# 环境安装

## Python

下载python：[python.org/getit/](https://www.python.org/getit/)

依赖：

pip install -U huggingface\_hub # 安装最新模型下载库

pip install -U hf-transfer # 安装最新模型下载加速库

## Llama.cpp

1. Llama.cpp

[Releases · ggml-org/llama.cpp · GitHub](https://github.com/ggml-org/llama.cpp/releases)

选择windows版本的CUDA版本

1. 安装Visual Studio

[Visual Studio: 面向软件开发人员和 Teams 的 IDE 和代码编辑器 (microsoft.com)](https://visualstudio.microsoft.com/zh-hans/)

1. 安装CUDA Toolkit

[CUDA Toolkit 12.9 Update 1 Downloads | NVIDIA Developer](https://developer.nvidia.com/cuda-downloads?target_os=Windows&target_arch=x86_64&target_version=11&target_type=exe_local)

添加环境变量：C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v12.9

1. 安装cmake

C++编译工具，选择[cmake-4.0.2-windows-x86\_64.zip](https://github.com/Kitware/CMake/releases/download/v4.0.2/cmake-4.0.2-windows-x86_64.zip)

[Download CMake](https://cmake.org/download/)

添加环境变量：D:\software\cmake-4.0.2-windows-x86\_64\bin

# 模型下载

## 选择模型

国外开源：[Models - Hugging Face](https://huggingface.co/models)

国内镜像：[Models - Hugging Face (hf-mirror.com)](https://hf-mirror.com/models)

下载指导书：[如何快速下载huggingface模型——全方法总结 - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/663712983)

## 下载模型（自动化代码）

|  |
| --- |
| import os  # 启用 hf\_transfer 加速 os.environ["HF\_HUB\_ENABLE\_HF\_TRANSFER"] = "1" os.environ["HF\_ENDPOINT"] = "https://hf-mirror.com" # 使用国内镜像  from huggingface\_hub import hf\_hub\_download  # 配置下载参数 # 举例：https://hf-mirror.com/tensorblock/Phind\_Phind-CodeLlama-34B-Python-v1-GGUF/blob/main/Phind-CodeLlama-34B-Python-v1-Q4\_K\_M.gguf # 模型的标题，如tensorblock/Phind\_Phind-CodeLlama-34B-Python-v1-GGUF REPO\_ID = "tensorblock/Phind\_Phind-CodeLlama-34B-Python-v1-GGUF" # 模型的版本，如Phind-CodeLlama-34B-Python-v1-Q4\_K\_M.gguf FILENAME = "Phind-CodeLlama-34B-Python-v1-Q4\_K\_M.gguf" # 可以修改为任何本地路径 DOWNLOAD\_DIR = "./models/Phind-CodeLlama-34B" # 创建下载目录 os.makedirs(DOWNLOAD\_DIR, exist\_ok=True)  # 使用 HfApi 下载文件 # api = HfApi() local\_path = hf\_hub\_download(  repo\_id=REPO\_ID,  filename=FILENAME,  repo\_type="model",  cache\_dir=DOWNLOAD\_DIR,  local\_dir=DOWNLOAD\_DIR,  # force\_download=False, # 重新下载 )  print(f"下载完成！文件保存至: {local\_path}") print(f"文件大小: {round(os.path.getsize(local\_path) / (1024 \*\* 3), 2)} GB") |

# 模型运行

## 编译

先生成MakeFile，代表要编译的内容；再启动编译

|  |
| --- |
| cd E:\file\python\llama  cmake -B build\_cuda -DLLAMA\_CUDA=ON -DLLAMA\_CUBLAS=ON -DLLAMA\_FLASH\_ATTN=ON -DCMAKE\_CUDA\_ARCHITECTURES="89"  # -DLLAMA\_CUBLAS=ON  # 作用：  # DLLAMA\_FLASH\_ATTN=ON  # 作用：启用Flash Attention优化  # DCMAKE\_CUDA\_ARCHITECTURES="89"  # 作用：指定目标GPU的计算能力架构,89对应4070 Ti的SM8.9  cmake --build build\_cuda --config Release –j 12  # 12核CPU编译 |

.\llama-cli.exe -m E:\file\python\MyCodeAi\models\Phind-CodeLlama-34B\Phind-CodeLlama-34B-Python-v1-Q4\_K\_M.gguf -ngl 38