ExLlamaV2

ExLlamav2 is a fast inference library for running LLMs locally on modern consumer-class GPUs.

It supports inference for GPTQ & EXL2 quantized models, which can be accessed on Hugging Face.

This notebook goes over how to run `exllamav2` within LangChain.

Additional information: ExLlamav2 examples

Installation

Refer to the official doc For this notebook, the requirements are :

- python 3.11
- langchain 0.1.7
- CUDA: 12.1.0 (see bellow)
- torch==2.1.1+cu121
- exllamav2 (0.0.12+cu121)

If you want to install the same exllamav2 version :

```
pip install https://github.com/turboderp/exllamav2/releases/download/v0.0.12/exllamav2-0.0.12+cu121-cp311-cp311-linux_x86_64.whl
```

if you use conda, the dependencies are :

```
- conda-forge::ninja
- nvidia/label/cuda-12.1.0::cuda
- conda-forge::ffmpeg
- conda-forge::gxx=11.4
```

Usage

You don't need an `API_TOKEN` as you will run the LLM locally.

It is worth understanding which models are suitable to be used on the desired machine.

TheBloke's Hugging Face models have a `Provided files` section that exposes the RAM required to run models of different quantisation sizes and methods (eg: Mistral-7B-Instruct-v0.2-GPTQ).

```
!pip install -U langchain huggingface_hub exllamav2 langchain_community
```

```
Requirement already satisfied: nvidia-nvtx-cu12==12.4.127 in /usr/local/lib/python3.11/dist-packages (from torch>=2.2.0->exllamav2) (1
Requirement already satisfied: nvidia-nvjitlink-cu12==12.4.127 in /usr/local/lib/python3.11/dist-packages (from torch>=2.2.0->exllamav
Requirement already satisfied: triton==3.2.0 in /usr/local/lib/python3.11/dist-packages (from torch>=2.2.0->exllamav2) (3.2.0)
Requirement already satisfied: sympy==1.13.1 in /usr/local/lib/python3.11/dist-packages (from torch>=2.2.0->exllamav2) (1.13.1)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python3.11/dist-packages (from sympy==1.13.1->torch>=2.2.0->exllam
Requirement already satisfied: cramjam>=2.3 in /usr/local/lib/python3.11/dist-packages (from fastparquet->exllamav2) (2.9.1)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas->exllamav2) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas->exllamav2) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas->exllamav2) (2025.2)
Requirement already satisfied: markdown-it-py>=2.2.0 in /usr/local/lib/python3.11/dist-packages (from rich->exllamav2) (3.0.0)
Requirement already satisfied: anyio in /usr/local/lib/python3.11/dist-packages (from httpx<1,>=0.23.0->langsmith<0.4,>=0.1.17->langch
Requirement already satisfied: httpcore==1.* in /usr/local/lib/python3.11/dist-packages (from httpx<1,>=0.23.0->langsmith<0.4,>=0.1.17
Requirement already satisfied: h11<0.15,>=0.13 in /usr/local/lib/python3.11/dist-packages (from httpcore==1.*->httpx<1,>=0.23.0->langs
Requirement already satisfied: jsonpointer>=1.9 in /usr/local/lib/python3.11/dist-packages (from jsonpatch<2.0,>=1.33->langchain-core<
Requirement already satisfied: mdurl~=0.1 in /usr/local/lib/python3.11/dist-packages (from markdown-it-py>=2.2.0->rich->exllamav2) (0.
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2->pandas->exllamav2) (1
Collecting mypy-extensions>=0.3.0 (from typing-inspect<1,>=0.4.0->dataclasses-json<0.7,>=0.5.7->langchain_community)
  Downloading mypy_extensions-1.0.0-py3-none-any.whl.metadata (1.1 kB)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.11/dist-packages (from jinja2->torch>=2.2.0->exllamav2) (3.0.
Requirement already satisfied: sniffio>=1.1 in /usr/local/lib/python3.11/dist-packages (from anyio->httpx<1,>=0.23.0->langsmith<0.4,>=
Downloading langchain_community-0.3.21-py3-none-any.whl (2.5 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 2.5/2.5 MB 42.1 MB/s eta 0:00:00
Downloading dataclasses_json-0.6.7-py3-none-any.whl (28 kB)
Downloading httpx_sse-0.4.0-py3-none-any.whl (7.8 kB)
Downloading pydantic_settings-2.8.1-py3-none-any.whl (30 kB)
