Machine and deep learning techniques in heavy-ion collisions with ALICE

Rüdiger Haake (CERN)
for the ALICE collaboration

(06.07.2017)
EPS-HEP 2017, Venice, Italy
b-jet tagging

Dielectron identification
b-jet tagging
b-jets with the ALICE detector

- Main interest of heavy-ion physics: **Quark-Gluon Plasma (QGP)**
- Hot & dense medium, strongly interacting w/ high-energy partons
- Jet measurement with ALICE down to low $p_T$

- Modification of b-jets different to udsg-jets
  - Larger energy loss for gluons than quarks (color charge)
  - “Dead cone effect”: For massive quarks, gluon bremsstrahlung suppressed at smaller angles w.r.t. parton direction

→ **b-jets interesting probe for the QGP**

Goal: Investigate parton energy loss mechanisms

- Here: Evaluation for p-Pb collisions as first step towards Pb-Pb collisions
  - Useful to study cold nuclear matter effects
  - Reference measurement for Pb-Pb collisions
b-jet identification

- B-hadrons decay in the (sub-)millimeter range \((c\tau \sim 500 \, \mu m)\),
  \(\rightarrow\) displaced from primary vertex

- Common discriminators:
  - Reconstructed secondary vertices
  - Track impact parameters

- Secondary vertex reconstruction:
  - Here: All three-track combinations considered (3-prong vertices)

“Conventional” approach:
Application of rectangular cuts on properties of most displaced vertices

Ansatz here: Apply ML techniques to several low-level inputs:
Constituents, secondary vertices, track impact parameters
Model design & input features

- Binary classification problem: b-jet *tagging*
- General design: **Multibranched, multilayered** neural network
  - Multiple subnetworks on several features:
    - 1D convolutional networks (**CNNs**)
  - Merged output fed to multilayered fully-connected network
  - Keras\(^1\) has been used for model creation & training
- Tested many different networks on different features

**Features**

- Array of secondary vertices, each:
  - \((x, y, z)\) rel. to primary vertex
  - Transverse plane distance & uncertainty: \(L_{xy}, \sigma_{xy}\)
  - Vertex track dispersion \(\sigma_{vtx}\), fit quality \(\chi^2\)
- Array of constituents: \(\eta, \phi, r\) (relative to jet axis), track impact parameters \(D, Z, \text{ and } j_T\)

\(^1\)F. Chollet et al., https://github.com/fchollet/keras
Simulation dataset

- $p$-Pb $\sqrt{s_{NN}} = 5.02$ TeV, PYTHIA6 + HIJING
- FastJet anti-$k_T$ jets, $R = 0.4$, tracks only, bgrd. corr.
- 200k training, 50k validation samples
- True jet type set with particle level information:
  - **B-hadron within $R = 0.4$:**
    - $\rightarrow$ b-jet
  - **If instead, C-hadron within $R = 0.4$:**
    - $\rightarrow$ c-jet
  - **Else:**
    - $\rightarrow$ light-flavor jet
Results: Mistagging vs. b-jet efficiency

- **Solid lines:** ML-based method (statistical uncertainty only)
- **Dashed lines:** Conventional, cut-based method

ML-assisted tagging method very promising
Mistagging efficiency much lower for c- and udsg-jets

1 cf. arXiv:1605.00143
Results: Mistagging efficiency vs. jet $p_T$

- **Solid symbols:** ML-based method (statistical uncertainty only)
- **Open symbols:** Conventional, cut-based method\(^1\)
- b-jet efficiency fixed (red)

- Also here: ML-assisted tagging method very promising

---

\(^1\) cf. arXiv:1605.00143
Results: Mistagging efficiency vs. jet $p_T$

- Mistagging efficiency for higher b-jet efficiency
- Solid symbols: c-efficiency
- Open symbols: udsg-efficiency

- Sample mostly udsg. About 90% udsg-, 5% c-jets
  - udsg efficiency should be below 0.5-1%
  - c efficiency should be below a 5-10%
- Higher b-jet efficiencies possible
Training control plots

- Accuracy, loss good control parameters
- Model shows slow learning up to high epoch counts
- Learning rate parameter has been lowered after 200 epochs: 
  \([10^{-4}, 10^{-5}]\)
- Not much to gain with longer training
Training control plots

- Accuracy, loss good control parameters
- Model shows slow learning up to high epoch counts
- Learning rate parameter has been lowered after 200 epochs: $[10^{-4}, 10^{-5}]$
- Not much to gain with longer training
Training control plots

