TMVA Deep Learning Developments - Inference Code Generation for Recurrent Neural Networks

Google Summer of Code 2021

Ahmat Hamdan

Mentors: Lorenzo Moneta, Sitong An

Deep Learning inference

In deep learning, there are two processes

- Training
- Inference

Deep learning frameworks are not ideal for inference.

In the machine learning comunity there is a focus on fast inference.
That's why

- ONNX (Open Neural Network Exchange)
- Inference Engines (ONNX Runtime, TensorRT, ...)

are under active development.

TMVA Inference Engine

In the TMVA (Toolkit for Multivariable Analysis) team, a fast inference engine that

- Takes ONNX files as input

and

- Produces a C++ script as output

Is under development.

See `https://github.com/root-project/root/pull/7544`

Goal of the project

Goal

- Implementation of recurrent neural networks operators as defined by the ONNX standards in the code generation format

Deliverables

- Production ready implementation of RNN, LSTM and GRU in the code generation format.
- Optimized implementation for CPU (using BLAS).

Recurrent Neural Networks (RNN)

Powerful neural network architecture for

- Temporal processing
- Sequential learning

The architecture of RNN

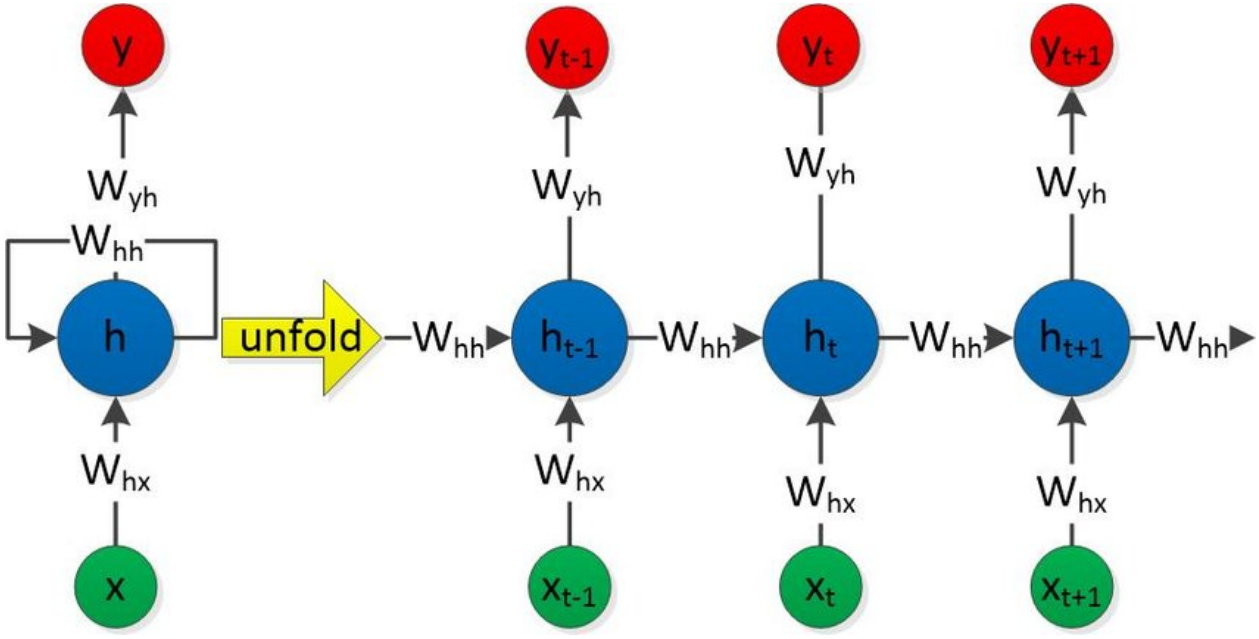

Figure: Jian Zheng 2017

RNN Equation

$$h_t = \sigma(x_t w_{xh}^T + h_{t-1} w_{hh}^T + b_h)$$

Where

- h_t, h_{t-1}: Hidden state
- x_t: Input matrix
- w_{xh}, w_{hh}: Weights
- b_h: Bias
- σ: Activation function
- t: Time step

Implementation of RNN 1/2

- The ROperator_RNN class is a derived from ROperator.
- The attributes fAttrActivationAlpha, fAttrActivationBeta, fAttrActivations, fAttrClip, fAttrDirection, fAttrHiddenSize and fAttrLayout.
- The names fNX, fNW, fNR, fNR, fNB, fNSequence_lens, fNInitial_h, fNY and fNY_h
- The shapes fShapeX, fShapeW, fShapeR, fShapeB, fShapeSequence_lens, fShapeInitial_h, fShapeY and fShapeY_h.
- The precision fType.

