Member-only story

12 things I wish I knew before starting to work with Hugging Face LLM

Insights and Tips for Navigating the Hugging Face LLM Landscape

Fabio Matricardi · Following
Published in Artificial Corner
8 min read · Jun 8

Listen Share More

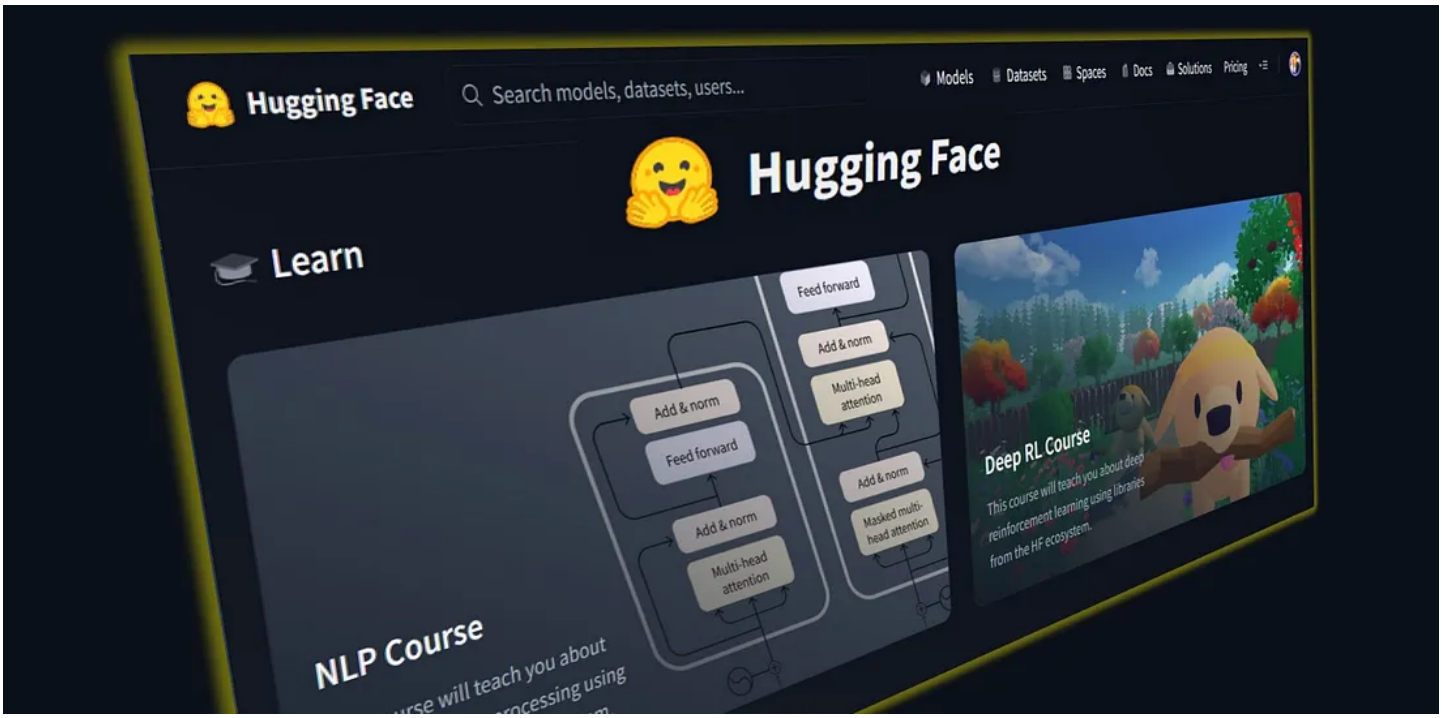


Image Created by the author

Hugging Face has become one of the most popular open-source libraries for Artificial Intelligence.

It is a treasure for every enthusiast of Natural Language Processing tasks.

When you access the Hugging Face's Language Model Hub you are in a complete new world of possibilities.

I started to experiment on my Google Colab Notebook every new feature I could. But the number of failures were greater than the success! **When you run your code, following tutorials and examples, and 8/10 times you get an error you just want to give up**! 😒

If you want to learn new tool or library, it is beneficial to know beforehand the potential issues that may arise: things that one wishes they had known before diving in.

In this article we will explore **12 thing**s every beginner should know. These tips will help you avoid common frustration and improve your progress with Hugging Face LLMs. *They are split into 4 main topics*:

1. 📚 Training course

The Hugging Face Free Course is a free course on NLP using the HuggingFace ecosystem. It focuses on teaching the ins and outs of NLP and how to accomplish state-of-the-art tasks in NLP.

When you register to their portal the first thing you are asked for is to joining the free training course. I immediately clicked on yes (it is free...).

The course is divided into three major modules, each divided into chapters or subsections.

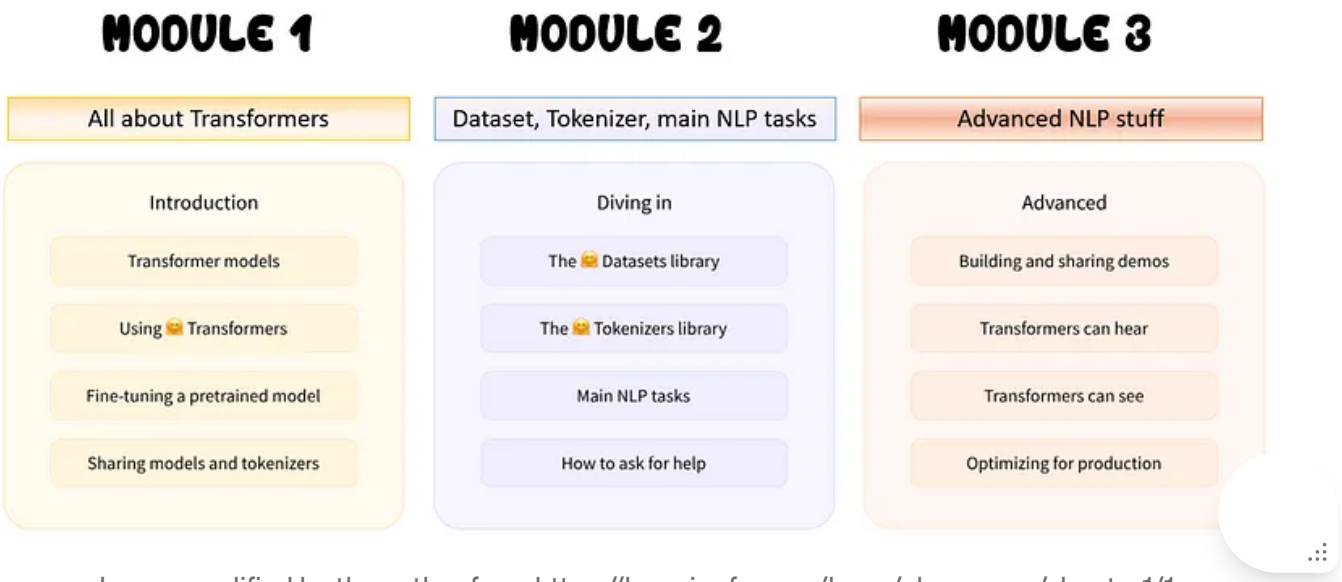


Image modified by the author from https://huggingface.co/learn/nlp-course/chapter1/1

The course is available in Pytorch and Tensorflow and can be followed along with Google Colab notebook.

