Exploring neural networks to improve b-jet tagging with the ALICE detector

Rüdiger Haake (CERN) for the ALICE collaboration

(21.03.2017)
1st IML Machine Learning Workshop, CERN
Talk outline

- Motivation: b-jets and their identification
- Model design
- Input features & data
- Results
- Summary & outlook
b-jets with the ALICE detector

• Conceptually, a jet is the final state of collimated hadrons that fragmented from a high-energy parton

• Jets can be used to shed light on the very early stage of a hadron collision

• The reconstructed jet observable is defined by the jet finding algorithm used to clusterize tracks into jets
  → “Charged jets”, charged part of a jet

• b-jets: jets arising from beauty quarks
Main interest of heavy-ion physics: **Quark-Gluon Plasma (QGP)**

- Hot and dense medium, strongly interacting with high-energy partons
- 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: Understand better the influence of the medium on parton energy loss

- Here: Measurement in 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 \((c\tau \sim 500 \mu m)\), → displaced from primary vertex

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

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

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

[Diagram of b-jet tagging in ALICE]
b-jet identification

- In addition: fragmentation pattern should be different for b-jets
  - Jet shapes as discriminators?
  - Constituents as discriminators in deep learning?

Qualitative representation of the constituent distribution around the jet axis:

- u-quark jets
- b-quark jets

PYTHIA6, 7 TeV, particle level jets, \( p_T = 30-40 \text{ GeV}/c \)
For illustration purposes

Not yet directly exploited in the conventional method in ALICE
Model design I

- Binary classification problem: b-jet tagging
- General design: Multibranched, multilayered neural network
  - Multiple subnetworks on several features
  - Output is merged and fed to multilayered fully-connected network
- Training done for whole network
- Keras\(^1\) has been used for model creation & training
- Several different networks on different features has been tested:
  - LSTMs, 2D convolutional networks on jet images ...

\(^1\)F. Chollet et al., https://github.com/fchollet/keras
Model design II

- Promising designs have been further refined with a parameter grid search
- Note: Due to limited time and computing performance, only a small fraction of all possible configurations has been tested
- A parameter correlation analysis has been done to check the relevance of potential input features
Model design

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

b-jet tagging in ALICE

Rüdiger Haake
Model design

Fully-connected network on high-level jet shapes
Model design

**Secondary vertices**
- \((v_x, v_y, v_z), \sigma_{v_x}, \chi^2, L, L_{xy}, \sigma_{xy}\)
  - Convolution 1D
  - Dropout per layer: 0.1
  - 64
  - Max pooling
  - 128
  - Conv. kernel: 4
  - 256
  - Conv. kernel: 2

**Jet constituents**
- \(\eta, \varphi, r\)
  - Impact param., \(j_t\)
  - Convolution 1D
  - Dropout per layer: 0.1
  - Conv. kernel: 4
  - 128
  - Max pooling
  - 64
  - Conv. kernel: 2
  - 64
  - Conv. kernel: 2
  - Max pooling
  - Pool length: 2

**High-level properties**
- Jet shapes, jet \(p_T, N_{\text{const}}\)
  - Fully-connected
  - Dropout per layer: 0.1
  - 128
  - 128
  - 128
  - 128

**Dropout: 0.1 for all layers & branches**

**Fully-connected**
- Dropout per layer: 0.25
  - 128
  - 128
  - 128
  - 128

**Sigmoid neuron for binary classification**
Model design

Secondary vertices
$(v_z, v_y, v_x, \sigma_{vz}, \sigma_{vy}, \chi^2, L, L_{xy}, \sigma_{xy})$

- Convolution 1D
  - Dropout per layer: 0.1
  - 64
  - Max pooling
  - 128
  - Conv. kernel: 4
  - 256
  - Conv. kernel: 2

Jet constituents
- $\eta, \varphi, r$
- Impact param., $j_r$

- Convolution 1D
  - Dropout per layer: 0.1
  - 128
  - Conv. kernel: 4
  - Max pooling
  - 128
  - Pool length: 2
  - 64
  - Conv. kernel: 2
  - 64
  - Conv. kernel: 2
  - Max pooling
  - Pool length: 2

High-level properties
- Jet shapes, jet $p_t$, $N_{const}$

- Fully-connected
  - Dropout per layer: 0.1
  - 128
  - 128
  - 128
  - 128

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

Merge (concatenation)

Sigmoid neuron
for binary classification

FC-network on top
Higher dropout here: 0.25
Model design

Secondary vertices

Jet constituents

High-level properties

Other model properties
- ADAM optimizer
- Loss: binary crossentropy
- Activation function: ReLU

Last neuron is sigmoid-activated
Input features

Basis
- FastJet anti-$k_T$ jets, resolution parameter $R = 0.4$, tracks only
- Underlying event corrected
- Jets fully contained within detector acceptance

Features
- High level parameters: Jet mass, radial moment, momentum dispersion, LeSub, track count, jet $p_T$ (see backup for definitions)
- Array of constituents: $\eta$, $\varphi$, $r$ (relative to jet axis), impact parameter $j_T$
- Array of secondary vertices (3-prong, dispersion < 0.05)
  Each vertex:
  - $(x, y, z)$ rel. to primary vertex
  - Transverse plane distance & uncertainty: $L_{xy}$, $\sigma_{xy}$
  - Vertex track dispersion $\sigma_{vtx}$, fit quality $\chi^2$
  - In addition: $L_{xy} / \sigma_{xy}$, total decay length $L \to$ information is in there, but it makes it easier for the network to learn
Simulation dataset I