• Accuracy, loss good control parameters
• Model shows slow learning up to high epoch counts
• Learning rate parameter has been lowered after 200 epochs: \([10^{-4}, 10^{-5}]\)
• Not much to gain with longer training
• AUC = Area Under ROC Curve
• AUC reveals slow, but constant learning up to 220 epochs
• Clearly separated score distribution
Dielectrons
Dielectron classification

- Dielectrons created at all stages of collision
- Negligible interaction after creation

Interesting probe for QGP

- Here: Focus on low-mass $e^+e^-$ identification
- Main goal of dielectron classification analysis: Reject background efficiently
Dielectron classification

- Sample is contaminated w/ combinatorial & $\gamma$-conversion pairs
- Use two neural networks to classify background
  1. Pairs from conversion
  2. Pairs with one electron from photon conversion
     Fully-connected, multilayered networks
- Cut such that signal is most significant
Dielectron classification

- Sample is contaminated with combinatorial & γ-conversion pairs
- Use two neural networks to classify background
  1. Pairs from conversion
  2. Pairs with one electron from photon conversion
     Fully-connected, multilayered networks
- Cut such that signal is most significant

Very promising gain in significance
On top: Improved electron identification

- Electron identification using several subdetectors in ALICE
  Measurement of $n\sigma$'s: How many standard deviations away from mean expected value

- MVA approach:
  Use Boosted Decision Tree (BDT) on $n\sigma$ values, track properties

- Performance evaluated in pp. Soon: PbPb

$n\sigma$ distribution for electrons TPC:
Summary
b-jet tagging

- Deep learning tagging method has been developed
- Performance evaluated in p-Pb MC simulations and compared to cut-based method
- Results are very promising
  - Tagging method allows much higher b-jet efficiencies
  - Lower mistagging rates
  
- Application on p-Pb data ongoing → promising results
- In addition: c-jet tagging to be explored
Summary

Low-mass dielectron classification
- Applied BDT to electron identification in pp, soon Pb-Pb
- Applied NN to classify dielectron background in Pb-Pb
- Very promising performance in MC simulations
- Electron identification in PbPb
- Application on pp, p-Pb, and Pb-Pb LHC Run 2 data

Several other ML-based analyses in ALICE ...
- For example, this morning at EPS-HEP: Charmed mesons & baryons pp & pPb (A. De Caro)

Thank you for your attention!
Backup
The ALICE detector

Machine learning in ALICE

Rüdiger Haake
MC-data adaption

- Method strongly relies on accuracy of MC
- Ansatz: Change MC such that it better reproduces data in our feature space
- Find differences: Use GBDT classifier to separate MC & data
- Cure differences: Reweight regions in feature-space to have same effective population in MC & data
- Using MC-data adaption leads to significantly better results

ROC curve of classifier: After reweighting, data and MC are hardly to separate in feature space

(A. Rogozhnikov, 1608.05806)
b-jets: Model design

Convolutional networks on fixed-length sequences of low-level parameters
b-jets: Model design

Convolutional networks on fixed-length sequences of low-level parameters

Dropout: 0.1 for all layers & branches

Sigmoid neuron for binary classification
b-jets: Model design

Secondary vertices

\[(v_x, v_y, v_z), \sigma_{xx}, \chi^2, L_{yy}, \sigma_{xy}\]

Convolution 1D
Dropout per layer: 0.1
Conv. kernel: 4
Max pooling
Pool length: 2
64
Conv. kernel: 2
32
Max pooling
Pool length: 2

Jet constituents

\[\eta, \varphi, r\]

Convolution 1D
Dropout per layer: 0.1
16
Max pooling
Pool length: 2
32
64
Conv. kernel: 2
96
Max pooling
Pool length: 2
96
Max pooling
64
Conv. kernel: 2

Impact parameters \(D_1, Z, j_r\)

Merge (concatenation)

Fully-connected
Dropout per layer: 0.25
L2 regularization
128
128
128
128

FC network on top
Higher dropout of 0.25 here

Sigmoid neuron for binary classification
b-jets: Model design

Secondary vertices

Jet constituents

Other model properties

- ADAM optimizer
- Loss: binary crossentropy
- Activation function: ReLU

Last neuron is sigmoid-activated
b-jets: Simulation dataset II

- Strictly separated samples for training, validation, and testing
- 200’000 (training), 50’000 (validation) for each class
  - Signal class: 100% b-jets
  - Background class: 10% c-jets & 90% udsg-jets

Note: This is for the network to adjust better to udsg-jets
The impact of using different percentages is small

- Testing statistics is higher:
  ~1.8M udsg-jets, ~500k c-jets, 580k b-jets