

作成も行なえる。Hugging Faceからは今回使用するモデルもダウンロードする。「hf auth login」でアカウント認証を行なうことで、ライセンス同意済みのモデルを入手できる。

```
curl -LsSf https://astral.sh/uv/install.sh | sh
source $HOME/.local/bin/env

mkdir -p ~/Documents/works && cd ~/Documents/works

uv venv --python 3.12
source .venv/bin/activate

uv pip install huggingface_hub

hf auth login --token hf_XXXXXXXXXXXXXXXXXX
# ダウンロードするモデルによっては認証が必要になる
```

ステップ3 | Dockerのインストール

Dockerをインストールしよう。Dockerはアプリケーションとその実行環境をひとまとめにした「コンテナ」と呼ばれる単位で管理するプラットフォームだ。vLLMはDockerコンテナとして提供されており、依存関係の複雑なインストール作業を省略できる。

```
sudo apt update
sudo apt install -y ca-certificates curl

sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg \
-o /etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc

echo "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] \
https://download.docker.com/linux/ubuntu \
$(. /etc/os-release && echo "$VERSION_CODENAME") \
stable" | \
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

sudo apt update
sudo apt install -y docker-ce docker-ce-cli containerd.io \
docker-buildx-plugin docker-compose-plugin

sudo usermod -aG docker $USER
newgrp docker
```

以下で動作確認を行ない、環境の準備は完了。次はLLMのダウンロードと起動となる。

```
docker run hello-world
```

ステップ4 | LLMモデルのダウンロード

LLMのモデルによってはGoogleのライセンス同意が必要がある。その場合は、事前にHugging Faceのモデルページで承認しておく。今回はGoogleの推論特化・4bit量子化モデルである「Gemma 4 26B-A4B-it AWQ INT4」を使用する。

```
hf download cyankiwi/gemma-4-26B-A4B-it-AWQ-4bit \
--local-dir ./gemma-4-26B-A4B-it-AWQ-4bit
```

ステップ5 | vLLMコンテナの起動

表5の設定でvLLMコンテナを起動する。

表5 vLLMコンテナの設定(モデル・推論設定)

項目	値
モデル	Gemma 4 26B-A4B-it AWQ INT4
アーキテクチャ	MoE(4B活性)
量子化	compressed-tensors (AWQ INT4)
dtype	float16
max_model_len	131,072(128K)
gpu_memory_utilization	0.9
KVキャッシュ	8.78GiB/372,445トークン
Vision	(image=1)
VRAM使用量	約28.2GB/32GB

```
docker run \
--name vllm-gemma4 \
--network=host \
--group-add=video \
--group-add=render \
--ipc=host \
--cap-add=SYS_PTRACE \
--security-opt seccomp=unconfined \
--device=/dev/kfd \
--device=/dev/dri \
--shm-size=16g \
-v ./gemma-4-26B-A4B-it-AWQ-4bit:/model \
-e VLLM_ROCM_USE_AITER=0 \
vllm/vllm-openai-rocm:v0.20.1 \
/model \
--served-model-name gemma-4-26B-A4B-it-AWQ-INT4 \
--dtype float16 \
--gpu-memory-utilization 0.90 \
--max-model-len 131072 \
--limit-mm-per-prompt '{"image":1,"audio":0}' \
--host 0.0.0.0 \
--port 8888
```

起動には数分かかる。ログに以下の一文が表示されれば正常起動だ。なお、上記で「-d」オプションを付けずに起動しているのはログを確認するため。実運用では「docker run -d」で起動する。