# Module 3 – Hands-On Exercise

This exercise is adapted from the Hugging Face Educational Toolkit.

## A Tour of the Hugging Face Hub

### Why the Hub?

Hugging Face's "The Hub" is a central platform where anyone can share and explore models, datasets, and ML demos. A single company won't "*solve AI*", but rather, a culture of sharing knowledge and resources just might. Because of this, the Hub aims to build the most extensive collection of open-source models, datasets, and demos.

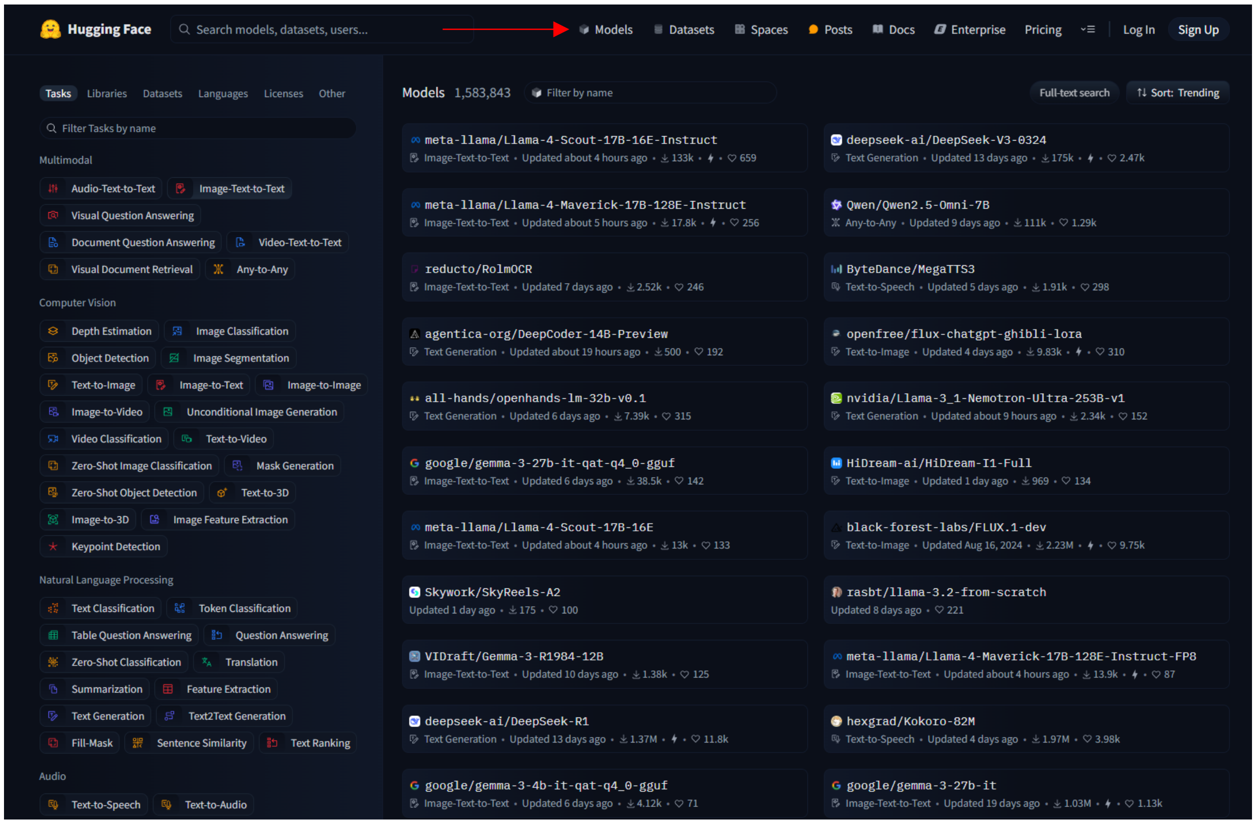
We gave you all the ingredients to make the transfer-learning omelet in the previous two exercises. To round out your skills, we must show you how to gather those ingredients yourself. The *Practicum AI* Beginner Series mentioned model zoos, repositories of model architectures, datasets, and pre-trained models. Hugging Face is just one such zoo, but its size and ease of use make it a *PracticumAI* favorite.

Here are some facts about the Hugging Face Hub:

- There are over a million public models!
- There are models for natural language processing, computer vision, audio, speech, reinforcement learning, and many more.
- There are language models for over 180 languages.
- Almost all modern ML libraries can leverage the Hub, from PyTorch and TensorFlow to advanced integrations with countless other libraries.

## Exploring Models

Let's kick off the exploration of models. You can access the models at hf.co/models. You will see the models sorted by "trending" by default (in the image below, the Sort menu is in the upper right-hand corner of the screen). As of this writing, the hottest model is the Llama-4-Scout. Undoubtedly, that will be different by the time you read this.



meta-llama/Llama-4-Scout-17B-16E-Instruct
Image-Text-to-Text • Updated about 4 hours ago • ↓ 133k • ⚡ • ♡ 659

deepseek-ai/DeepSeek-V3-0324
Text Generation • Updated 13 days ago • ↓ 175k • ⚡ • ♡ 2.47k

meta-llama/Llama-4-Maverick-17B-128E-Instruct
Image-Text-to-Text • Updated about 5 hours ago • ↓ 17.8k • ⚡ • ♡ 256

Qwen/Qwen2.5-Omni-7B
Any-to-Any • Updated 9 days ago • ↓ 111k • ♡ 1.29k

reducto/RolmOCR
Image-Text-to-Text • Updated 7 days ago • ↓ 2.52k • ♡ 246

ByteDance/MegaTTS3
Text-to-Speech • Updated 5 days ago • ↓ 1.91k • ♡ 298

agentica-org/DeepCoder-14B-Preview
Text Generation • Updated about 19 hours ago • ↓ 500 • ♡ 192

openfree/flux-chatgpt-ghibli-lora
Text-to-Image • Updated 4 days ago • ↓ 9.83k • ⚡ • ♡ 310

all-hands/openhands-lm-32b-v0.1
Text Generation • Updated 6 days ago • ↓ 7.39k • ♡ 315

nvidia/Llama-3_1-Nemotron-Ultra-253B-v1
Text Generation • Updated about 9 hours ago • ↓ 2.34k • ♡ 152

google/gemma-3-27b-it-qat-q4_0-gguf
Image-Text-to-Text • Updated 6 days ago • ↓ 38.5k • ♡ 142

HiDream-ai/HiDream-I1-Full
Text-to-Image • Updated 1 day ago • ↓ 969 • ♡ 134

meta-llama/Llama-4-Scout-17B-16E
Image-Text-to-Text • Updated about 4 hours ago • ↓ 13k • ♡ 133

black-forest-labs/FLUX.1-dev
Text-to-Image • Updated Aug 16, 2024 • ↓ 2.23M • ⚡ • ♡ 9.75k

