Neural Turing Machine - OpenNN

QueenField
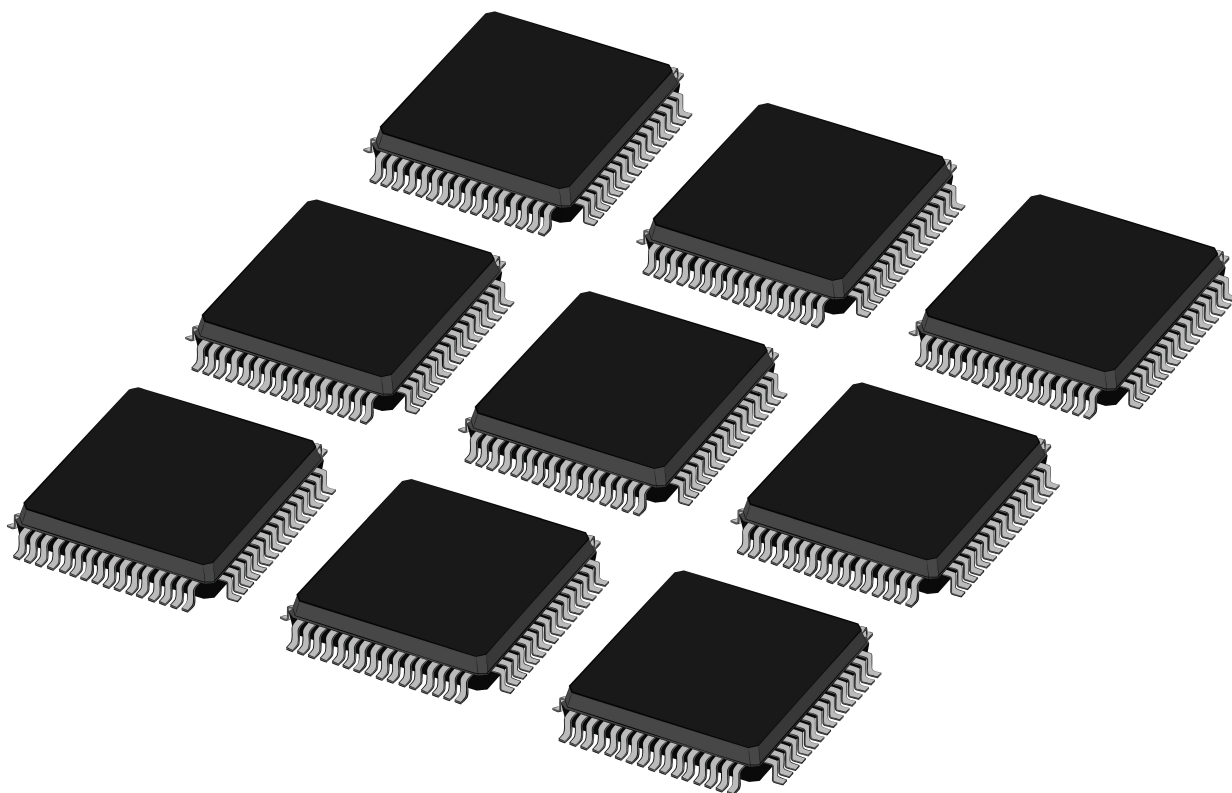

Figure 1: QueenField

1. INTRODUCTION

OpenNN is a software library written in C++ for advanced analytics. It implements neural networks, the most successful machine learning method.

- The main advantage of OpenNN is its high performance.

- This library outstands in terms of execution speed and memory allocation. It is constantly optimized and parallelized in order to maximize its efficiency.

- Some typical applications of OpenNN are business intelligence (customer segmentation, churn prevention. . .), health care (early diagnosis, microarray analysis. . .) and engineering (performance optimization, predictive maitenance. . .).

The documentation is composed by tutorials and examples to offer a complete overview about the library.

- The documentation can be found at the official OpenNN site.

- CMakeLists.txt are build files for CMake, it is also used by the CLion IDE.

- The .pro files are project files for the Qt Creator IDE, which can be downloaded from its site. Note that OpenNN does not make use of the Qt library.