Downloading marshmallow-3.26.1-py3-none-any.whl (50 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 50.9/50.9 kB 4.6 MB/s eta 0:00:00
Downloading python_dotenv-1.1.0-py3-none-any.whl (20 kB)
Downloading typing_inspect-0.9.0-py3-none-any.whl (8.8 kB)
Downloading mypy_extensions-1.0.0-py3-none-any.whl (4.7 kB)
Installing collected packages: python-dotenv, mypy-extensions, marshmallow, httpx-sse, typing-inspect, pydantic-settings, dataclasses-
Successfully installed dataclasses-json-0.6.7 httpx-sse-0.4.0 langchain_community-0.3.21 marshmallow-3.26.1 mypy-extensions-1.0.0 pyda
```

```
!git clone https://github.com/langchain-ai/langchain.git
```

```
Cloning into 'langchain'...
remote: Enumerating objects: 234693, done.
remote: Counting objects: 100% (829/829), done.
remote: Compressing objects: 100% (342/342), done.
remote: Total 234693 (delta 613), reused 492 (delta 487), pack-reused 233864 (from 2)
Receiving objects: 100% (234693/234693), 403.70 MiB | 31.50 MiB/s, done.
Resolving deltas: 100% (177404/177404), done.
Updating files: 100% (7411/7411), done.
```

```
import os

from huggingface_hub import snapshot_download
from langchain_community.llms.exllamav2 import ExLlamaV2
from langchain_core.callbacks import StreamingStdOutCallbackHandler
from langchain_core.prompts import PromptTemplate
from langchain.chains import LLMChain
#from libs.langchain.langchain.chains.llm import LLMChain
```

```
# function to download the gptq model
def download_GPTQ_model(model_name: str, models_dir: str = "./models/") -> str:
    """Download the model from hugging face repository.

    Params:
    model_name: str: the model name to download (repository name). Example: "TheBloke/Capybara
    """
    # Split the model name and create a directory name. Example: "TheBloke/CapybaraHermes-2.5-

    if not os.path.exists(models_dir):
        os.makedirs(models_dir)

    _model_name = model_name.split("/")
    _model_name = "_".join(_model_name)
    model_path = os.path.join(models_dir, _model_name)
    if _model_name not in os.listdir(models_dir):
        # download the model
```

```
        snapshot_download(
            repo_id=model_name, local_dir=model_path, local_dir_use_symlinks=False
        )
    else:
        print(f"{model_name} already exists in the models directory")

    return model_path
```

```
from exllamav2.generator import (
    ExLlamaV2Sampler,
)

settings = ExLlamaV2Sampler.Settings()
settings.temperature = 0.85
settings.top_k = 50
settings.top_p = 0.8
settings.token_repetition_penalty = 1.05

model_path = download_GPTQ_model("TheBloke/Mistral-7B-Instruct-v0.2-GPTQ")

callbacks = [StreamingStdOutCallbackHandler()]

template = """Question: {question}

Answer: Let's think step by step."""

prompt = PromptTemplate(template=template, input_variables=["question"])

# Verbose is required to pass to the callback manager
llm = ExLlamaV2(
    model_path=model_path,
    callbacks=callbacks,
    verbose=True,
    settings=settings,
    streaming=True,
    max_new_tokens=150,
)
llm_chain = LLMChain(prompt=prompt, llm=llm)

question = "What Football team won the UEFA Champions League in the year the iphone 6s was rele

output = llm_chain.invoke({"question": question})
print(output)
```

```
Building C++/CUDA extension ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 100% 0:14:01 0:00:00
Loading exllamav2_ext extension (JIT)...
/usr/local/lib/python3.11/dist-packages/huggingface_hub/utils/_auth.py:94: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (https://huggingface.co/settings/tokens), set it as secre
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.
  warnings.warn(
/usr/local/lib/python3.11/dist-packages/huggingface_hub/file_download.py:933: UserWarning: `local_dir_use_symlinks` parameter is depreca
For more details, check out https://huggingface.co/docs/huggingface_hub/main/en/guides/download#download-files-to-local-folder.
  warnings.warn(
