finetuned models, you will find that basically the entire list is made up of Llama 2 derivatives.

While Llama 2 is not fully open, it is very permissive for the vast majority of users.

> If, on the Llama 2 version release date, the monthly active users of the products or services made available by or for Licensee, or Licensee's affiliates, is greater than 700 million monthly active users in the preceding calendar month, you must request a license from Meta

Hosting a Llama 2 Backed API

Llama 2 models come in 3 different sizes: 7B, 13B, and 70B parameters.

The 70 Billion parameter version requires multiple GPUs so it won't be possible to host for free.

Out of the 13B and 7B version, the 13B version is more powerful but requires some compression (quantization or reducing float precision) to fit on a single mid-range GPU. Luckily the Llama cpp library makes this fairly trivial!

The basic outline to hosting a Llama 2 API will be as follows:

1. Use Google Colab to get access to an Nvidia T4 GPU for free!
2. Use Llama cpp to compress and load the Llama 2 model onto GPU.
3. Create a FastAPI server to provide a REST API to the model.
4. Use Ngrok to expose the FastAPI endpoints via a public URL.

The full Colab code can be found below:

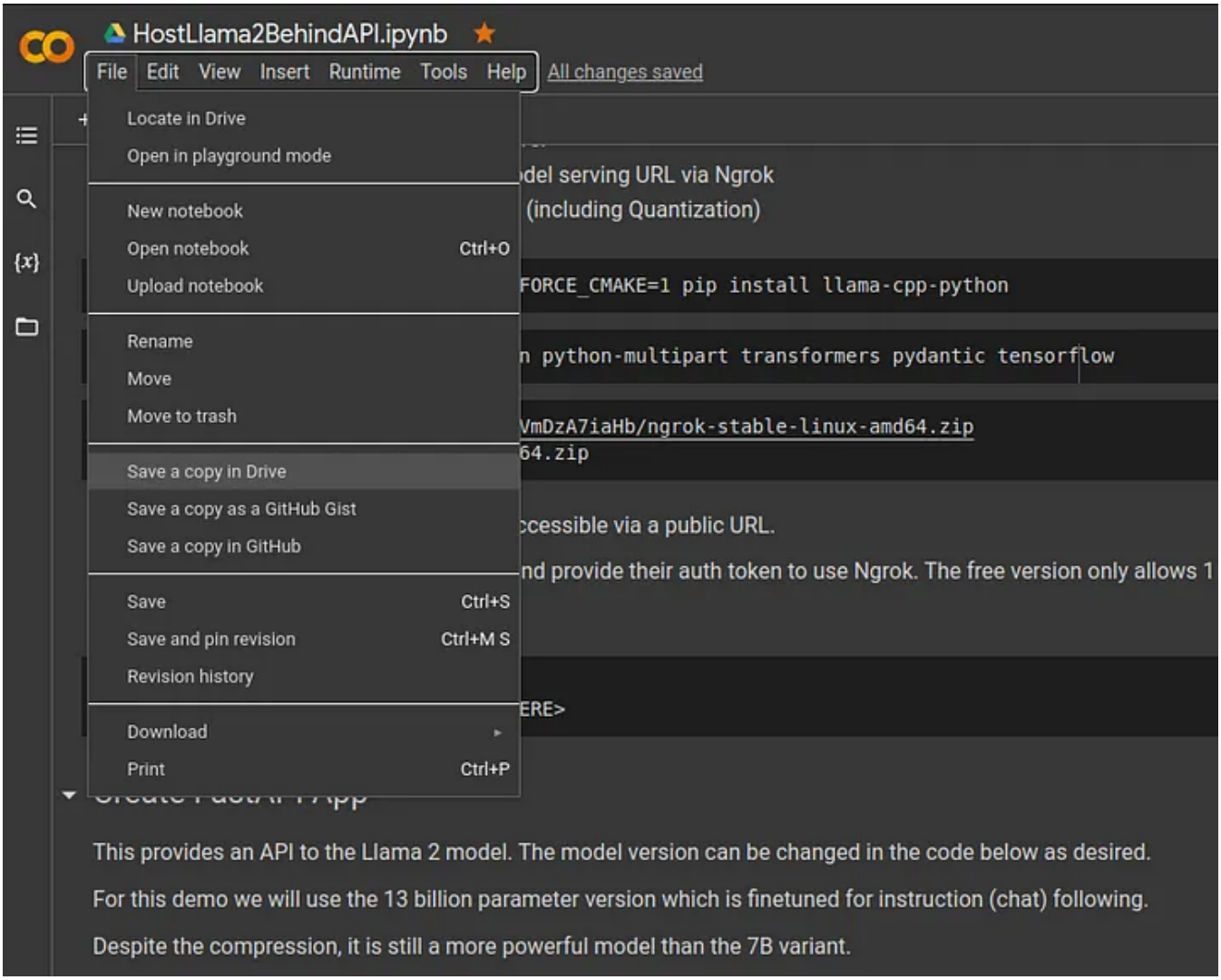
HostLlama2API

How to provide an API to Llama 2 model for free

colab.research.google.com

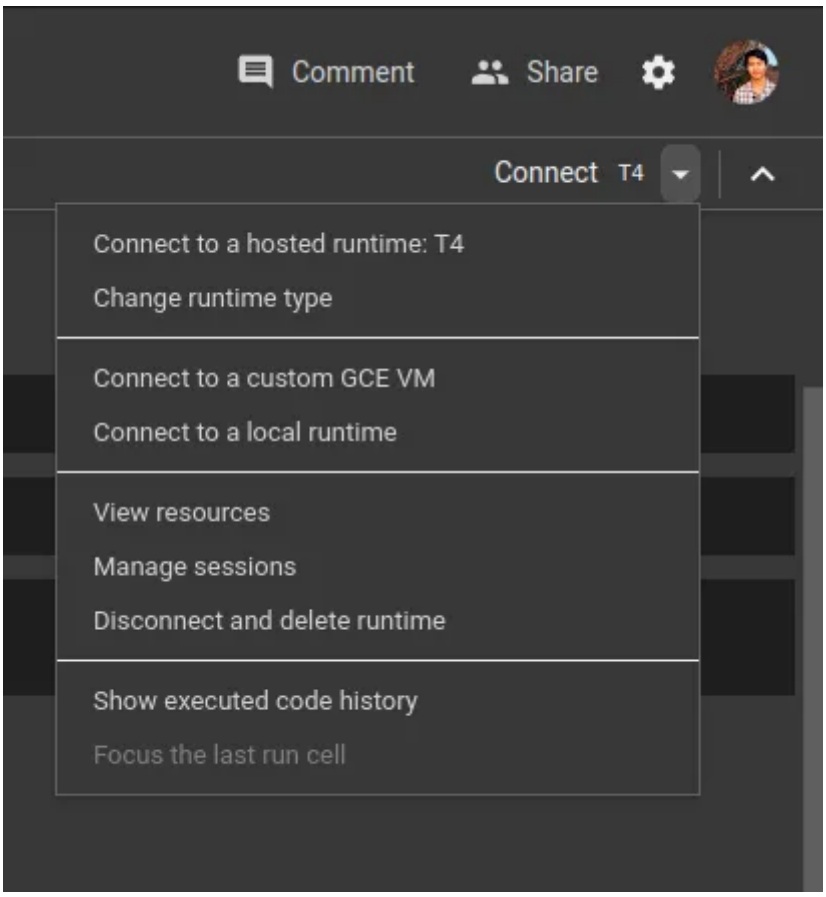
Working with Google Colab

Start off by making a copy of the example notebook:



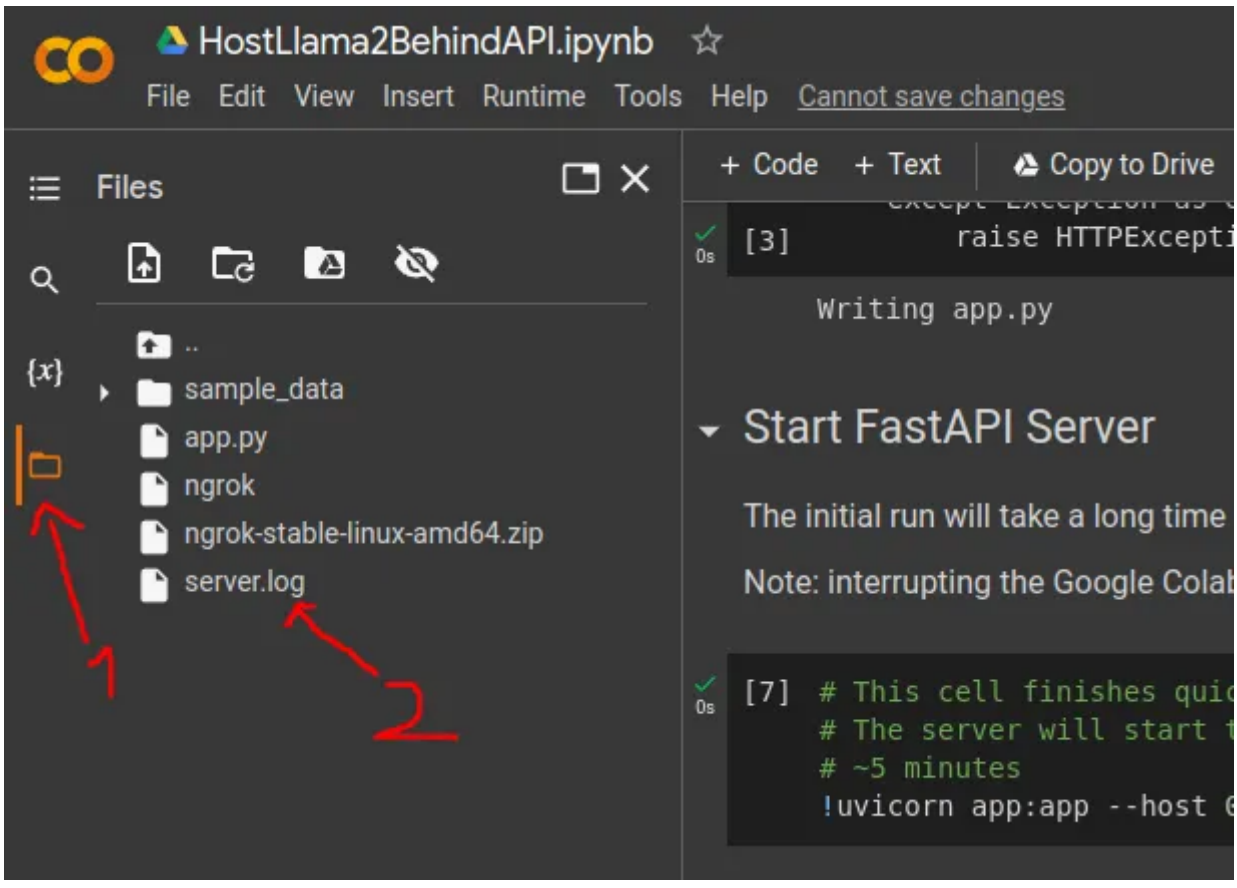
On the top left corner, select File -> Save a copy in Drive. This will open a new Colab owned by you.

Double check that the notebook is set to use a T4 Nvidia GPU:



On the top right corner, there are options to change the runtime hardware.

To view the files on the instance (such as the server.log), check the left sidebar:

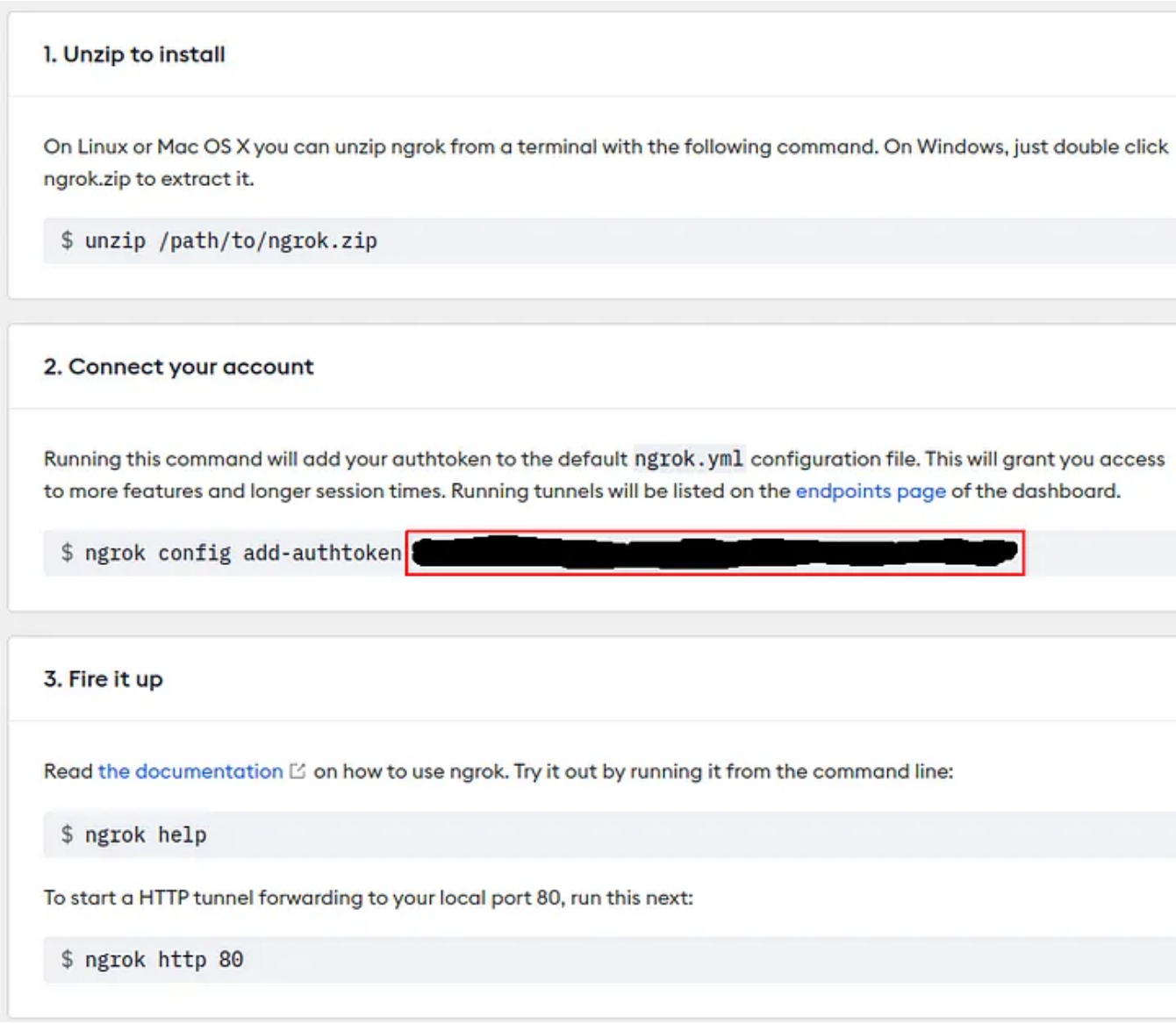


Create Public URL with Ngrok

The last tool we need is Ngrok which provides a way to make the model serving endpoint accessible via a public URL.

Users are required to make a free account which only allows 1 localtunnel. (You don't need to download or set up anything, you just need an account for the user auth key)

You can sign up using email or via GitHub or Google accounts. Feel free to skip the 2-factor authentication setup.



Grab the Auth token on the Ngrok setup page, marked by the red box in the image

Place the user Auth token in the Colab code where it says:

```
!./ngrok authtoken <YOUR-NGROK-TOKEN-HERE>
```

This is the only change required for the Notebook, easy-peasy!

