

ADAMS

Advanced **D**ata mining **A**nd **M**achine learning **S**ystem

Module: adams-cntk

CNTK

Peter Reutemann Dale Fletcher

December 20, 2017

©2017



THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato



Except where otherwise noted, this work is licensed under
<http://creativecommons.org/licenses/by-sa/4.0/>

Contents

1	CNTK	5
1.1	Installation	5
2	Flow	7
3	Open MPI	9
4	Intel MKL	11
	Bibliography	13

Chapter 1

CNTK

CNTK, the Microsoft Cognitive Toolkit[2], is a deeplearning C++ library which can be configured via the BrainScript Network Builder¹ or Python. It comes with binaries for 64-bit Windows and Linux. It has experimental bindings for Java as well, but only for prediction time.

1.1 Installation

Follow the installation instructions for CNTK:

<https://docs.microsoft.com/en-us/cognitive-toolkit/Setup-CNTK-on-your-machine>

For the Java bindings to work, you need to do the following:

- Download binaries for your platform.
- Create a new directory to contain all the libraries
- Copy the libraries from the *lib* and *dependencies/lib* directories into the newly created directory. For older Linux versions that required a custom build of the openmpi libraries, copy these libraries as well.
- Either supply the new library directory to the JVM using *-Djava.library.path=...*, or, when using the *Launcher* class, (package *adams.core.management*), you can use the *ADAMS_LIBRARY_PATH* environment variable to supply the directory. The launcher will automatically add the content of the variable to the command-line.

¹<https://docs.microsoft.com/en-us/cognitive-toolkit/BrainScript-Network-Builder>

Chapter 2

Flow

The following standalines are available:

- *CNTKSetup* – for overriding the global CNTK settings.

The following sources are available:

- *CNTKModelGenerator* – uses the specified model generator to output model specification strings.

The following transformers are available:

- *CNTKModelApplier* – uses the specified model applier to the incoming data.
- *CNTKModelInfo* – outputs information about the incoming model.
- *CNTKModelReader* – reads a CNTK model from disk.

Chapter 3

Open MPI

When installing the LLVM version of the Open MPI library on Ubuntu (and derivatives), CNTK will not find the `libiomp5.so` library, as it is called `libomp.so.5`. You can simply symlink the library as follows:

```
cd /usr/lib/x86_64-linux-gnu
sudo ln -s libomp.so.5 libiomp5.so
```


Chapter 4

Intel MKL

Intel provides high-performance BLAS libraries for download as well[?]:

After installation, you have to add the libraries to your environment variables for deeplearning4j to pick up.

First, locate the directory that contains the MKL runtime library:

- Linux/OSX: `libmkl_rt.so`
- Windows: `mkl_rt.dll`

Second, add this directory to your environment variables:

- Linux/OSX: add the path to `LD_LIBRARY_PATH`

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/path/to/mkl_rt
```

- Windows: add the path to your `%PATH%` variable, either through the control panel or on the command prompt:

```
set PATH=%PATH%;path\to\mkl_rt
```

Furthermore, for Windows you also need to add the OpenMP runtime libraries to the path. Locate the directory that contains the `libiomp5md.dll` library and add this directory to your `%PATH%` environment variable as well.

Bibliography

- [1] *ADAMS* – Advanced Data mining and Machine learning System
<https://adams.cms.waikato.ac.nz/>
- [2] *CNTK* – Microsoft Cognitive Toolkit (CNTK), an open source deep-learning toolkit.
<https://github.com/Microsoft/CNTK>