Development of a Flavor Tagging Algorithm using Deep Learning on ILC

1st general meeting of ILC-Japan Physics Working Group

2022/11/25

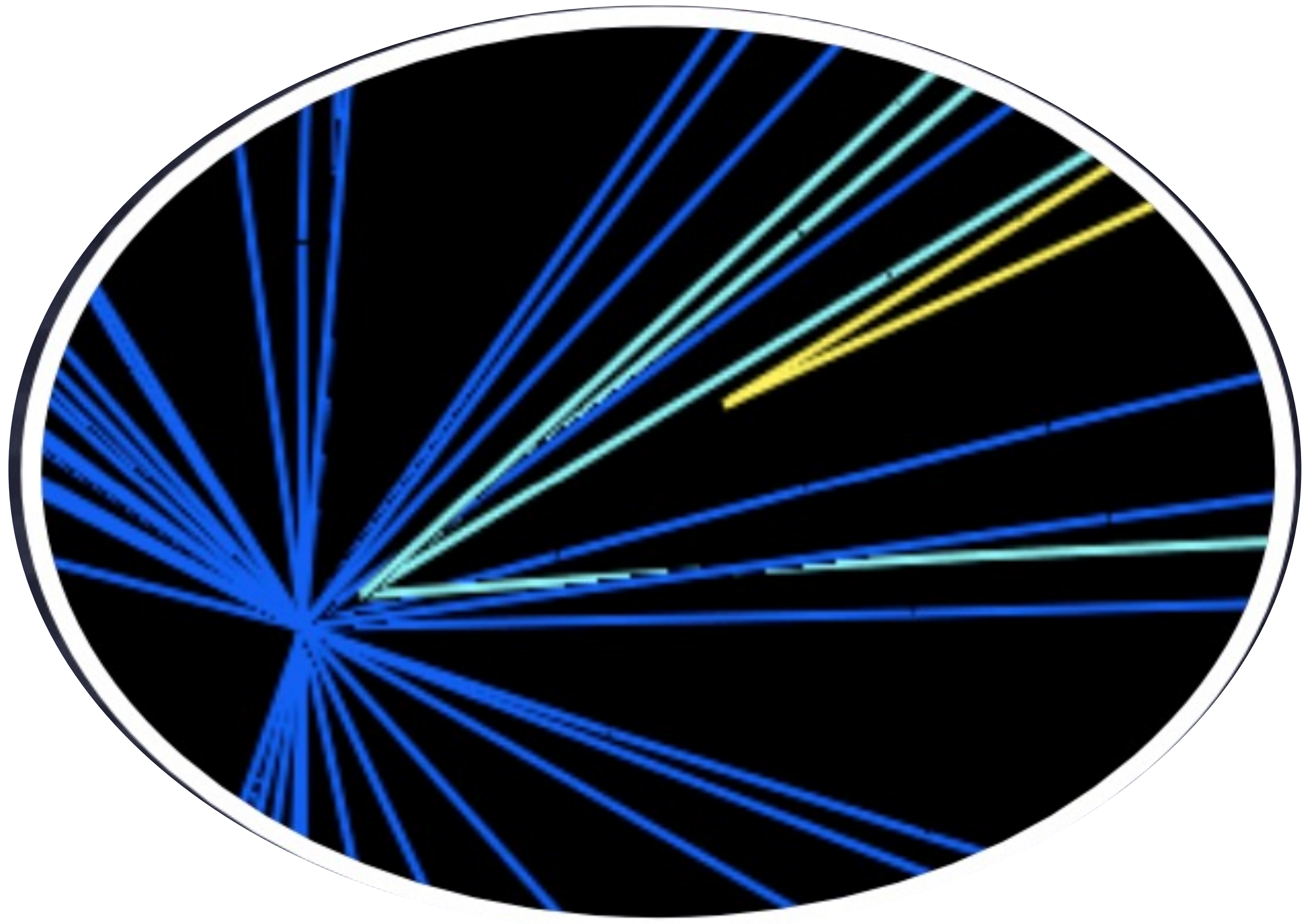
T.Onoe[A], T.Suehara[A], K.Kawagoe[A], T.Yoshioka[A],
H.Nagahara[B], Y.Nakashima[B], N.Takemura[C]
(Kyushu Univ.[A], Osaka Univ.[B], Kyushu Institute of Technology[C])