The training course is extensive and well organized. There are also quizzes at the end of each chapter to test understanding

I whish I knew...

1. If you are a beginner, you can start using pre-trained models with the Hugging Face Transformers library: follow few tutorials and that is enough.
2. If you are a more advanced user you can even fine-tune and customize these models for specific NLP tasks: in this scenario it is a good idea to complete the entire course.
3. The best way to understand is to **test yourself on a Google Colab Notebook, experimenting new things**. If you only follow this long course without testing things yourself you will not benefit from it. Use the material as a reference and refer to the official documentation as much as you can.

2. 🧰 Transformers and Pipelines

Transformers are your toolbox to interact with all the Hugging Face models. You don't even have to download them: if you create an `API Token` you can call the *Inference API* to do your job (like you may have done with *ChatGPT*).

With **transformers** you don't need to know immediately complicated techniques to use LLMs. With the pretrained models you can perform many common tasks:

📝 Natural Language Processing: text classification, named entity recognition, question answering, language modeling, summarization, translation, multiple choice, and text generation.
🖼️ Computer Vision: image classification, object detection, and segmentation.
🗣️ Audio: automatic speech recognition and audio classification.
🐙 Multimodal: table question answering, optical character recognition, information extraction from scanned documents, video classification, and visual question answering.

I whish I knew...

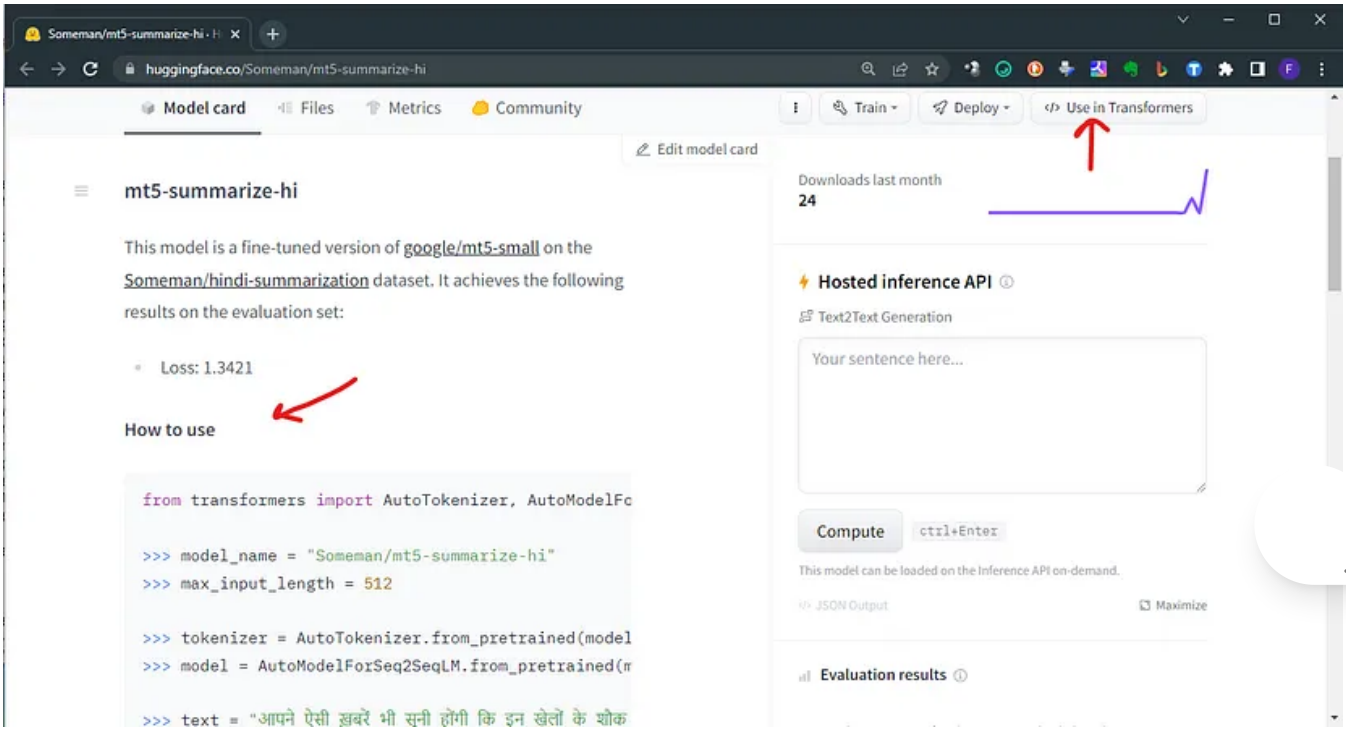
4. Use *AutoTokenizer* and *AutoModelForSeq2Seq*: initially I thought that you need t use specific Transformers and Tokenizers for each Model family (for example if you use a T5 model family, you need to use specific transformers for them)

```
from transformers import T5Tokenizer, T5ForConditionalGeneration

tokenizer = T5Tokenizer.from_pretrained("t5-small")
model = T5ForConditionalGeneration.from_pretrained("t5-small")
```

