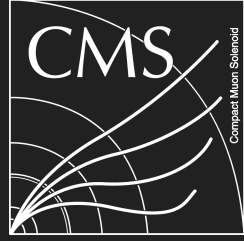


A3D3 @ UC San Diego

Russell Marroquin, Javier Duarte
UC San Diego

A3D3 High-Throughput AI Methods and Infrastructure Workshop
University of Washington
July 10, 2023

Outline

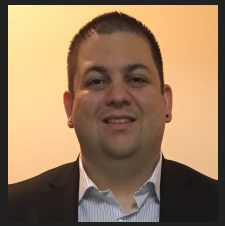
1. Overview
2. Motivation for LLPs and Current Limitations
3. Introducing LLP Tagger
4. Current Status of LLP Tagger
5. Anomaly Detection
6. Conclusion & Next Steps

Outline

1. Overview
2. Motivation for LLPs and Current Limitations
3. Introducing LLP Tagger
4. Current Status of LLP Tagger
5. Anomaly Detection
6. Conclusion & Next Steps

A3D3 Group Members @ UCSD

Javier Duarte (PI)



Daniel Diaz (postdoc)

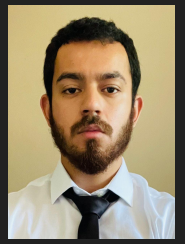
Melissa Quinnan (postdoc)

Raghav Kansal (grad)

Farouk Mokhtar (grad)

Anthony Aportela (grad)

Zihan Zhao (grad)



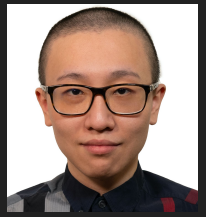
Russell Marroquin (grad)

Billy Li (grad)

Rounak Sen (MS)

Venkat Krishnamohan (MS)

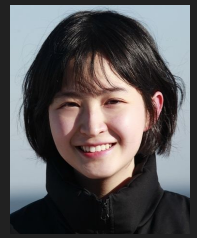
Andrew Skivington (postbac)



Zichun Hao (undergrad)

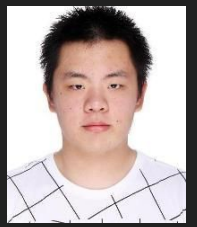
Rohan Shenoy (undergrad)

Sukanya Krishna (undergrad)



Anni Li (undergrad)

Zhaoyu Zhang (undergrad)



Michael Zhang (undergrad)

Micah de la Pena (undergrad)

Carlos Pareja (undergrad)

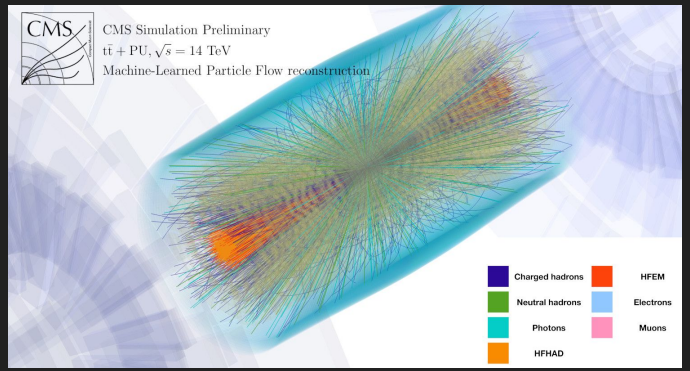
Scully (Co-I)

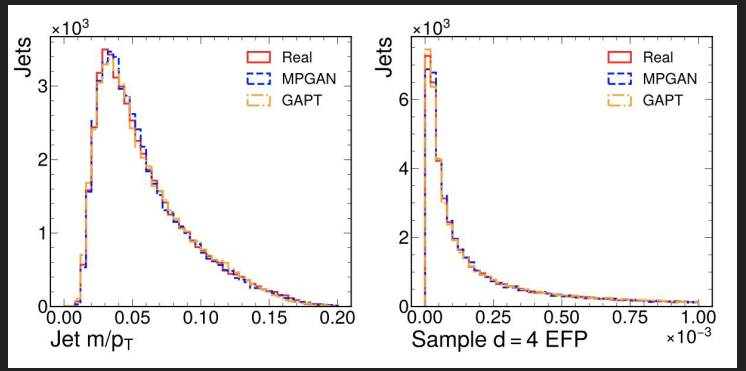
Mulder (Co-I)

Overview of A3D3 @ UCSD

Group broadly interested in

1. Enhancing core algorithms in particle physics with ML techniques
   - Machine-learned particle-flow reconstruction
     - Personnel: Farouk Mokhtar, Michael Zhang [arXiv:2203.00330, arXiv:2303.17657]
   - Generative ML for simulation
     - Personnel: Raghav Kansal, Anni Li, Zhaoyu Zhang, Venkat Krishnamohan, Rounak Sen [arXiv:2211.10295]
   - Equivariant ML
     - Personnel: Zichun Hao, Raghav Kansal [arXiv:2212.07347]
   - Self-supervised learning
     - Personnel: Zihan Zhao, Carlos Pareja, Micah de la Pena, Farouk Mokhtar, Raghav Kansal

