Machine Learning at CMS

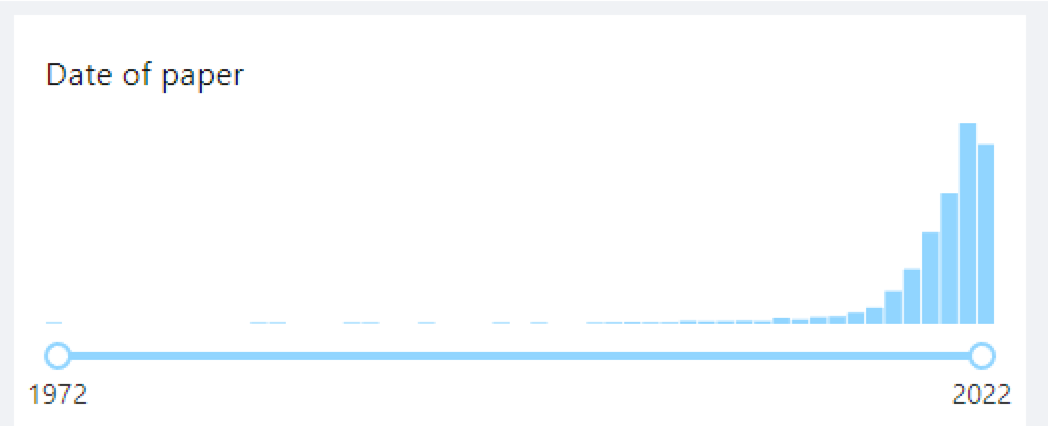
Jin Wang

18th September 2022

机器学习技术在高能物理中的应用研讨会

Machine learning in HEP

- Modern machine learning techniques, including deep learning, is rapidly being applied, adapted, and developed for high energy physics
  - Significant amount of ML publication in recent years in HEP [Inspire search]



  - Many are very mature, integrated and already used in HEP
  - Many are very interesting/promising R&D project

- A comprehensive list of ML approaches used and developed in HEP
  - Living Review of Machine Learning for Particle Physics

Machine learning in CMS

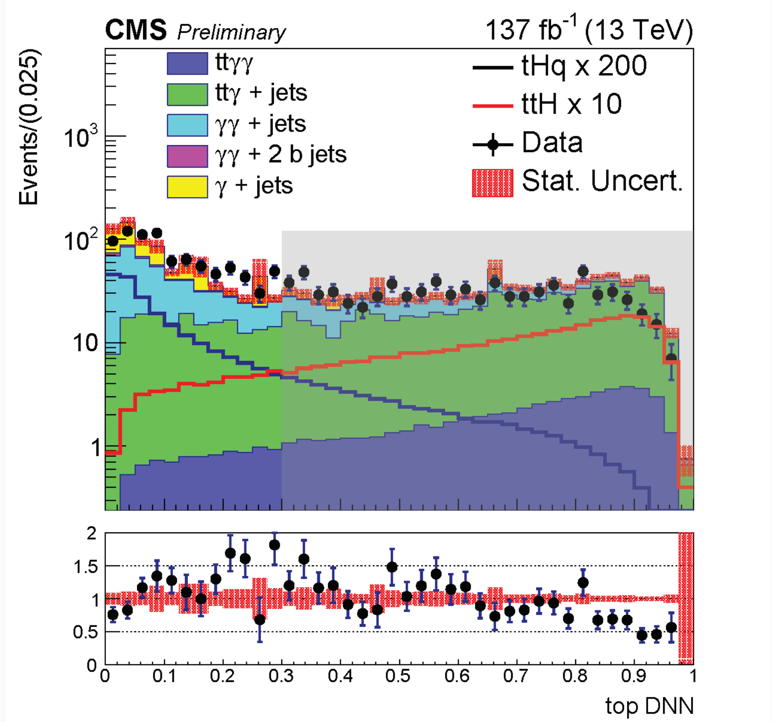
- Machine learning techniques are extensively used and explored in CMS
  - ML techniques: classification, regression, unsupervised, generative models etc.
  - Analysis
    - Event classification and signal extraction: BDTs, DNN, CNN, RNN, GNN
    - High-level object reconstruction/tagging (e.g. Higgs, top etc)
    - Likelihood-free techniques to explore EFT
    - Use ML to reduce the impact of systematic uncertainties
  - Reconstruction
    - Object construction/identification: jet/tau tagging, electron/photon/muon reco/id etc.
    - Global event interpretation: pileup mitigation, end-to-end ɣ-reconstruction
    - Detector geometry: HGCal reconstruction
  - Trigger
    - L1 trigger: hardware based fast classification
    - Model compression techniques
    - Displaced muons, anomaly detection, HGCal Taus, vertexing
  - Simulation
    - Generative models for faster/accurate simulation algorithms
    - Autoregressive, flow-based, diffusion based, variational autoencoders, GAN’s

ML in CMS analysis

Analysis: Top DNN in $H \to \gamma\gamma$

- The magnitude top Yukawa coupling y_t can be constrained through measurements of the $t\bar{t}H$ cross section.
  - But, not sensitive to the sign of y_t.
- Studying tHq production allows us to constrain the sign as well: tHq production cross section greatly enhanced if $y_t = -y_t^{\text{SM}}$.

