Fast Inference of Machine Learning Models with SOFIE

Source Code

Lorenzo Moneta, Sanjiban Sengupta, CERN, Geneva, Switzerland
Ioanna-Maria Panagou, University of Thessaly, Volos, Greece

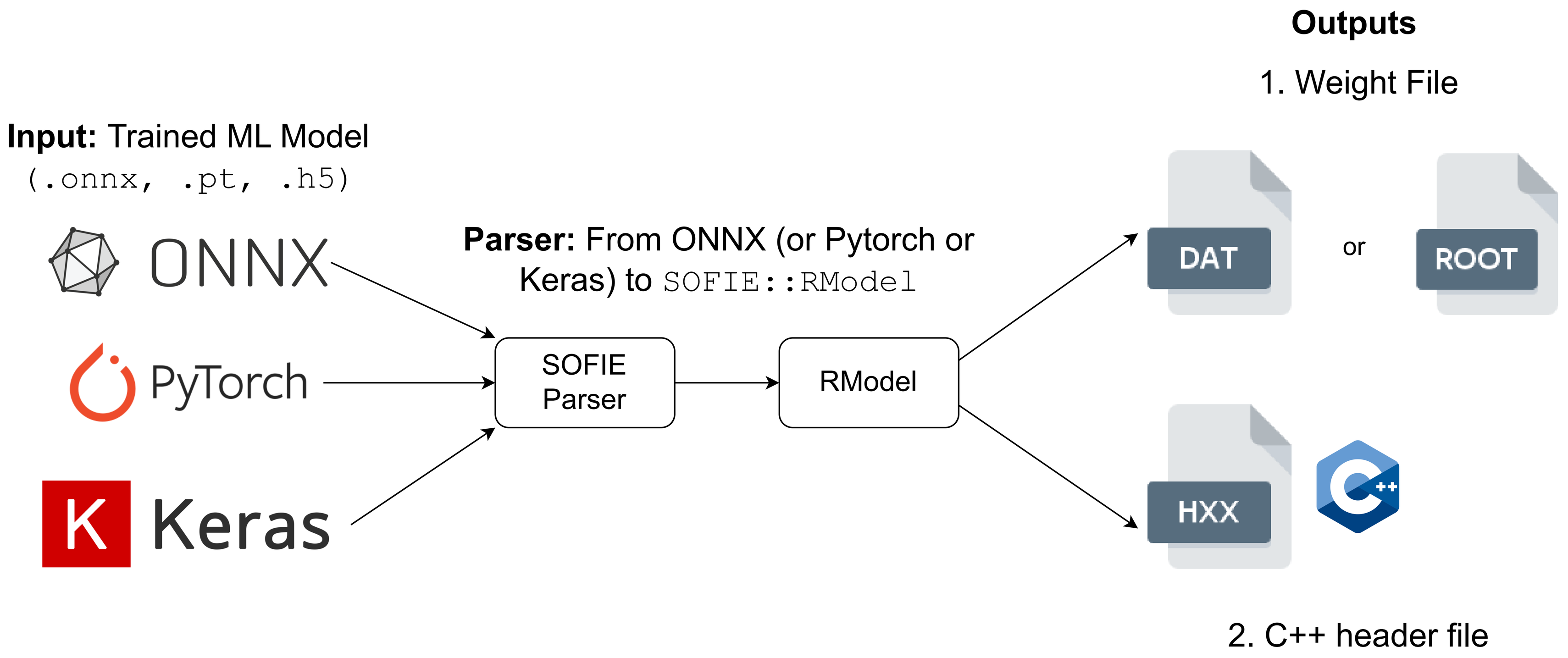
root.cern

Motivation

- Popular machine learning libraries, such as Keras and PyTorch, provide functionality for inference, but **support only their own models** and are constrained by **heavy dependencies**
- SOFIE [2] creates standalone C++ inference code for a model with **limited dependencies** (only on BLAS libraries), which can be included in any other C++ project.
- SOFIE supports several types of deep learning models, including now message passing **GNN**.
- SOFIE can generate **SYCL** code that can run on various GPUs and is dependent only on Intel MKL BLAS and portBLAS libraries.

Description

- SOFIE accepts input in the form of a pre-trained machine learning model, presented in `.onnx`, `.pt` or `.h5` format and transforms the input model into an equivalent graph of operators.
- The code generation step produces a **C++ header file** with the inference function in C++ and a weight file in `.dat` or `.root` format.



- The generated code can be easily **integrated in C++ applications** or compiled on the flight using the ROOT JIT capabilities of CLING and **used in Python code**.
- The model can also be evaluated within ROOT **RDataFrame**.