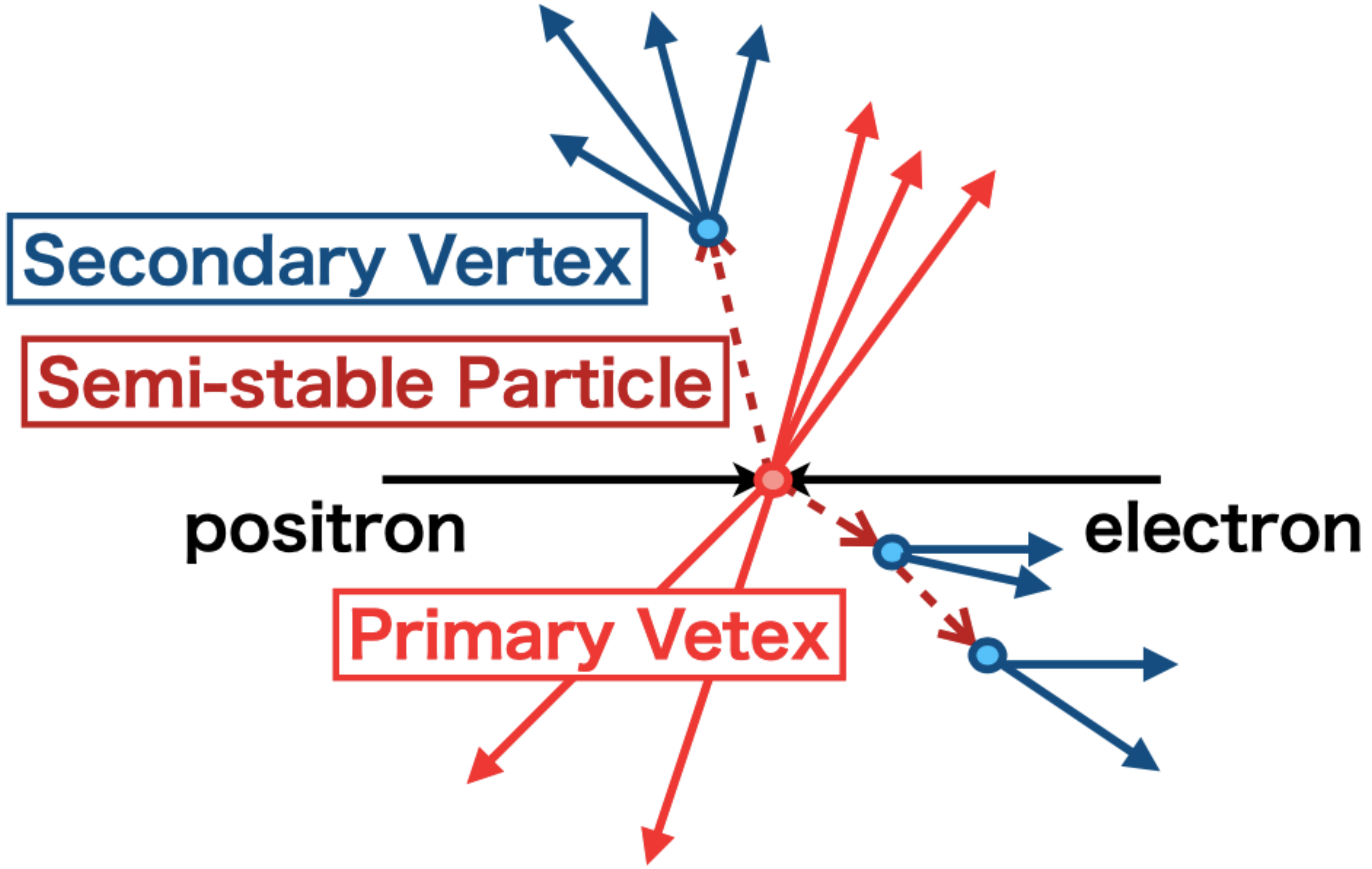
Contents

Introduction

Flavor Tagging

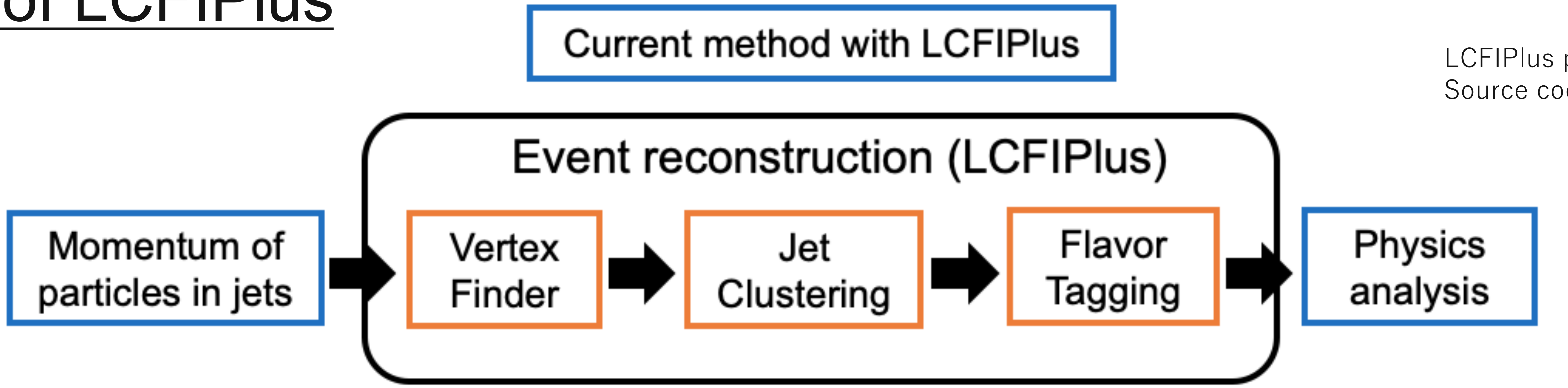
- Jets are bandles of hadrons originated by quarks and gluons.
- It is important to identify quarks/gluons (b/c/g/uds) of the origins of the jets.
  e.g. separation of H → bb, cc, gg
- A flavor tagging is the algorithm which classify the quarks.
- b/c hadrons can fly before their decay, because of their finite lifetimes
  → Vertex finding is important for the flavor tagging





Introduction

Structure of LCFIPlus



LCFIPlus paper: NIM A 808 (2016) 109-116
Source codes: https://github.com/lcfiplus/LCFIPlus

- Vertex Finder : Find primary and secondary vertices
- Jet Clustering : Reconstruct jets by clustering particles
- Flavor Tagging : Classify jets as b/c/others

・In the LCFIPlus, the flavor tagging algorithm is based on the Boosted Decision Trees, which is traditional ML.