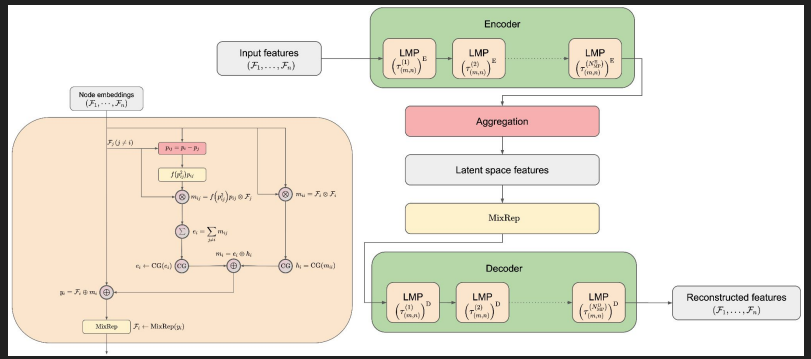


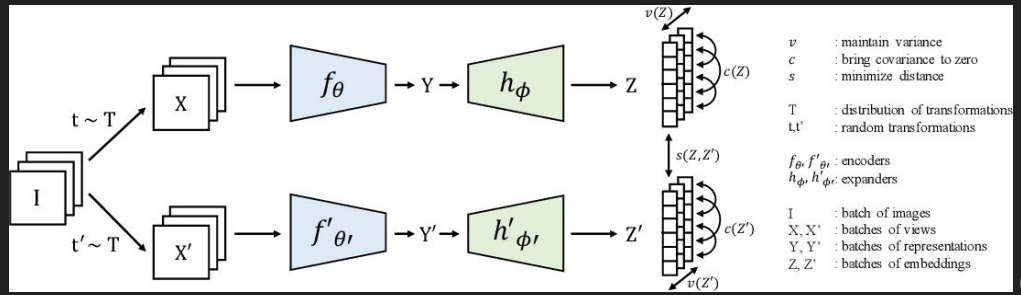


Overview of A3D3 @ UCSD

Group broadly interested in

1. Enhancing core algorithms in particle physics with ML techniques
   - Machine-learned particle-flow reconstruction
     - Personnel: Farouk Mokhtar, Michael Zhang [arXiv:2203.00330, arXiv:2303.17657]
   - Generative ML for simulation
     - Personnel: Raghav Kansal, Anni Li, Zhaoyu Zhang, Venkat Krishnamohan, Rounak Sen [arXiv:2211.10295]
   - Equivariant ML
     - Personnel: Zichun Hao, Raghav Kansal [arXiv:2212.07347]
   - Self-supervised learning
     - Personnel: Zihan Zhao, Carlos Pareja, Micah de la Pena, Farouk Mokhtar, Raghav Kansal

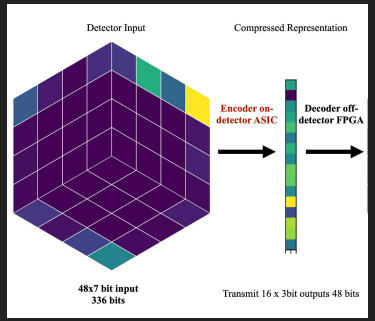




Overview of A3D3 @ UCSD

Group broadly interested in

1. Enhancing core algorithms in particle physics with ML techniques
   - Machine-learned particle-flow reconstruction
     - Personnel: Farouk Mokhtar, Michael Zhang [arXiv:2203.00330, arXiv:2303.17657]
   - Generative ML for simulation
     - Personnel: Raghav Kansal, Anni Li, Zhaoyu Zhang, Venkat Krishnamohan, Rounak Sen [arXiv:2211.10295]
   - Equivariant ML
     - Personnel: Zichun Hao, Raghav Kansal [arXiv:2212.07347]
   - Self-supervised learning
     - Personnel: Zihan Zhao, Carlos Pareja, Micah de la Pena, Farouk Mokhtar, Raghav Kansal
2. ML on FPGAs for real-time applications including CMS level-1 (L1) trigger
   - Data compression for CMS Phase-2 HGCAL
     - Personnel: Rohan Shenoy [arXiv:2306.04712]



Overview of A3D3 @ UCSD

Group broadly interested in

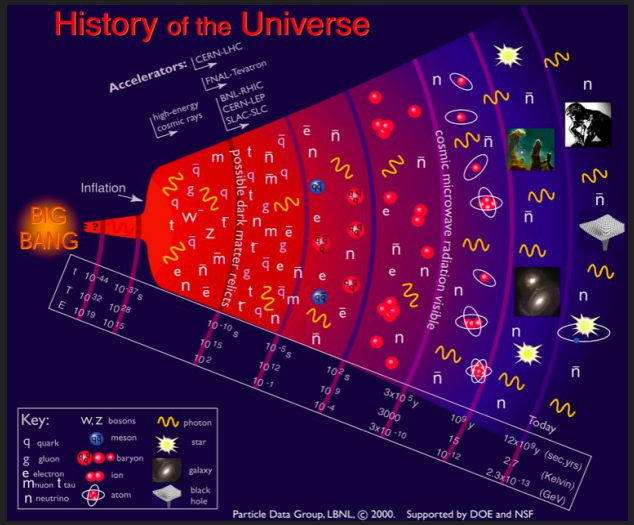
1. Enhancing core algorithms in particle physics with ML techniques
   - Machine-learned particle-flow reconstruction
     - Personnel: Farouk Mokhtar, Michael Zhang [arXiv:2203.00330, arXiv:2303.17657]
   - Generative ML for simulation
     - Personnel: Raghav Kansal, Anni Li, Zhaoyu Zhang, Venkat Krishnamohan, Rounak Sen [arXiv:2211.10295]
   - Equivariant ML
     - Personnel: Zichun Hao, Raghav Kansal [arXiv:2212.07347]
   - Self-supervised learning
     - Personnel: Zihan Zhao, Micah de la Pena, Farouk Mokhtar, Raghav Kansal
2. ML on FPGAs for real-time applications including CMS level-1 (L1) trigger
   - Data compression for CMS Phase-2 HGCAL
     - Personnel: Rohan Shenoy [arXiv:2306.04712]
   - LLP jet tagging for CMS Phase-2 L1 trigger
     - Personnel: Russell Marroquin, Anthony Aportela, Daniel Diaz
   - Anomaly detection in CMS Run 3 L1 trigger
     - Personnel: Andrew Skivington, Sukanya Krishna, Melissa Quinnan [arXiv:2108.03986]