Skywork/SkyReels-A2
Updated 1 day ago • ↓ 175 • ♡ 100

rasbt/llama-3.2-from-scratch
Updated 8 days ago • ♡ 221

VIDraft/Gemma-3-R1984-12B
Image-Text-to-Text • Updated 10 days ago • ↓ 1.38k • ♡ 125

meta-llama/Llama-4-Maverick-17B-128E-Instruct-FP8
Image-Text-to-Text • Updated about 4 hours ago • ↓ 13.9k • ⚡ • ♡ 87

deepseek-ai/DeepSeek-R1
Text Generation • Updated 13 days ago • ↓ 1.37M • ⚡ • ♡ 11.8k

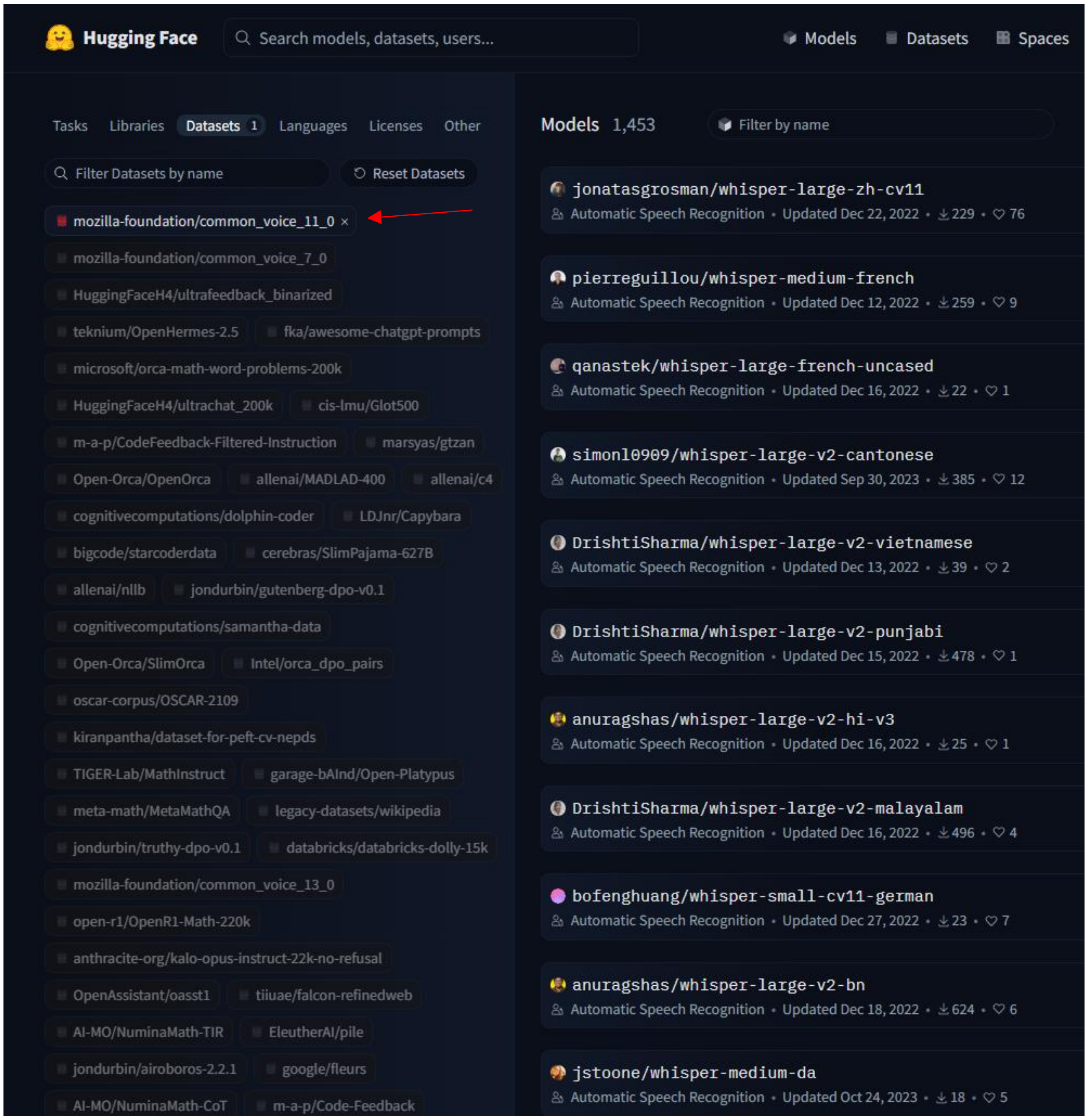
hexgrad/Kokoro-82M
Text-to-Speech • Updated 4 days ago • ↓ 1.97M • ♡ 3.98k

google/gemma-3-4b-it-qat-q4_0-gguf
Image-Text-to-Text • Updated 6 days ago • ↓ 4.12k • ♡ 71

google/gemma-3-27b-it
Image-Text-to-Text • Updated 19 days ago • ↓ 1.03M • ⚡ • ♡ 1.13k

On the top left, you'll see categories that you can use to filter the models. In the image above you can see that the Task category is selected. Here's a quick overview of the more critical model filtering categories:

- **Tasks:** There is support for dozens of tasks in different domains, such as multimodal, computer vision, natural language processing, audio, tabular, and more.
- **Libraries:** Although the Hub was originally for Hugging Face's own transformers models, it has integration with dozens of libraries. You can find PyTorch, Keras, spaCy, Scikit-learn models, and many more.
- **Datasets:** The Hub also hosts hundreds of thousands of datasets, as you'll find more about later. In this Model view, you can filter *models pre-trained on the dataset of interest.*



Hugging Face  Search models, datasets, users...  Models  Datasets  Spaces

Tasks  Libraries  Datasets 1  Languages  Licenses  Other

Filter Datasets by name  Reset Datasets

mozilla-foundation/common_voice_11_0 ×

mozilla-foundation/common_voice_7_0
HuggingFaceH4/ultrafeedback_binarized
teknium/OpenHermes-2.5
fka/awesome-chatgpt-prompts
microsoft/orca-math-word-problems-200k
HuggingFaceH4/ultrachat_200k
cis-lmu/Glot500
m-a-p/CodeFeedback-Filtered-Instruction
marsyas/gtzan
Open-Orca/OpenOrca
allenai/MADLAD-400
allenai/c4
cognitivecomputations/dolphin-coder
LDJnr/Capybara
bigcode/starcoderdata
cerebras/SlimPajama-627B
allenai/nllb
jondurbin/gutenberg-dpo-v0.1
cognitivecomputations/samantha-data
Open-Orca/SlimOrca
Intel/orca_dpo_pairs
oscar-corpus/OSCAR-2109
kiranpantha/dataset-for-peft-cv-nepds
TIGER-Lab/MathInstruct
garage-bAInd/Open-Platypus
meta-math/MetaMathQA
legacy-datasets/wikipedia
jondurbin/truthy-dpo-v0.1
databricks/databricks-dolly-15k
mozilla-foundation/common_voice_13_0
open-r1/OpenR1-Math-220k
anthracite-org/kalo-opus-instruct-22k-no-refusal
OpenAssistant/oasst1
tiiuae/falcon-refinedweb
AI-MO/NuminaMath-TIR
EleutherAI/pile
jondurbin/airoboros-2.2.1
google/fleurs
AI-MO/NuminaMath-CoT
m-a-p/Code-Feedback

