Root Storage of Deep Learning Models in TMVA

Sub-Org: Root-Project

Mentors: Lorenzo Moneta, Sitong An, Anirudh Dagar

SANJIBAN SENGUPTA

github.com/sanjibansg

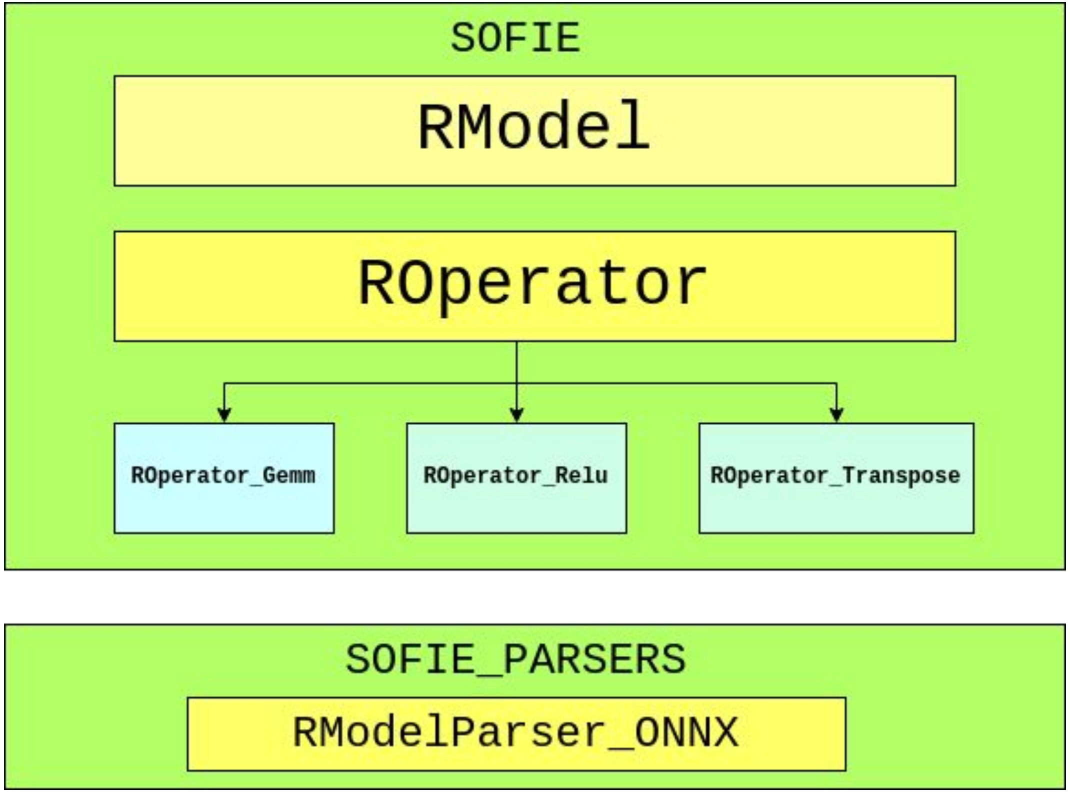
✉ sanjiban.sg@gmail.com

About ROOT TMVA

TMVA

- Toolkit for Multivariate Analysis
- Provides a Machine Learning environment for training, testing and evaluation of multivariate methods
- **Latest Development**
    - **SOFIE** *(System for Optimized Fast Inference code Emit)*
        - Intermediate Representation of trained deep learning models
            - Stores configuration & model weights of trained deep learning models following ONNX standards
        - Generation of Fast inference code having least latency and few dependencies.
            - Takes models as inputs and produces C++ header files containing easily invokable functions that can be included and utilized in a “plug-and-go” style.

