Contribution ID: **257** Type: **Poster**

Efficient data movement for Machine Learning inference in heterogeneous CMS software

Efficient data processing using machine learning relies on heterogeneous computing approaches, but optimizing input and output data movements remains a challenge. In GPU-based workflows data already resides on GPU memory, but machine learning models requires the input and output data to be provided in specific tensor format, often requiring unnecessary copying outside of the GPU device and conversion steps. To address this, we present an interface that allows seamless conversion of Structure of Arrays (SoA) data into lists of PyTorch tensors without explicit data movement. Our approach computes the necessary strides for various data types, including scalars and rows of vectors, matrices, allowing PyTorch tensors to directly access the data on the GPU memory. The introduced metadata structure provides a flexible mechanism for defining the columns to be used and specifying the order of the resulting tensor list. This user-friendly interface minimizes the amount of code required, allowing direct integration with machine learning models. Implemented within the CMS computing framework and using the Alpaka library for heterogeneous applications, this solution significantly improves GPU efficiency. By avoiding unnecessary CPU-GPU transfers, it accelerates model execution while maintaining flexibility and ease of use.

Significance

References

Experiment context, if any

CMS experiment

Author: CMS COLLABORATION

Presenter: CMS COLLABORATION

Session Classification: Poster session with coffee break

Track Classification: Track 1: Computing Technology for Physics Research