- p-Pb dataset PYTHIA6 Perugia 2011 + HIJING
- Enhanced b-/c-quark production, $\sqrt{s_{NN}} = 5.02$ TeV
- Sample truth (i.e. jet type) is set with geometrical matching and particle level information:
  - If a B-hadron is found within $R = 0.7$ of the jet, it is considered a b-jet.
  - If instead of a B-hadron, a C-hadron is found, the jet is considered a c-jet.
  - All other jets are tagged as light-flavor jets.
Simulation dataset II

- Strictly separated samples for training, validation, and testing
- 200’000 (training), 40’000 (validation) for each class
  - Signal class: 100% b-jets
  - Background class: 20% c-jets & 80% 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:
  ~1.2M udsg-jets, ~150k c-jets
Results
Observables

• Goal at this stage: **Performance evaluation & comparison to conventional, cut-based method**

• Indicator: b-jet tagging & c-/udsg-mistagging efficiencies
• The higher the b-jet efficiency, the higher the mistagging efficiencies
  → Need to find optimum working point

• Here: b-jet tagging efficiency directly set by cutting on score
  score/efficiency relation known in MC simulations
• In the following, the mistagging efficiencies will be shown for several working points

• Note: jet $p_T$ from geometrical matching with particle-level jets
The present ML-assisted tagging method is very promising, compared to conventional method
– mistagging efficiency lower for c- and udsg-jets
– mistagging efficiencies rise less steep when considering higher b-jet tagging efficiency
Mistagging efficiencies vs. jet $p_T$

- Mistagging efficiency vs. jet $p_T$
- Solid symbols represent efficiencies with present ML-based method
- Open symbols show conventional, cut-based performance
- For the sake of comparison: Between 20-50 GeV/c b-jet efficiency set to values used in cut-based method

The present ML-assisted tagging method is very promising, performance better over whole jet $p_T$-range
Mistagging efficiencies vs. jet $p_T$

- Mistagging efficiency vs. jet $p_T$ for higher b-jet efficiencies (ML-based method only)
- Solid symbols: c-efficiency
- Open symbols: udsg-efficiency

- As expected, higher b-jet efficiencies lead to much higher mistagging rates
- ML-based method expected to allow higher b-jet efficiencies than cut-based methods while showing same mistagging efficiencies
- A future analysis could compare several working points
Model shows slow learning up to roughly 250 epochs.
After 250 epochs, model remains stable.
Learning rate successively lowered: [0.001, 0.0005, 0.0002, 0.0001]
Strong loss differences due to regularization.
Accuracy shows slight overfitting, but AUC (see next slide) and loss still fine.
Training control plots

- **AUC = Area Under ROC Curve**
- AUC reveals slow, but constant learning up to 250 epochs
- Tests show that the performance cannot be improved by just learning more epochs
- Interesting: Learning onset between epochs 60-80

![Graph showing ROC curve and AUC](image)

**ALICE Simulation**
- PYTHIA + HIJING, p-Pb \( \sqrt{s_{NN}} = 5.02 \text{ TeV} \)
- FastJet, anti-\( k_T \), \( R = 0.4 \), \( |\eta_{\text{jet}}| < 0.5 \)
- \( 15 \leq p_{T,\text{jet}}^{\text{gen}} \leq 120 \text{ GeV/c} \)
Summary

- ML-assisted tagging method has been developed
- Mixture of deep/shallow learning
  - Conventional FC networks on high-level parameters
  - Deep convolutional networks on low-level parameters
- Performance evaluated in p-Pb MC simulations and compared to cut-based method

Results are very promising
- Tagging method might allow higher b-jet efficiencies
- Lower mistagging rates
- Systematic uncertainties still to be assessed
- Data applicability to be checked → Better generalization with data from different generator/decayer?
Outlook

- Next step: Apply tagging method in p-Pb collisions and test performance on data
- Idea: Use both conventional and ML-assisted method and compare performance in a paper
  Strength of ALICE: reach down to low $p_T$

- On agenda: Train model with higher statistics at high $p_T$

- Tagging bias: All tagging methods (ML or cut-based) potentially bias the sample in an unwanted way
  → Careful examination of tagged sample

Thank you for your attention!
Backup
Jet shape definitions

**Radial moment**
“$p_T$-weighted width of jet”

$$g = \sum \frac{p_{T,\text{const}}}{p_{T,\text{jet}}} |\Delta R_{\text{const}}|$$

**Momentum dispersion**
Contains direct information about jet fragmentation

$$p_T D = \sqrt{\sum p_{T,\text{const}}^2 / \sum p_{T,\text{const}}}$$

**LeSub**
Difference between leading and subleading constituent

$$\text{LeSub} = p_{T,\text{lead.}} - p_{T,\text{sub.}}$$

**Jet mass**
Connected to virtually of initial parton that showered into jet

$$M = \sqrt{E^2 - p_T^2 - p_Z^2}$$