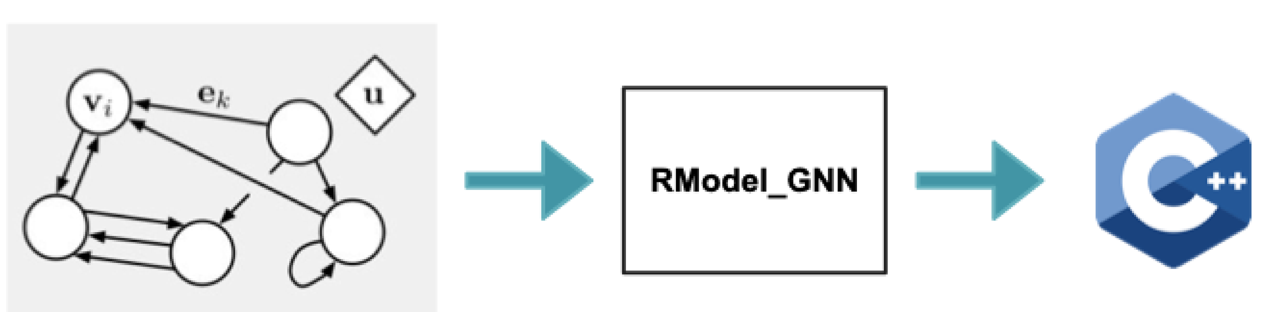
ONNX Supported Operators

Operators implemented in ROOT	CPU	GPU
Perceptron (GEMM)	✓	✓
Convolution (1D, 2D, 3D)	✓	✓
DeConvolution (1D, 2D, 3D)	✓	✓
Recurrent (RNN, GRU, LSTM)	✓	
Activations (Relu, Selu, Swish, LeakyRelu, Tanh,...)	✓	✓
Pooling (MaxPool, AveragePool,...)	✓	✓
BatchNorm, LayerNorm	✓	✓
Binary Op (Add, Sum, Mul, Div,..)	✓	✓
Unary Op (Neg,Sqrt,Exp,..)	✓	✓
Reshape, Flatten, Concat, Reduce, Gather	✓	✓
Transpose, Slice, Squeeze, Unsqueeze	✓	✓
Custom operator	✓	