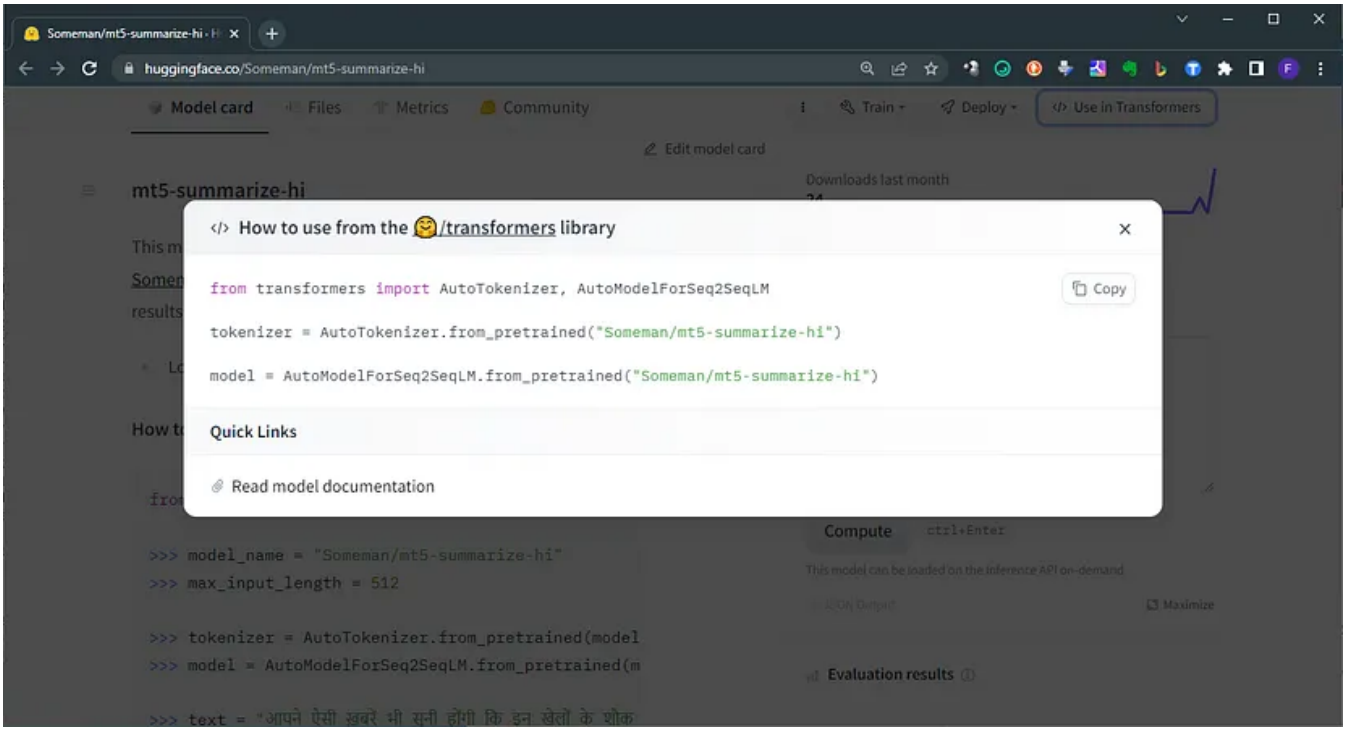
But not all models have clear indication on Hugging Face model card to how to use them. For all pre-trained model you can declare a simple statement:

```
from transformers import AutoTokenizer, AutoModelForCausalLM

#replace "databricks/dolly-v2-3b" with "yourpathto/hfmodel..."
tokenizer = AutoTokenizer.from_pretrained("databricks/dolly-v2-3b")
model = AutoModelForCausalLM.from_pretrained("databricks/dolly-v2-3b")
```

5. On the model card of every model, you have a quick guide to use in transformers



Model Card: how to use section



The Use in Transformers button on the top right corner

3. 🤖 What Model should I pick?

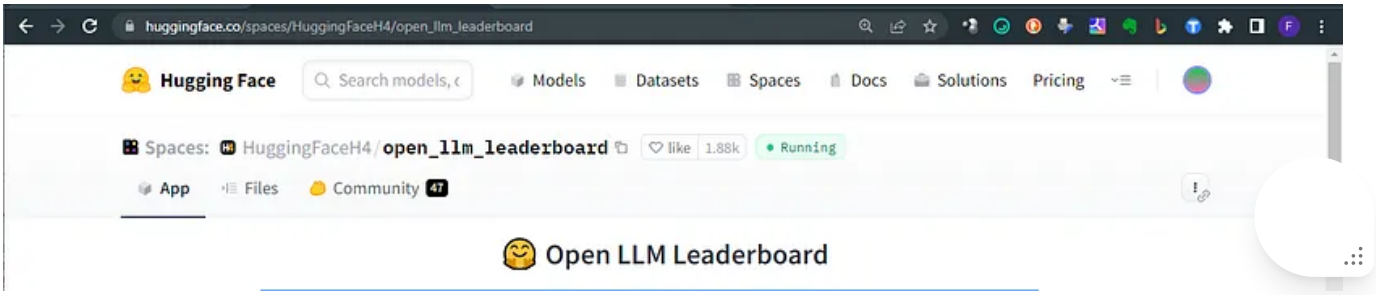
The Hugging Face Hub is a platform with over 120k models, 20k datasets, and 50k demo apps (Spaces), all open source and publicly available, in an online platform where people can easily collaborate and build ML together.

With such a huge number it is difficult to pick the right one. In the beginning I was browsing trough them at random. Testing them this way, was a big mistake and a

waste of time.

I whish I knew...