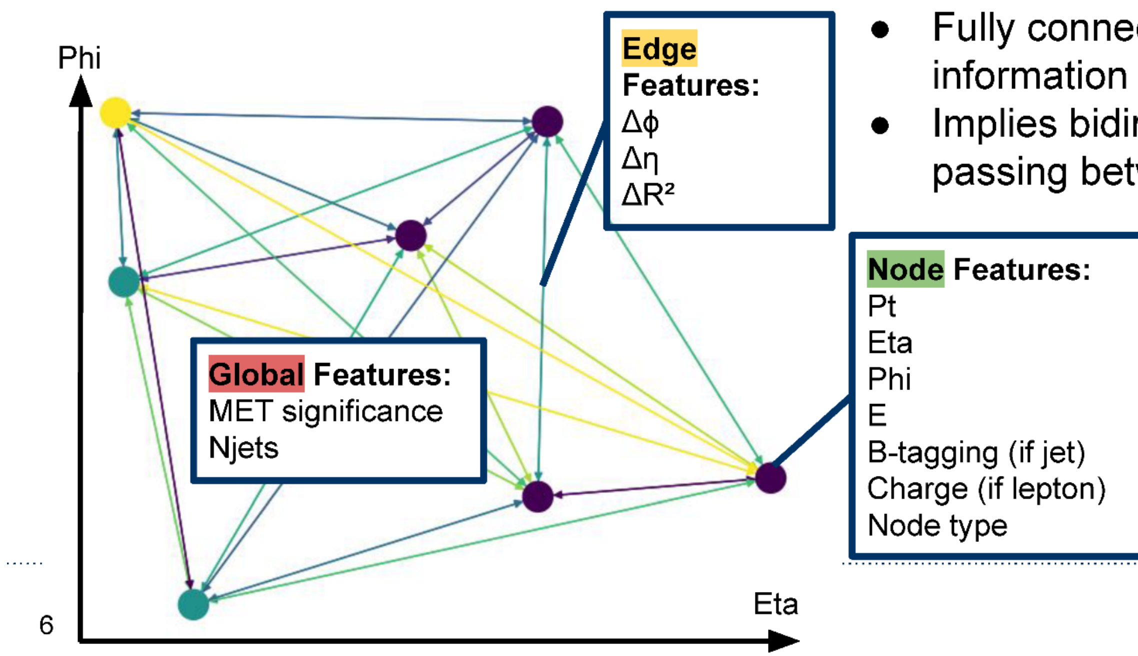
- CMS-PAS-HIG-19-015 employs dedicated signal regions for both ttH and tHq.
- Similar final states between these two processes make them very difficult to distinguish experimentally.
- Dedicated "Top DNN" is trained to separate between ttH and tHq.
  - Same architecture as DNNs used in $t\bar{t}H$ analysis.
  - Shown to significantly outperform a BDT trained for the same task.



Analysis: GNN for HEP events

- GNN: particularly well-suited for processes with high multiplicity and complex structure
  - Message Passing - Local and global sharing of information around the graph

Representing HEP Events as Graphs



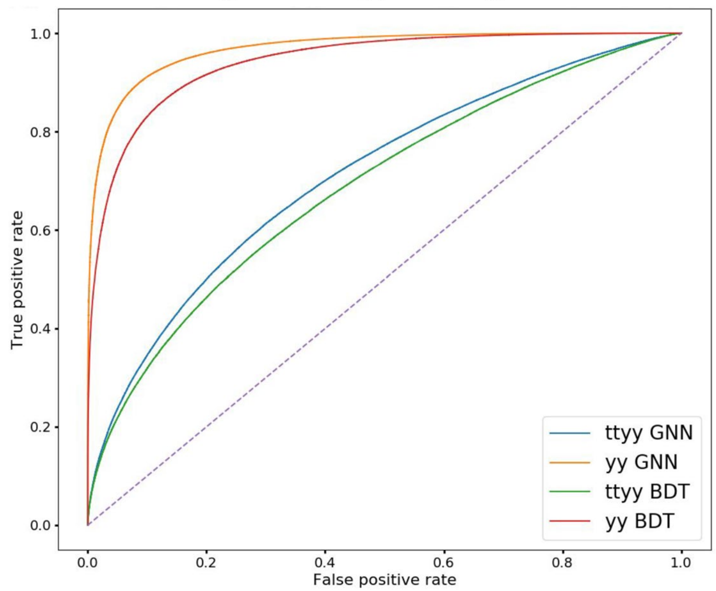
- Store 4-momenta and other information as node features
- Fully connected graph with angular information stored as edge features
- Implies bidirectional message passing between all nodes
- Event level variables can be used as global features