Fetching 10 files: 100%                                    10/10 [00:26<00:00,  5.70s/it]
Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back to regular HTTP download. For better perfo
WARNING:huggingface_hub.file_download:Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back to r
special_tokens_map.json: 100%                                    72.0/72.0 [00:00<00:00, 7.10kB/s]

tokenizer.json: 100%                                    1.80M/1.80M [00:00<00:00, 6.95MB/s]

config.json: 100%                                    1.08k/1.08k [00:00<00:00, 48.2kB/s]

generation_config.json: 100%                                    111/111 [00:00<00:00, 1.90kB/s]

quantize_config.json: 100%                                    186/186 [00:00<00:00, 4.03kB/s]

README.md: 100%                                    23.1k/23.1k [00:00<00:00, 478kB/s]
Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back to regular HTTP download. For better perfo
WARNING:huggingface_hub.file_download:Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back to r
.gitattributes: 100%                                    1.52k/1.52k [00:00<00:00, 61.3kB/s]

model.safetensors: 100%                                    4.16G/4.16G [00:26<00:00, 206MB/s]

tokenizer_config.json: 100%                                    1.46k/1.46k [00:00<00:00, 143kB/s]

tokenizer.model: 100%                                    493k/493k [00:00<00:00, 6.54MB/s]
{'token_repetition_penalty': 1.05, 'token_repetition_range': -1, 'token_repetition_decay': 0, 'token_frequency_penalty': 0.0, 'token_pre
stop_sequences []
<ipython-input-8-bad80366ee82>:30: LangChainDeprecationWarning: The class `LLMChain` was deprecated in LangChain 0.1.17 and will be remo
  llm_chain = LLMChain(prompt=prompt, llm=llm)
 We know that the iPhone 6s was released on September 25, 2015. The UEFA Champions League final match is usually held in May of each yea

Let's see which team won the UEFA Champions League in May 2015 or earlier:

1. Barcelona (2014-15)
2. Real Madrid (2013-14)
3. Bayern Munich (2012-13)
4. Chelsea (2011-12)
5. Inter Milan (2009-10)

None of these teams won the UEFA Champions League in May 2015 or{'question': 'What Football team won the UEFA Champions League in the ye
```

```python
import gc

import torch

torch.cuda.empty_cache()
gc.collect()
!nvidia-smi
```

```
Thu Apr 10 03:35:20 2025
+-----------------------------------------------------------------------------------------+
| NVIDIA-SMI 550.54.15              Driver Version: 550.54.15      CUDA Version: 12.4     |
|-----------------------------------------+------------------------+----------------------+
| GPU  Name                 Persistence-M | Bus-Id          Disp.A | Volatile Uncorr. ECC |
| Fan  Temp   Perf          Pwr:Usage/Cap |           Memory-Usage | GPU-Util  Compute M. |
|                                         |                        |               MIG M. |
|=========================================+========================+======================|
|   0  Tesla T4                       Off |   00000000:00:04.0 Off |                    0 |
| N/A   47C    P0             26W /   70W |    8278MiB /  15360MiB |      6%      Default |
|                                         |                        |                  N/A |
+-----------------------------------------+------------------------+----------------------+

+-----------------------------------------------------------------------------------------+
| Processes:                                                                              |
|  GPU   GI   CI        PID   Type   Process name                              GPU Memory |
|        ID   ID                                                               Usage      |
|=========================================================================================|
+-----------------------------------------------------------------------------------------+
```