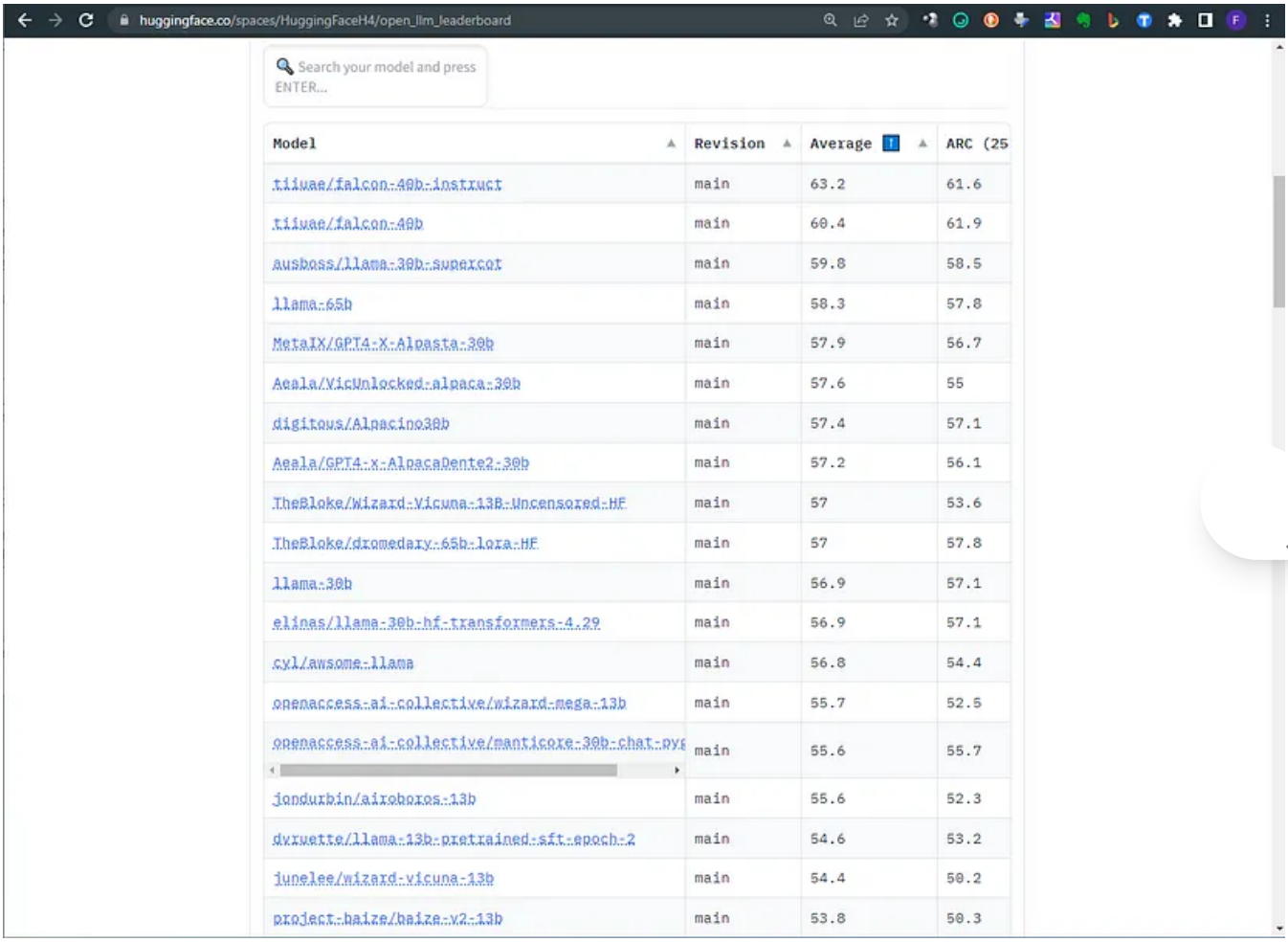
6. You can start from the Leaderboard to understand the more performing models in the community.



https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard

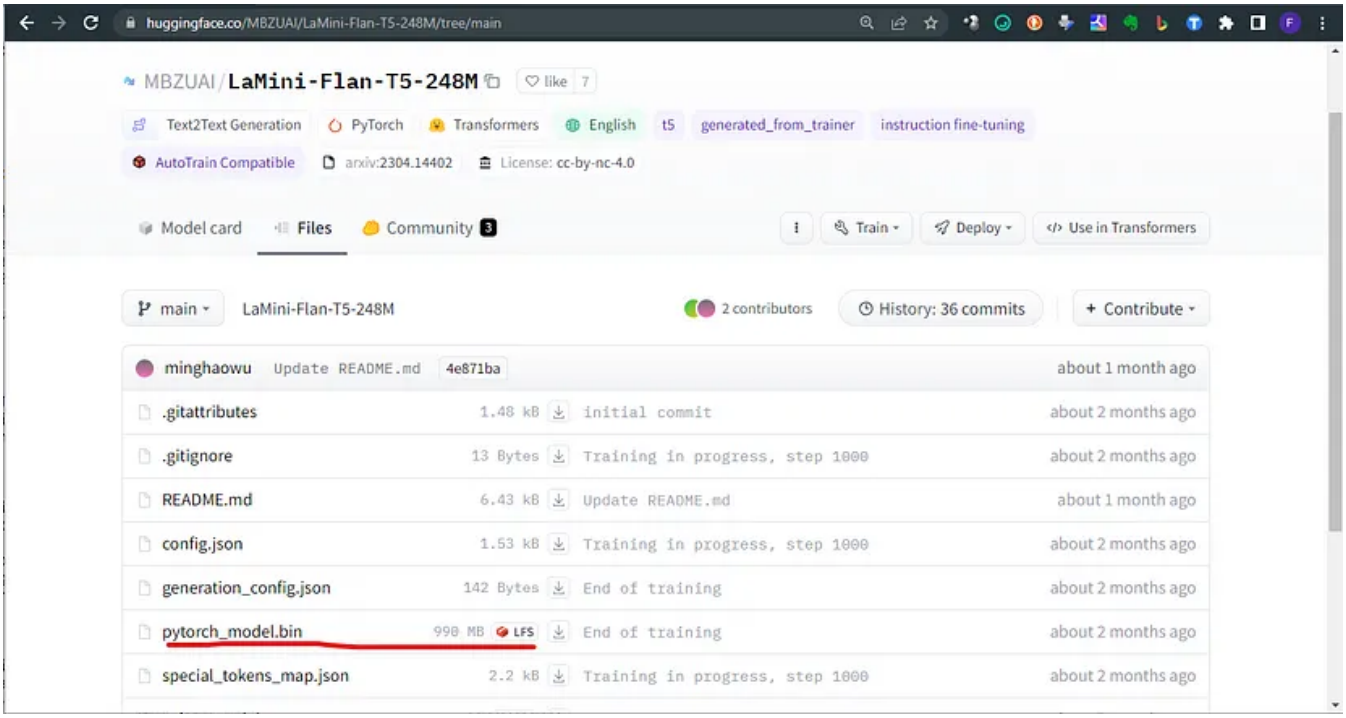
> ⚠️ *With the plethora of large language models (LLMs) and chatbots being released week upon week, often with grandiose claims of their performance, it can be hard to filter out the genuine progress that is being made by the open-source community and which model is the current state of the art. The 🤗 Open LLM Leaderboard aims to track, rank and evaluate LLMs and chatbots as they are released.*

