Event reweighting with particle transformer networks

Andrew Cudd[1], Roger Huang[3], Masaki Kawaue[2], Ben Nachman[4], Tatsuya Kikawa[2], Vinicius Mikuni[3], Callum Wilkinson[3]

[1]University of Colorado Boulder, [2]Kyoto University, [3]Lawrence Berkeley National Lab., [4]SLAC National Accelerator Lab.

Introduction: Often in particle physics it is useful to transform one distribution into another using weights applied to events, commonly referred to as reweighting. Commonly this reweighting function is restricted in the amount of information used to generate weights, limiting the accuracy to either specific regions of phase space or to only a few variables or dimensions. We present a method based on Deep neural networks using Classification for Tuning and Reweighting (DCTR) [1] using a transformer-based neural network architecture to perform reweighting utilizing the kinematic and flavor information of all particles.

Calculating Event Weights:

- Train a model to classify between two datasets **A** & **B** using the weighted cross-entropy loss (L)
- Each event **x** has a true label **p** and receives a prediction **q** from the network

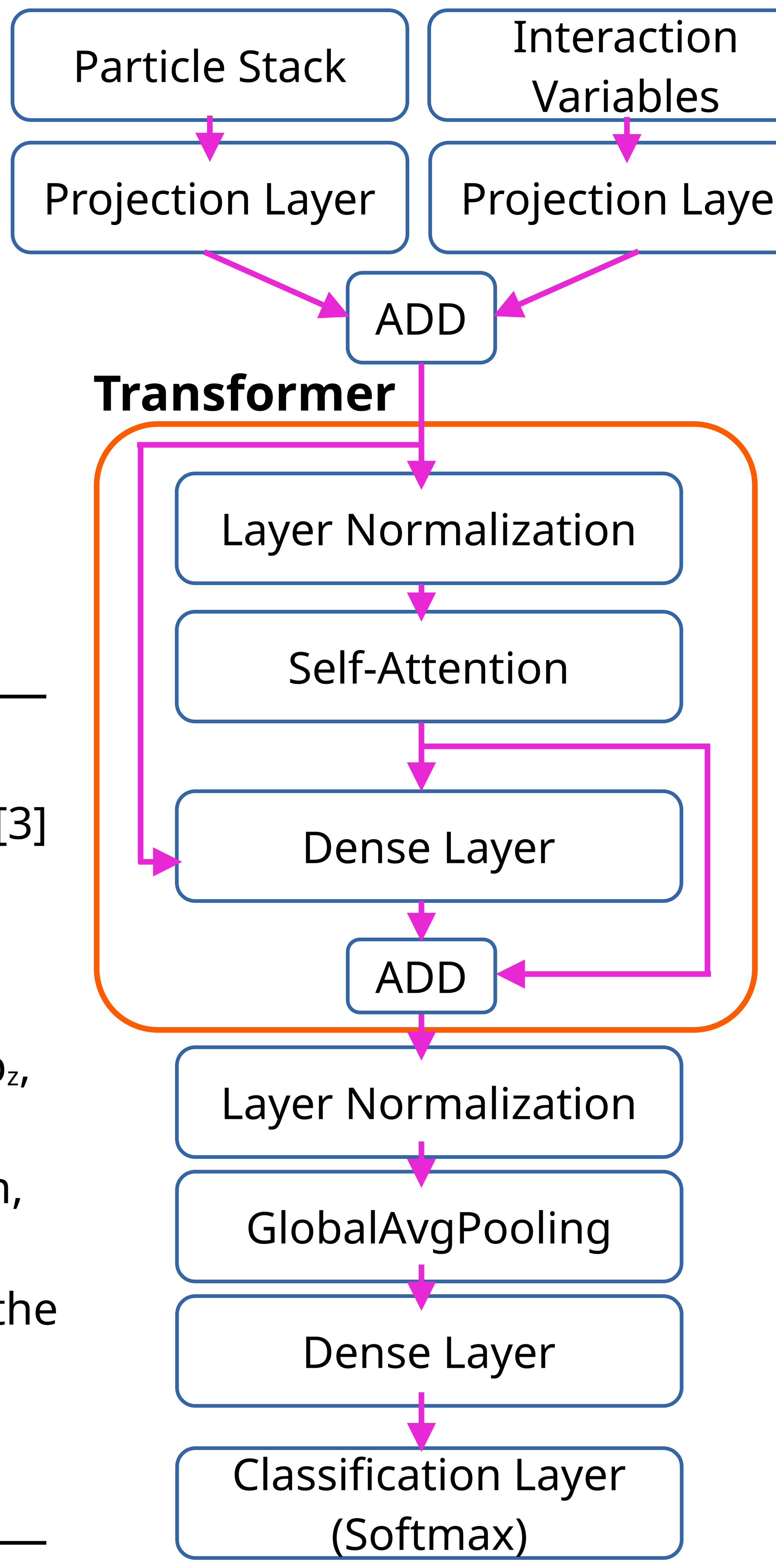
$$L_i = p_i \log q_i + (1 - p_i)\log(1 - q_i)$$