✓ **Purpose on this study**

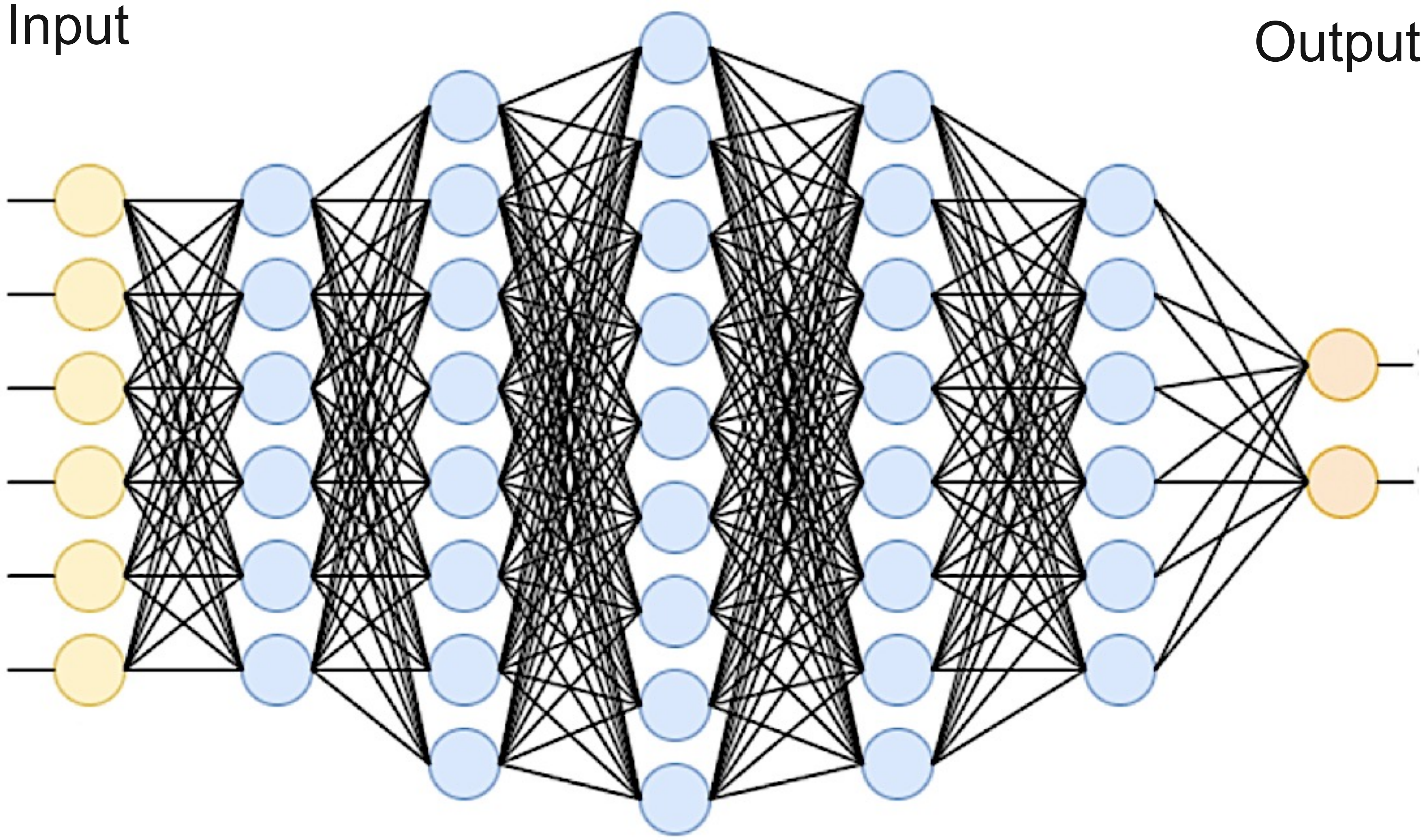
- Improve the performance of the flavor tagging by introducing deep-learning techniques.
- Combine vertex finding and flavor tagging in single DNN structure or GNN structure.

Introduction

Deep Learning

- One of the supervised machine learning technologies.
- “Deep” refers to the use of multiple layers.
- Machines learn by using deep neural networks (DNNs) and extract features from big data.

- There are various practical networks.

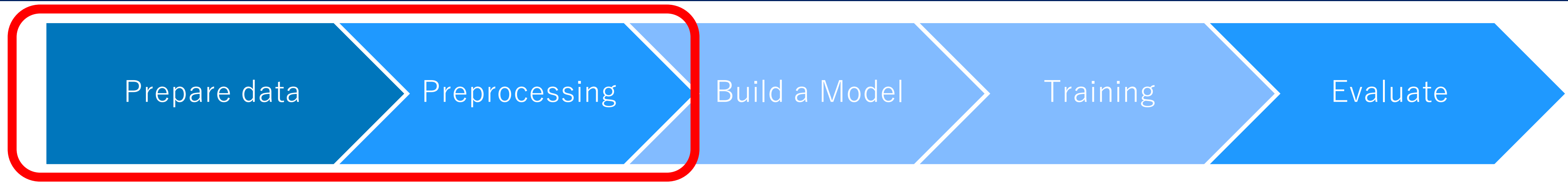


- To use deep learning, it is necessary to follow the flow.