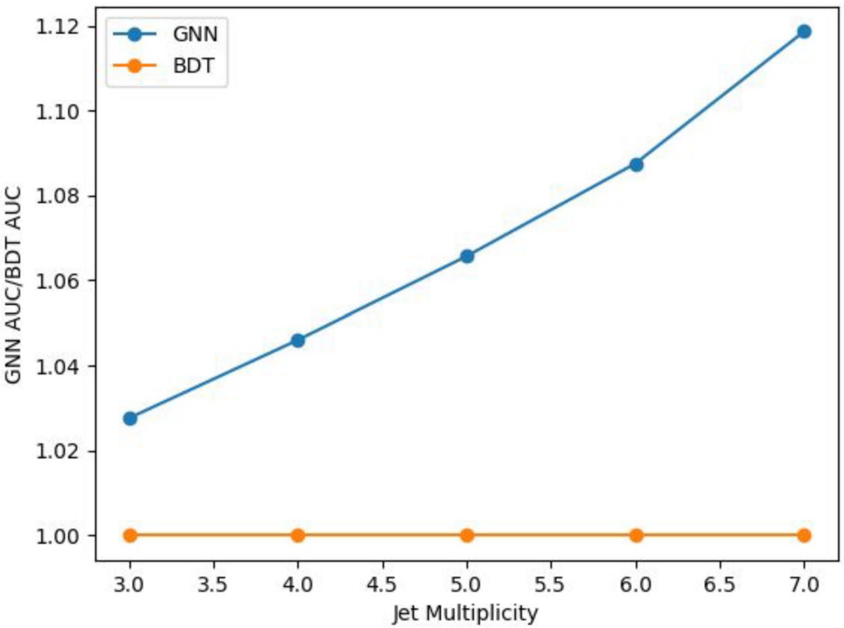
Ryan Roberts

Analysis: GNN performance

- Significant improvement in the GNN performance comparing to the BDT

ttH signal vs. yy background



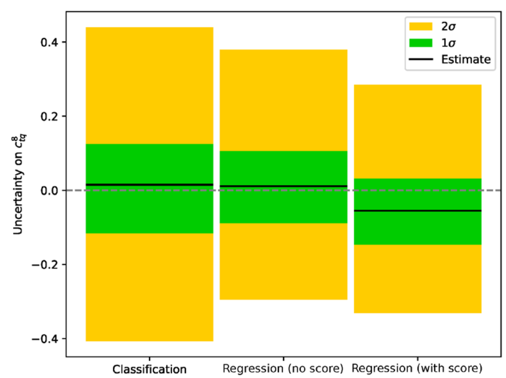


Ryan Roberts

- GNN's natural way of representing information and flexible number of objects will lead to performance increasing with multiplicity/event complexity

Analysis: Constraining Effective Field Theories with ML

- Efficiently train neural networks that precisely estimate likelihood ratios
  - Calculate the full true parton-level likelihood starting from N simulated events
  - Capture the information in the fully differential cross sections, including all correlations between observables
- Approach 1: classification
  - Train a neural network to classify between two types of events with different poi
  - Classifier output s is a probability, then transform into likelihood ratio
  - Parameterize the network
- Approach 2: regression
  - Train neural network to output the likelihood ratio
  - Use joint likelihood ratio r_{joint} and score t_{joint} obtained from matrix elements for training data



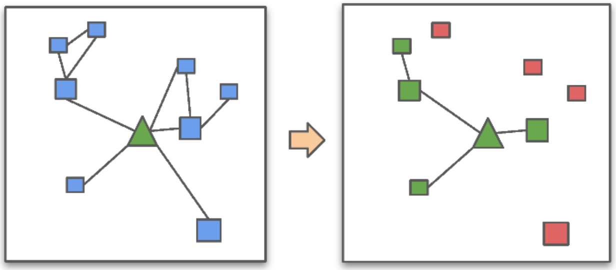
CMS: ML4EFT package: https://bib-pubdb1.desy.de/record/425819

ML in CMS reconstruction

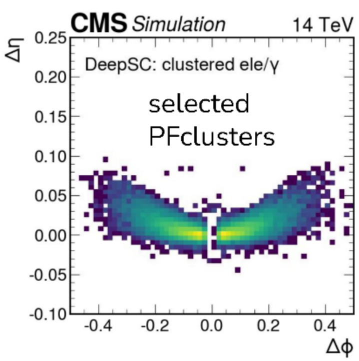
ML in CMS reconstruction

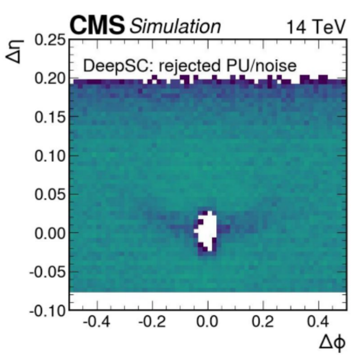
- ML in object forming
  - Tracking
  - Calorimetry clustering
- ML in object level applications
  - Jets/tau etc. tagging
  - Energy regression
  - Object identification
- ML in global event interpretation
  - Particle Flow
  - Pileup mitigation
  - Missing energy
  - End-to-End merged photon reconstruction
- ML in complex detector geometries
  - HGCAL reconstruction

Reconstruction: ECAL DeepSC

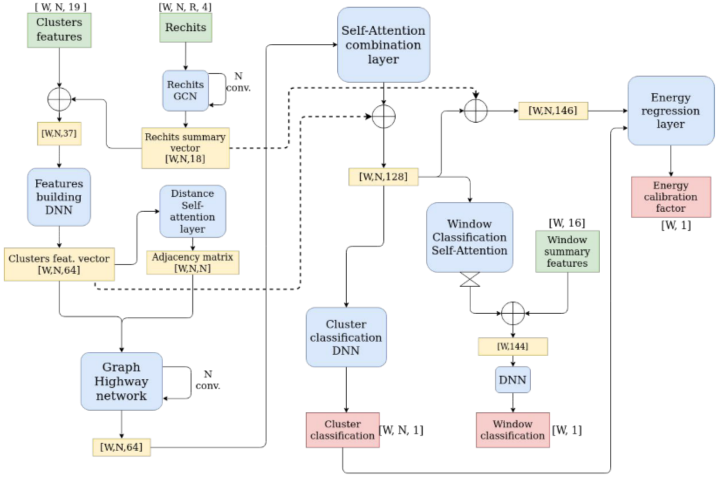


- ECAL DPG effort to improve the **SuperClustering** step
  - Linking of PFClusters to recover Bremsstrahlung or photon conversion
  - Base object for ele/gamma reconstruction, ECAL calibration, input to PF
  - Classical algo very efficiency, not very pure wrt noise/PU
- Target a **replacement of the current algorithm** in CMS reconstruction sequence
  - Seeded algorithm, working in small window of the detector
  - Implemented in CMSSW and evaluated the performance on **final Electrons/photons**
- **Architecture**:
  - Graph convolution network + attention layers
  - **Targets**: clusters selection, window classification