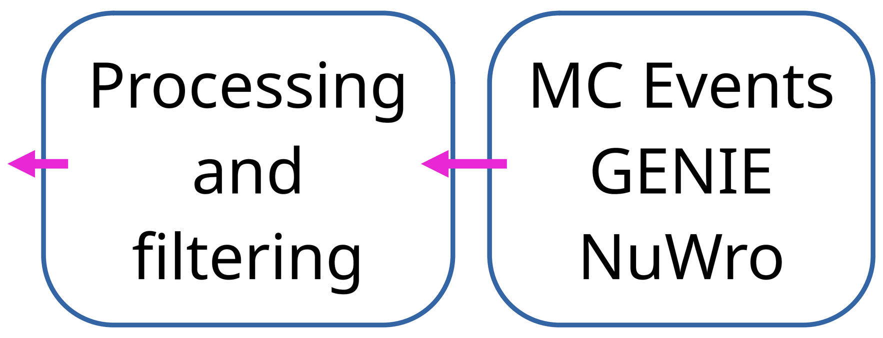
- Model predictions can be used to approximate the ratio of the datasets and used as weights for reweighting

$$\mathcal{L} = p_A(x_i)/p_B(x_i) \approx q_i/(1 - q_i)$$

Sample Preparation and Inputs:

- Events generated using several GENIE [2] models and NuWro [3] → O(5E6) to O(50E6) total events (N) in each sample
- Inputs to the network are a particle stack containing outgoing particles and a set of interaction variables
- Particle stack contains max V particles described by (E, p_x, p_y, p_z, PDG) with total shape (N, V, 5)
- Events with fewer than V particles are padded to a fixed length, and PDG codes are one-hot encoded
- Interaction vector contains kinematic quantities that apply to the entire event → (q_0, q_3, Q^2, W, E_ν)
- Events are preprocessed to normalize the distribution to zero mean and unit standard deviation