Prepare data → Preprocessing → Build a Model → Training → Evaluate

Data information for DNN

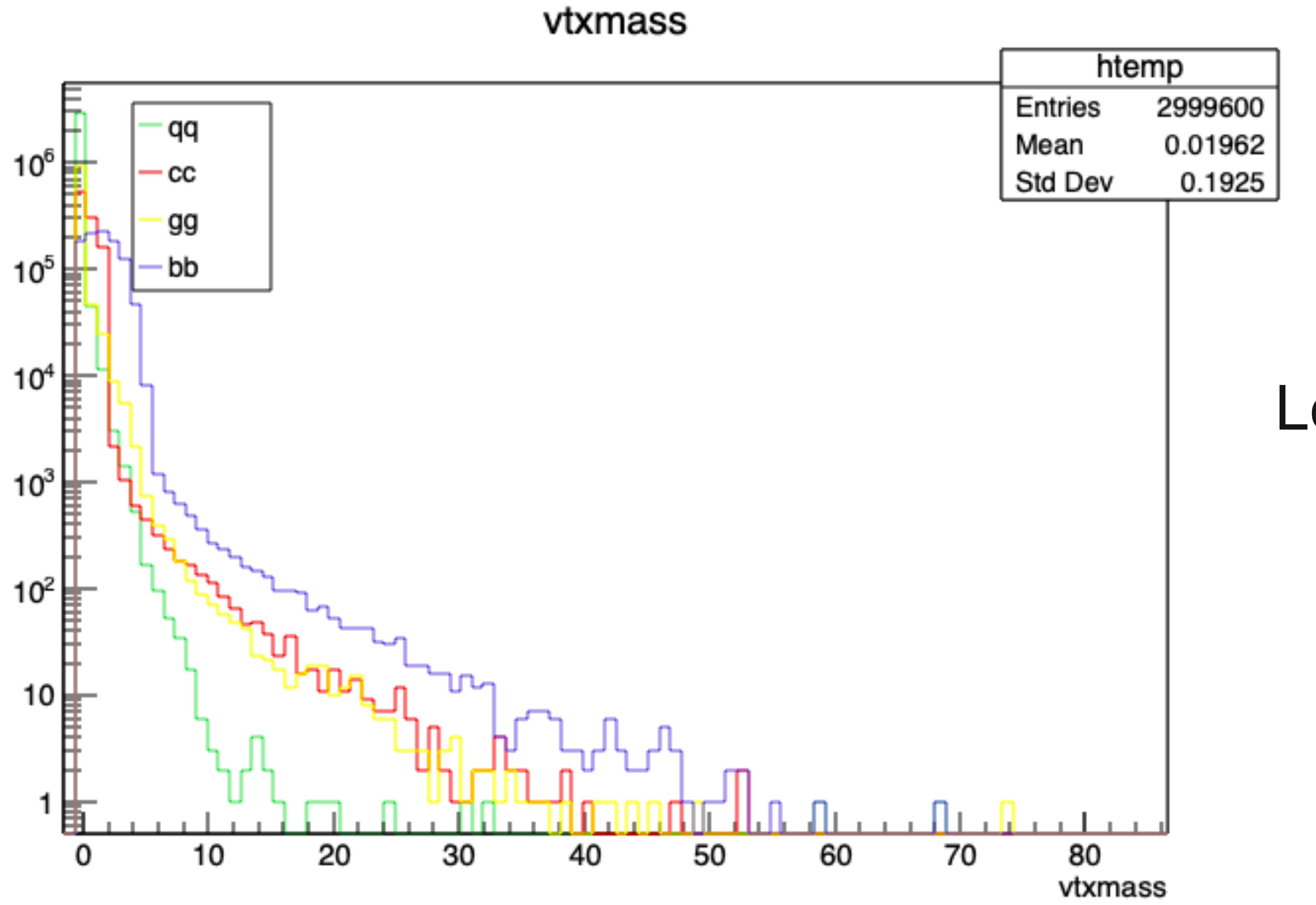
Prepare data



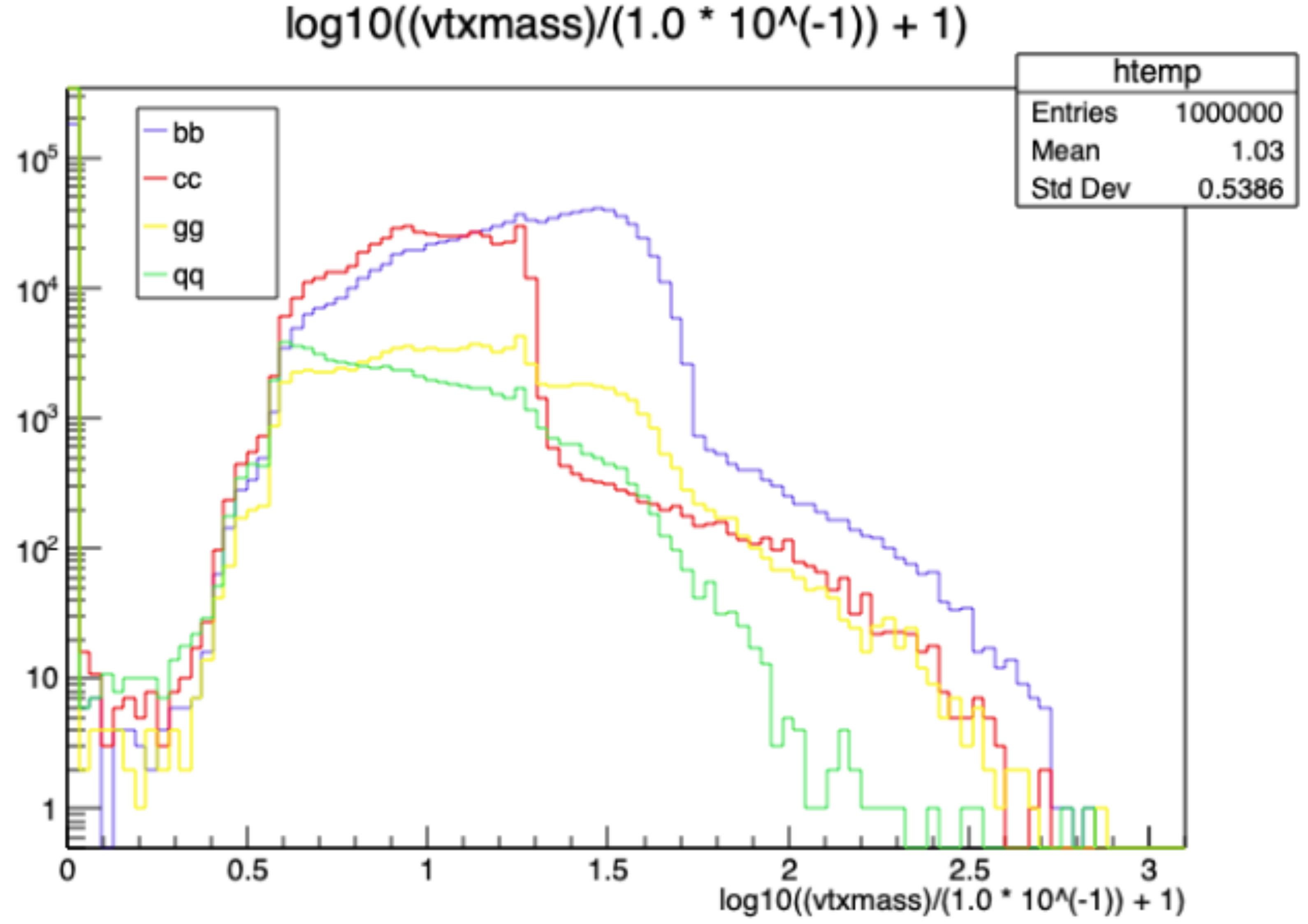
- 4 million events data from ILD full simulation (250GeV, bb:cc:gg:qq = 1:1:1:4)
- 42 variables from vertex finder is used for training
  (e.g. number of vertices, position/mass/probability/number-of-tracks of each vertex, displacement of tracks from the interaction point etc.)

Preprocessing

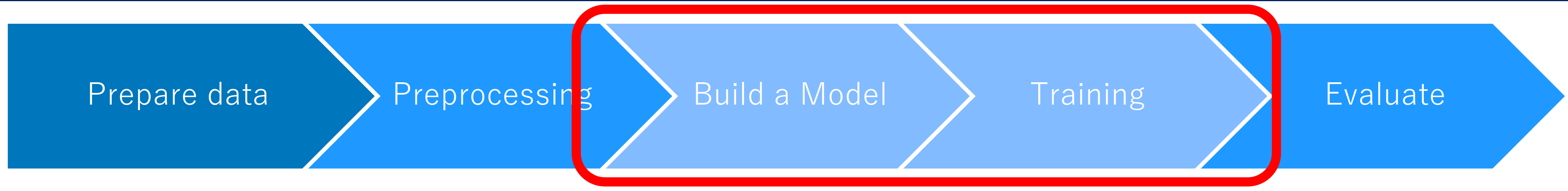
Before feeding to the input of the network by transformation of the variables distribution of the input variables should be flatten and scaled.