Focus of this talk

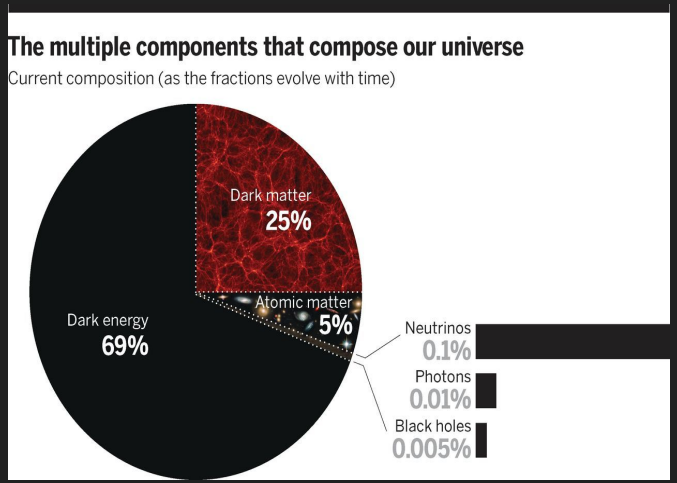
Outline

1. Overview
2. Motivation and Current Limitations
3. Introducing LLP Tagger
4. Current Status of LLP Tagger
5. Anomaly Detection
6. Conclusion & Next Steps

We Wonder…



…why is there more matter than antimatter in our universe?



…what is Dark Matter and Dark Energy?

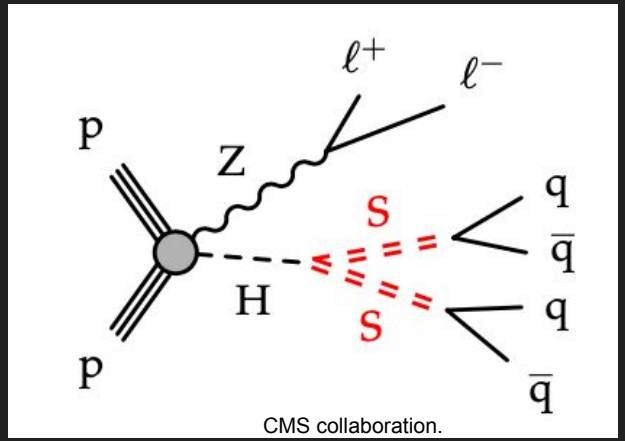
Thus, there has to be more to the Standard Model…

Anomalies

- “Anomalies” refers to anything not well modeled by our simulation of the SM
  - Often discarded during event selection
- These anomalous signals could be:
  1. Detector flaws
  2. New Physics (BSM signals)
- **What if we have been discarding interesting physics when selecting events?**

Long-Lived Particles (LLPs)

- Here, we refer to beyond the standard model (BSM) signals.
- Predicted to have relatively longer lifetimes than SM particles. Thus, they have decay vertices farther from the collision point [2].
- Several theories predict their existence. To name a few:
  - Supersymmetry
  - Twin Higgs
  - Dark Sector Models



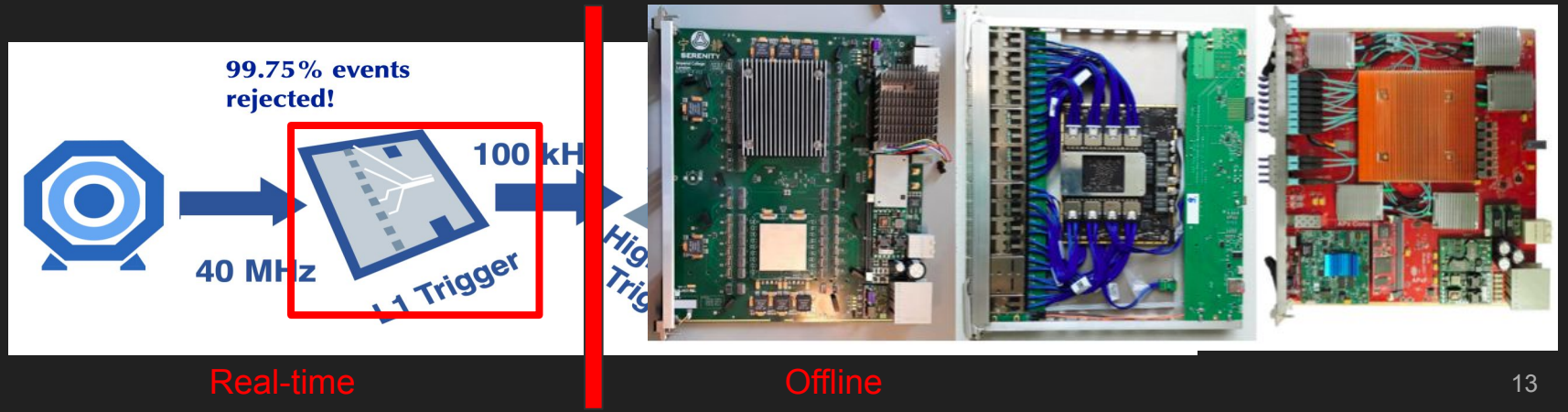
CMS collaboration.