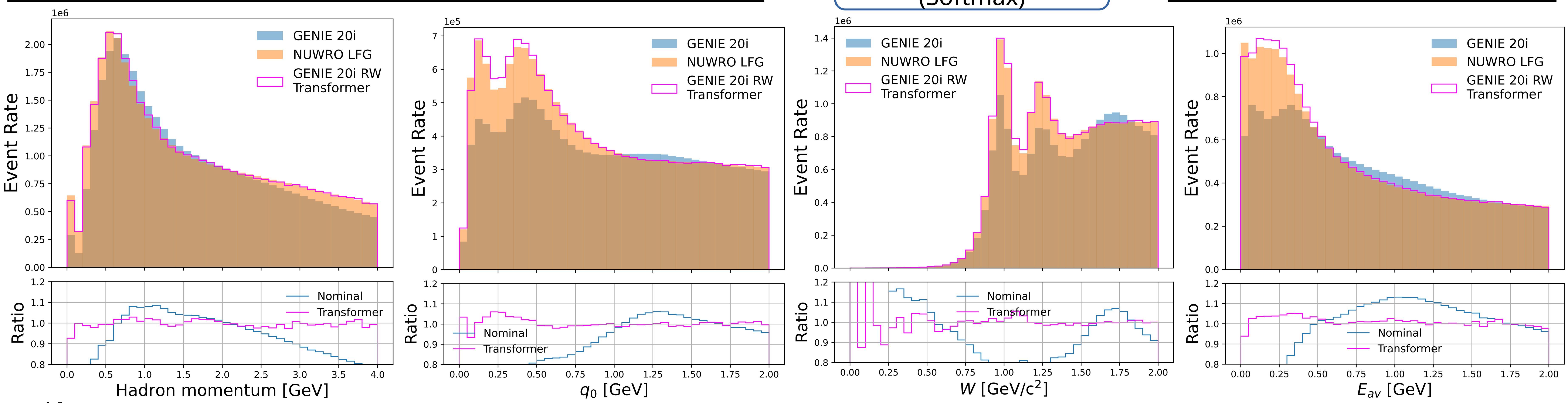


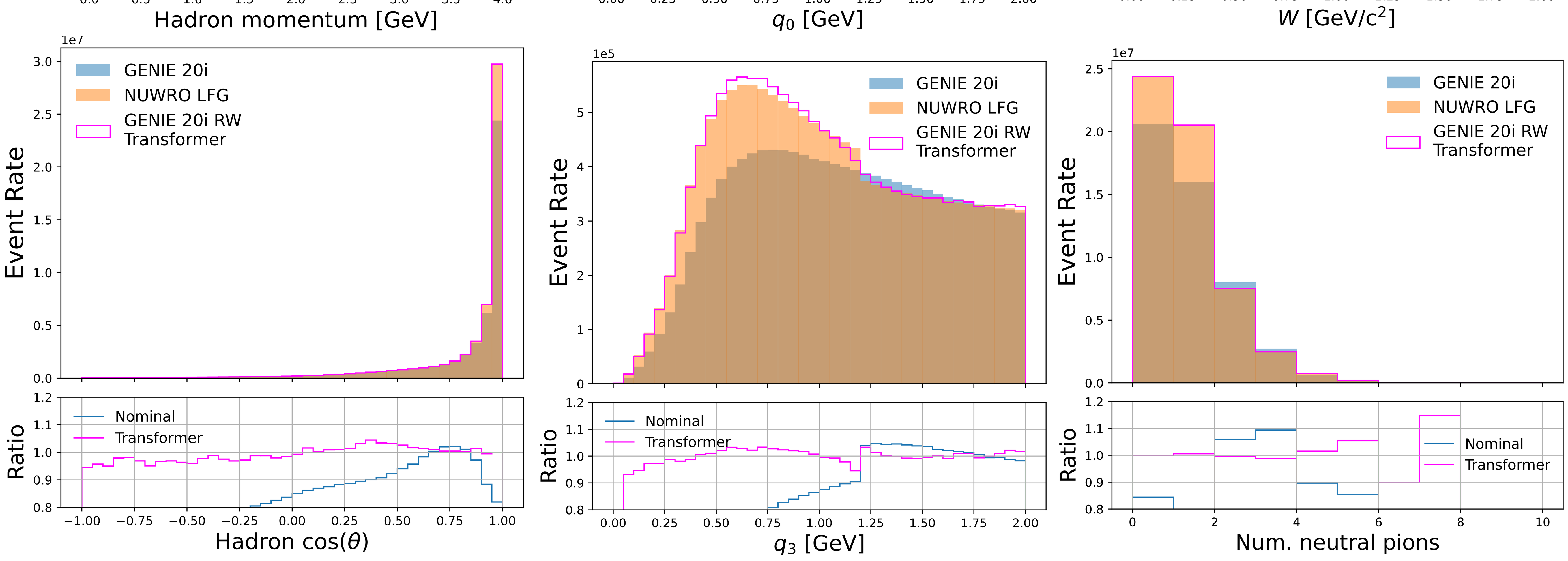
Model Architecture:

- Particle stack and interaction variables are first passed through a projection/embedding layer
- Once projected to some latent space, input vectors are added
- Transformer with 6 layers, 2 heads, model dimension of 128
- Final stage of classification head to predict particle label
- Masking to process variable number of particle in an event

Model Training:

- Parallel training on 4 NVIDIA A100 GPUs via Horovod
- Reduced learning rate on plateau with early stopping
- O(100) epochs of training for ~2 to ~8 hours of runtime





Summary:

- Reweighting works across many dimensions simultaneously
- Able to reweight continuous and discrete variables
- Unbinned event-by-event weights!
- Further improvements planned, e.g. the training procedure

Supported by the U.S. Dept. of Energy, Office of Science, Office of High Energy Physics; Japan JSPS; and the National Energy Research Scientific Computing Center (NERSC)

[1] Phys. Rev. D 101, 091901; [2] Nucl. Instrum. Methods Phys. Res., Sect. A 614, 87 (2010); [3] Nucl. Phys. B, Proc. Suppl. 229–232, 499 (2012)