About TMVA SOFIE



SOFIE

RModel

ROperator

ROperator_Gemm

ROperator_Relu

ROperator_Transpose

SOFIE_PARSERS

RModelParser_ONNX

Project Description

Executed Tasks

- Serialization of RModel
- SOFIE Converter for Keras
- SOFIE Converter for PyTorch
- Tests & Tutorials

Serializing

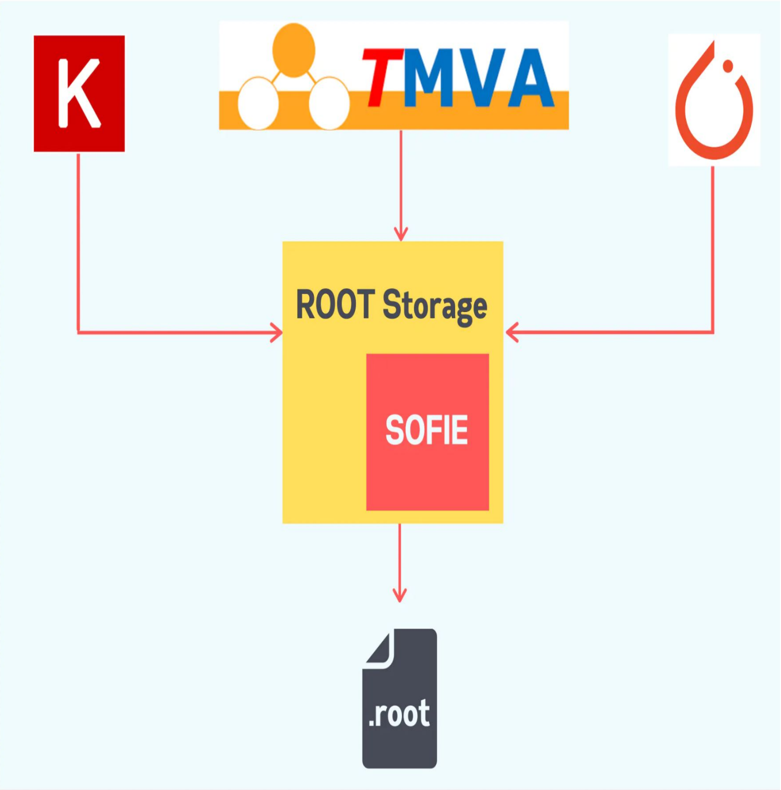
```
 [+] Modified RModel, ROperator
```

Keras Converter

```
 [+] SOFIE::PyKeras::Parse()
 [+] SOFIE::PyKeras::ConvertToRoot()
```

PyTorch Converter

```
 [+] SOFIE::PyTorch::Parse()
 [+] SOFIE::PyTorch::ConvertToRoot()
```



Serialization of RModel

Interface for writing root file

```
TFile file("model.root","CREATE");
using namespace TMVA::Experimental;
SOFIE::RModelParser_ONNX Parser;
SOFIE::RModel model = Parser.Parse("./example_model.onnx");
model.Write("model");
file.Close();
```

Interface for reading root file

```
TFile file("model.root","READ");
using namespace TMVA::Experimental;
SOFIE::RModel *model;
file.GetObject("model",model);
file.Close();
```

SOFIE Keras Converter

Interface

```
using TMVA::Experimental::SOFIE;

//Parser returns a RModel object
RModel model = PyKeras::Parse("trained_model_dense.h5");

//Converter writes a ROOT file directly
PyKeras::ConvertToRoot("trained_model_dense.h5");
```

SOFIE PyTorch Converter

Interface

```
using TMVA::Experimental::SOFIE;

//Building the vector for input shapes
std::vector<size_t> s1{120,1};
std::vector<std::vector<size_t>> inputShape{s1};

//Parser returns a RModel object
RModel model = PyTorch::Parse("trained_model_dense.pt",inputShape);


//Converter write3s a ROOT file directly
PyTorch::ConvertToRoot("trained_model_dense.pt",inputShape);
```

ROOT Storage of BDT

Expected Tasks

- Class for intermediate representation of TMVA trained BDT models
- Parse function for translating .xml files generated after training
- Mapping interface to TMVA Tree Inference

Interface

```
using TMVA::Experimental;

RootStorage::BDT model;
bool usePurity = true;

//Parser loads the BDT model from .xml to RootStorage::BDT object
model.Parse("BDT_Model.weights.xml",usePurity);
```

Conclusion

- Project Page
  summerofcode.withgoogle.com/projects/#5424575602491392
- Final Report
  github.com/sanjibansg/GSoC21-RootStorage/wiki
- Code Implementations
  github.com/root-project/root/pulls?q=author:sanjibansg
- Documentation Blog
  blog.sanjiban.ml/series/gsoc