Support for Missing operators can be added on user requests

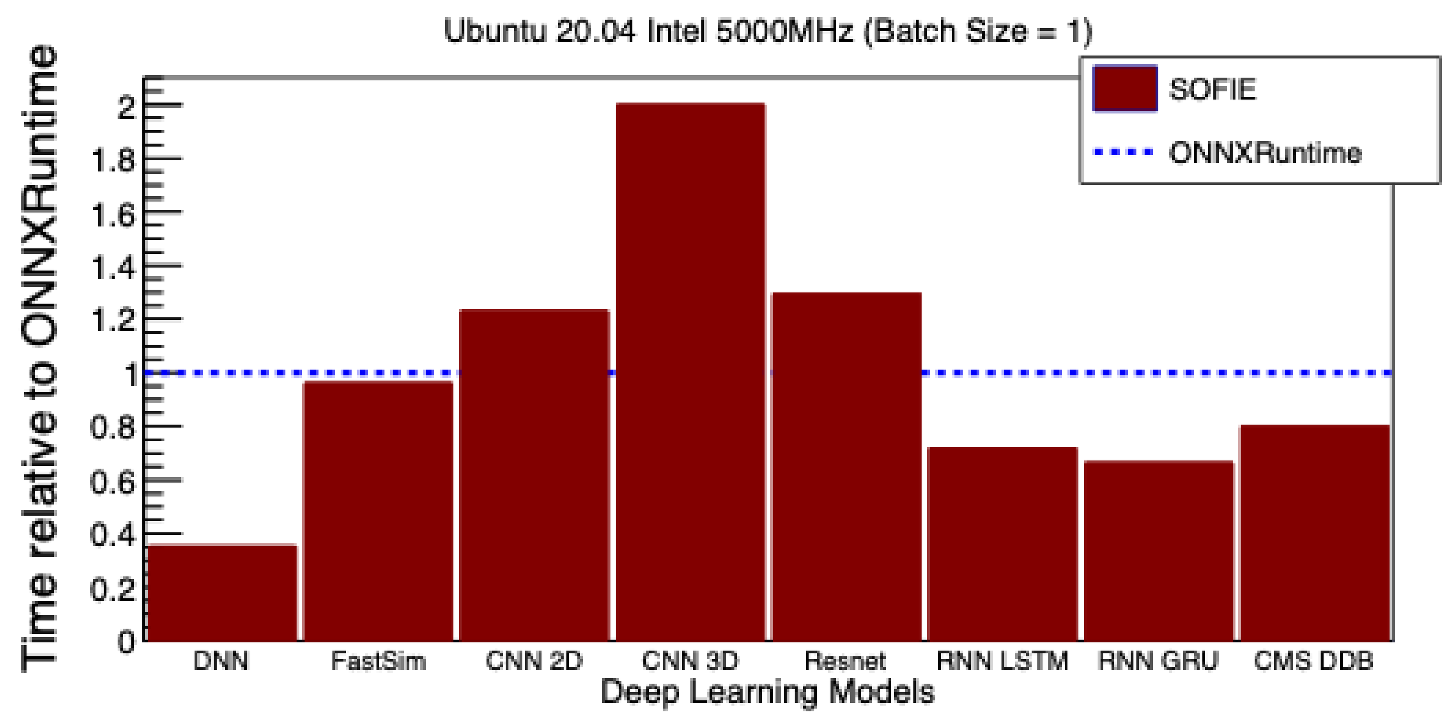
GNN Support

- SOFIE can generate C++ code from GNN models based on the Graph Nets library [1]

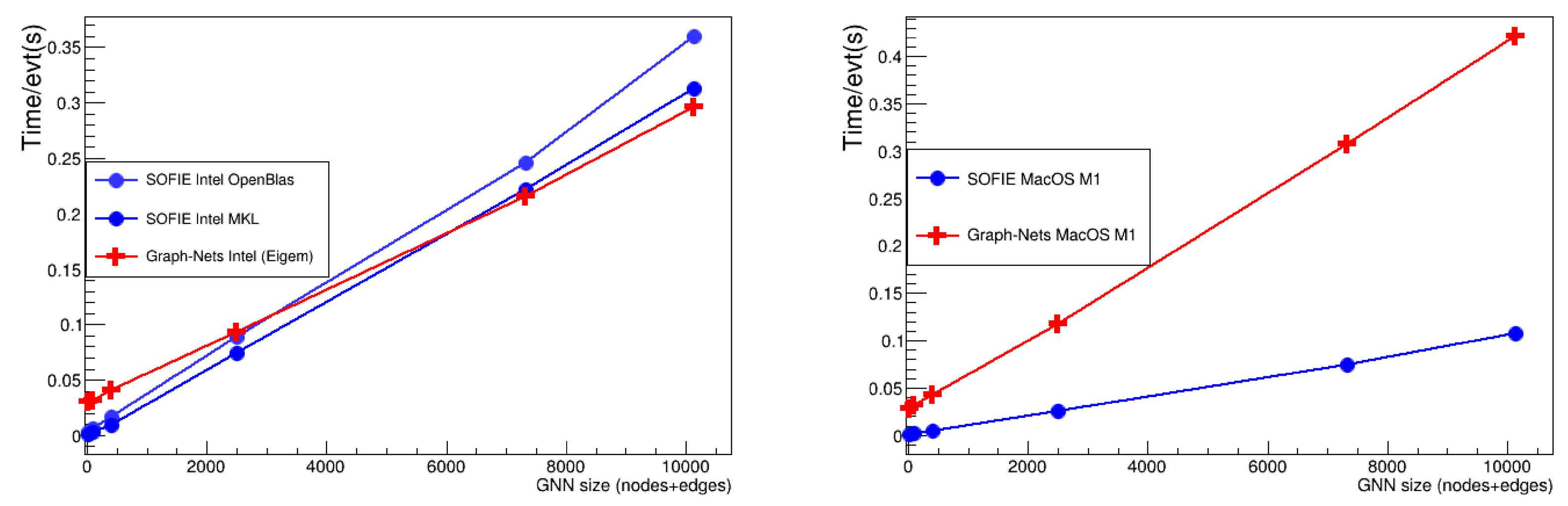


Benchmarks on CPU

- We tested 8 different deep learning models with various complexity.
- We compare the CPU time to evaluate the models using the C++ code generated by **SOFIE** or by using directly **ONNXRuntime**.