Models 1,453  Filter by name

jonatasgrosman/whisper-large-zh-cv11
Automatic Speech Recognition • Updated Dec 22, 2022 • 229 • 76

pierreguillou/whisper-medium-french
Automatic Speech Recognition • Updated Dec 12, 2022 • 259 • 9

qanastek/whisper-large-french-uncased
Automatic Speech Recognition • Updated Dec 16, 2022 • 22 • 1

simonl0909/whisper-large-v2-cantonese
Automatic Speech Recognition • Updated Sep 30, 2023 • 385 • 12

DrishtiSharma/whisper-large-v2-vietnamese
Automatic Speech Recognition • Updated Dec 13, 2022 • 39 • 2

DrishtiSharma/whisper-large-v2-punjabi
Automatic Speech Recognition • Updated Dec 15, 2022 • 478 • 1

anuragshas/whisper-large-v2-hi-v3
Automatic Speech Recognition • Updated Dec 16, 2022 • 25 • 1

DrishtiSharma/whisper-large-v2-malayalam
Automatic Speech Recognition • Updated Dec 16, 2022 • 496 • 4

bofenghuang/whisper-small-cv11-german
Automatic Speech Recognition • Updated Dec 27, 2022 • 23 • 7

anuragshas/whisper-large-v2-bn
Automatic Speech Recognition • Updated Dec 18, 2022 • 624 • 6

jstoone/whisper-medium-da
Automatic Speech Recognition • Updated Oct 24, 2023 • 18 • 5

In the example image above, I've selected the *mozilla-foundation/common_voice_11_0* dataset. The list on the right are models tagged as pre-trained on that dataset. Nice, right?

- **Languages:** These are the languages that audio-and/or-text-capable models have been tagged as pre-trained on.

## Model Search Questions

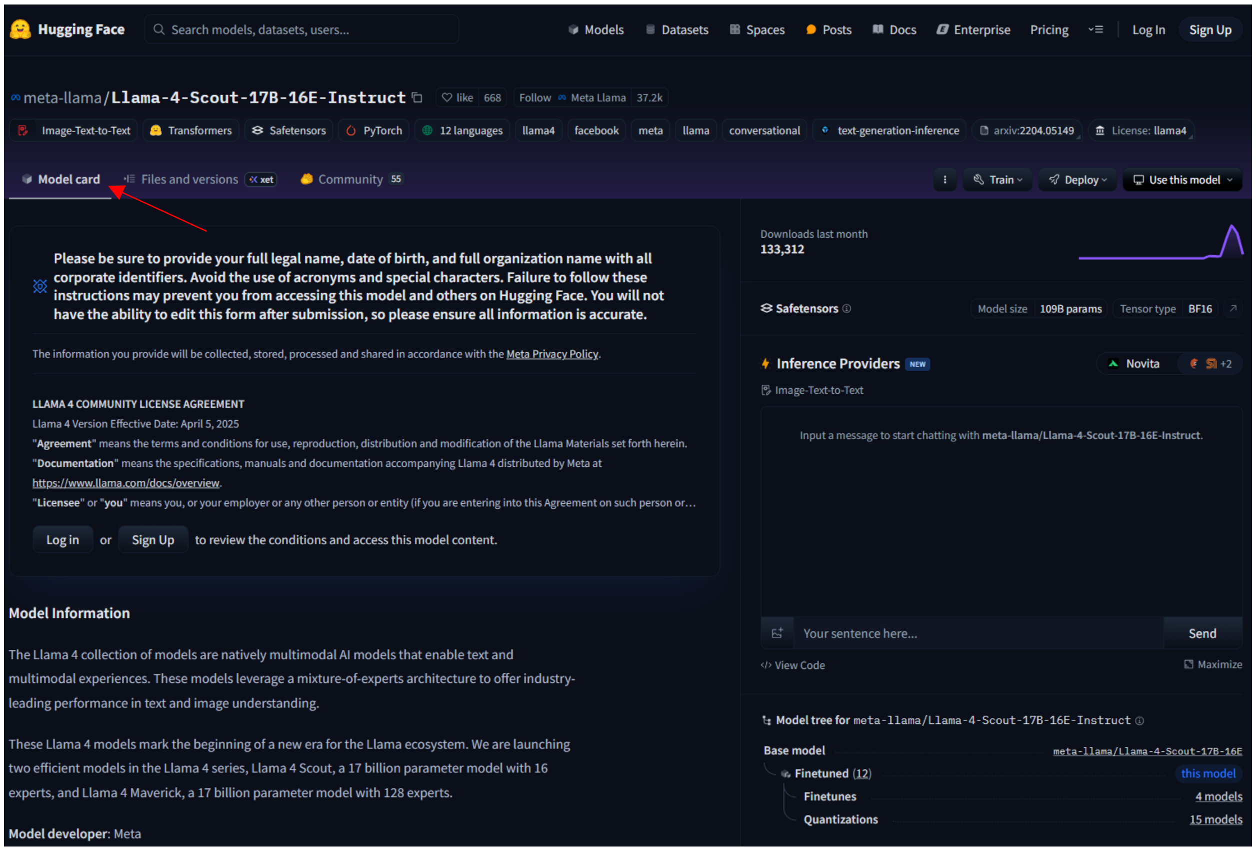
**Q1.** What tags would you select to find an image-and-text input, text output model pre-trained on an English language dataset?

**Q2.** You have some time series data and are mostly experienced using the PyTorch library. You want to try a transformer-based model because you suspect you have long-range dependencies in your data. Which tags would you use to find some candidate models?

## Exploring a Model

Now, let's drill down to looking at individual models. *Click on whatever model is trending most for you now* (we'll use Llama-4-Scout since that's what's big as of this writing).

The website will take you to the **model card** when you click a model. A model card is a tool that documents models, providing helpful information about the models and are essential for discoverability and reproducibility.



**Please be sure to provide your full legal name, date of birth, and full organization name with all corporate identifiers. Avoid the use of acronyms and special characters. Failure to follow these instructions may prevent you from accessing this model and others on Hugging Face. You will not have the ability to edit this form after submission, so please ensure all information is accurate.**

The information you provide will be collected, stored, processed and shared in accordance with the Meta Privacy Policy.

**LLAMA 4 COMMUNITY LICENSE AGREEMENT**

Llama 4 Version Effective Date: April 5, 2025

"**Agreement**" means the terms and conditions for use, reproduction, distribution and modification of the Llama Materials set forth herein.