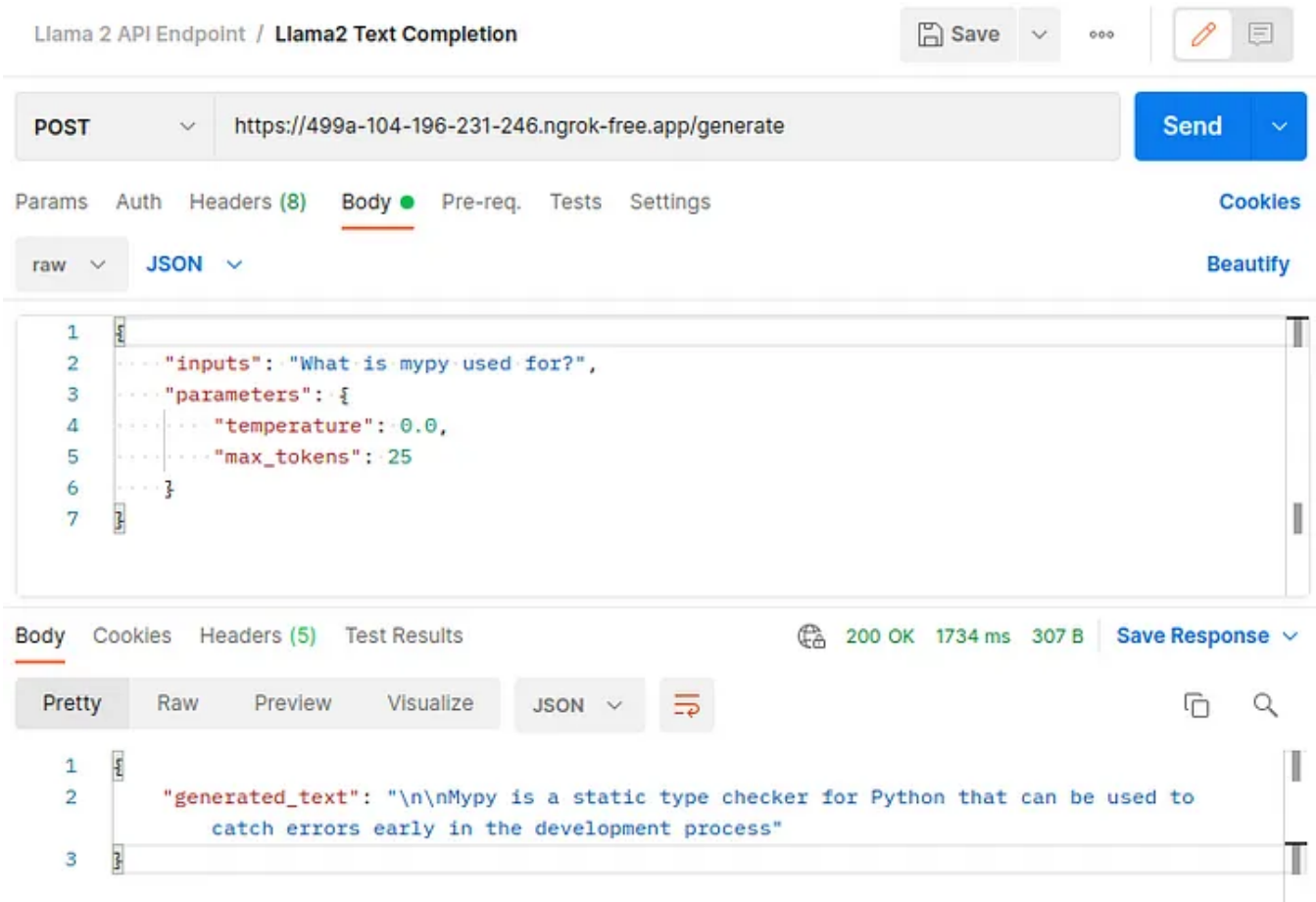
The rest of the flow is fairly straightforward, please refer to the notebook.

Hitting the Llama 2 Model API

Now you can easily hit your Llama 2 model with a simple request!

```
curl --location --request POST 'https://499a-104-196-231-246.ngrok-free.app/ger
--header 'Content-Type: application/json' \
--data-raw '{
    "inputs": "What is mypy used for?",
    "parameters": {
        "temperature": 0.0,
        "max_tokens": 25
    }
}'
```

Here is how the request/response might look if you use Postman:

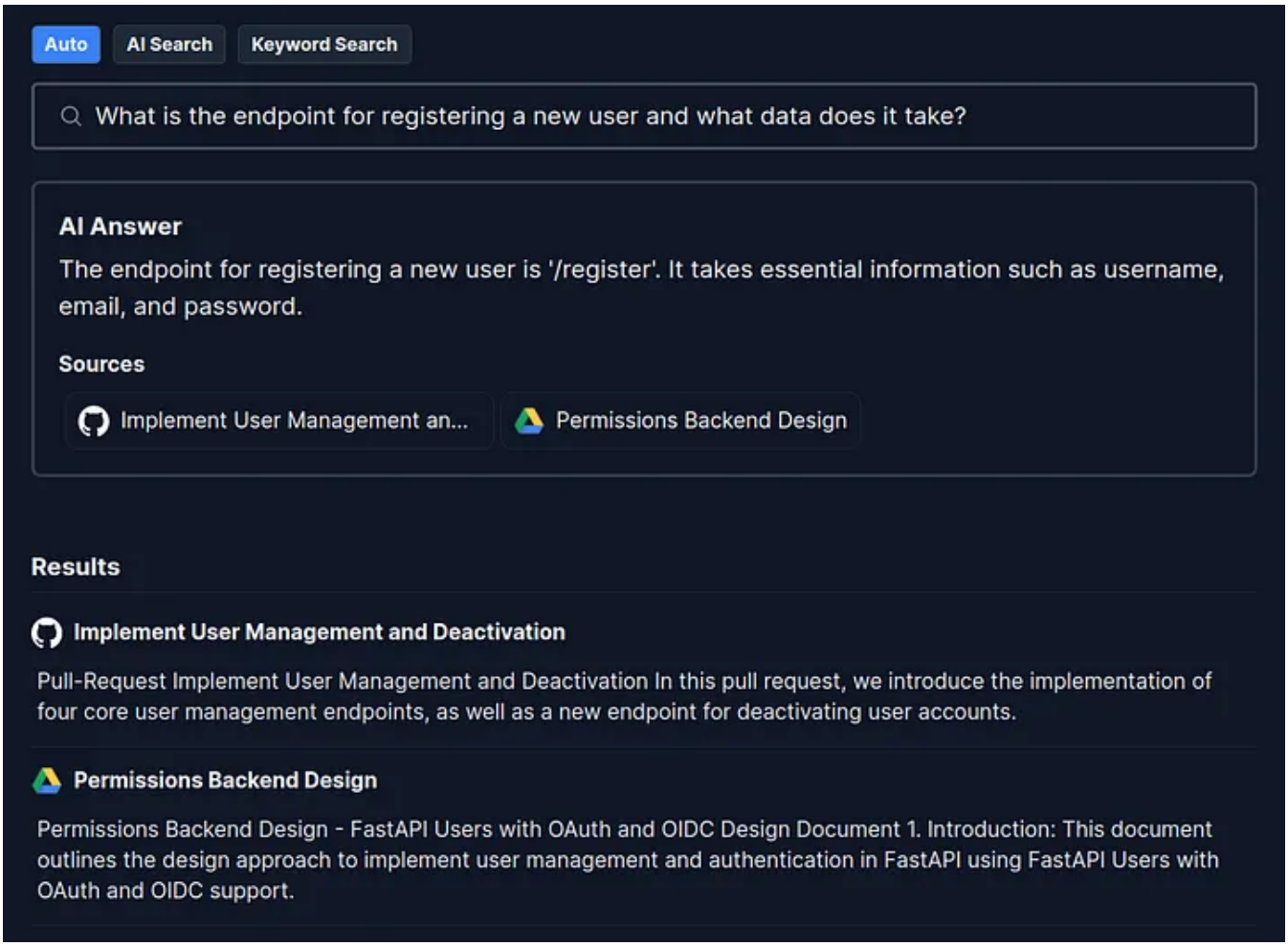


Be sure to use your own Ngrok public URL

Next Steps

One of the most prominent and broadly applicable use cases of host-hosted LLMs is to enrich the model's internal knowledge with your private data so that it can answer user queries with full context like a knowledgeable teammate.

I walk through how to set up a free and opensource project for doing this in this follow up post.



Question-Answer result from Danswer connected to GitHub and GoogleDrive

As of August 2023, Danswer is the only free and fully opensource project for LLM based question-answering that is fully featured for production use.

If you liked this post, I highly recommend checking out Danswer as well!

Danswer Code: https://github.com/danswer-ai/danswer

Danswer Docs: https://docs.danswer.dev/

Closing Notes

While it is incredibly valuable to be able to host your own LLM at no cost, there are some caveats to using Google Colab:

- This guide (and Google Colab) is intended more for development use, to have a permanent endpoint, you probably want to invest in dedicated hardware as Google Colab will reclaim the instance after a period of inactivity.
- The higher end GPUs like the A100s are not available on the free tier.
- On the free tier, you can only claim an instance for up to 12 hours in a single session.

Llama 2 NLP Large Language Models Hosting Open Source