OpenNN is developed by Artelnics, a company specialized in artificial intelligence.

For Windows Users!

1. Settings → Apps → Apps & features → Related settings, Programs and Features → Turn Windows features on or off → Windows Subsystem for Linux

2. Microsoft Store → INSTALL UBUNTU

type:

```
sudo apt update
sudo apt upgrade
```

2. PROJECTS

2.1. OpenNN Examples

2.1.1. simple function regression

type:

```
source BUILD-x86
./simple_function_regression-x86.run
```

2.1.2. simple pattern recognition

type:

```
source BUILD-x86
./simple_pattern_recognition-x86.run
```

2.1.3. airfoil self noise

type:

```
source BUILD-x86
./airfoil_self_noise-x86.run
```

2.1.4. airline passengers

type:

```
source BUILD-x86
./airline_passengers-x86.run
```

2.1.5. breast cancer

type:

```
source BUILD-x86
./breast_cancer-x86.run
```

2.1.6. iris plant

type:

```
source BUILD-x86
./iris_plant-x86.run
```

2.1.7. logical operations

type:

```
source BUILD-x86
./logical_operations-x86.run
```

2.1.8. pima indians diabetes

type:

```
source BUILD-x86
./pima_indians_diabetes-x86.run
```

2.1.9. urinary inflammations diagnosis

type:

```
source BUILD-x86
./urinary_inflammations_diagnosis-x86.run
```

2.1.10. yacht hydrodynamics design

type:

```
source BUILD-x86
./yacht_hydrodynamics_design-x86.run
```

2.1.11. yacht hydrodynamics production

type:

```
source BUILD-x86
./yacht_hydrodynamics_production-x86.run
```

2.1.12. leukemia

type:

```
source BUILD-x86
./leukemia-x86.run
```

2.1.13. pollution forecasting

type:

```
source BUILD-x86
./pollution_forecasting-x86.run
```

2.1.14. temperature forecasting

type:

```
source BUILD-x86
./temperature_forecasting-x86.run
```

2.1.15. mnist

type:

```
source BUILD-x86
./mnist-x86.run
```

2.2. Neural Turing Machine

2.2.1. PU

type:

```
source BUILD-x86
./PU-x86.run
```

2.2.2. SoC

type:

```
source BUILD-x86
./SoC-x86.run
```

2.2.3. MPSoC

type:

```
source BUILD-x86
./MPSoC-x86.run
```

3. WORKFLOW

3.1. Hardware

3.1.1. RISC-V GNU C/C++

type:

```
sudo apt install autoconf automake autotools-dev curl python3 libmpc-dev \
libmpfr-dev libgmp-dev gawk build-essential bison flex texinfo gperf \
libtool patchutils bc zlib1g-dev libexpat-dev
```

type:

```
git clone --recursive https://github.com/riscv/riscv-gnu-toolchain

cd riscv-gnu-toolchain

./configure --prefix=/opt/riscv-elf-gcc
sudo make clean
sudo make

./configure --prefix=/opt/riscv-elf-gcc
sudo make clean
sudo make linux

./configure --prefix=/opt/riscv-elf-gcc --enable-multilib
sudo make clean
sudo make linux
```

3.1.2. OpenRISC GNU C/C++

type:

```
sudo apt install git libgmp-dev libmpfr-dev libmpc-dev zlib1g-dev texinfo \
build-essential flex bison
```