"**Documentation**" means the specifications, manuals and documentation accompanying Llama 4 distributed by Meta at https://www.llama.com/docs/overview.

"**Licensee**" or "**you**" means you, or your employer or any other person or entity (if you are entering into this Agreement on such person or…

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

**Model Information**

The Llama 4 collection of models are natively multimodal AI models that enable text and multimodal experiences. These models leverage a mixture-of-experts architecture to offer industry-leading performance in text and image understanding.

These Llama 4 models mark the beginning of a new era for the Llama ecosystem. We are launching two efficient models in the Llama 4 series, Llama 4 Scout, a 17 billion parameter model with 16 experts, and Llama 4 Maverick, a 17 billion parameter model with 128 experts.

**Model developer**: Meta

Input a message to start chatting with **meta-llama/Llama-4-Scout-17B-16E-Instruct**.

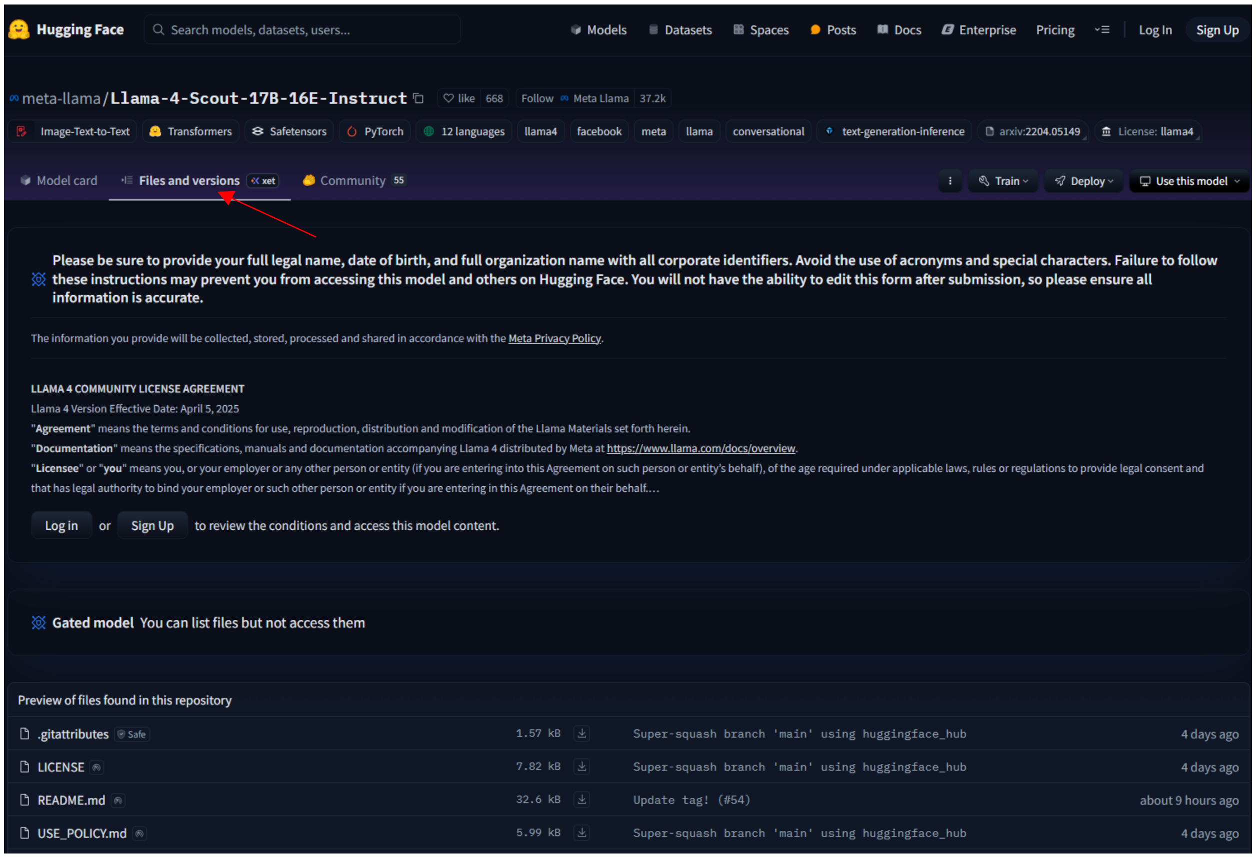
**Model tree for** `meta-llama/Llama-4-Scout-17B-16E-Instruct`

| | |
|---|---|
| **Base model** | `meta-llama/Llama-4-Scout-17B-16E` |
| **Finetuned** (12) | this model |
| **Finetunes** | 4 models |
| **Quantizations** | 15 models |

The interface has many components, so let's go through them:

- At the top, you can find different **tags** for things such as the task (*text generation, image classification*, etc.), frameworks (*PyTorch*, *TensorFlow*, etc.), the model's language (*English*, *Arabic*, *etc.*), and license (*e.g. MIT*).
- At the right column, once you're signed in, you can frequently explore the model directly in the browser using the *Inference Providers API*. The model you selected will determine what kinds of inference you can do. This is a good way to get a quick feel for a model's operation.
- In the middle, you can go through the model card content. Typically, models produced by industry and academic leaders will be well documented, which can give you a solid understanding of how the model is implemented, its intended use, etc.

Where does all this data come from? At Hugging Face, everything is open-sourced and based on Git repositories. You can click the *Files and Versions* tab, which will allow you to see all the repository files, including the model weights. The model card is a markdown file **(README.md)** containing metadata, such as the tags on top of the content.



**Please be sure to provide your full legal name, date of birth, and full organization name with all corporate identifiers. Avoid the use of acronyms and special characters. Failure to follow these instructions may prevent you from accessing this model and others on Hugging Face. You will not have the ability to edit this form after submission, so please ensure all information is accurate.**

The information you provide will be collected, stored, processed and shared in accordance with the Meta Privacy Policy.

**LLAMA 4 COMMUNITY LICENSE AGREEMENT**
Llama 4 Version Effective Date: April 5, 2025
"**Agreement**" means the terms and conditions for use, reproduction, distribution and modification of the Llama Materials set forth herein.
"**Documentation**" means the specifications, manuals and documentation accompanying Llama 4 distributed by Meta at https://www.llama.com/docs/overview.
"**Licensee**" or "**you**" means you, or your employer or any other person or entity (if you are entering into this Agreement on such person or entity's behalf), of the age required under applicable laws, rules or regulations to provide legal consent and that has legal authority to bind your employer or such other person or entity if you are entering in this Agreement on their behalf.…

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

**Gated model** You can list files but not access them

Preview of files found in this repository

| | | | |
|---|---|---|---|
| .gitattributes Safe | 1.57 kB | Super-squash branch 'main' using huggingface_hub | 4 days ago |
| LICENSE | 7.82 kB | Super-squash branch 'main' using huggingface_hub | 4 days ago |
| README.md | 32.6 kB | Update tag! (#54) | about 9 hours ago |
| USE_POLICY.md | 5.99 kB | Super-squash branch 'main' using huggingface_hub | 4 days ago |

Since all model pages are Git-based repositories, you get version control out of the box. Just as with GitHub, you can do things such as Git cloning, adding, committing, branching, and pushing. If you've never used Git, we suggest taking our Beginner's Series!