Implementation of RNN 2/2

- ROperator_RNN(...) is a constructor.
- TypeInference(...) infers the type of the tensors.
- ShapeInference(...) infers the shape of the tensors.
- Initialize(...) initializes the model.
- Generate(...) generates the code of the RNN operator.

LSTM Equation

$$
\begin{aligned}
i_t &= \sigma(x_t w_{xi}^T + h_{t-1} w_{hi}^T + b_i) \\
f_t &= \sigma(x_t w_{xf}^T + h_{t-1} w_{hf}^T + b_f) \\
o_t &= \sigma(x_t w_{xo}^T + h_{t-1} w_{ho}^T + b_o) \\
\tilde{c}_t &= \sigma(x_t w_{xc}^T + h_{t-1} w_{hc}^T + b_c) \\
c_t &= f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \\
h_t &= o_t \odot \phi(c_t)
\end{aligned}
$$

Where

- i_t: Input gate
- f_t: Forget gate
- o_t: Output gate
- $\tilde{c}_t, c_t$: Cell state
- h_t, h_{t-1}: Hidden state
- x_t: Input matrix
- $w_{xi}, w_{xf}, w_{xo}, \ldots$: Weights
- b_i, b_f, b_o, b_c: Bias
- σ, ϕ: Activation functions
- t: Time step

Implementation of LSTM

- The ROperator_LSTM class is derived from ROperator.
- The attributes fAttrActivationAlpha, fAttrActivationBeta, fAttrActivations, fAttrClip, fAttrDirection, fAttrHiddenSize, fInputForget and fAttrLayout
- The names fNX, fNW, fNR, fNR, fNB, fNSequence_lens, fNInitial_h, fNInitial_c, fNP, fNY, fNY_h, and Y_c.
- The shapes fShapeX, fShapeW, fShapeR, fShapeB, fShapeSequence_lens, fShapeInitial_h, fShapeInitial_c, fShapeP, fShapeY, fShapeY_h and fShapeY_c.
- The precision fType.

GRU Equation

$$z_t = \sigma(x_t w_{xz}^T + h_{t-1} w_{hz}^T + b_z)$$
$$r_t = \sigma(x_t w_{xr}^T + h_{t-1} w_{hr}^T + b_r)$$
$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \phi(x_t w_{xh} + (r_t \odot h_{t-1}) w_{hh}^T + b_h)$$

Where

- x_t: Input matrix
- z_t: Update gate
- r_t: Reset gate
- h_t, h_{t-1}: Hidden state
- σ, ϕ: Activation functions
- $w_{xz}, w_{hz}, w_{xr}, w_{hr}, w_{xh}, w_{hh}$: Weights
- b_z, b_r, b_h: Bias
- t: Time step

Implementation of GRU

- The ROperator_GRU class is derived from ROperator.
- The attributes fAttrActivationAlpha, fAttrActivationBeta, fAttrActivations, fAttrClip, fAttrDirection, fAttrHiddenSize, fAttrLayout and fAttrLinearBeforeReset.
- The names fNX, fNW, fNR, fNR, fNB, fNSequence_lens, fNInitial_h, fNY and fNY_h.
- The shape of the tensors fShapeX, fShapeW, fShapeR, fShapeB, fShapeSequence_lens, fShapeInitial_h, fShapeY and fShapeY_h.
- The precision fType.

Parse the RNN, LSTM and GRU nodes

- make_ROperator_RNN(...) parses the RNN node
- make_ROperator_LSTM(...) parses the LSTM node
- make_ROperator_GRU(...) parses the GRU node

Thank you!

https://github.com/axmat/TMVAFastInferencePrototype
https://github.com/axmat/TMVAInference