IRIS-HEP Fellowship Proposal:
Deep Learning Implementations for Sustainable Matrix Element Method Calculations

Mihir Katare[1]

[1]University of Illinois at Urbana-Champaign

April 2021

1 Project Description

The utilization of multivariate statistical analysis methods in the analysis of experimental and simulated particle physics data has allowed us to gain insight on discoveries of new physics. Methods like neural networks, decision trees, and other machine learning techniques have gained a lot of traction due to their ability to model particle physics phenomena with high accuracy. However, methods like neural networks are effectively black-box methods with limited transparency and interpretability. As such, a lot of effort is usually spent on ensuring the results are generalized, as well as extensively testing whether the network has learnt the underlying physics of the process.

On the other hand, another statistical analysis technique, the Matrix Element Method (MEM) - which is largely the focus of this paper - does not suffer from these issues. The Matrix Element Method calculates the probability of an event due to the process ξ, with final observed momenta $\mathbf{x}$, under theory parameters $\boldsymbol{\alpha}$ as:

$$P_\xi(\mathbf{x}|\boldsymbol{\alpha}) = \frac{1}{\sigma_\xi^{\text{fiducial}}(\boldsymbol{\alpha})} \int d\Phi(\boldsymbol{y}_{\text{final}}) dx_1 dx_2 \frac{f(x_1)f(x_2)}{2sx_1x_2} |\mathscr{M}_\xi(\boldsymbol{y}_{\text{final}}|\boldsymbol{\alpha})|^2 \delta^4(\boldsymbol{y}_{\text{initial}} - \boldsymbol{y}_{\text{final}}) W(\mathbf{x}, \boldsymbol{y}_{\text{final}}) \quad (1)$$

This is a first principle approach that directly involves the underlying physics into our calculations. The Matrix Element $\mathscr{M}_\xi(\boldsymbol{y}_{\text{final}}|\boldsymbol{\alpha})$ (usually to leading-order(LO)) depends on the momenta of the participating particles, and so it exhaustively incorporates the kinematics of the process ξ. The matrix element also has a clear physical meaning in terms of transition probability from a state to a continuum of states within quantum field theory[3], which makes the results from MEM very interpretable and transparent.

The drawback of MEM is the significant amount of computationally intensive calculations involved in its execution. The numerical integral in equation (1) is high-dimensional and usually involves sharply-peaked integrands in phase space. Also if there is missing information for certain particles involved in the process, then integrating over it increases the dimensionality further[3]. To counteract this drawback and enable MEM to be a viable technique in searches for new physics, we propose the implementation of deep learning techniques to accurately and efficiently approximate the numerical integration from equation (1) - providing the much required speedup over the traditional approach while preserving its interpretability.

This project aims to develop the framework for the aforementioned deep learning implementation. Full ME calculations will be performed using a conventional importance sampling Monte-Carlo method (like VEGAS), and then used as training data. I will design a deep neural network and perform supervised learning using this dataset. Data augmentation procedures will be devised in order to generalize the results from our network. After training to sufficient accuracy, I will begin investigation on convenient interface options that will allow users to easily approximate calculations from equation (1) using the deployed networks instead of conventional methods. I will also explore generative network architectures that have shown promising results in evaluating integrals from high energy physics processes [1]. All of the developed code will be made available on a GitHub repository.

Research projects that involve MEM are hindered by the computational and time cost required by full ME calculations. By using the model implemented in our project, the ME calculations used in the debugging, testing, and exploratory phase could be replaced by a good approximation and significantly sped up. For the final

runs, the full ME calculations can be used to give precise results. This would make the workflow for research involving MEM much more efficient.

This project will be carried out under the supervision and mentorship of Dr. Mark Neubauer (University of Illinois at Urbana-Champaign) and Dr. Matthew Feickert (University of Illinois at Urbana-Champaign).

2 Proposed Timeline

- **Week 1 to 3:**
  Familiarize with the process of data production and management and review literature on comparable research. Come up with data processing and augmentation techniques, specific to the working dataset. Create a data generation module for more efficient memory utilization. Create a public GitHub repository for the project with initial documentation.

- **Week 4 and 5:**
  Design initial deep neural network (DNN) and train/validate it on the processed data. Provide analysis of initial baseline results from this network as well as insights on improving performance by hyperparameter tuning and architecture modification.

- **Week 6 and 7:**
  Employ hyperparameter tuning and modify architecture for improving accuracy and investigate further improvements in model performance. Address pitfalls of comparable research (like lack of generalization in some cases)[2] and attempt to fix the same using techniques like augmentation.

- **Week 8 and 9:**
  Begin investigating the development of an interface to further train the deployed models or use the final trained weights for prediction. Make final updates to the repository and associated documentation.

- **Week 10 and 11:**
  Begin exploring other deep learning methods like GAN's that have shown promising results for numerical integral estimation in similar fields of study [1]. Compare the results of these exploratory methods with the DNN results in terms of accuracy and training time.

- **Week 12:**
  Depending on the completion of the main project, perform exploratory research on topics like next-to-leading-order calculations using ML techniques or automatic differentiation to improve the interpolation of grid points. Compile final results and methodologies in the form of a report and presentations.

References

[1] Joshua Bendavid. *Efficient Monte Carlo Integration Using Boosted Decision Trees and Generative Deep Neural Networks*. 2017. arXiv: 1707.00028 [hep-ph].

[2] F. Bury and C. Delaere. "Matrix element regression with deep neural networks — Breaking the CPU barrier". In: *Journal of High Energy Physics* 2021.4 (Apr. 2021). ISSN: 1029-8479. DOI: 10.1007/jhep04(2021)020. URL: http://dx.doi.org/10.1007/JHEP04(2021)020.

[3] Mark Neubauer. *msneubauer/CWP-Sustainable-Matrix-Element-Method- DNN: Release of 2017 whitepaper*. Version v1.0. Aug. 2020. DOI: 10.5281/zenodo.4008241. URL: https://doi.org/10.5281/zenodo.4008241.