No evidence of LLPs yet…

Current Limitations

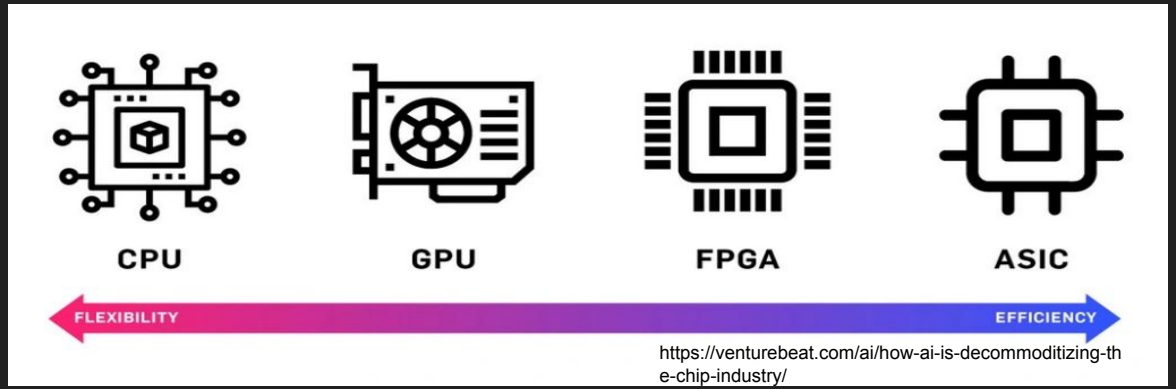
- In part, our inability to find BSM signals may occur at event selection levels called triggers [1].
  - Triggers save events during live collisions and after for further and offline analysis.

- Triggers may be biased towards “promptly” decaying particles, i.e. SM signals
  - This potentially discards events where BSM signals are found [2]
- Starting 2026, the CMS detector will undergo major upgrades in preparation for the HL-LHC…
  - At UCSD, we aim to implement BSM-signal-oriented triggers based on Machine Learning (ML) on the Level 1 (L1)



What Are FPGAs?

- Field Programmable Gate Arrays (FPGAs) allow the acquisition of data at the sub-microsecond and high rates previously mentioned [3]. Currently:
  - The L1 trigger handles data from 40 million collisions per second
  - Events are selected at a latency of 3.4 micro-seconds [2]



https://venturebeat.com/ai/how-ai-is-decommoditizing-the-chip-industry/

- More flexible, allowing reconfiguration
- Handles smaller and simpler functions
- Low processing capacity

- Flexible, allowing reconfiguration
- Can handle complex functions
- High processing capacity

- New device needed for reconfiguration
- Can handle more complex functions
- Very high processing capacity

Outline

1. Overview
2. Motivation for LLPs and Current Limitations
3. LLP Jet Tagger @ Level-1
4. Current Status of LLP Jet Tagger
5. Anomaly Detection
6. Conclusion & Next Steps

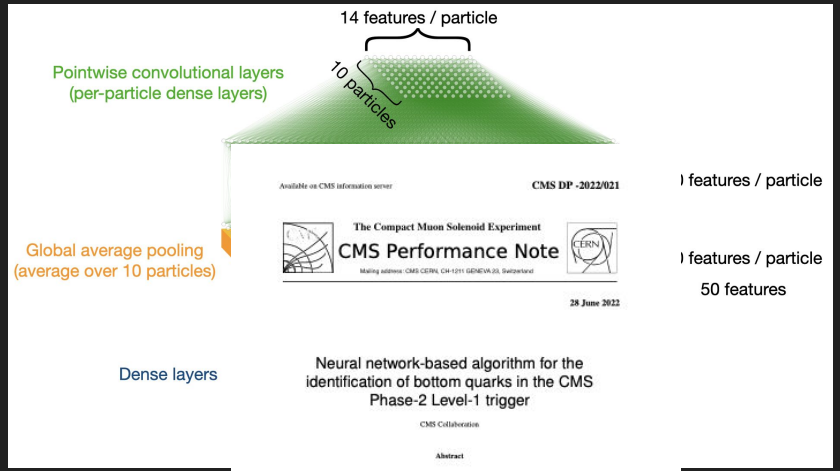
LLP Jet Tagger Overview

- The LLP jet tagger is an ML algorithm trained to trigger on LLP signatures
- Our goal is to develop this algorithm to be:
  1. Efficient enough to infer LLP signatures
  2. Fast enough to meet latency requirements
  3. Compact enough to fit in the limited resources of the trigger hardware, i.e. FPGAs

LLP Jet Tagger

Model architecture:

- Based on two Conv1D layers which act as featurizers for inputs from each jet.
  - Particles are clustered into jets using the seeded cone algorithm.
  - Each jet contains 10 particles with 14 features:
    - One-hot encoding
    - pT, η, φ scaled relative to jet
    - dx, dy, dz
- Conv1D layers are followed by downsizing and dense layers to produce a value between 0 and 1, i.e. likelihood of being LLP jet



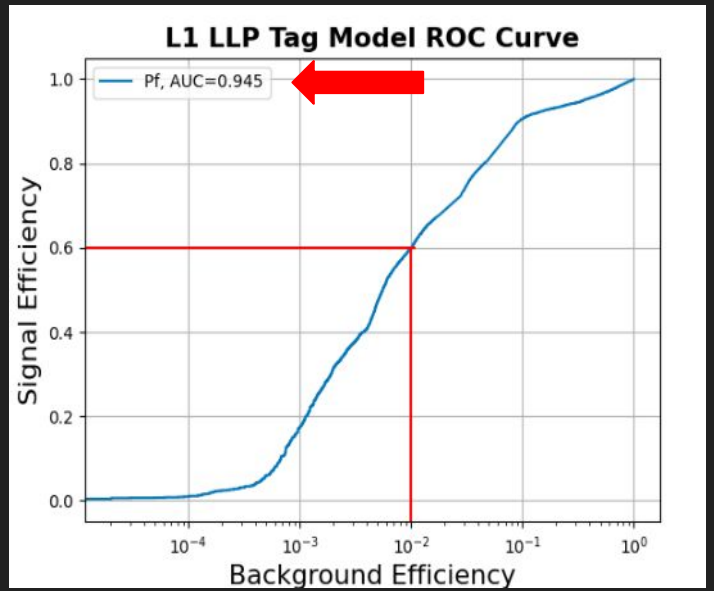
Available on CMS information server
CMS DP -2022/021
The Compact Muon Solenoid Experiment
CMS Performance Note
28 June 2022
Neural network-based algorithm for the identification of bottom quarks in the CMS Phase-2 Level-1 trigger
CMS Collaboration
Abstract
The Phase-2 upgrade of the CMS detector for the High Luminosity upgrade of the LHC (HL-LHC) includes the introduction of many new capabilities into the Level-1 trigger, including tracking and the new high- granularity calorimeter. The inclusion of tracking in particular offers the possibility of developing an algorithm to identify jets originating from bottom quarks (b-tagging) for use in the Level-1 trigger for the first time at CMS. However, only the outer tracker contributes information to the Level-1 trigger, while the sensitive pixel detector which is used for offline b-tagging algorithm does not contribute. In order to make effective use of the new information CMS will implement specialized versions of the particle flow (PF) and Pileup Per Particle Identification (PUPPI) algorithms. The particles reconstructed using the PUPPI algorithm will then be used to construct higher level objects, such as jets. The results presented here show the performance of an algorithm using a neural network (NN) that is feasible to implement on current trigger hardware in the Correlator Layer-2 system where jets are built. This implementation is also capable of operating within the budgeted latency requirements of the Level-1 trigger environment. The algorithm runs on the PUPPI particles in each jet, and is capable of discriminating between jets originating from bottom quarks and jets originating from light quarks or gluons. The expected performance and physics implication of this algorithm is shown using Monte Carlo samples with 200 pileup interactions, simulating the conditions of the HL-LHC.

Architecture was inspired by a b-jet tagger… [4]

Preliminary Results

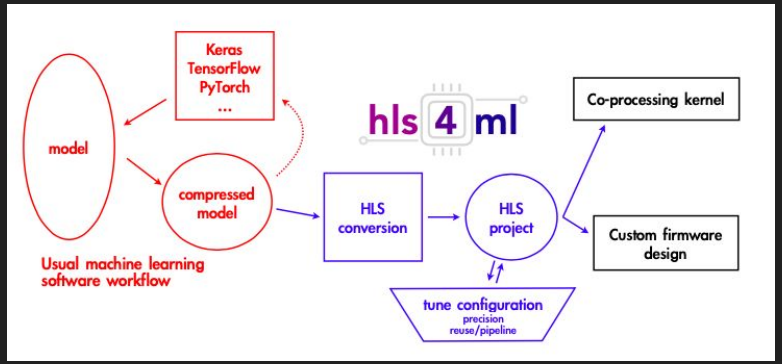
- We use the receiver operating characteristic (ROC) Curve to describe the performance of our tagger.
  - The ROC curve tells us the efficiency of our model at a given background rejection rate.
  - The area under the curve (AUC) is a number from 0 to 1 and suggests the overall performance of the model.

- We aim to improve this performance given that background at the L1 will be very large
  - We want the signal efficiency to be higher, specifically on the region inside square



Firmware

- ML models must be converted to the firmware needed to run on FPGAs
  - Keras or PyTorch → High Level Synthesis (HLS)
- To do this, we employ the *hls4ml* tool
  - This is a tool specifically for implementing trained ML models on FPGAs [3]
  - A typical workflow is shown on the figure
- hls4ml also provides resource utilization estimates, allowing us to tweak our ML algorithm further if resource and latency requirements are not met…