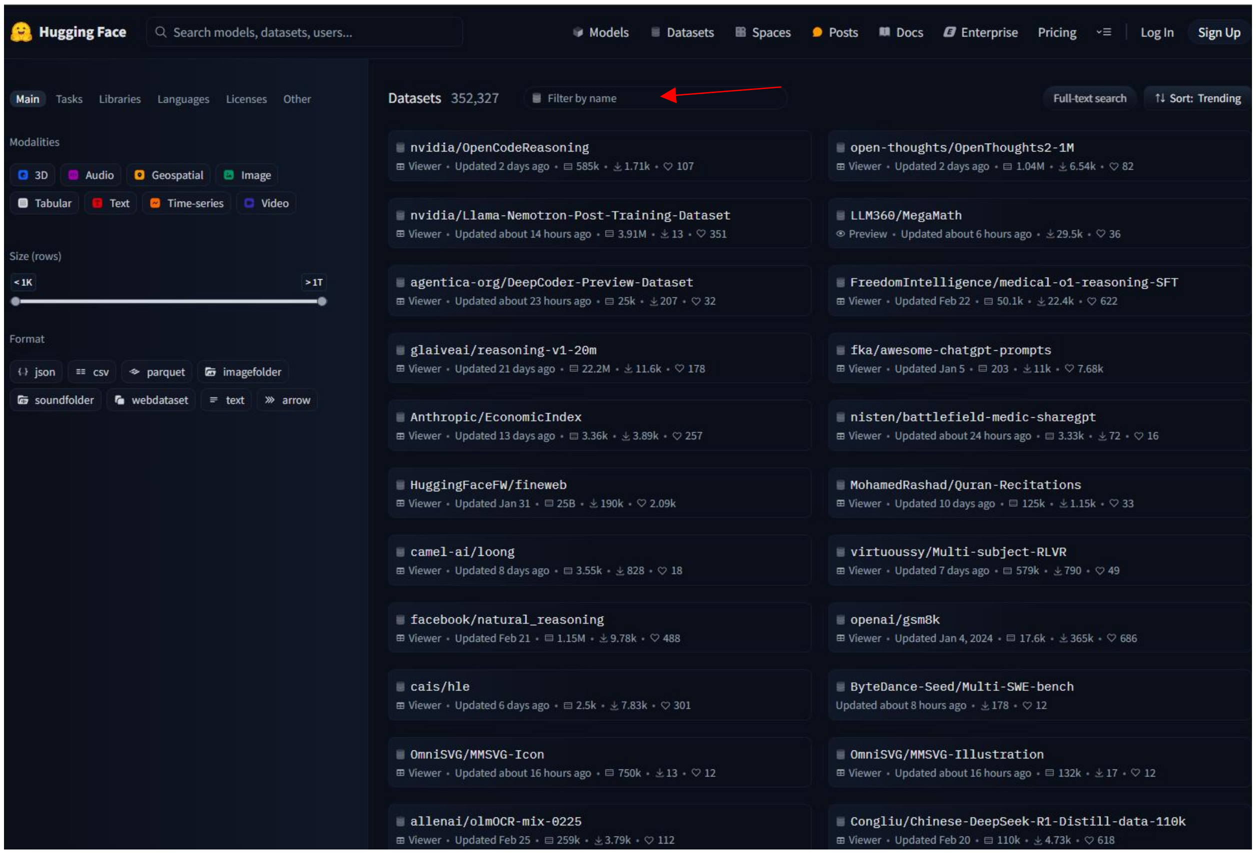
## Model Page Questions

**Q3.** How many parameters does the model you're looking at have? *NOTE: The Llama-Scout-4 model above has this shown. Poorly documented model pages may not list these; you might need to return and find a better-documented model!*

**Q4.** Where would you go to get developer and community support if you had an issue with the model you're looking at? *NOTE: As mentioned above, The Llama-Scout-4 model has this shown. Poorly supported model pages may not list this; you might need to return and find a better-supported model!*

## Exploring Datasets

The Hub hosts hundreds of thousands of open-sourced datasets, which are free to use and span several domains. Like models, you can head to https://hf.co/datasets. On the left, you can find different category filters based on the dataset's task, license, and size (in numbers of samples or "rows").



Datasets 352,327

nvidia/OpenCodeReasoning
Viewer • Updated 2 days ago • 585k • 1.71k • 107

open-thoughts/OpenThoughts2-1M
Viewer • Updated 2 days ago • 1.04M • 6.54k • 82

nvidia/Llama-Nemotron-Post-Training-Dataset
Viewer • Updated about 14 hours ago • 3.91M • 13 • 351

LLM360/MegaMath
Preview • Updated about 6 hours ago • 29.5k • 36

agentica-org/DeepCoder-Preview-Dataset
Viewer • Updated about 23 hours ago • 25k • 207 • 32

FreedomIntelligence/medical-o1-reasoning-SFT
Viewer • Updated Feb 22 • 50.1k • 22.4k • 622

glaiveai/reasoning-v1-20m
Viewer • Updated 21 days ago • 22.2M • 11.6k • 178

fka/awesome-chatgpt-prompts
Viewer • Updated Jan 5 • 203 • 11k • 7.68k

Anthropic/EconomicIndex
Viewer • Updated 13 days ago • 3.36k • 3.89k • 257

nisten/battlefield-medic-sharegpt
Viewer • Updated about 24 hours ago • 3.33k • 72 • 16

