```
!apt-get update
!apt-get install -y build-essential cmake


!CMAKE_ARGS="-DGGML_CUDA=ON" pip install llama-cpp-python --no-cache-dir


! CMAKE_ARGS="-DGGML_CUDA=on"


!nvcc --version


!nvidia-smi


! pip install llama-cpp-python


! git clone https://github.com/ggerganov/llama.cpp


!pip install -r /content/llama.cpp/requirements.txt



import os
from huggingface_hub import snapshot_download
from llama_cpp import Llama


model_name = "BAAI/bge-large-en-v1.5"


base_model = "./original_model/"
quantized_model = "./quantized_model/"


snapshot_download(repo_id=model_name, local_dir=base_model , local_dir_use_symlinks=False)


original_model = quantized_model+'bge-large-en-Q4_K_M.gguf'
print(original_model)


!mkdir ./quantized_model/


!python llama.cpp/convert-hf-to-gguf.py ./original_model/ --outtype f8 --outfile ./quantized_model/bge-large-en-1.5.gguf


!python /content/llama.cpp/convert_hf_to_gguf.py ./original_model/ --outtype bf16 --outfile ./quantized_model/bge-large-en-1.5.gguf


texts = "This is an example"
model = Llama("/content/quantized_model/bge-large-en-1.5.gguf", embedding=True)
embed = model.embed(texts)
embed


import os

file_path = '/content/quantized_model/bge-large-en-1.5.gguf'

# Check if the file exists
if os.path.exists(file_path):
    # Get the file size in bytes
    file_size = os.stat(file_path).st_size
    # Convert bytes to gigabytes (GB)
    file_size_gb = file_size / (1024 * 1024 * 1024)

    print(f"Size of '{file_path}': {file_size} bytes ({file_size_gb:.2f} GB)")
else:
    print(f"File '{file_path}' not found.")


import os

file_path = '/content/original_model/onnx/model.onnx'

# Check if the file exists
if os.path.exists(file_path):
    # Get the file size in bytes
    file_size = os.stat(file_path).st_size
    # Convert bytes to gigabytes (GB)
    file_size_gb = file_size / (1024 * 1024 * 1024)
```

```
        print(f"Size of '{file_path}': {file_size} bytes ({file_size_gb:.2f} GB)")
    else:
        print(f"File '{file_path}' not found.")
```