Check on the best hits your favorite and try it with the inference API



Hugging Face Leaderboard

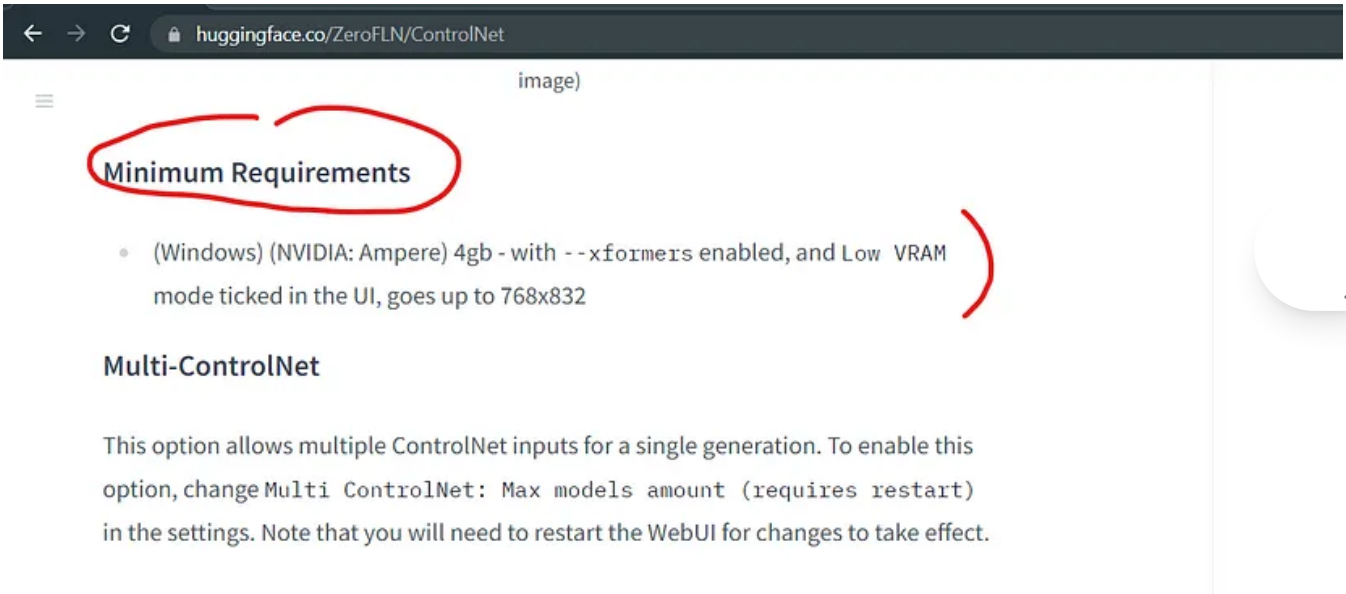
7. **Your hardware is the constraint:** if you don't have a GPU you must stick to small models. Go to the file section and look at the weight of the .bin file.



https://huggingface.co/MBZUAI/LaMini-Flan-T5-248M/tree/main Files section

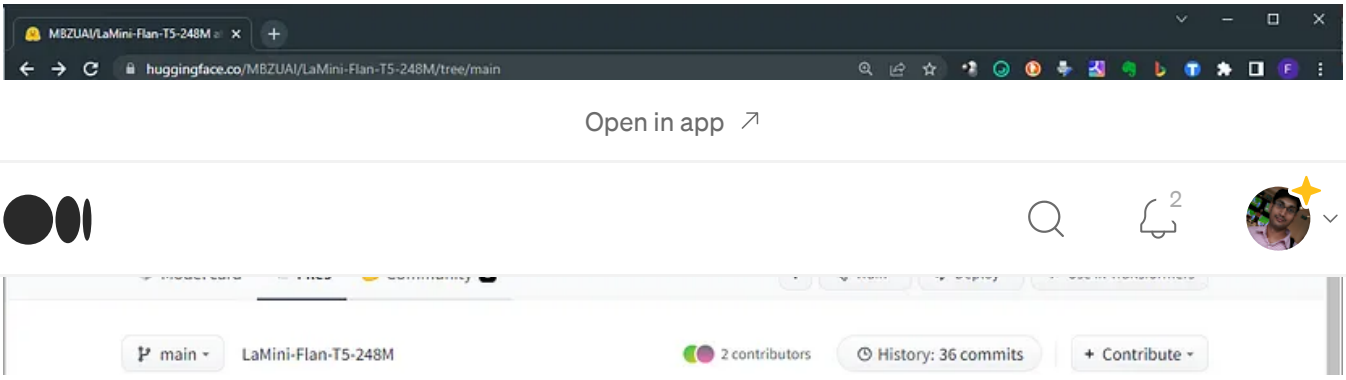
This is my favorite model: consider that if you run on CPU only, **your free VRAM required is doubled the size of the `model.bin` file.**

Sometimes in the model card it is also mentioned the minimum spec required.



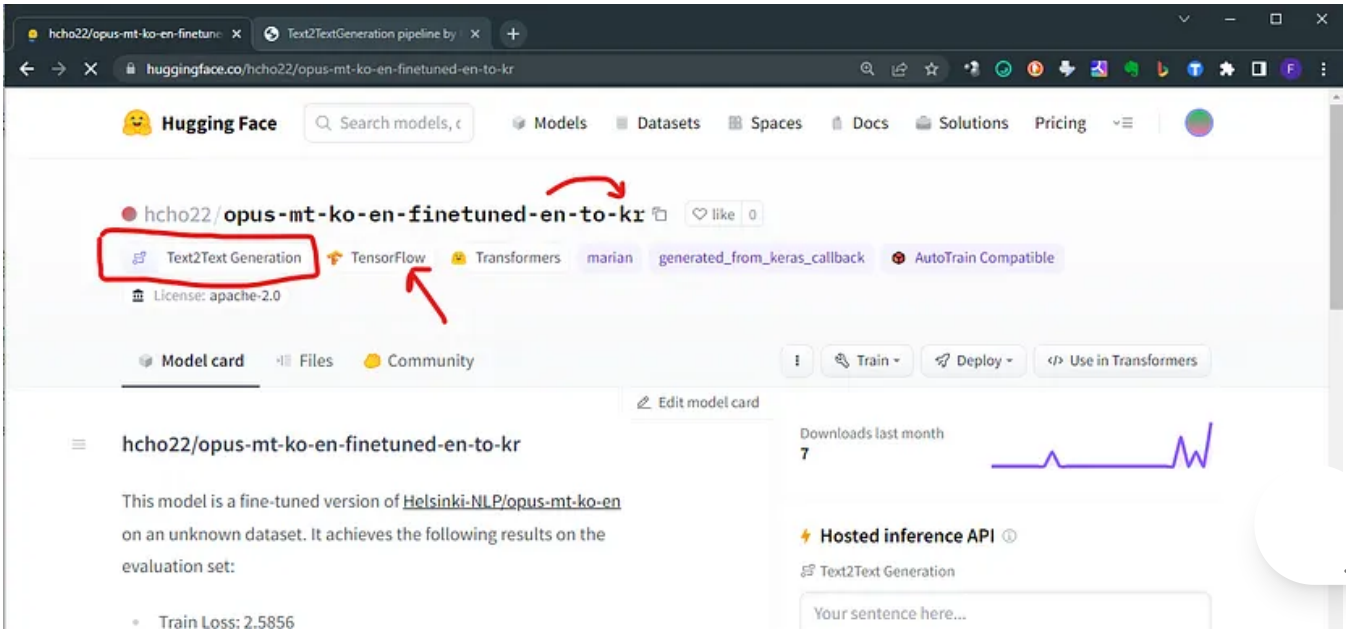
Requirements for https://huggingface.co/ZeroFLN/ControlNet

8. Pick your model based on the task you want to perform! You cannot have on your PC ChatGPT... But you can have multiple small models that perform different task. A summarizer, a text generetor, a translator and so on. You find the tasks related to the models on the model card page itself.