Logarithmic transformation



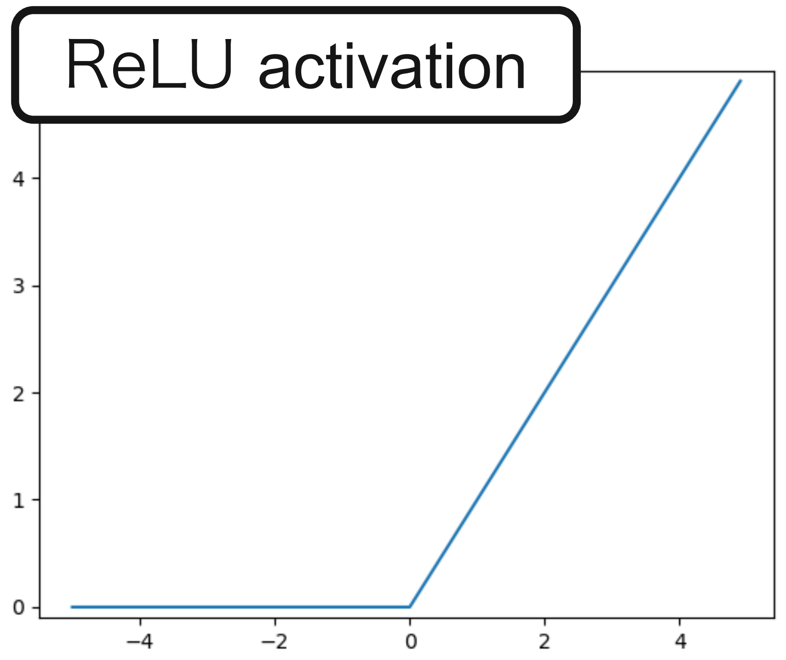
DNN Model & Training

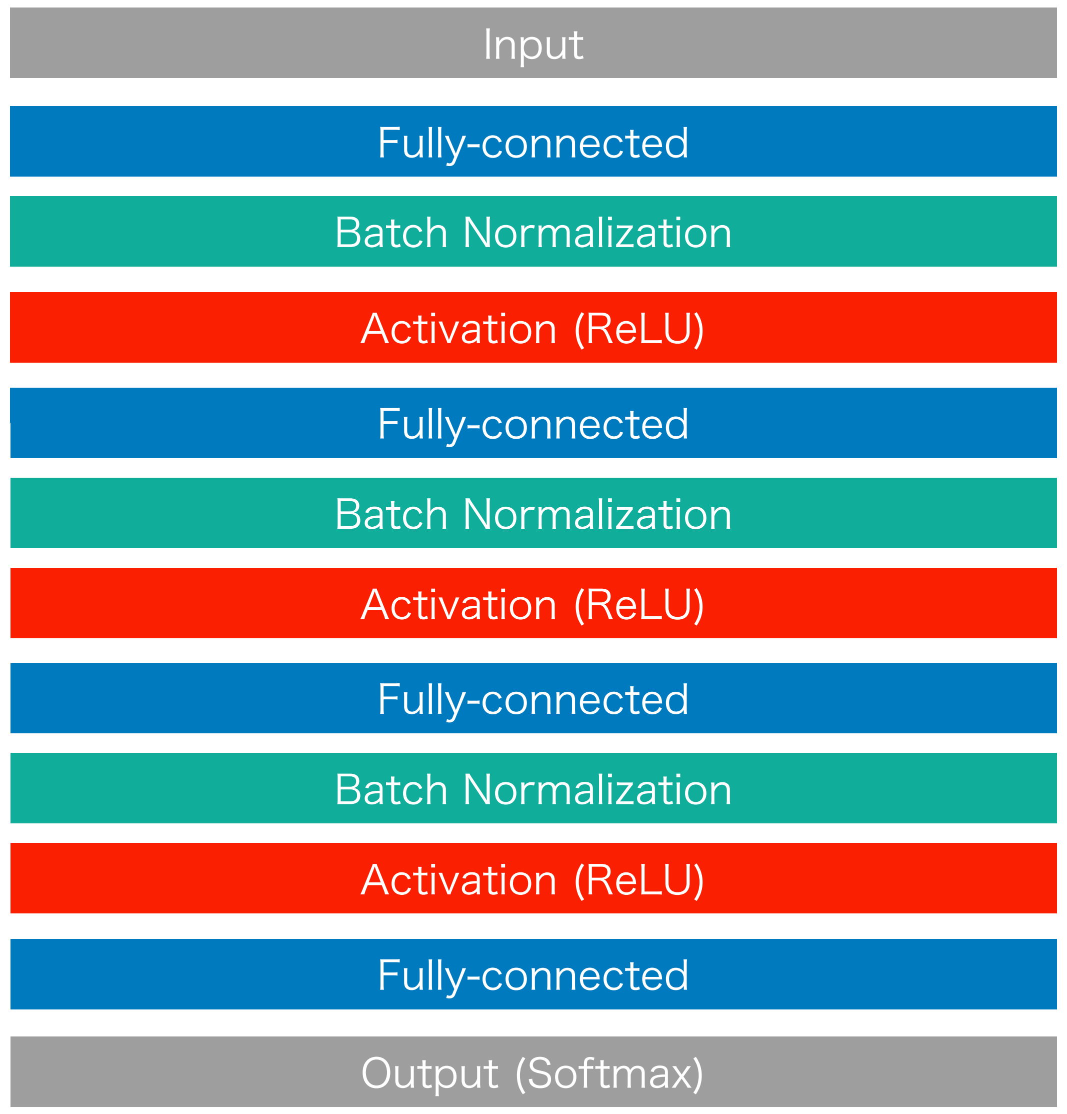


Build a Model

・Simplest DNN model as first trial and basis for comparison to modern networks.

- Input : 42 parameters
- 4 fully-connected layer with batch normalization and ReLU activation
- Output : 3 categories (b-, c-, uds- likeness)





Training

- Loss function : Categorical cross entropy
- Optimization : Adam (Learning rate : 0.01)
- The number of training : 100 epochs
- Batch size : 1024

$$\text{Loss} = -\sum_{i=1}^{\substack{\text{output}\\ \text{size}}} y_i \cdot \log \hat{y}_i$$