HuggingFaceFW/fineweb
Viewer • Updated Jan 31 • 25B • 190k • 2.09k

MohamedRashad/Quran-Recitations
Viewer • Updated 10 days ago • 125k • 1.15k • 33

camel-ai/loong
Viewer • Updated 8 days ago • 3.55k • 828 • 18

virtuoussy/Multi-subject-RLVR
Viewer • Updated 7 days ago • 579k • 790 • 49

facebook/natural_reasoning
Viewer • Updated Feb 21 • 1.15M • 9.78k • 488

openai/gsm8k
Viewer • Updated Jan 4, 2024 • 17.6k • 365k • 686

cais/hle
Viewer • Updated 6 days ago • 2.5k • 7.83k • 301

ByteDance-Seed/Multi-SWE-bench
Updated about 8 hours ago • 178 • 12

OmniSVG/MMSVG-Icon
Viewer • Updated about 16 hours ago • 750k • 13 • 12

OmniSVG/MMSVG-Illustration
Viewer • Updated about 16 hours ago • 132k • 17 • 12

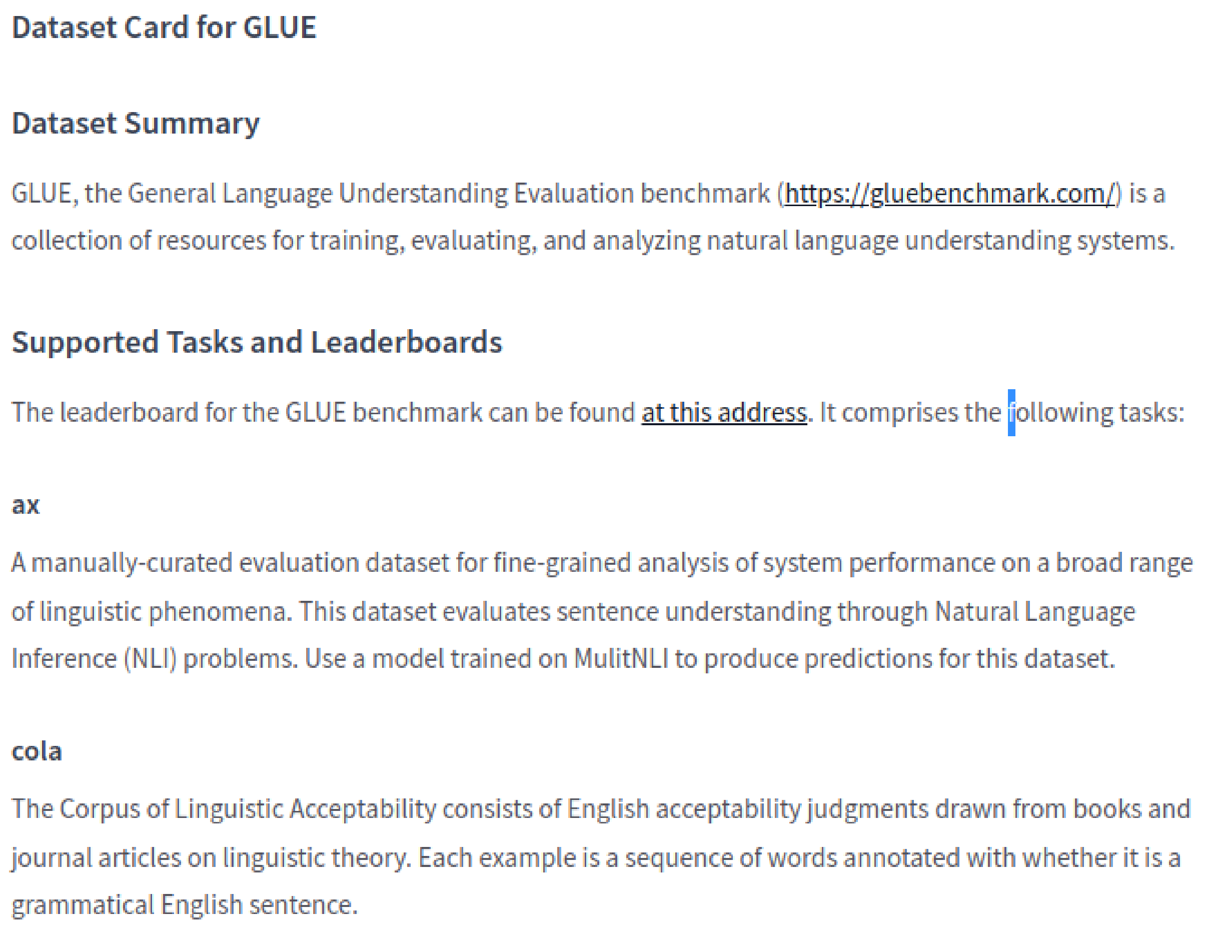
allenai/olmOCR-mix-0225
Viewer • Updated Feb 25 • 259k • 3.79k • 112

Congliu/Chinese-DeepSeek-R1-Distill-data-110k
Viewer • Updated Feb 20 • 110k • 4.73k • 618

Let's explore the GLUE dataset, a famous dataset used to test the performance of NLP models.

1. In the "Filter by name" box directly above the dataset lists, type "nyu-mll/glue"
2. Hit *Enter*
3. Click on the *nyu-mll/glue* link

Like model repositories, you have a dataset card that documents the dataset. If you scroll down a bit, you will find things such as the summary, the structure, and more.



Dataset Card for GLUE

Dataset Summary

GLUE, the General Language Understanding Evaluation benchmark (https://gluebenchmark.com/) is a collection of resources for training, evaluating, and analyzing natural language understanding systems.

Supported Tasks and Leaderboards

The leaderboard for the GLUE benchmark can be found at this address. It comprises the following tasks:

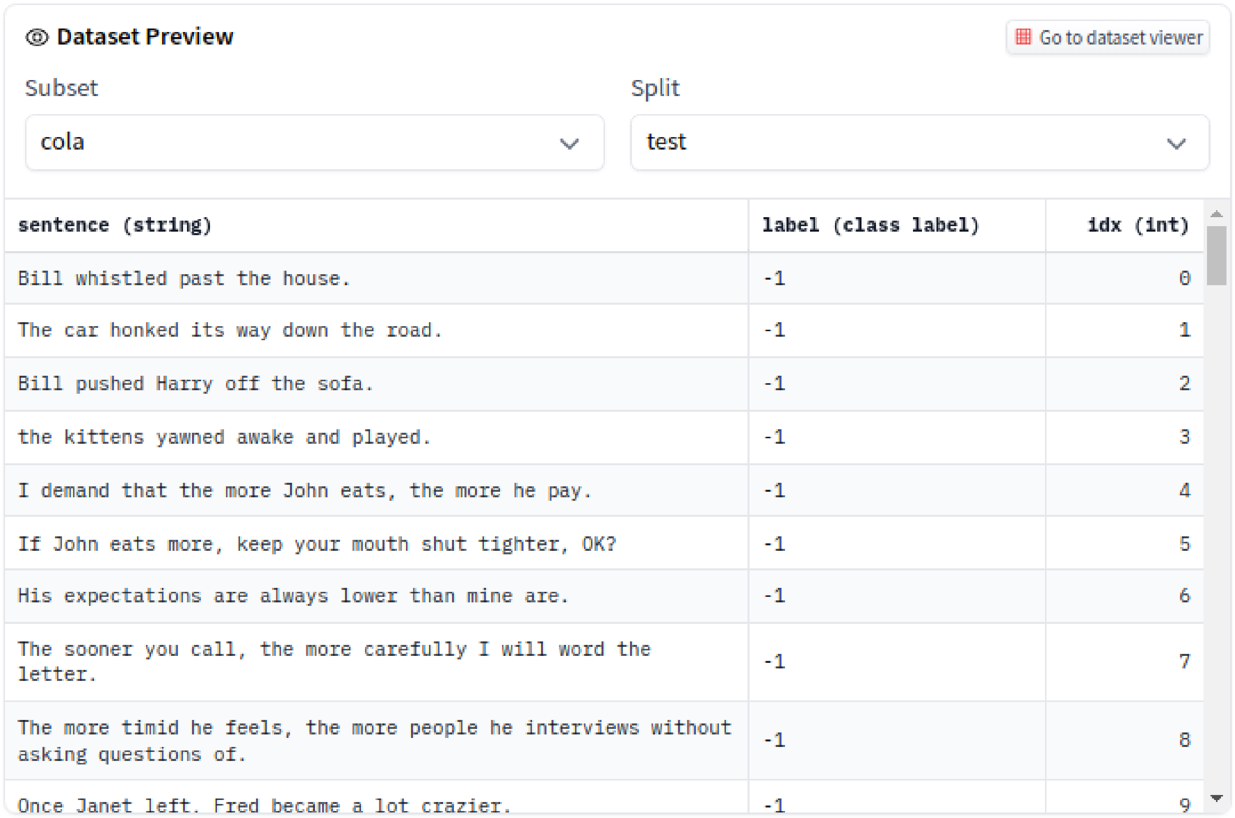
ax

A manually-curated evaluation dataset for fine-grained analysis of system performance on a broad range of linguistic phenomena. This dataset evaluates sentence understanding through Natural Language Inference (NLI) problems. Use a model trained on MulitNLI to produce predictions for this dataset.

cola

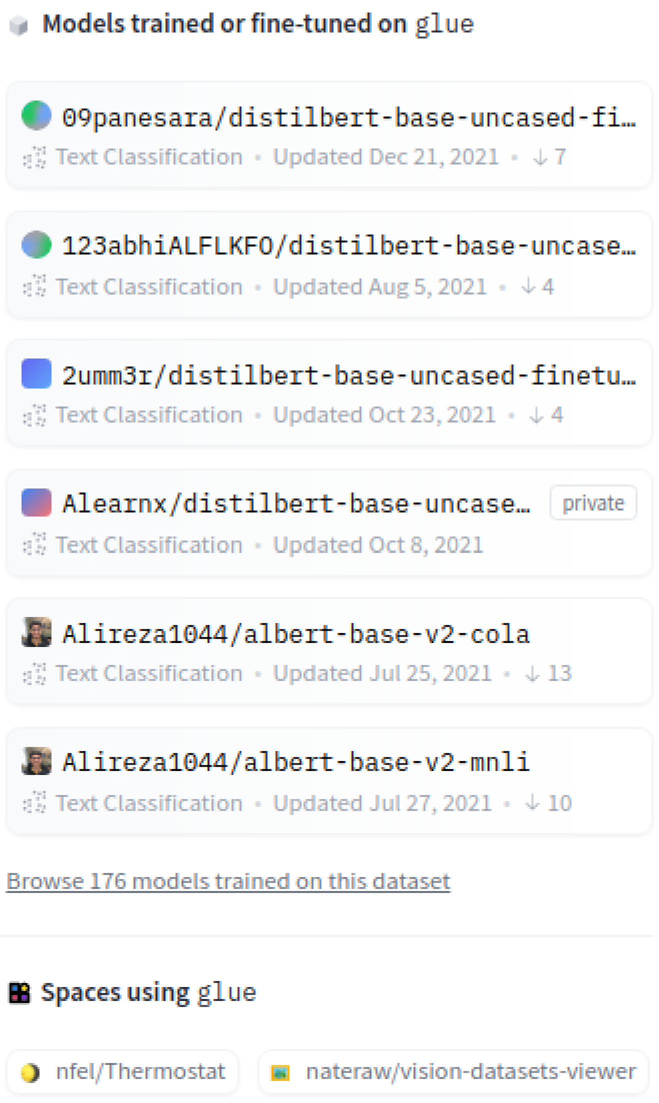
The Corpus of Linguistic Acceptability consists of English acceptability judgments drawn from books and journal articles on linguistic theory. Each example is a sequence of words annotated with whether it is a grammatical English sentence.

- At the top, you can explore a slice of the dataset directly in the browser. The GLUE dataset is divided into multiple sub-datasets (or subsets) that you can select, such as COLA and QNLI.



| sentence (string) | label (class label) | idx (int) |
|---|---|---|
| Bill whistled past the house. | -1 | 0 |
| The car honked its way down the road. | -1 | 1 |
| Bill pushed Harry off the sofa. | -1 | 2 |
| the kittens yawned awake and played. | -1 | 3 |
| I demand that the more John eats, the more he pay. | -1 | 4 |
| If John eats more, keep your mouth shut tighter, OK? | -1 | 5 |
| His expectations are always lower than mine are. | -1 | 6 |
| The sooner you call, the more carefully I will word the letter. | -1 | 7 |
| The more timid he feels, the more people he interviews without asking questions of. | -1 | 8 |
| Once Janet left, Fred became a lot crazier. | -1 | 9 |

- At the right of the dataset card, you can see a list of models trained or fine-tuned on this dataset.



Models trained or fine-tuned on glue

09panesara/distilbert-base-uncased-fi…
Text Classification • Updated Dec 21, 2021 • ↓ 7

123abhiALFLKFO/distilbert-base-uncase…
Text Classification • Updated Aug 5, 2021 • ↓ 4

2umm3r/distilbert-base-uncased-finetu…
Text Classification • Updated Oct 23, 2021 • ↓ 4

Alearnx/distilbert-base-uncase… private
Text Classification • Updated Oct 8, 2021

Alireza1044/albert-base-v2-cola
Text Classification • Updated Jul 25, 2021 • ↓ 13

Alireza1044/albert-base-v2-mnli
Text Classification • Updated Jul 27, 2021 • ↓ 10

Browse 176 models trained on this dataset

Spaces using glue

nfel/Thermostat    nateraw/vision-datasets-viewer

### Dataset Exploration Questions

Search for the Common Voice dataset. Answer these questions:

**Q5:** What tasks can the Common Voice dataset be used for?

**Q6:** How many languages are covered in this dataset?

**Q7:** Which are the dataset splits?

## Bringing It All Together

With a mighty tool like Hugging Face at your disposal, you now have a launching point for nearly any project. Sometimes building a model from layers and training from scratch *is* the way to go, but your journey should always start with exploring if anyone's already done most of the work for you. "Well begun is half done" – *Typically attributed to Aristotle.*

