



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DIPARTIMENTO DI
SCIENZE STATISTICHE "PAOLO FORTUNATI"



10 - Hugging Face and Transformers Library

Giovanni Della Lunga
giovanni.dellalunga@unibo.it

Halloween Conference in Quantitative Finance

Bologna - October 25-26-27, 2023

Introduction

What is Hugging Face?

- Hugging Face is a community specializing in Natural Language Processing (NLP) and artificial intelligence (AI).
- Founded in 2016, the company has made significant contributions to the field of NLP by democratizing access to state-of-the-art machine learning models and tools.



Hugging Face

What is Hugging Face?

- Hugging Face has a strong community focus.
- They provide a platform where researchers and developers can share their trained models, thereby fostering collaboration and accelerating progress in the field.



Hugging Face

Hugging Face Library

Hugging Face Transformers as a Library

- Hugging Face's Transformers library is an open-source library for NLP and machine learning.
- It provides a wide variety of pre-trained models and architectures like BERT, GPT-X, T5, and many others.
- The library is designed to be highly modular and easy to use, allowing for the quick development of both research and production projects.
- It supports multiple languages and tasks like text classification, question-answering, text generation, translation, and more.

Transformers

- Transformers is a Python library that makes downloading and training state-of-the-art ML models easy.
- Although it was initially made for developing language models, its functionality has expanded to include models for computer vision, audio processing, and beyond.
- Two big strengths of this library are: 1) it easily integrates with Hugging Face's Models, Datasets, and Spaces repositories, and 2) the library supports other popular ML frameworks such as PyTorch and TensorFlow.
- This results in a simple and flexible all-in-one platform for downloading, training, and deploying machine learning models and apps.

Models

- There is a massive repository of pre-trained models available on Hugging Face (more than 250000 at the time of writing this).
- Almost all these models can be easily used via Transformers.

Models

The screenshot shows the Hugging Face website interface. At the top, there's a navigation bar with tabs for 'Models', 'Datasets', 'Spaces', 'Docs', 'Solutions', 'Pricing', 'Log In', and 'Sign Up'. The 'Models' tab is selected and highlighted with a red box. Below the navigation bar, the main content area is divided into two sections. On the left, there's a large hero section with the Hugging Face logo (a yellow emoji) and the text 'The AI community building the future.' followed by 'The platform where the machine learning community collaborates on models, datasets, and applications.' On the right, there's a list of models under the heading 'Models 469,541'. The list includes various models such as 'meta-llama/llama-2-70b', 'stabilityai/stable-diffusion-xl-base-0.9', 'openchat/openchat', 'lillyasviel/ControlNet-v1-1', 'cerspense/zeroscope_v2_XL', 'meta-llama/llama-2-13b', 'tiiuae/falcon-40b-instruct', 'WizardLM/WizardCoder-15B-V1.0', 'CompVis/stable-diffusion-v1-4', and 'stabilityai/stable-diffusion-2-1'. Each model entry shows its name, a brief description, and some statistics like 'Updated 4 days ago' and '1.25k' likes.

Models

Hugging Face Search models, datasets, users...

Models 334,298 Filter by name

new Full-text search 11 Sort: Trending

Tasks Libraries Datasets Languages Licenses Other

Filter Tasks by name

Multimodal

- Feature Extraction Text-to-Image
- Image-to-Text Text-to-Video
- Visual Question Answering
- Document Question Answering
- Graph Machine Learning

Computer Vision

- Depth Estimation Image Classification
- Object Detection Image Segmentation
- Image-to-Image Unconditional Image Generation
- Video Classification Zero-Shot Image Classification

Natural Language Processing

- Text Classification Token Classification
- Table Question Answering Question Answering
- Zero-Shot Classification Translation
- Summarization Conversational
- Text Generation Text2Text Generation