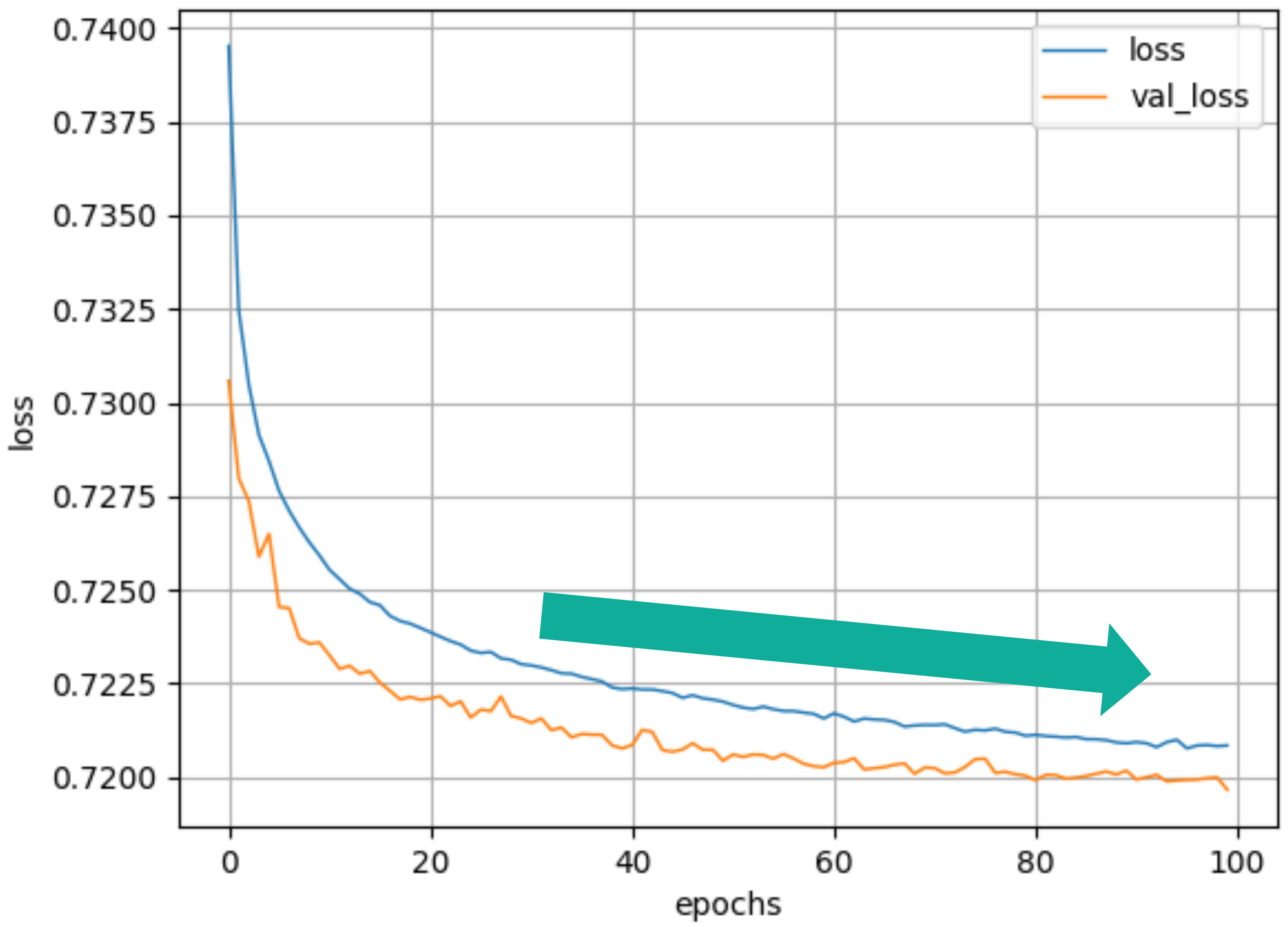
Results of DNN

Prepare data → Preprocessing → Build a Model → Training → Evaluate

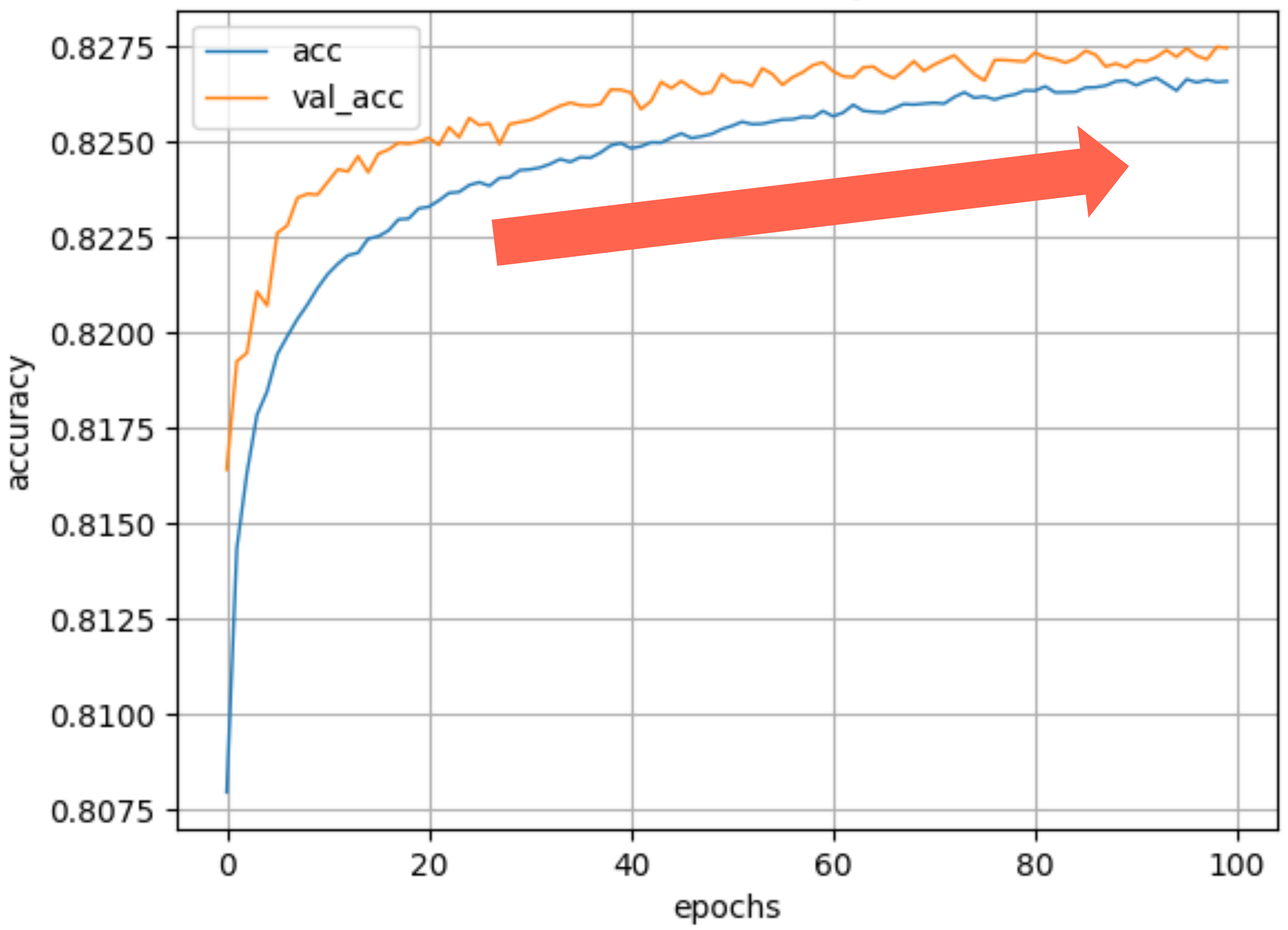
Evaluate

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

