

Compile Libtorch+MKL on WSL2

Hao Deng

December 2023

1 Introduction

We will go through how to compile Libtorch + MKL on windows in this paper. Compiling Libtorch + MKL directly on Windows will fail because it requires a Linux environment. Thus, we will compile it on WSL2 (Windows Subsystem for Linux), which provides us with a virtual Linux environment. If you haven't install WSL2 and Ubuntu, install them before proceeding.

Machine Specification:

1. Windows 10
2. WSL 2
3. Ubuntu 22.04
4. 16 GB RAM

2 Install Pytorch

In a Windows PowerShell, run `ubuntu` to enter ubuntu system.

Enter a conda environment by running: `conda activate <name>`.

Download the source code of Pytorch and update the existing checkout by running:

```
git clone --recursive https://github.com/pytorch/pytorch
cd pytorch
git submodule sync
git submodule update --init --recursive
```

Install all required dependencies: `pip install -r requirements.txt`.

3 Install MKL

Go to <https://www.intel.com/content/www/us/en/developer/tools/oneapi/onemkl-download.html>.

Select Linux operating system and offline installer, then a link to download MKL will appear (you don't have to sign up).

Below the link are the instructions to install MKL in command line, which look like the following:

Command Line Download

Command Line Installation Parameters

```
1 wget https://registrationcenter-download.intel.com/akdlm/IRC_NAS/86d6a4c1-c998-4c6b-9fff-  
ca004e9f7455/1_onemkl_p_2024.0.0.49673_offline.sh  
2  
3 sudo sh ./1_onemkl_p_2024.0.0.49673_offline.sh
```

Installation Instructions for Linux*

Step 1: From the console, locate the downloaded install file.

Step 2: Use `$ sudo sh ./<installer>.sh` to launch the GUI Installer as the root.

Optionally, use `$ sh ./<installer>.sh` to launch the GUI Installer as the current user.

Step 3: Follow the instructions in the installer.

Step 4: Explore the [Get Started Guide](#).

Follow the instructions to install MKL.

The default installation path is `:/opt/intel`.

4 Compile Libtorch with MKL

Create a script called "libtorch_mkl.sh" with the following content in the directory where Pytorch locates:

```
rm -rf build  
mkdir build  
cd build  
# Set environment variable for MKL  
export MKL_HOME=/opt/intel/oneapi/mkl/2024.0/  
# cmake  
# To compile debug version, change -DCMAKE_BUILD_TYPE:STRING=Release \ ->  
#                                     -DCMAKE_BUILD_TYPE:STRING=Debug \  
cmake -G Ninja \  
-DBUILD_PYTHON=OFF \  
-DUSE_CUDA=OFF \  
-DUSE_CUDNN=OFF \  
-DUSE_NCCL=OFF \  
-DUSE_OPENMP=ON \  
-DUSE_BLAS=ON \  
-DUSE_XNNPACK=OFF \  
-DBUILD_CAFFE2=OFF \  
-DUSE_FBGEMM=OFF \  
-DUSE_PYTORCH_QNNPACK=OFF \  
-DUSE_EIGEN_FOR_BLAS=OFF \  
-DUSE_QNNPACK=OFF \  
-DUSE_KINETO=OFF \  
-DUSE_NNPACK=OFF \  
-DUSE_MKL=ON \  
-DUSE_MKLDNN=OFF \  
-DBUILD_CAFFE2_OPS=OFF \  
-DBUILD_SHARED_LIBS:BOOL=ON \  
-DCMAKE_BUILD_TYPE:STRING=Release \  
-DCMAKE_INSTALL_PREFIX:PATH=../libtorch_mkl \  
-DCMAKE_C_COMPILER=/usr/bin/gcc \  

```

```
-DCMAKE_CXX_COMPILER=/usr/bin/g++ \  
../pytorch  
ninja install
```

Run the script by: `bash libtorch_mkl.sh`.

After finishing running the script, a resulting directory called "libtorch_mkl" will be created and we finished compiling libtorch with MKL on WLS2.