Models

- microsoft/phi-1.5**
Text Generation · Updated 1 day ago · ± 25.5k · ♥ 540
- Deci/DeciLM-6b**
Text Generation · Updated 3 days ago · ± 401 · ♥ 130
- tiiuae/falcon-180B-chat**
Text Generation · Updated 11 days ago · ± 6.99k · ♥ 348
- meta-llama/Llama-2-7b**
Text Generation · Updated Jul 19 · ± 2.49k
- coqui/XTTS-v1**
Text-to-Speech · Updated 2 days ago · ± 4 · ♥ 83
- microsoft/phi-1**
Text Generation · Updated 6 days ago · ± 1.95k · ♥ 75
- baichuan-inc/Baichuan2-13B-Chat**
Text Generation · Updated 5 days ago · ± 280k · ♥ 158
- warp-ai/wuerstchen**
Text-to-Image · Updated 4 days ago · ± 2.48k · ♥ 55
- tiiuae/falcon-180B**
Text Generation · Updated 11 days ago · ± 45.3k · ♥ 647
- stabilityai/stable-diffusion-xl-base-1.0**
Text-to-Image · Updated 13 days ago · ± 1.39M · ♥ 2.65k
- lillyasviel/sd_control_collection**
Updated 8 days ago · ♥ 350
- Deci/DeciLM-6b-instruct**
Text Generation · Updated about 1 hour ago · ± 298 · ♥ 89
- meta-llama/Llama-2-7b-chat-hf**
Text Generation · Updated Aug 9 · ± 581k · ♥ 1.17k
- monster-labs/control_v1p_sd15_qrcode_monster**
Updated Jul 21 · ± 2.7k · ♥ 466
- THUDM/chatglm2-6b**
Updated 9 days ago · ± 468k · ♥ 1.7k
- meta-llama/Llama-2-7b-chat-hf**
Text Generation · Updated Aug 9 · ± 199k · ♥ 1.32k

Models

- To see what navigating the repository looks like, let's consider an example.
- Say we want a model that can do text generation, but we want it to be available via the Transformers library so we can use it in one line of code (as we did above).
- We can easily view all models that fit these criteria using the "Tasks" and "Libraries" filters.
- A model that meets these criteria is the newly released Llama 2. More specifically, Llama-2-7b-chat-hf, which is a model in the Llama 2 family with about 7 billion parameters, optimized for chat, and in the Hugging Face Transformers format.
- We can get more information about this model via its model card, which is shown in the next slide.

Models

The screenshot shows the Hugging Face model page for **meta-llama/Llama-2-7b-chat-hf**. Red arrows and text labels highlight key features:

- Org and Model Name:** Points to the model identifier `meta-llama/Llama-2-7b-chat-hf` in the header.
- Model Tags:** Points to the horizontal list of tags including `Text Generation`, `PyTorch`, `Safetensors`, `Transformers`, `English`, `llama`, `facebook`, `meta`, `llama-2`, `text-generation-inference`, and `arxiv:2307.09288`.
- Model Info:** Points to the main description of the Llama 2 model.
- QuickStarts:** Points to the `Train`, `Deploy`, and `Use in Transformers` buttons.

Model Info: Llama 2 is a collection of pretrained and fine-tuned generative text models ranging in scale from 7 billion to 70 billion parameters. This is the repository for the 7B fine-tuned model, optimized for dialogue use cases and converted for the Hugging Face Transformers format. Links to other models can be found in the index at the bottom.

QuickStarts: The page includes buttons for `Train`, `Deploy`, and `Use in Transformers`, along with a `QuickStarts` section.

Models

huggingface.co/meta-llama/Llama-2-7b

Hugging Face Search models, datasets, users...

Models Datasets Spaces Docs Solutions Pricing Log In Sign Up

meta-llama / **Llama-2-7b** like 2.49k

Text Generation PyTorch English facebook meta llama llama-2 arxiv:2307.09288

Model card Files and versions

Access Llama 2 on Hugging Face

This is a form to enable access to Llama 2 on Hugging Face **after you have been granted access from Meta**. Please visit the [Meta website](#) and accept our license terms and acceptable use policy before submitting this form. Requests will be processed in 1-2 days.

Your Hugging Face account email address **MUST** match the email you provide on the Meta website, or your request will not be approved.

Log in or Sign Up to review the conditions and access this model content.