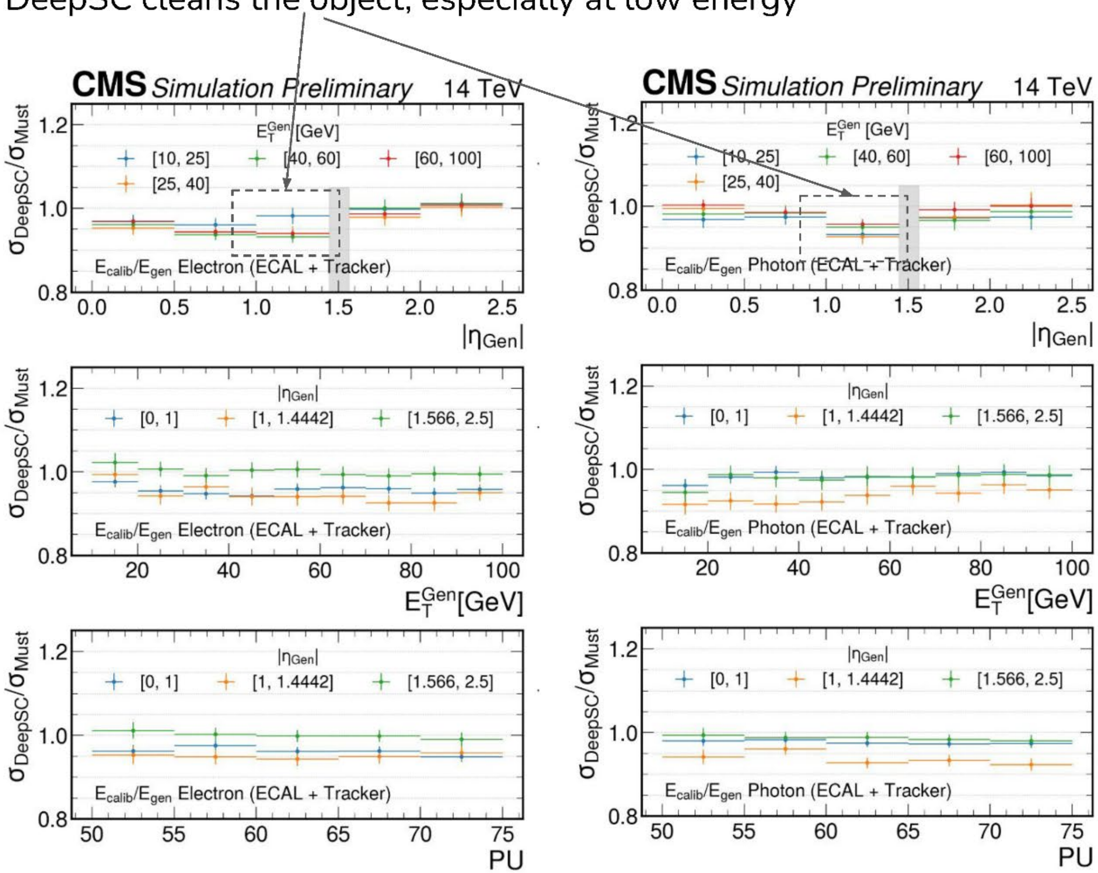
rejected PFclusters

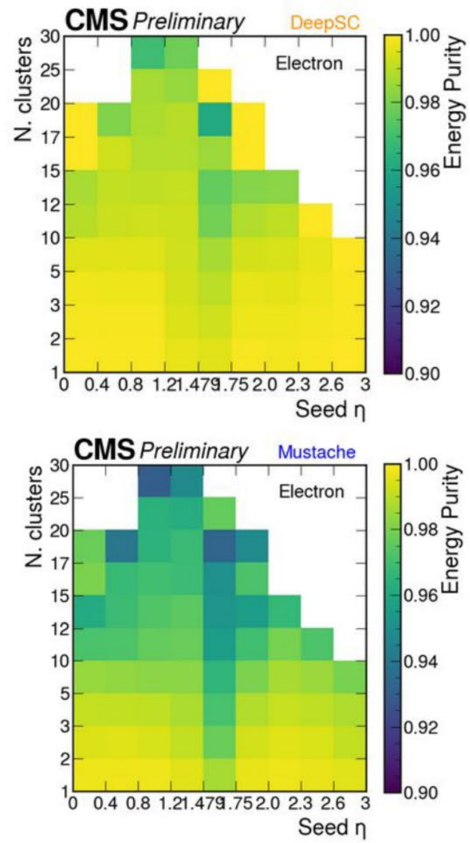


DeepSC

Reconstruction: ECAL DeepSC

Improvements in the final resolution (after regression) where the material budget is larger → DeepSC cleans the object, especially at low energy