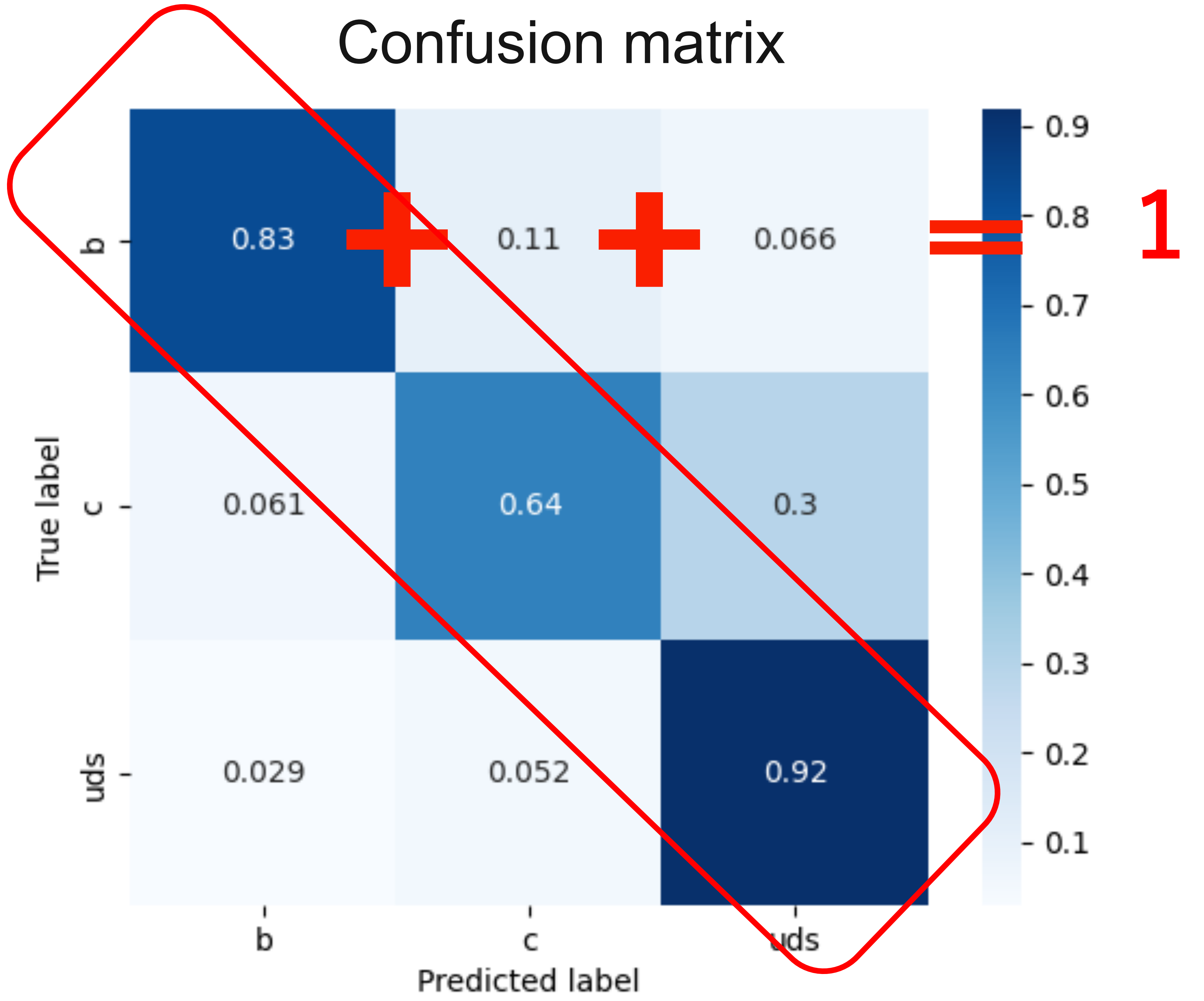
Loss



Accuracy



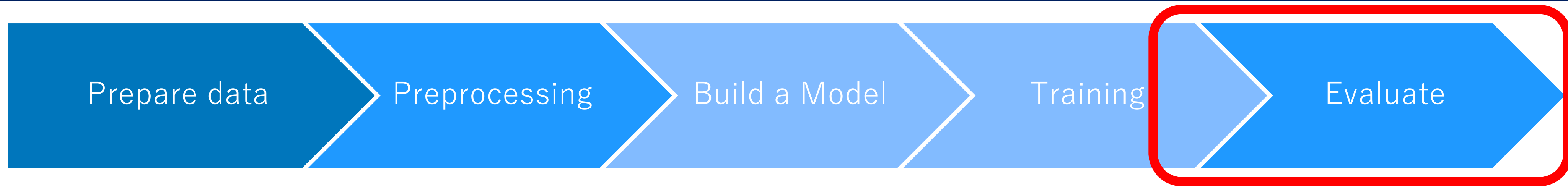
Confusion matrix



From plots, we can see that the model has comparable performance on both train and validation datasets (labeled test).