Llama 2

Llama 2 is a collection of pretrained and fine-tuned generative text models ranging in scale from 7 billion to 70 billion parameters. This is the repository for the 7B pretrained model. Links to other models can be found in the index at the bottom.

Downloads last month 0

Hosted inference API

Text Generation

Inference API has been turned off for this model.

Spaces using meta-llama/Llama-2-7b 11

- qiantong-xu/toolbench-leaderboard
- AironHeart/llama2
- bhaskartripathi/llama-2-70b-chatbot
- hoyinli/demo-app
- mikeee/gradio-chatinterface
- Araeynn/llama2
- osanseviero/streaming-example
- realchenyuy/llama2-playground
- KingPinX/llbot-1
- scp4950/graph
- agbhu/meta-llama-Llama-2-7b-hf

Model Details

Models

Hugging Face Search models, datasets, users...

Models Datasets Spaces Docs Solutions Pricing

meta-llama / **Llama-2-7b** like 2.49k

Text Generation PyTorch English facebook meta llama llama-2 arxiv:2307.09288

Model card Files and versions Use with library

Gated model You have been granted access to this model

Llama 2

Llama 2 is a collection of pretrained and fine-tuned generative text models ranging in scale from 7 billion to 70 billion parameters. This is the repository for the 7B pretrained model. Links to other models can be found in the index at the bottom.

Model Details

Note: Use of this model is governed by the Meta license. In order to download the model weights and tokenizer, please visit the [website](#) and accept our License before requesting access here.

Meta developed and publicly released the Llama 2 family of large language models (LLMs), a collection of pretrained and fine-tuned generative text models ranging in scale from 7 billion to 70 billion parameters. Our fine-tuned LLMs, called Llama-2-Chat, are optimized for dialogue use cases. Llama-2-Chat models outperform open-source chat models on most benchmarks we tested, and in our human evaluations for helpfulness and safety, are on par with some popular closed-source

Downloads last month
0

Hosted inference API

Text Generation
Inference API has been turned off for this model.

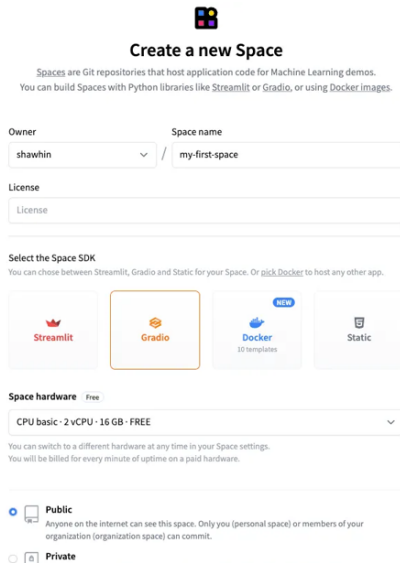
Spaces using meta-llama/Llama-2-7b 11

- qiantong-xu/toolbench-leaderboard
- AironHeart/llama2
- bhaskartripathi/Llama-2-70b-chatbot
- hoyinli/demo-app
- mikeel/gradio-chatinterface
- Araeynn/llama2
- osanseviero/streaming-example
- realchengyu/llama2-playground
- KingPinX/ibot-1
- scp4550/grah
- agbhu/meta-llama-Llama-2-7b-hf

Hugging Face Spaces

Hugging Face Spaces

- To go one step further, we can quickly deploy this UI via Hugging Face Spaces.
- These are Git repositories hosted by Hugging Face and augmented by computational resources.



Create a new Space

Spaces are Git repositories that host application code for Machine Learning demos. You can build Spaces with Python libraries like [Streamlit](#) or [Gradio](#), or using [Docker images](#).

Owner: shawhin / Space name: my-first-space

License: License

Select the Space SDK

You can choose between [Streamlit](#), [Gradio](#) and [Static](#) for your Space. Or [pick Docker](#) to host any other app.