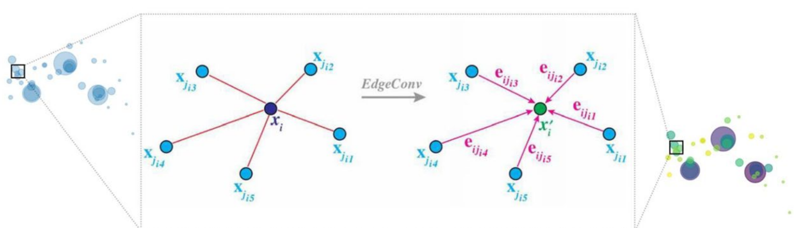


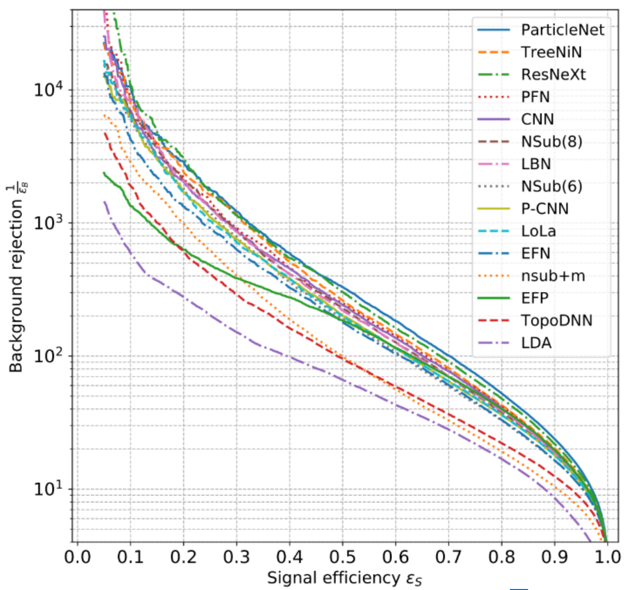


DeepSC

Reconstruction: jet tagger - ParticleNet

- Some of the jet tagging architectures tested in CMS
  - DeepCSV (DNN) (paper)
  - DeepJet (RNN) (2008.10519)
  - ParticleNet (EdgeConv) (1902.08570)
  - Point clouds transformers (2202.03772)
- **ParticleNet**
  - EdgeConv GNN based architecture on jet constituents
    - Edge convolution and the dynamic graph CNN (DGCNN) method [arXiv:1801.07829]
    - Applied on the "point cloud" data structure





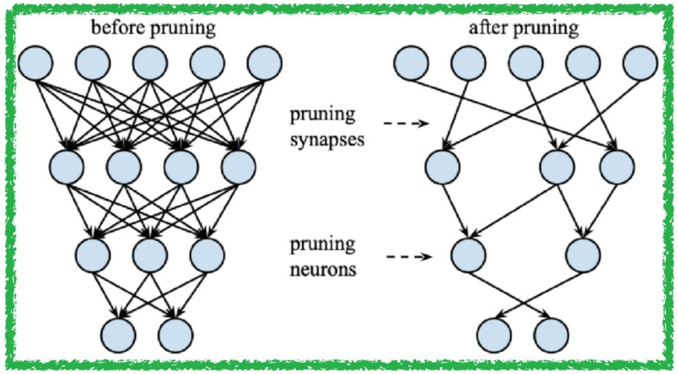
  - Full documentation and training framework (Weaver) available
- Next generation of ParticleNet (Huilin Qu)

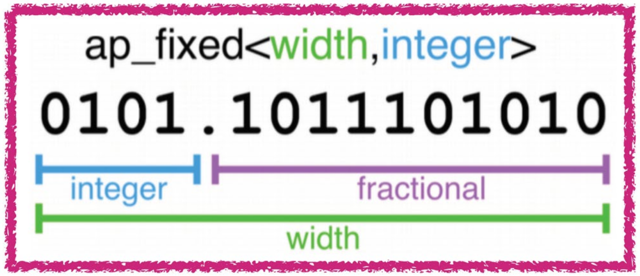
Tagger review

ML in CMS Triggers

ML in CMS Triggers

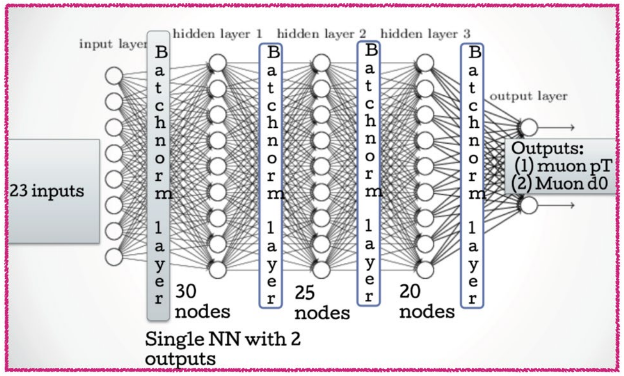
- ML in L1 trigger has substantial hardware constraints
  - Total L1 trigger latency is currently 4 μs, is 12 μs in Phase 2
  - Algorithms must be kept within available system resources, latency limitations
    - Most algorithms are limited to less than 1 μs
    - Need pruning method to reduce the complexity of the arichitecture
- Running ML on L1 trigger typically requires fixed-point arithmetic, not floating point
  - Different methods of quantizing (post-training quantization, quantization-aware-training)
- Algorithms are wired onto the chip
  - Programming traditionally done with low-level hardware languages
  - Possible to translate C to Verilog/VHDL using High Level Synthesis (HLS) tools

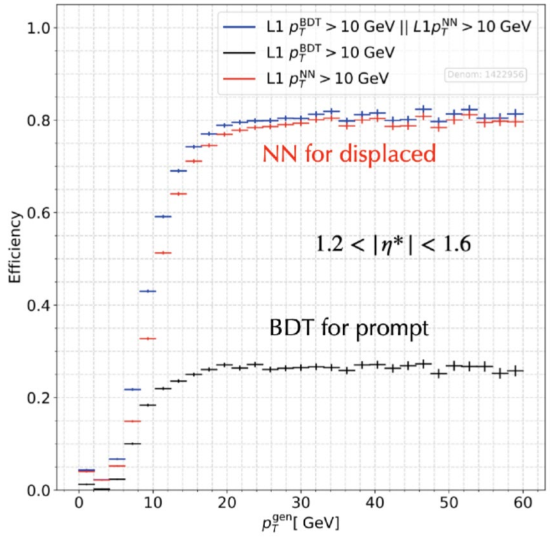




ML in L1 Trigger: Displaced Muons

- Long-lived neutral particles that may decay at macroscopic distances from the primary vertex (heavy Higgs, SUSY models etc.)
  - No info in the tracker but will be observed with displaced muons
- Use ML in LT trigger to reconstruct displaced muons
  - BDT already developed for prompt pT assignment
- NN capable of significantly improved efficiency for displaced muons
  - Outputs are pT and d0