Resource Utilization Estimates

```
================================================================
== Utilization Estimates
================================================================
* Summary:
+---------------------+---------+-------+---------+---------+------+
|         Name        | BRAM_18K| DSP48E|    FF   |   LUT   | URAM |
+---------------------+---------+-------+---------+---------+------+
|DSP                  |        -|      -|        -|        -|     -|
|Expression           |        -|      -|        0|        6|     -|
|FIFO                 |        -|      -|        -|        -|     -|
|Instance             |        1|  26247|   372853|   744177|     -|
|Memory               |        -|      -|        -|        -|     -|
|Multiplexer          |        -|      -|        -|     2613|     -|
|Register             |        -|      -|    28035|        -|     -|
+---------------------+---------+-------+---------+---------+------+
|Total                |        1|  26247|   400888|   746796|     0|
+---------------------+---------+-------+---------+---------+------+
|Available SLR        |     1344|   3072|   864000|   432000|   320|
+---------------------+---------+-------+---------+---------+------+
|Utilization SLR (%)  |       ~0|    854|       46|      172|     0|
+---------------------+---------+-------+---------+---------+------+
|Available            |     5376|  12288|  3456000|  1728000|  1280|
+---------------------+---------+-------+---------+---------+------+
|Utilization (%)      |       ~0|    213|       11|       43|     0|
+---------------------+---------+-------+---------+---------+------+
```

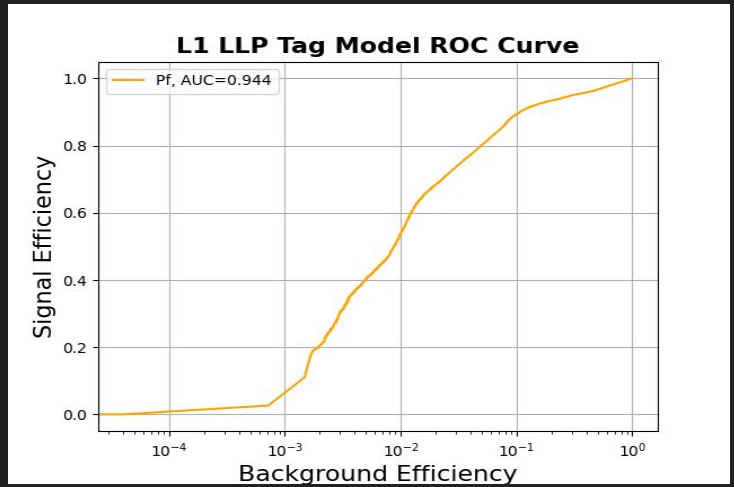
```
================================================================
== Performance Estimates
================================================================
+ Timing (ns):
    * Summary:
    +--------+-------+----------+------------+
    |  Clock | Target| Estimated| Uncertainty|
    +--------+-------+----------+------------+
    |ap_clk  |   5.00|     4.325|        0.62|
    +--------+-------+----------+------------+

+ Latency (clock cycles):
    * Summary:
    +-----+-----+-----+-----+----------+
    |  Latency  |  Interval | Pipeline |
    | min | max | min | max |   Type   |
    +-----+-----+-----+-----+----------+
    |   46|   46|   10|   10| function |
    +-----+-----+-----+-----+----------+
```

Digital Signal Processing (arithmetic) blocks (DSPs) usage is high. We turn to quantization…

Quantization

- Often calculations by ML algorithms are performed using 32-bit floating points. This is not needed to achieve optimal performance.
- Quantization can reduce the precision of the these calculations (weights, biases, etc.) in the neural network without significant loss of performance
  - Helps reduce the resource consumption of the ML model on the FPGA



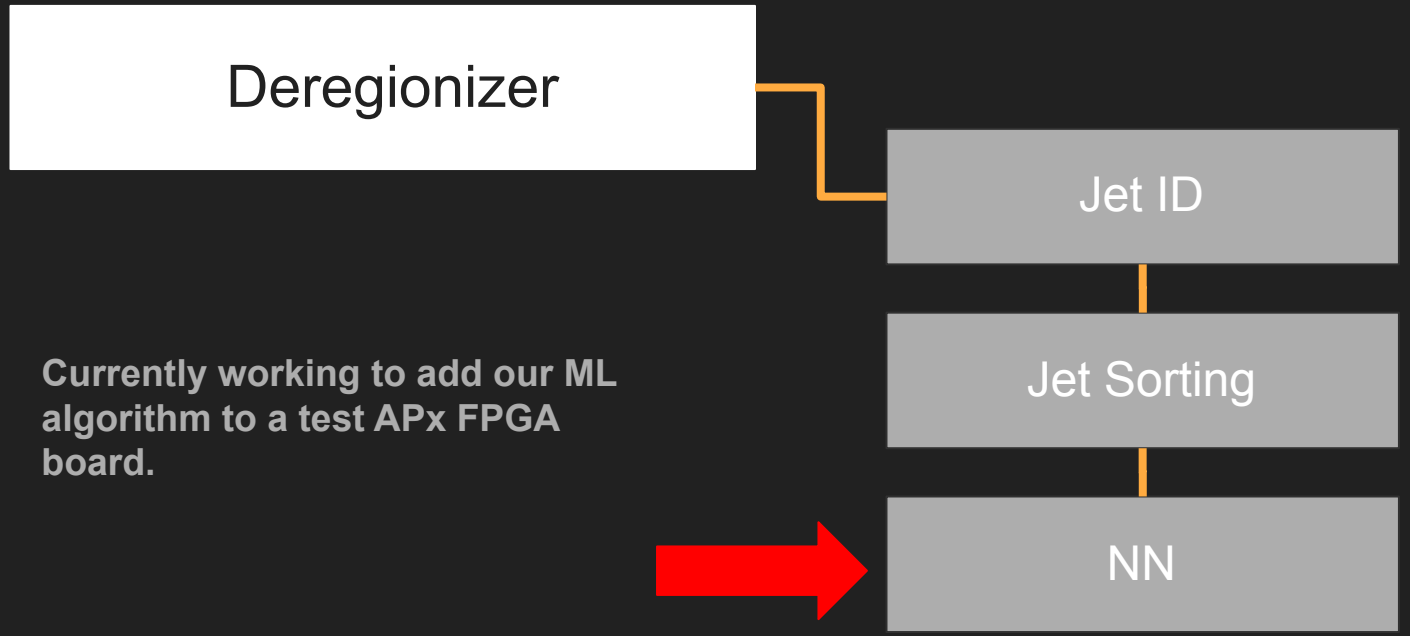
AUC value differs by 0.01 from previous model.