Streamlit Gradio Docker (NEW) Static

Space hardware: Free

CPU basic · 2 vCPU · 16 GB · FREE

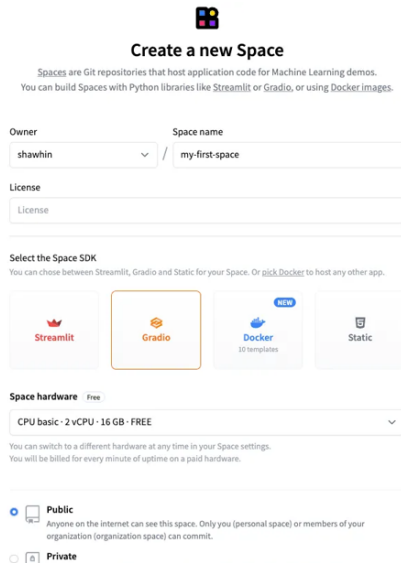
You can switch to a different hardware at any time in your Space settings. You will be billed for every minute of uptime on a paid hardware.

☒ Public
Anyone on the internet can see this space. Only you (personal space) or members of your organization (organization space) can commit.

☐ Private
Only you (personal space) or members of your organization (organization space) can see and

Hugging Face Spaces

- To make a new Space, we first go to the Spaces page and click "Create new space".
- Then, configure the Space by giving it the name e.g. "my-first-space" and selecting Gradio as the SDK. Then hit "Create Space".



Create a new Space

Spaces are Git repositories that host application code for Machine Learning demos. You can build Spaces with Python libraries like [Streamlit](#) or [Gradio](#), or using [Docker images](#).

Owner: shawhin / Space name: my-first-space

License: License

Select the Space SDK

You can choose between Streamlit, Gradio and Static for your Space. Or [pick Docker](#) to host any other app.

Streamlit Gradio Docker 10 templates Static

Space hardware: Free

CPU basic · 2 vCPU · 16 GB · FREE

You can switch to a different hardware at any time in your Space settings. You will be billed for every minute of uptime on a paid hardware.

☒ Public
Anyone on the internet can see this space. Only you (personal space) or members of your organization (organization space) can commit.

☐ Private
Only you (personal space) or members of your organization (organization space) can see and

Hugging Face Spaces

The screenshot shows the Hugging Face Spaces homepage. At the top, there's a navigation bar with links to Models, Datasets, Spaces, Docs, Solutions, and Pricing. Below this is a search bar and a 'Create new Space' button. The main section is titled 'Spaces' with the tagline 'Discover amazing ML apps made by the community!'. There's a 'Spaces of the week' section featuring a grid of eight featured Spaces, each with a title, a brief description, a creator, and a timestamp. The featured Spaces are: LoRA Roulette (multimodalart, 1 day ago), Blind Chat (mithril-security, 3 days ago), Upside-Down-Diffusion (AP123, 3 days ago), DeciDiffusion-v1-0 (Deci, 19 days ago), Nougat Transformers (hf-vision, 6 days ago), Stable Diffusion 2-1 (stabilityai, 17 days ago), Explore Clinical & Biomedical Language Models (hf4h, 17 days ago), and Lilac (lilacai, 5 days ago).

Space Name	Creator	Running on	Heart Count	Time Ago
LoRA Roulette	multimodalart	ZE80	98	1 day ago
Blind Chat	mithril-security	T4	81	3 days ago
Upside-Down-Diffusion	AP123	T4	128	3 days ago
DeciDiffusion-v1-0	Deci	A10G	66	19 days ago
Nougat Transformers	hf-vision	T4	54	6 days ago
Stable Diffusion 2-1	stabilityai	CPU UPGRADE	9.12k	17 days ago
Explore Clinical & Biomedical Language Models	hf4h	CPU UPGRADE	83	17 days ago
Lilac	lilacai	CPU UPGRADE	26	5 days ago

Hugging Face Spaces

- Next, we need to upload **app.py** and **requirements.txt** files to the Space.
- **The app.py file houses the code we used to generate the Gradio UI, and the requirements.txt file specifies the app's dependencies.**
- Finally, we push the code to the Space just like we would to GitHub. The end result is a public application hosted on Hugging Face Spaces.