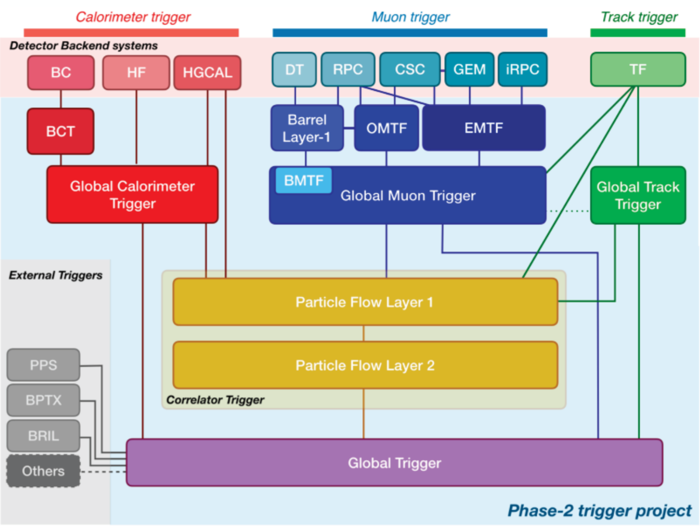
Sergo J. et al.
Fermilab

Other ML application in L1 Trigger

- Anomaly Detection
  - Design algorithms generically for signals of not-yet-theorized models or in regions of parameter space not currently favored
  - Need trigger to ensure we maintain events for later analysis
  - DNN based approach reaches required latency, resource are reasonable
  - Ongoing efforts with CNN and GNN
- Stay tuned for many active ML work in triggers for phase 2

Phase 2

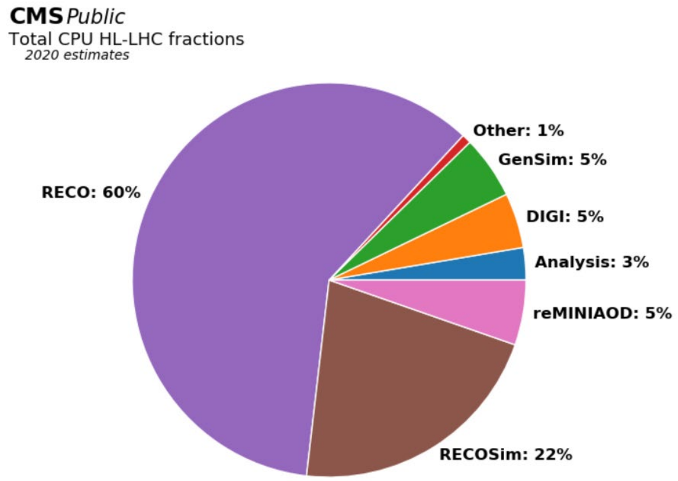
- HGCal taus
- Vertexing
- Track quality
- Electron ID
- b-tagging
- LLP jet tagging
- q/g-tagging
- NNPuppi taus
- ML MET
- Topological triggers

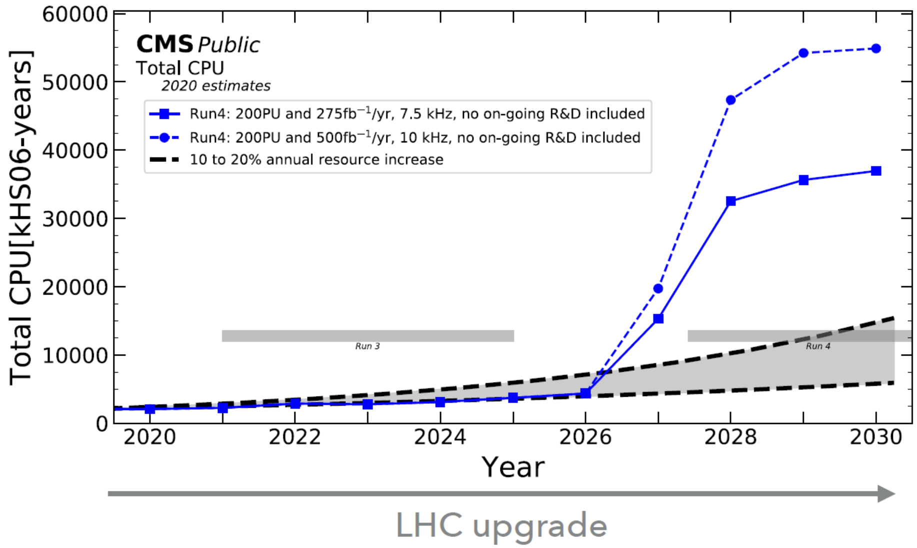


ML in CMS Simulation

ML in CMS Simulation

- Beginning of Run 2: full detector simulation (Geant4) took ~40% of grid CPU resources for CMS & ATLAS [arXiv:1803.04165]