MBZUAI/LaMini-Flan-T5–248M can do text2text generation...

The one here below is a model dedicated to translations: specifically from english to korean. Remember that in Hugging Face Hub translation models are usually working only for a pair, and a specific order. This one is from English to Korean (en-to-ko)



https://huggingface.co/hcho22/opus-mt-ko-en-finetuned-en-to-kr

The model card is telling us a lot of things:

```
1. The base model: marian
2. The Machine Leraning framework: Tensorflow
3. The specialized task: Text2Text-generation
```

9. In case the **weights** of the model are in `.h5` format you need to install tensorflow (like in the example above)

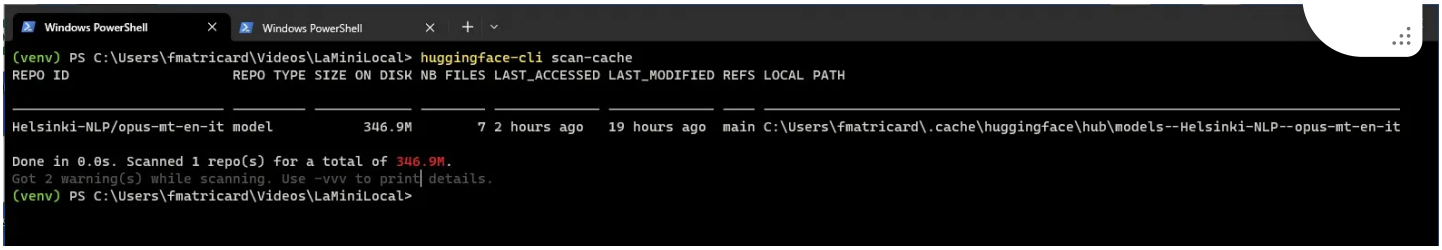
```
pip install tensorflow
```

Remember that you need to specify the *tensorflow* framework when you call your *Model*, with `from_tf=True`. It is something like this:

```
repo_id = “hcho22/opus-mt-ko-en-finetuned-en-to-kr”
model_ttKR = AutoModelForSeq2SeqLM.from_pretrained(repo_id, from_tf=True)
```

10. The models that are downloaded automatically when you declare them, are stored in a special *cached* directory on your Computer. This means that you can copy/paste them in your project folder! Run in your terminal this command to get the list and the paths of all cached models. Learn more here.

```
huggingface-cli scan-cache
```



run on terminal: huggingface-cli scan-cache

4. 🦜🔗 LangChain and Text2Text-generation

LangChain is a library that helps developers build applications powered by large language models (LLMs). It provides a framework for connecting LLMs to other sources of data, such as the internet or personal files, and allows developers to chain together multiple commands to create more complex applications.

I whish I knew...

11. Start immediately to work with LangChain: the docs and tutorials are really good! 🦜🔗 can be used to build applications powered by LLMs such as chatbots, question-answering systems, summarization systems, and code generation systems. It is a powerful tool that is easy to use and provides a wide range of features.

12. Modern models (like the T5 family) have a pipeline called the **Text2TextGeneration**: this is the pipeline for text to text generation using seq2seq models. **Text2TextGeneration is a single pipeline for all kinds of NLP tasks** like Question answering, sentiment classification, question generation, translation, paraphrasing, summarization, etc.

Conclusions

Working with Hugging Face's Language Models (LLMs) can be a challenging yet rewarding experience for AI enthusiast like you and me.

After exploring the 12 things I wish I knew before starting to work with Hugging Face LLMs, I hope you have some good swiss-knife hacks in your pockets. Enjoy testing the amazing resources of the Hugging Face ecosystem.

Ready to start?

You can check yourself how to do your first AI generation

Your first AI generation

How to start with open source LLM: from knowledge to application.

artificialcorner.com

If you want to start with a Google Colab project and learn cool NLP tasks with Hugging Face models, have a look here:

Answering Question About your Documents Using LangChain (and NOT OpenAI)

How to use Hugging Face LLM (open source LLM) to talk to your documents, pdfs and also articles from webpages.

artificialcorner.com

If you feel ready to run it on your computer, follow these 7 steps. Do not worry, CPU only is enough!

A 7 steps guide to install and run LaMini-LM on your computer

LaMini is your small ChatGPT running on a CPU: learn how to ask questions with python

artificialcorner.com

If this story provided value and you wish to show a little support, you could:

1. Clap 50 times for this story (this really, really helps me out)
2. Sign up for a Medium membership using my link — ($5/month to read unlimited Medium stories)
3. Follow me on Medium
4. Read my latest articles https://medium.com/@fabio.matricardi