Hugging Face Spaces

The screenshot shows a web browser window with the URL `huggingface.co/spaces/gdlunga/halloween-unibo-2023/tree/main`. The page displays the Hugging Face interface with a search bar and navigation links for Models, Datasets, Spaces, Docs, Solutions, Pricing, and a user profile. Below the navigation bar, the space name `gdlunga/halloween-unibo-2023` is shown with a 'Like' button and a 'Running' status indicator. The main content area shows a file tree for the `main` branch of the `halloween-unibo-2023` space, listing files with their sizes and commit history.

File	Size	Commit	Time
<code>gdlunga</code>		Create requirements.txt	1 minute ago
<code>.gitattributes</code>	1.52 kB	initial commit	about 1 hour ago
<code>README.md</code>	239 Bytes	initial commit	about 1 hour ago
<code>app.py</code>	527 Bytes	Create app.py	2 minutes ago
<code>requirements.txt</code>	25 Bytes	Create requirements.txt	1 minute ago

Hugging Face Spaces

The screenshot shows a web browser window with the URL `huggingface.co/spaces/gdlunga/halloween-unibo-2023/blob/main/requirements.txt`. The page is the Hugging Face interface for a Space. At the top, there's a navigation bar with the Hugging Face logo, a search bar, and links to Models, Datasets, Spaces, Docs, Solutions, Pricing, and a user profile. Below this, the Space name `gdlunga/halloween-unibo-2023` is displayed with a 'Stopped' status. The main content area shows the file `requirements.txt` with a 'main' branch selector. The file content is as follows:

```
1. gradio
2. transformers
3. torch
```

Metadata for the file includes the author `gdlunga`, the file name `Create requirements.txt`, a commit hash `4b803f7`, and a timestamp of `8 days ago`. Below the file content, there are options to view the raw file, history, blame, edit, delete, and a 'No virus' status. The file size is noted as `25 Bytes`.

Hugging Face Spaces

The screenshot shows a web browser window displaying a Hugging Face Space. The URL is `huggingface.co/spaces/gdlunga/halloween-unibo-2023/blob/main/app.py`. The page header includes the Hugging Face logo, a search bar, and navigation links for Models, Datasets, Spaces, Docs, Solutions, Pricing, and a user profile. Below the header, the Space name `gdlunga/halloween-unibo-2023` is shown with status indicators (Like, Stopped, Logs). The main content area displays the `main` branch of the `halloween-unibo-2023/app.py` file. The code is a Python script that uses the `transformers` library to create a question-answering pipeline with a RoBERTa model. It defines a function `answer_question` that takes a question and context, tokenizes them, and uses the pipeline to generate an answer. The script is launched as a Gradio interface.

```

1 from transformers import AutoModelForQuestionAnswering, AutoTokenizer, pipeline
2 import gradio as gr
3 import ast
4
5 mdl_name = "deepset/roberta-base-squad2"
6 my_pipeline = pipeline('question-answering', model=mdl_name, tokenizer = mdl_name)
7
8 def answer_question(question, context):
9     text = '{"question": ' + question + ', "context": ' + context + '}'
10    di = ast.literal_eval(text)
11    response = my_pipeline(di)
12    return response
13
14 gr.Interface(answer_question, inputs=["text", "text"], outputs = "text").launch()

```

Hugging Face Spaces

The screenshot shows a web browser window with the URL `huggingface.co/spaces/gdlunga/halloween-unibo-2023/edit/main/app.py`. The code editor displays the following Python code:

```

5
6 md1_name = "deepset/roberta-base-squad2"
7 my_pipeline = pipeline('question-answering', model=md1_name, tokenizer = md1_name)
8
9 def answer_question(question, context):
10     text = '{"question": ' + question + ', 'context': ' + context + '}'
11     di = ast.literal_eval(text)
12     response = my_pipeline(di)
13     return response
14
15 grad.Interface(answer_question, inputs=["text", "text"], outputs = "text").launch()

```

Below the code editor, there are two radio buttons for committing changes:

- ☒ Commit directly to the main branch
- ☐ Open as a pull request to the main branch

The "Commit changes" section includes a text input field with the value "Update app.py". Below this, there are tabs for "Edit" and "Preview". The "Edit" tab is active, showing a text area for an extended description. A blue arrow points to the "Commit changes to main" button at the bottom left of the interface.