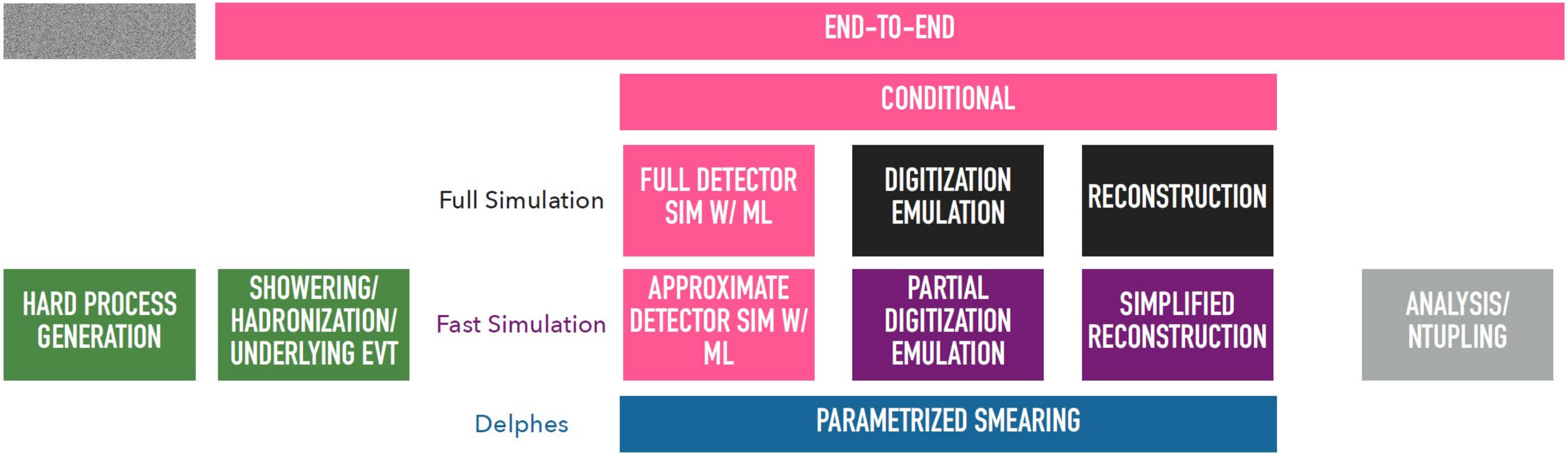


- Computing demands increase nonlinearly with increasing “pileup” in LHC
- Detector upgrades for HL-LHC: increased complexity [arXiv:2004.02327]
- Further technical improvements expected to be limited [arXiv:2005.00949]

Need more processing power or smarter algorithms like deep learning for simulation

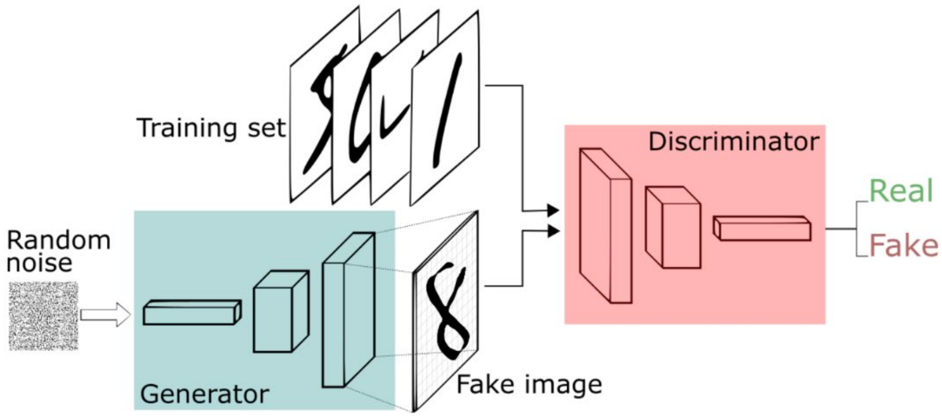
ML in CMS Simulation

- Several different strategies:
  - Replace (part of) FullSim: increase speed, preserve accuracy
  - Replace (part of) FastSim: decrease speed (slightly), increase accuracy
  - Conditional: map generated → reconstructed events
  - End-to-end: map random noise → reconstructed events directly



ML in CMS Simulation

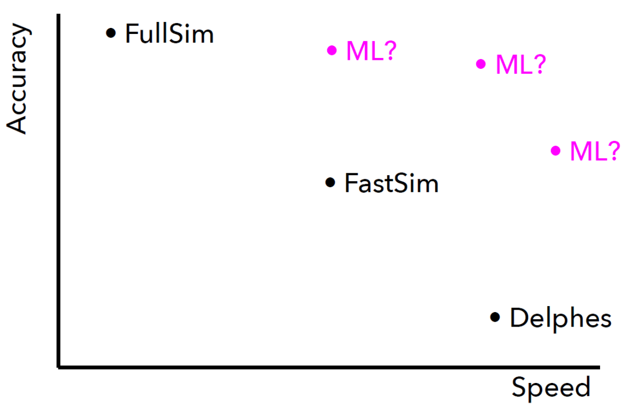
- Regression with feedforward network: arXiv:2010.01835
  - Directly map inputs (gen.) to outputs (reco.) probabilistically
- Generative adversarial networks (GANs)
  - Train two neural networks in tandem
    - one to generate realistic “fake” data
    - the other to discriminate “real” from “fake” data
  - arXiv:1406.2661, arXiv:1912.04958