type:

```
git clone git://sourceware.org/git/binutils-gdb.git binutils
git clone https://github.com/openrisc/or1k-gcc.git gcc
git clone git://sourceware.org/git/newlib-cygwin.git newlib
git clone git://sourceware.org/git/binutils-gdb.git gdb

export PATH=/opt/or1k-elf-gcc/bin:${PATH}

mkdir build-binutils; cd build-binutils
../binutils/configure --target=or1k-elf --prefix=/opt/or1k-elf-gcc \
--disable-itcl --disable-tk --disable-tcl --disable-winsup --disable-gdbtk \
--disable-libgui --disable-rda --disable-sid --disable-sim --disable-gdb \
--with-sysroot --disable-newlib --disable-libgloss --with-system-zlib
make
sudo make install
cd ..

mkdir build-gcc-stage1; cd build-gcc-stage1
../gcc/configure --target=or1k-elf --prefix=/opt/or1k-elf-gcc \
--enable-languages=c --disable-shared --disable-libssp
make
sudo make install
cd ..

mkdir build-newlib; cd build-newlib
../newlib/configure --target=or1k-elf --prefix=/opt/or1k-elf-gcc
make
sudo make install
cd ..

mkdir build-gcc-stage2; cd build-gcc-stage2
../gcc/configure --target=or1k-elf --prefix=/opt/or1k-elf-gcc \
--enable-languages=c,c++ --disable-shared --disable-libssp --with-newlib
make
sudo make install
cd ..

mkdir build-gdb; cd build-gdb
../gdb/configure --target=or1k-elf --prefix=/opt/or1k-elf-gcc --disable-itcl \
--disable-tk --disable-tcl --disable-winsup --disable-gdbtk --disable-libgui \
--disable-rda --disable-sid --with-sysroot --disable-newlib --disable-libgloss \
--disable-gas --disable-ld --disable-binutils --disable-gprof --with-system-zlib
make
sudo make install
cd ..
```

3.2. Software

3.2.1. Spike Simulator for RISC-V

type:

```
export PATH=/opt/riscv-elf-gcc/bin:${PATH}

git clone --recursive https://github.com/riscv/riscv-isa-sim

cd riscv-isa-sim
mkdir build
cd build
../configure --prefix=/opt/riscv-elf-gcc
make
sudo make install
```

3.2.2. Or1ksim Simulator for OpenRISC

type:

```
git clone --recursive https://github.com/openrisc/or1ksim

cd or1ksim
mkdir build
cd build
../configure --prefix=/opt/or1ksim --program-prefix=or1k-
make
sudo make install
```

4. CONCLUSION

4.1. OpenNN Examples

4.1.1. simple function regression

type:

```
source BUILD-RISCV
spike pk simple_function_regression-riscv.elf
```

4.1.2. simple pattern recognition

type:

```
source BUILD-RISCV
spike pk simple_pattern_recognition-riscv.elf
```

4.1.3. airfoil self noise

type:

```
source BUILD-RISCV
spike pk airfoil_self_noise-riscv.elf
```

4.1.4. airline passengers

type:

```
source BUILD-RISCV
spike pk airline_passengers-riscv.elf
```

4.1.5. breast cancer

type:

```
source BUILD-RISCV
spike pk breast_cancer-riscv.elf
```

4.1.6. iris plant

type:

```
source BUILD-RISCV
spike pk iris_plant-riscv.elf
```

4.1.7. logical operations

type:

```
source BUILD-RISCV
spike pk logical_operations-riscv.elf
```

4.1.8. pima indians diabetes

type:

```
source BUILD-RISCV
spike pk pima_indians_diabetes-riscv.elf
```

4.1.9. urinary inflammations diagnosis

type:

```
source BUILD-RISCV
spike pk urinary_inflammations_diagnosis-riscv.elf
```

4.1.10. yacht hydrodynamics design

type:

```
source BUILD-RISCV
spike pk yacht_hydrodynamics_design-riscv.elf
```

4.1.11. yacht hydrodynamics production

type:

```
source BUILD-RISCV
spike pk yacht_hydrodynamics_production-riscv.elf
```

4.1.12. leukemia

type:

```
source BUILD-RISCV
spike pk leukemia-riscv.elf
```

4.1.13. pollution forecasting

type:

```
source BUILD-RISCV
spike pk pollution_forecasting-riscv.elf
```

4.1.14. temperature forecasting

type:

```
source BUILD-RISCV
spike pk temperature_forecasting-riscv.elf
```

4.1.15. mnist

type:

```
source BUILD-RISCV
spike pk mnist-riscv.elf
```

4.2. Neural Turing Machine

4.2.1. PU

type:

```
source BUILD-RISCV
spike pk PU-riscv.elf
```

4.2.2. SoC

type:

```
source BUILD-RISCV
spike pk SoC-riscv.elf
```

4.2.3. MPSoC

type:

```
source BUILD-RISCV
spike pk MPSoC-riscv.elf
```