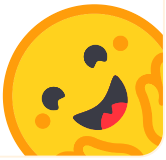

Installation

Transformers works with PyTorch, TensorFlow 2.0, and Flax. It has been tested on Python 3.9+, PyTorch 2.0+, TensorFlow 2.6+, and Flax 0.4.1+.

Virtual environment

A virtual environment helps manage different projects and avoids compatibility issues between dependencies. Take a look at the Install packages in a virtual environment using pip and venv guide if you're unfamiliar with Python virtual environments.

venv uv

Create and activate a virtual environment in your project directory with venv.

```
python -m venv .env
source .env/bin/activate
```

Python

You can install Transformers with pip or uv.

pip uv

pip is a package installer for Python. Install Transformers with pip in your newly created virtual environment.

```
pip install transformers
```

For GPU acceleration, install the appropriate CUDA drivers for PyTorch and TensorFlow.

Run the command below to check if your system detects an NVIDIA GPU.

```
nvidia-smi
```

To install a CPU-only version of Transformers and a machine learning framework, run the following command.

PyTorch TensorFlow Flax

```
pip install 'transformers[torch]'
uv pip install 'transformers[torch]'
```

Test whether the install was successful with the following command. It should return a label and score for the provided text.

```
python -c "from transformers import pipeline; print(pipeline('sentiment-analysis')('hu
[{'label': 'POSITIVE', 'score': 0.9998704791069031}]
```

Source install

Installing from source installs the *latest* version rather than the *stable* version of the library. It ensures you have the most up-to-date changes in Transformers and it's useful for experimenting with the latest features or fixing a bug that hasn't been officially released in the stable version yet.

The downside is that the latest version may not always be stable. If you encounter any problems, please open a GitHub Issue so we can fix it as soon as possible.

Install from source with the following command.

```
pip install git+https://github.com/huggingface/transformers
```

Check if the install was successful with the command below. It should return a label and score for the provided text.

```
python -c "from transformers import pipeline; print(pipeline('sentiment-analysis')('hu
[{'label': 'POSITIVE', 'score': 0.9998704791069031}]
```

Editable install

An editable install is useful if you're developing locally with Transformers. It links your local copy of Transformers to the Transformers repository instead of copying the files. The files are added to Python's import path.

```
git clone https://github.com/huggingface/transformers.git
cd transformers
pip install -e .
```

> You must keep the local Transformers folder to keep using it.

Update your local version of Transformers with the latest changes in the main repository with the following command.

```
cd ~/transformers/
git pull
```

conda

conda is a language-agnostic package manager. Install Transformers from the conda-forge channel in your newly created virtual environment.

```
conda install conda-forge::transformers
```

Set up

After installation, you can configure the Transformers cache location or set up the library for offline usage.

Cache directory

When you load a pretrained model with from_pretrained(), the model is downloaded from the Hub and locally cached.

Every time you load a model, it checks whether the cached model is up-to-date. If it's the same, then the local model is loaded. If it's not the same, the newer model is downloaded and cached.

The default directory given by the shell environment variable `TRANSFORMERS_CACHE` is `~/.cache/huggingface/hub`. On Windows, the default directory is `C:\Users\username\.cache\huggingface\hub`.

Cache a model in a different directory by changing the path in the following shell environment variables (listed by priority).

1. HF_HUB_CACHE or `TRANSFORMERS_CACHE` (default)
2. HF_HOME
3. XDG_CACHE_HOME + `/huggingface` (only if `HF_HOME` is not set)

Older versions of Transformers uses the shell environment variables `PYTORCH_TRANSFORMERS_CACHE` or `PYTORCH_PRETRAINED_BERT_CACHE`. You should keep these unless you specify the newer shell environment variable `TRANSFORMERS_CACHE`.

Offline mode

To use Transformers in an offline or firewalled environment requires the downloaded and cached files ahead of time. Download a model repository from the Hub with the snapshot_download method.

> Refer to the Download files from the Hub guide for more options for downloading files from the Hub. You can download files from specific revisions, download from the CLI, and

> even filter which files to download from a repository.

```
from huggingface_hub import snapshot_download

snapshot_download(repo_id="meta-llama/Llama-2-7b-hf", repo_type="model")
```

Set the environment variable `HF_HUB_OFFLINE=1` to prevent HTTP calls to the Hub when loading a model.

```
HF_HUB_OFFLINE=1 \
python examples/pytorch/language-modeling/run_clm.py --model_name_or_path meta-llama/L
```

Another option for only loading cached files is to set `local_files_only=True` in from_pretrained().

```
from transformers import LlamaForCausalLM

model = LlamaForCausalLM.from_pretrained("./path/to/local/directory", local_files_only
```

<> Update on GitHub

← Transformers

Quickstart →