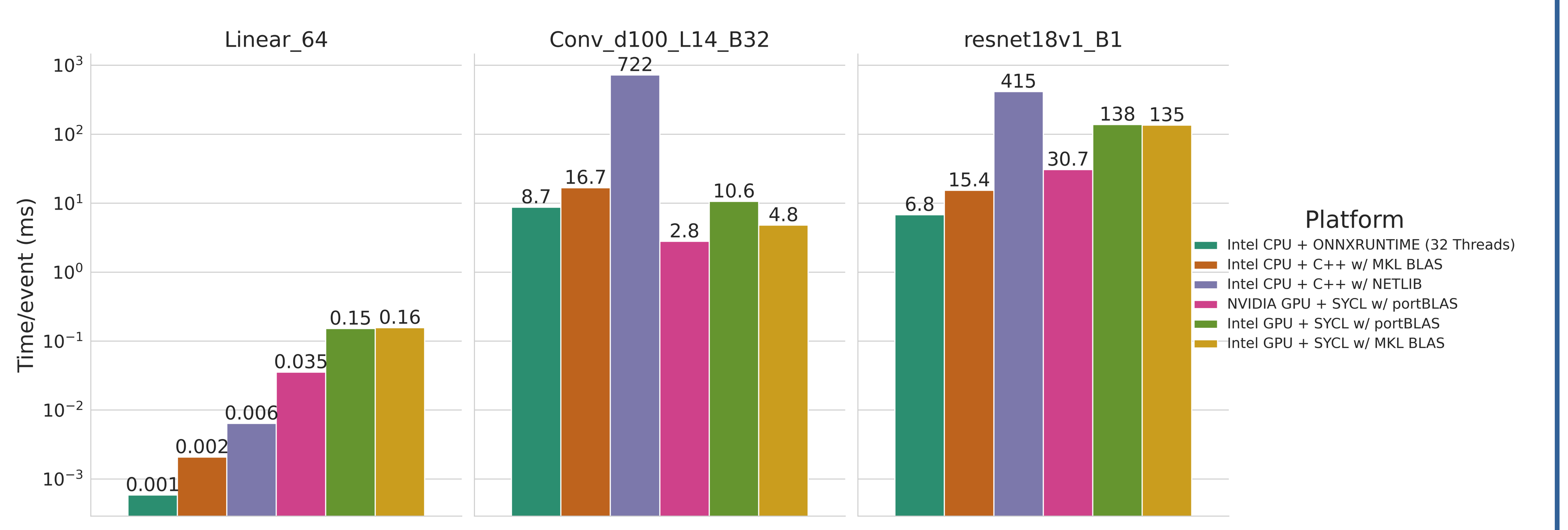


- We tested also the CPU performance for **GNN models** varying the number of nodes and edges on Linux and MacOS architectures

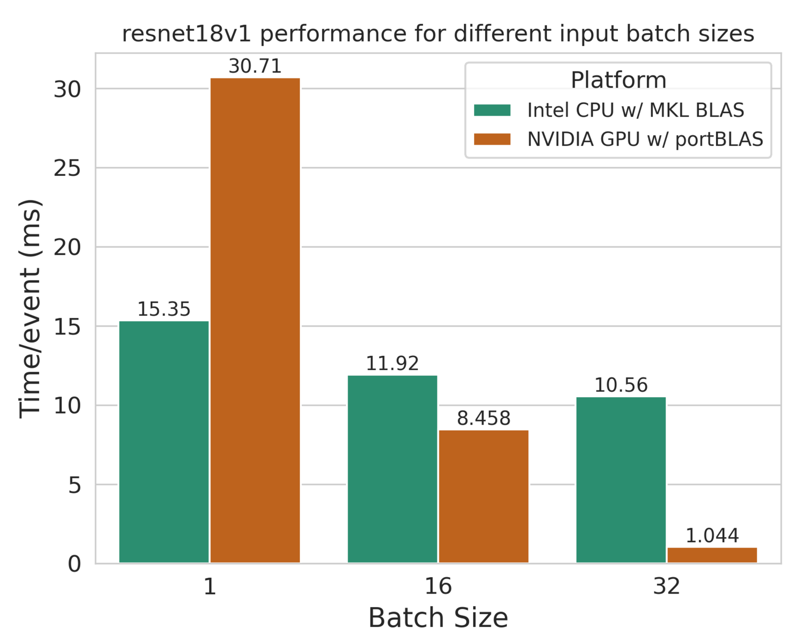


Benchmarks on GPU

- We tested 6 different configurations for various platforms and execution backends using the **SYCL** code extension of SOFIE.



- There is significant **correlation between performance improvement and model size**.
  - Models with fewer layers and lower computational complexity, such as Linear_64 exhibit inferior performance on GPU compared to models with more extensive layer counts, such as Convolutional or resnet models.



- The performance for the same model (resnet18v1_81) varies considerably with the input **batch size**.

KEY REFERENCES

[1] DeepMind. *Graph Nets Library*. URL: `https://github.com/google-deepmind/graph_nets`.

[2] Lorenzo Moneta Sitong An et al. "SOFIE: C++ Code Generation for Fast Inference of Deep Learning Models in ROOT/TMVA". In: *Journal of Physics: Conference Series* 2438.1 (Feb. 2023), p. 012013. DOI: `10.1088/1742-6596/2438/1/012013`. URL: `https://dx.doi.org/10.1088/1742-6596/2438/1/012013`.

ACKNOWLEDGEMENTS

CERN

MORE INFORMATION

Lorenzo Moneta
CERN
Lorenzo.Moneta@cern.ch