Hugging Face Spaces

The screenshot shows a web browser window displaying a Hugging Face Space. The address bar shows the URL `huggingface.co/spaces/gdlunga/halloween-unibo-2023`. The page header includes the Space name, a 'Like' button (0 likes), a 'Run' button (labeled 'RunIng'), and a 'Logs' button. The main interface has a 'question' input field, a 'context' input field, and a 'Submit' button. To the right of the input fields is an 'output' area. At the bottom of the page, there is a footer that reads 'Use via API' and 'Built with Gradio'.

question

context

Clear Submit

output

Use via API Built with Gradio

Managing secrets and environment variables

- If your app requires environment variables (for instance, secret keys or tokens), do not hard-code them inside your app!
- Instead, go to the ****Settings**** page of your Space repository and add a new variable or secret.
- Use variables if you need to store non-sensitive configuration values and secrets for storing access tokens, API keys, or any sensitive value or credentials.

Repository variables and secrets ⓘ

Variables Public

✕ MODEL_REPO_ID · Updated less than a minute ago

View

Delete

Secrets Private

🔒 HF_TOKEN · Updated less than a minute ago

Replace

Delete

🔒 OPENAI_API_KEY · Updated less than a minute ago

Replace

Delete

Managing secrets and environment variables

The screenshot shows a web browser window displaying the Hugging Face profile of Giovanni Della Lunga. The browser's address bar shows the URL `huggingface.co/gdlunga`. The profile page includes a profile picture of a golden robot head, the name **Giovanni Della Lunga**, and the username `gdlunga`. Below the profile information, there are links for "polyhedron-gdl", "Research interests" (None yet), and "Organizations" (None yet). The main section is titled "Spaces 4" and lists four Spaces:

- Halloween Unibo 2023**: A Space with a bar chart icon.
- Chat Pdf 1**: A Space with a fish icon, marked as "private" and "Stopped".
- Llama 2**: A Space with a trophy icon, marked as "private" and "Stopped".
- Transformers 1**: A Space with a lightning bolt icon, marked as "Stopped".

At the bottom of the Spaces list, there are sections for "Models" (None yet) and "Datasets". The browser's address bar at the bottom shows the URL `https://huggingface.co/spaces/gdlunga/halloween-unibo-2023`.

Managing secrets and environment variables

The screenshot shows the Hugging Face Spaces interface for the 'halloween-unibo-2023' space. The 'Settings' tab is highlighted with a red box, and a blue arrow points to it from the right. The interface shows the space is running, has 1 contributor, and 3 commits. A table lists the files:

File	Size	Action	Commit	Time
.gitattributes	1.52 kB	Download	initial commit	15 days ago
README.md	239 Bytes	Download	initial commit	15 days ago
app.py	527 Bytes	Download	Create app.py	15 days ago
requirements.txt	25 Bytes	Download	Create requirements.txt	15 days ago

Managing secrets and environment variables

huggingface.co/spaces/gdlunga/halloween-unibo-2023/settings

Webmail Register it YouTube Netflix Prime Video Google Drive Unibo Polyhedron Trello Overleaf https://projecthub...

useful if your Space uses GPU and has an out of memory error since this will fully stop the existing Space before restarting.

Change space visibility Make private

This space is currently **public**. Anyone on the internet can see this space. Only you (personal space) or members of your organization (organization space) can commit.

Variables and secrets New variable New secret

Variables **Public**

No variables

Secrets **Private**

No secrets

Rename or transfer this space

All links to this space will automatically redirect to the new location, including git operations. However, to avoid confusion, we recommend updating any existing local clones to point to the new repository URL. To do so, you can use the following command:

```
git remote set-url origin [NEW_URL]
```

New owner: **gdlunga** / New name: **New space name**

[I understand, move this space](#)