Artificial Intelligence | Python | Local Gpt | Hugging Face | Transformers

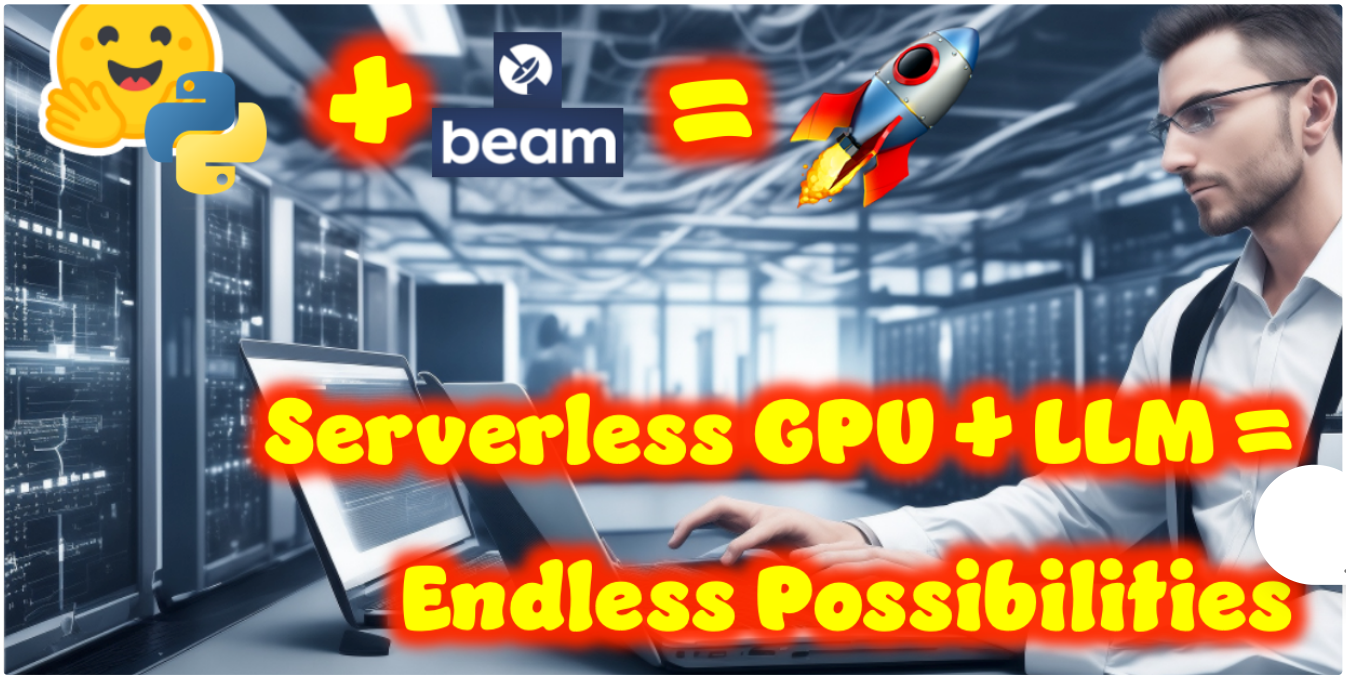
Following

Written by Fabio Matricardi

2.7K Followers · Writer for Artificial Corner

passionate educator, curious industrial automation engineer. Learning Leadership and how to build my own AI. contact me at fabio.matricardi@gmail.com

More from Fabio Matricardi and Artificial Corner



Fabio Matricardi in Artificial Corner

Compute without Constraints: Serverless GPU + LLM = Endless Possibilities

Spark conversations through the cloud with serverless GPUs powering your most advanced Hugging Face large language models.

17 min read · Aug 29

280 1

The PyCoach in Artificial Corner

My Honest Review of Some Paid AI Tools I've Used So Far

Here's why I canceled and kept some of these subscriptions so far.

✦ · 6 min read · Aug 15

2.5K 36

Diana Dovgopol in Artificial Corner

I Used ChatGPT (At Work) for 6 Months. Here's How to 10X Your Productivity

ChatGPT can help automate the most boring parts of anyone's job.

✦ · 7 min read · Jun 7

2K 28

Fabio Matricardi in Artificial Corner

Your Local LLM on your Network with FastAPI—part 2

Learn how to run your local FREE Hugging Face Language Model with Python, FastAPI and Streamlit.

10 min read · Aug 3

534 2

Recommended from Medium

Nick Hilton

The End of the Subscription Era is Coming

You’re overpaying for your porn (and journalism)

10 min read · Aug 30

9.2K 172

Sachin Kulkarni

Generative AI with Enterprise Data

Create business value add Enterprise knowledge to Large Language Models

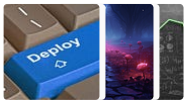
6 min read · Jul 25

174 5

Lists

ChatGPT

21 stories · 151 saves



Predictive Modeling w/ Python

20 stories · 380 saves

ChatGPT prompts

24 stories · 372 saves

AI Regulation

6 stories · 119 saves

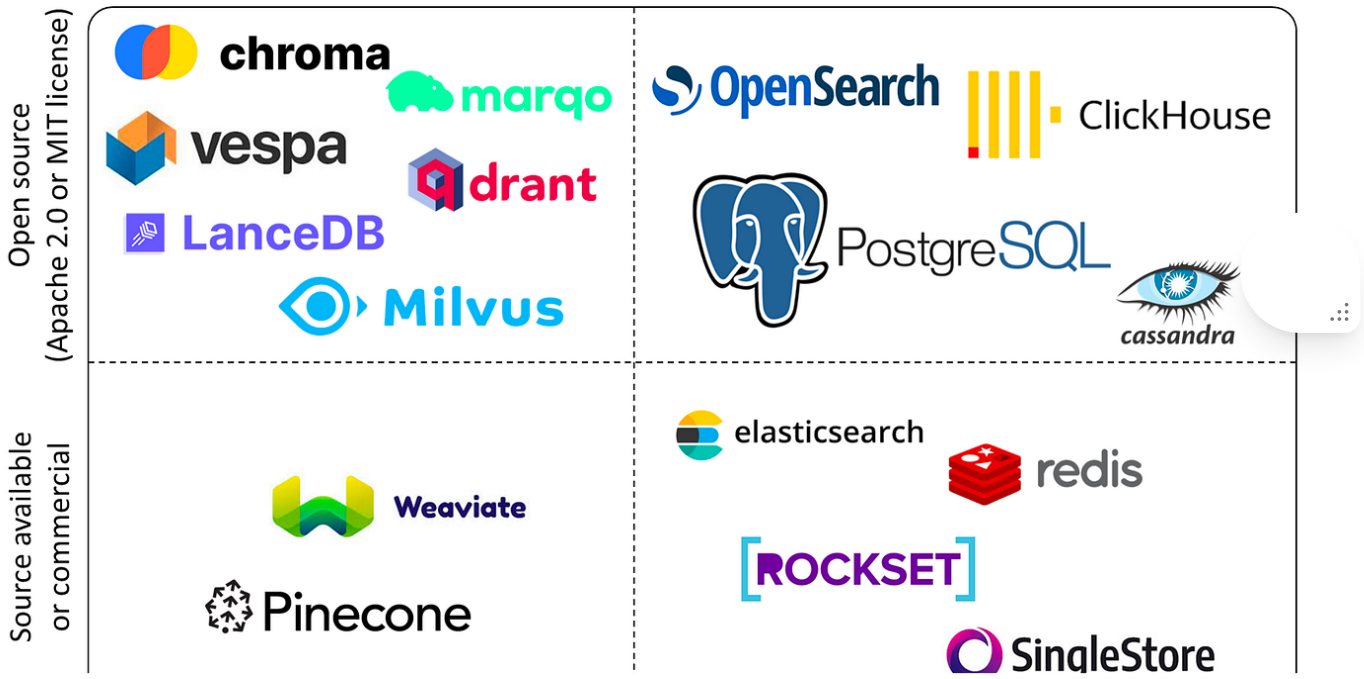
Moshe Sipper, Ph.D. in The Generator

Jailbreaking Large Language Models: If You Torture the Model Long Enough, It Will Confess!

A Cautionary Tale...