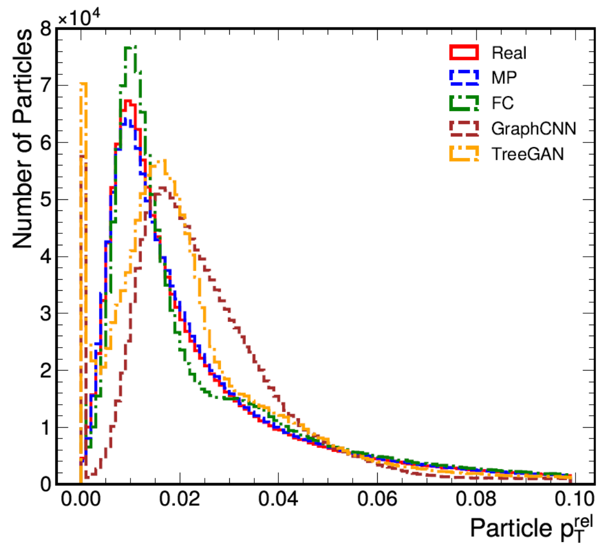


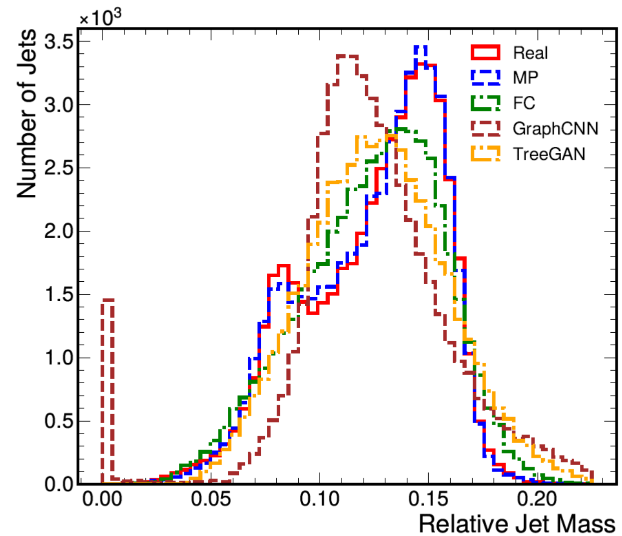
- Graph-based GAN to generate particle clouds: arXiv:2012.00173,arXiv:2106.11535
- Variational autoencoders, diffusions models , CALOFLOW, MPGAN

ML in CMS Simulation

- Need to define evaluation metrics to
  - check the quality of generated data
  - compare generative models
- Traditional method for evaluation
  - Evaluating physics simulations by comparing physical distributions







- ML method for evaluation
  - High-performing classifier learns salient hidden features from data
  - E.g. Frechet distance arXiv:2106.11535

CMS ML groups

Machine learning groups in CMS

- The goal of the CMS ML Group is to enable, support, guide, and foster ML developments in computing, POGs, and PAGs

- Information organized and gathered from a variety of sources
  - Machine learning forums and workshops
  - Communications with external teams developing ML applications
  - Dedicated talks/feedbacks from analysis/object/detector/computing groups and statistic community

- 3 subgroups to document/train ML knowledge, integrate production ready ML applications and keep track of ML R&D efforts

CMS ML knowledge group

- Goal of the Knowledge sub-group
  - Collect, maintain and disseminate knowledge of machine learning algorithms
  - Development and maintenance of CMS machine learning benchmarks
  - Comparing and tracking the performance of algorithms, platforms and ML frameworks on a set of benchmark
  - On-demand technical discussion with working groups

- Knowledge Sources
  - Papers and talks about ML implementations in CMS and HEP
- Experts List
  - Collections of experts in different areas of ML who are open to answering questions

- Documentation
  - https://cms-ml.github.io/documentation/

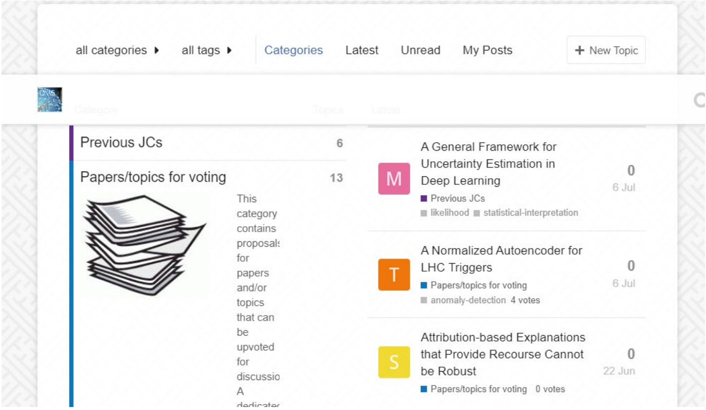
CMS ML production group

- The focus of the Production sub-group
  - Delivering production-level training and inference for CMS ML algorithms
- Develop and maintain of ML application/inference workflows for CMS
  - Broad development of inference engines for CMS *TensorFlow, MXNet, ONNX, PyTorch, hls4ml*
  - Work closely with CMS framework experts, liason to the CMS framework and software/computing groups
  - Handling integration issues
- Development of training tutorials, help with training facilities
- Common code repository for ML tools
  - https://github.com/cms-ml/cmsml

CMS ML innovation group

- The goal of the Innovation sub-Group
  - Identify and apply new machine learning techniques to CMS challenges
  - Discuss the relevance of new outside ideas
  - Help with the adaptation and implementation of specific models
  - Develop specific methods for CMS that will lead to technical publications
  - Lead organization of ML-oriented hackathons and challenges

- ML **Journal club**
  - To discuss bleeding-edge ML ideas already or not yet pursued by CMS
  - Proposals for papers/topics that can be upvoted for discussion
    - https://cms-ml-journalclub.web.cern.ch



Summary

Summary

- Many active machine learning projects within CMS
  - Growing usage of more advanced ML techniques in various analysis areas
  - Object tagging/reconstruction ever improving with deep learning
    - GNN playing a big role, increasing amount of regression applications
  - Significant opportunity to accelerate simulations using machine learning
  - Many ongoing developments in Level-1 trigger using ML
    - Improvements can have significant impact on acceptance/performance

- Well established ML groups in CMS to document, apply ML techniques and explore new ideas
  - Good connection with experts/analysis/object/detector/computing groups

- Stay tuned for more dedicated CMS ML talks in the future