

# Technisches

Technologien, Hosting, Webseite, Messages, Bug-Reports, Aufbau auf Vorarbeit

- [📄 Link collection for the technical prep work](#)
- [📄 How to Find a GPU Hosting Service – a Guide by ViraaJ Akuthota](#)
- [📄 Tipps zu KI und LLMs](#)

# ☐☐ Link collection for the technical prep work

- Open Source Guide: <https://opensource.guide/de/>
- Trends auf GitHub: <https://github.com/trending>
- AlternativeTo, Crowdsourced Software Recommendations: <https://alternativeto.net/>
- CHAOSS: <https://chaoss.community/software/>
- You can apply for virtual private servers financed by the Open Tech Fund:  
<https://www.eclips.is/>

## Markdown & co

- *Tables Generator* for Markdown: [https://www.tablesgenerator.com/markdown\\_tables](https://www.tablesgenerator.com/markdown_tables)
- *Pandoc* converts various formats (like markdown, HTML, LaTeX, docx,...) into each other.  
Great to build simple tool chains around documents: <https://pandoc.org/>

# 📄 How to Find a GPU Hosting Service – a Guide by Viraa Akuthota

*For his project "[Human Rights Predictor](#)" (Round 15) our grantee Viraa Akuthota was looking for a GPU hosting service. Here he explains how he went about it:*

To fine-tune models and create embeddings on large corpuses of qualitative data, a high amount of GPU RAM (VRAM) is required. For example, fine-tuning BERT on a dataset of 15k cases that vary in size creates roughly 100k-200k sequences at a 512 token limit. This requires approximately 140 GB of VRAM. This hardware requirement means such tasks cannot be conducted on most consumer-grade machines. I conducted an exercise to hopefully identify an affordable and relatively easy-to-use cloud compute option. During this search, I faced many difficulties. The benefits and disadvantages of the majority of service providers I reviewed can be found in the table below.

Overall, the production system I landed on is to utilize:

- PaperSpace's Core using a Windows Server instance to avoid using the terminal as much as possible.
- Always available Multi-GPU instances, for example, 4 x A6000 Nvidia GPUs with 192 GB VRAM total for roughly \$7 USD an hour.
- Approximately \$3 USD per month for 50 GB persistent storage, making offline costs negligible.
- For Linux users, they have a Python ML template which will save time installing python, packages, cuda, etc.

Before production, I utilise either Google Colab or HuggingFace:

- For testing fine-tuning or creating embeddings, I believe Google Colab's free T4 instance provides the highest amount of VRAM for any free tier.
- For testing LLMs, HuggingFace's serverless inference free tier allows you to utilize a variety of LLMs such as LLAMA 405B. However, the Pro tier at \$9 USD per month increases the rate limit on this inference. I receive approximately 300 API calls per hour.

Provider	Benefits	Disadvantages	GPU Limit
----------	----------	---------------	-----------

Amazon EC2	<ul style="list-style-type: none"> <li>• Relatively affordable compared to other cloud providers</li> </ul>	<ul style="list-style-type: none"> <li>• Requires familiarity with AWS</li> <li>• Application for quotas is not straightforward and the approval process takes time</li> </ul>	<ul style="list-style-type: none"> <li>• Essentially unlimited</li> </ul>
Amazon Notebooks	<ul style="list-style-type: none"> <li>• Easy to set up an ML system</li> <li>• Relatively affordable compared to other cloud providers</li> </ul>	<ul style="list-style-type: none"> <li>• Notebooks are limited to certain GPU sizes, essentially under 100GB VRAM.</li> <li>• Even if you have a quota for the underlying resource it will not work for a notebooks</li> </ul>	<ul style="list-style-type: none"> <li>• Under 100GB VRAM</li> </ul>
Microsoft Azure		<ul style="list-style-type: none"> <li>• The registration system and console is sufficiently complicated that I did not utilise this service.</li> <li>• Quota application process did not seem straight forward.</li> </ul>	<ul style="list-style-type: none"> <li>• Essentially unlimited</li> </ul>
Google Cloud		<ul style="list-style-type: none"> <li>• Unable to secure access to a high-end GPU as they were <b>ALWAYS</b> unavailable</li> </ul>	

Google Colab	<ul style="list-style-type: none"> <li>• Very easy to use and set up</li> </ul>	<ul style="list-style-type: none"> <li>• Relatively more expensive</li> <li>• Not guaranteed access to the most powerful GPUs that is claimed to be accessible even with premium services</li> </ul>	<ul style="list-style-type: none"> <li>• A100 GPU with 40GB VRAM, if available, which is rare</li> </ul>
Paperspace Notebooks	<ul style="list-style-type: none"> <li>• Very easy to use and set up</li> <li>• Multiple 'free' GPU availability with unlimited hours at the premium option</li> </ul>		<ul style="list-style-type: none"> <li>• PaperSpace has plans which provide various systems at 6 hours of continuous use at a mix of free or paid options. The free options still require a base payment plan to be purchased</li> <li>• For the premium plan, a single P5000 15gb VRAM machine is available for free.</li> <li>• A 'core' machine can also be purchased where you can pay per hour without having to pay for a monthly plan. I currently have 4 x A6000 48gb VRAM for \$7.56 an hour.</li> </ul>
Paperspace Server/Console	<ul style="list-style-type: none"> <li>• Always available multi-GPU instance</li> <li>• ML template server instances</li> <li>• Easy server setup</li> </ul>	<ul style="list-style-type: none"> <li>• More expensive than the big players</li> <li>• Some of the ML template server instances come with certain issues with libraries</li> </ul>	<ul style="list-style-type: none"> <li>• Essentially unlimited</li> </ul>



# ☐☐ Tipps zu KI und LLMs

Hier veröffentlichen wir fortlaufend Tipps und Tricks rund um die Entwicklung von KI-Anwendungen:

Zum Training und Finetuning von LLMs gibt es den [Unsloth's Instruct Modell Trainer](https://github.com/unslothai/unsloth), der kostenlos ist, auf einer ebenfalls kostenlosen Google Colab Instanz betrieben werden kann und sehr gute Ergebnisse produziert. Hier geht es zum Repository:

<https://github.com/unslothai/unsloth?tab=readme-ov-file>.