5 min read · Sep 7

609 7



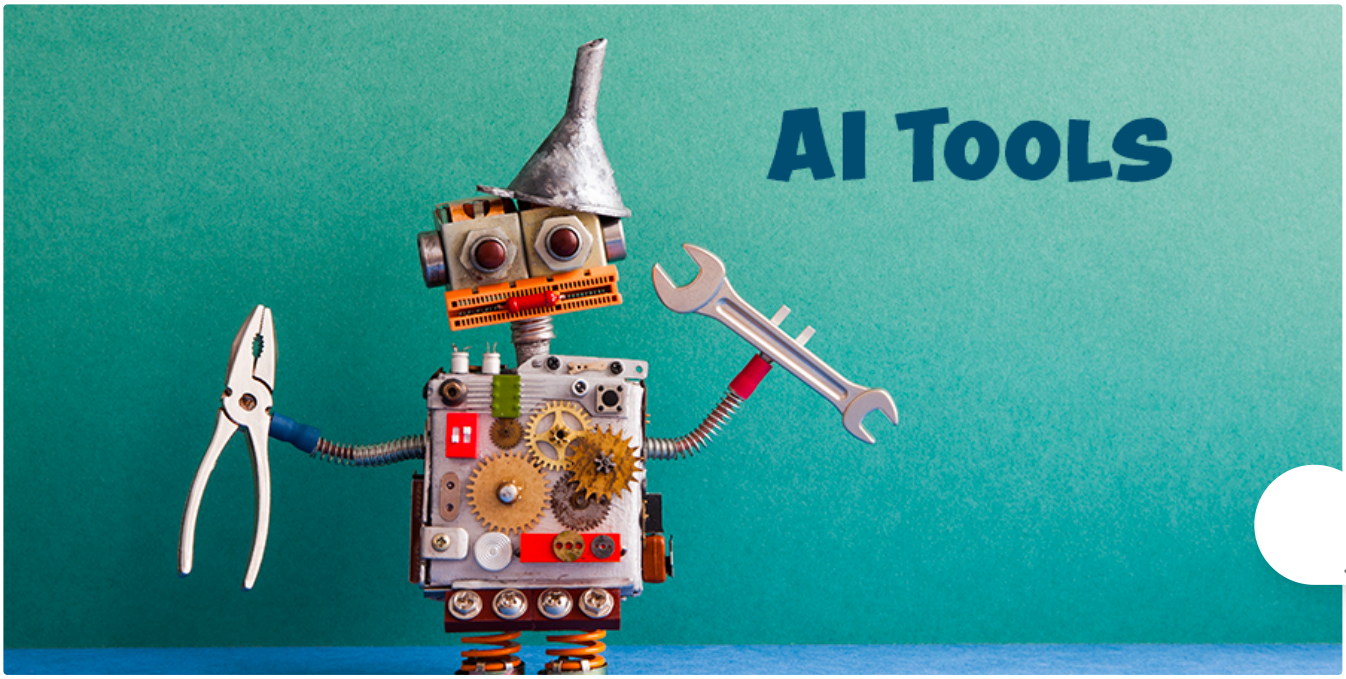
Yingjun Wu in Data Engineer Things

Why You Shouldn't Invest In Vector Databases?

Delving into the vector database field from three perspectives: technology, applications, and the market.

8 min read · Jun 7

702 14



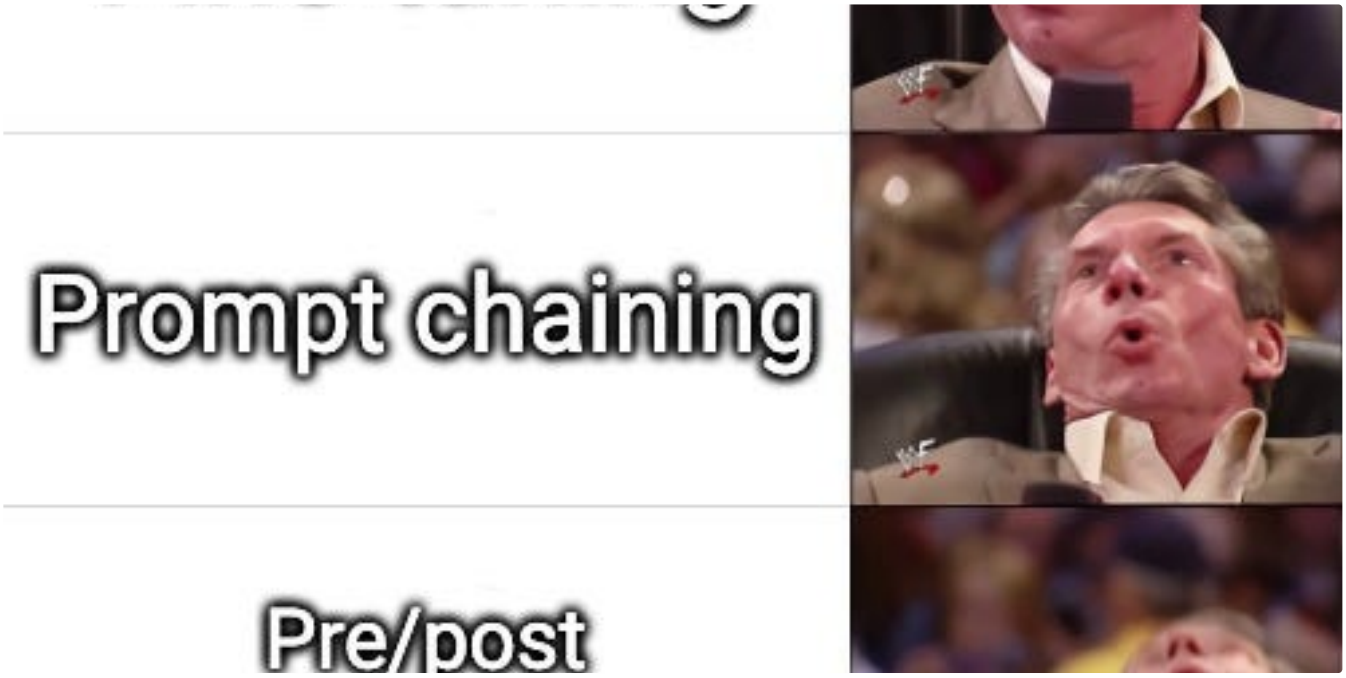
Digital Giraffes

7 Awesome and Free AI Tools You Should Know

We collected 7 free artificial intelligence(AI) tools, most of them easy to use and some more sophisticated... like building ML models.

5 min read · Nov 17, 2022

7K 164



Jason Fan

How we cut the rate of GPT hallucinations from 20%+ to less than 2%

tl;dr: Instead of fine-tuning, we used a combination of prompt chaining and pre/post-processing to reduce the rate of hallucinations by an...

4 min read · Mar 8

645 11