## Hands-on Exercise

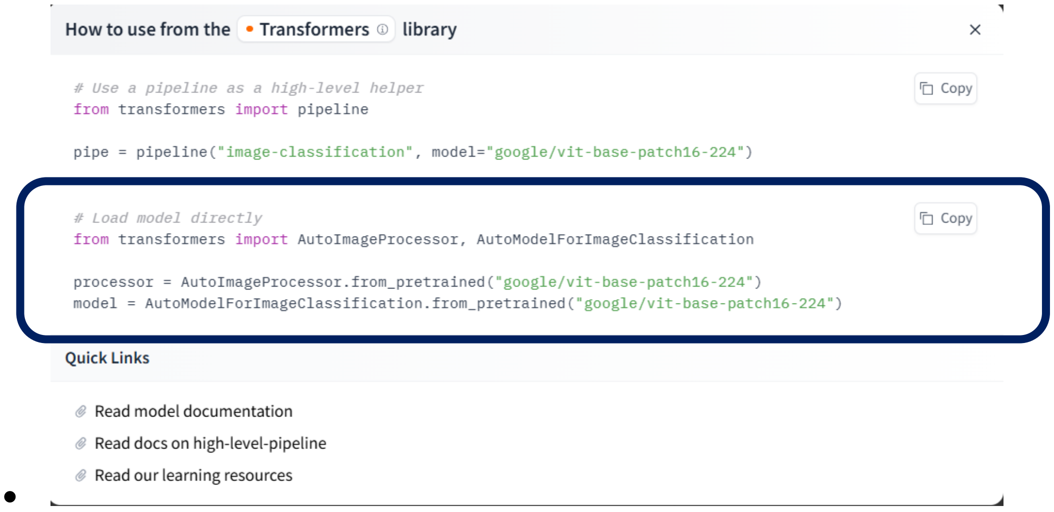
In Notebook 1, 01_transfer_learning_concepts.ipynb, we compared:

- Training EfficientNet-B0 from scratch on the AgriNet dataset
- Finetuning EfficientNet-B0 *pre-trained on ImageNet* with the AgriNet dataset

We discovered that finetuning led to better results because the pre-trained model already "understood" basic image features and could generalize better with a smaller dataset (AgriNet).

Now, let's leverage your new knowledge of the Hugging Face Hub. Your task is to find a *different* pre-trained image classification model on the Hub, load it directly using the *transformers* library, and fine-tune it on the AgriNet dataset.

### 1. *Explore and Select a Model:*

- Go to the Hugging Face Hub models page: https://hf.co/models.
- Use the filters (Task: Image Classification, Libraries: PyTorch, Datasets: filter for models pre-trained on ImageNet or some other large dataset).
- Browse the filtered models. Look for established architectures *other than* EfficientNet-B5 (e.g., ResNet variants, ViT variants, MobileNet variants, etc.). Choose one that is well-documented and has a Community Support page.
- Note that some models will require an account and authentication to download and use.
- In JupyterLab, *make a copy of Notebook 1*. Rename this copy 03.0_transfer_learning_capstone.ipynb
- In your new notebook, create a Markdown cell above the Import Libraries section and list the Hugging Face Model ID of the selected model. Provide a brief description of why you selected that model.
- On the model page, click the "Use this model" button and select "Transformers"
- The pop-up for the "google/vit-base-patch16-224" is below. You will want to use the code in the "# Load model directly" section.
- 
How to use from the Transformers library

```
# Use a pipeline as a high-level helper
from transformers import pipeline

pipe = pipeline("image-classification", model="google/vit-base-patch16-224")
```

```
# Load model directly
from transformers import AutoImageProcessor, AutoModelForImageClassification

processor = AutoImageProcessor.from_pretrained("google/vit-base-patch16-224")
model = AutoModelForImageClassification.from_pretrained("google/vit-base-patch16-224")
```

Quick Links

- Read model documentation
- Read docs on high-level-pipeline
- Read our learning resources

2. *Update Imports:*

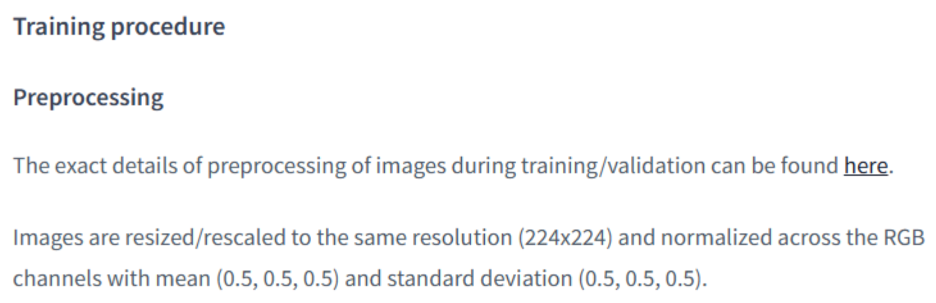
- Navigate to **Section 1 (Import libraries)** in your *03.0* notebook.
- Add the following import statement with the other imports:

```
# Import Hugging Face Transformers
from transformers import AutoImageProcessor,
AutoModelForImageClassification
import torch.nn as nn
```

- Run the import cell to make sure the library is available.

3. *Modify data loaders (Section 4):*

- Depending on the input dimensions of the images the model was trained on, you may need to modify the data loaders. In the case of "google/vit-base-patch16-224", the model card shows:



**Training procedure**

**Preprocessing**

The exact details of preprocessing of images during training/validation can be found here.

Images are resized/rescaled to the same resolution (224x224) and normalized across the RGB channels with mean (0.5, 0.5, 0.5) and standard deviation (0.5, 0.5, 0.5).

- We are set for dimensions since it's the same 224x224 but we should update the normalization steps using the values in the model card. For your model, you should be able to find at least the input shape.

4. *Modify Pre-trained Model Loading (Section 8):*

- Navigate to what was originally **Section 8 (Load the EfficientNet-B0 model with pre-trained weights)**. We will repurpose this section to load *your* chosen Hub model.
- **Replace** the code block in this section with the lines from the model pop-up to load the model (e.g. from the image above for "google/vit-base-patch16-224", we'd use these lines:

```
processor = AutoImageProcessor.from_pretrained(
   "google/vit-base-patch16-224")
vit_model = AutoModelForImageClassification.from_pretrained(
   "google/vit-base-patch16-224")
```

- Then either in the same cell, or in a new cell add:

```
# For compatibility with the existing code, we need to
# wrap the Hugging Face model.
class HuggingFaceModelWrapper(nn.Module):
    def __init__(self, hf_model):
        super(HuggingFaceModelWrapper, self).__init__()
        self.hf_model = hf_model

    def forward(self, x):
        outputs = self.hf_model(x)
        return outputs.logits  # Extract logits
                              # for compatibility

# Freeze all layers except the classifier
for param in vit_model.parameters():
    param.requires_grad = False

wrapped_vit_model = HuggingFaceModelWrapper(vit_model)

# Replace the classification layer
wrapped_vit_model.hf_model.classifier = nn.Linear(
    wrapped_vit_model.hf_model.classifier.in_features,
num_classes
)
```

- Run this cell. It should load your selected model architecture with a new classification head that is ready for AgriNet.

*Run All:* Execute all cells in your modified 03.0 notebook from top to bottom to ensure everything works and to get your final results. Debug any errors that arise, paying close attention to the FIX_ME sections. You did it!!!