Outline

1. Overview
2. Motivation for LLPs and Current Limitations
3. LLP Jet Tagger @ Level-1
4. Current Status of LLP Jet Tagger
5. Anomaly Detection
6. Conclusion & Next Steps

Deployment

Workflow of the L1 trigger on the FPGA board is as follows:



Currently working to add our ML algorithm to a test APx FPGA board.

Outline

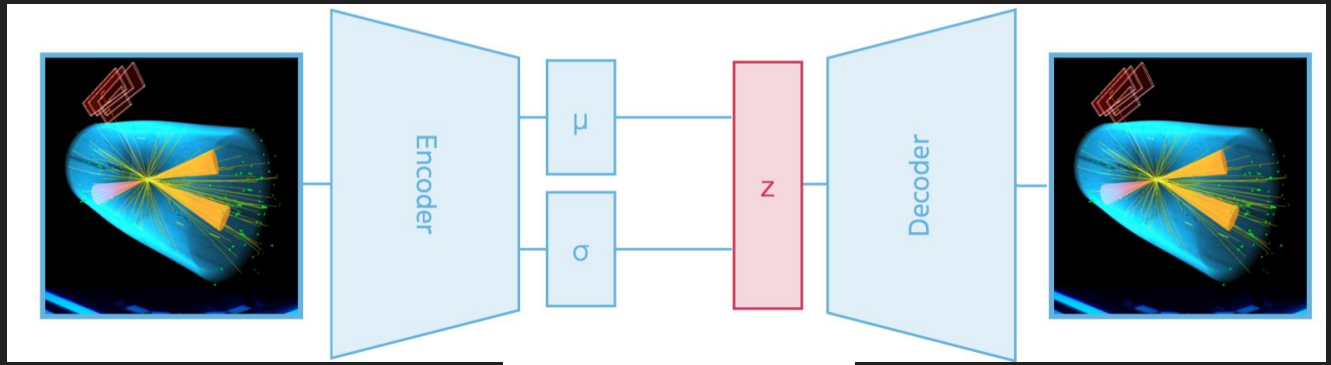
1. Overview
2. Motivation for LLPs and Current Limitations
3. LLP Jet Tagger @ Level-1
4. Current Status of LLP Jet Tagger
5. Anomaly Detection
6. Conclusion & Next Steps

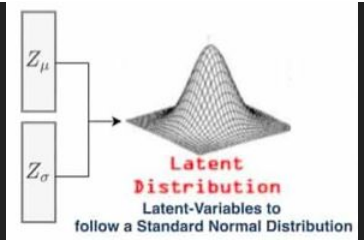
Anomaly Detection @ L1

- The goal is to develop an ML-based anomaly detection algorithm for the L1 Trigger
- **Basic idea:**
  - Trained on ZeroBias data, i.e. pileup
  - Events that are similar to the bulk of data get low anomaly score → not stored
  - Events that are NOT similar to bulk of data get high anomaly score and are saved as interesting events
- **Strategy:** use unsupervised algorithms to detect non-SM-like anomalies → **Variational Autoencoder**

Variational Autoencoders

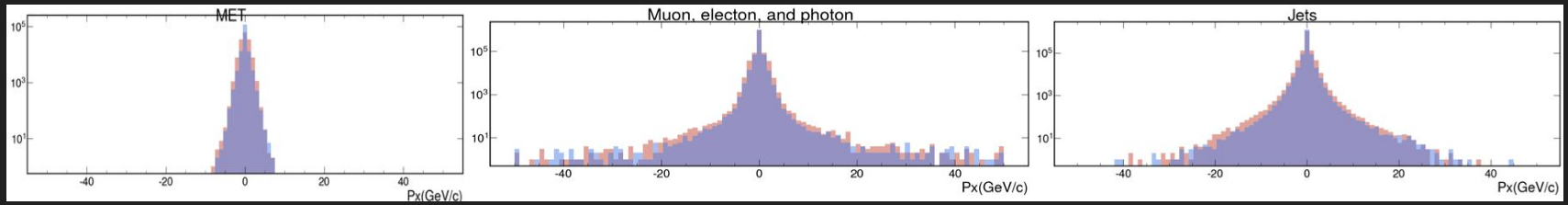
- Autoencoders (AEs) compress input to a smaller dimensional latent space then decompress and calculate difference
- **Variational Autoencoders (VAEs)** model the latent space as a probability distribution; possible to detect anomalies purely with latent space variables

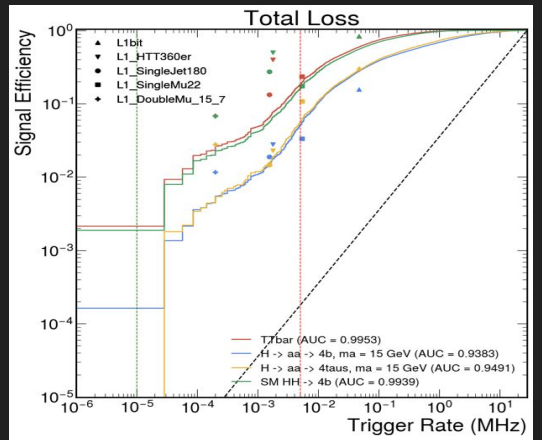




Reconstruction & AD

Reconstruction across different physics objects:





Summary: Good reconstruction is across physics objects in px; Anomaly detection performance

Outline

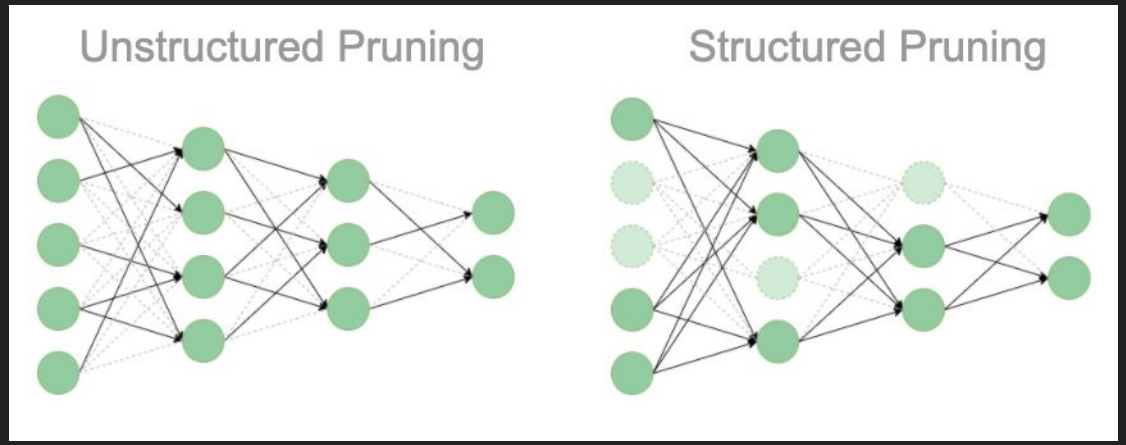
1. Overview
2. Motivation for LLPs and Current Limitations
3. LLP Jet Tagger @ Level-1
4. Current Status of LLP Jet Tagger
5. Anomaly Detection
6. Conclusion & Next Steps

Conclusion

- We may be limited to detect BSM signals because of triggering biases at event selection levels inside detectors called triggers.
- The CMS detector will undergo major upgrades starting 2026 in preparation for the HL-LHC, which opens up the opportunity to implement ML on the triggers.
- Our goals at UCSD in the context of the Level 1 trigger are to develop BSM-signal-oriented ML algorithms that are:
  1. Efficient enough to infer LLP signatures
  2. Fast enough to meet latency requirements
  3. Compact enough to fit in the limited resources of the trigger hardware

Next Steps

- Looking into pruning for our ML models…
- Pruning essentially penalizes the model for having too many non-zero weights
  - This is a dynamic process during the training phase where the model itself learns to set some of the weights to zero, leading up to less multiplications → less resources used on FPGAs



Studies have shown that pruning leads improved performance…

References

[1] Juliette Alimena, *et al.,* **Searching for Long-Lived Particles Beyond the Standard Model at the Large Hadron Collider,** arXiv:1903.04497**.**

[2] D. Contardo, *et al.*, **Technical Proposal for the Phase-II Upgrade of the Compact Muon Solenoid**, doi: 10.17181/CERN.VU8I.D59J**.**

[3] J. Duarte, *et al.*, **Fast Inference of Deep Neural Networks in FPGAs for Particle Physics**, arXiv:1804.06913**.**

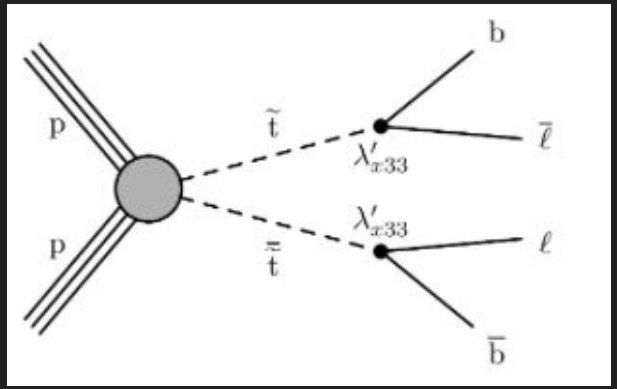
[4] CMS collaboration, **Neural Network-Based Algorithm for the Identification of Bottom Quarks in the CMS Phase-2 Level-1 trigger**, https://cds.cern.ch/record/2814728/files/DP2022_021.pdf**.**

Contact me: rmarroquinsolares@ucsd.edu

Backup

Pre-processing:

- Trained LLP tagger on a relatively small dataset with ~100k events.
- Signal dataset is constructed from a simulation based on Stop production in pp collisions at 14 TeV.
- Background is modeled from from QCD at 14 TeV.



Results are promising…