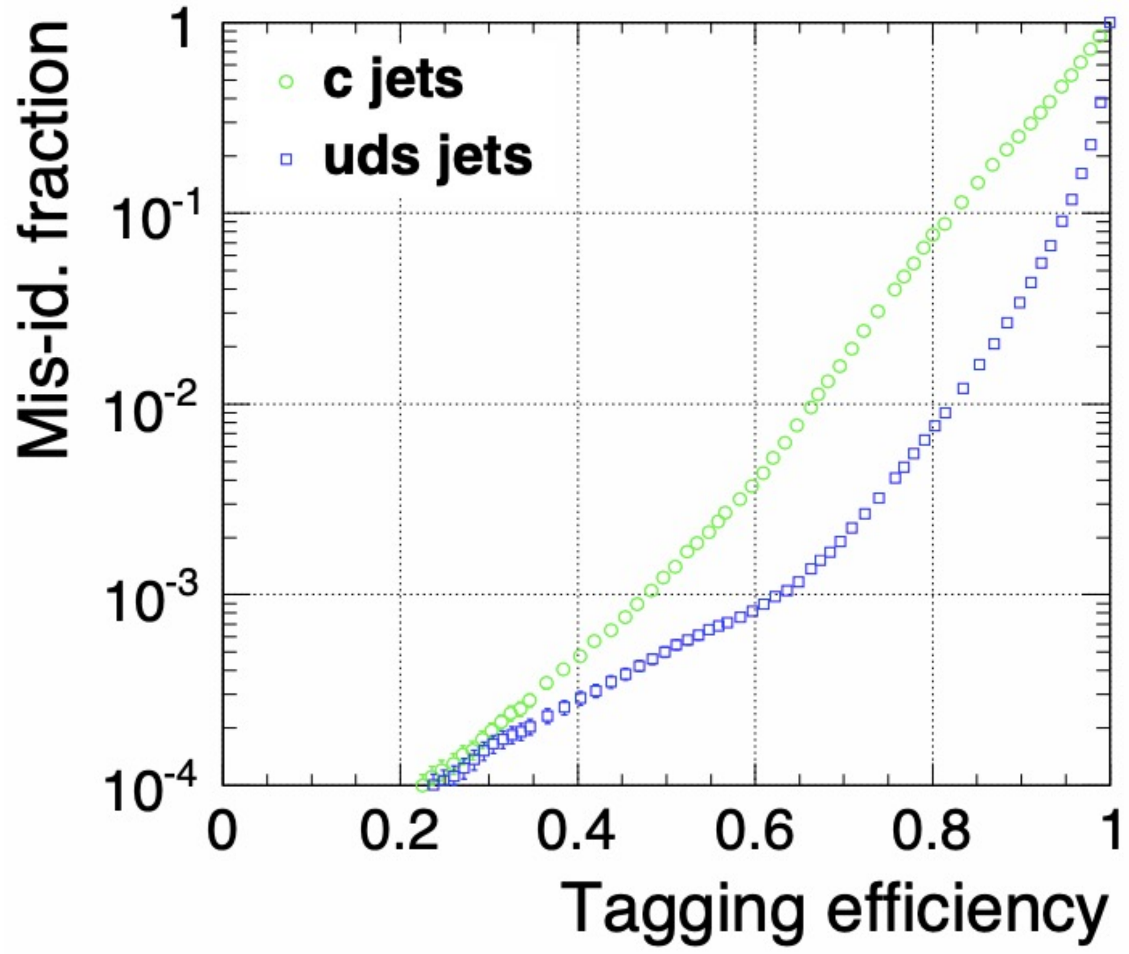
uds events are classified well, but c events are classified at 65% accuracy.

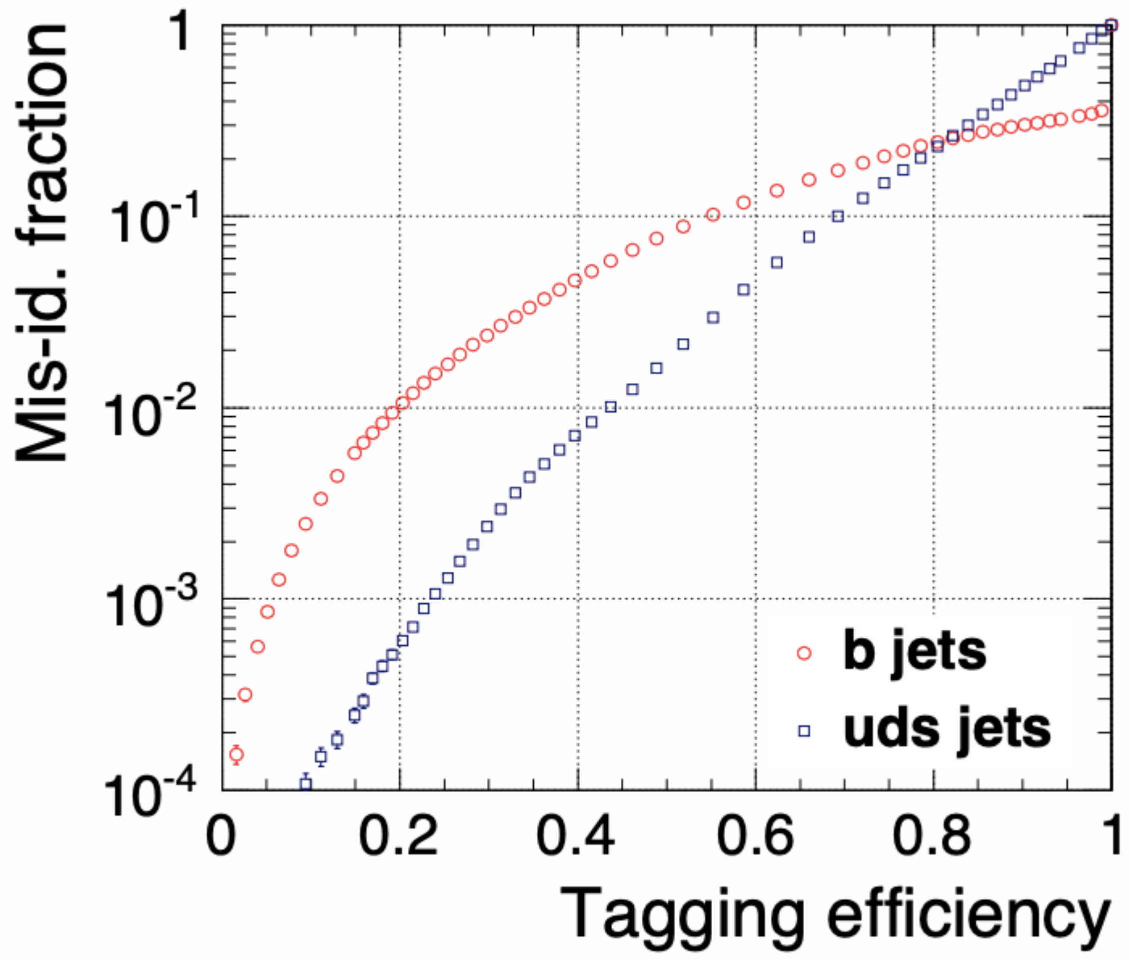
Results of DNN



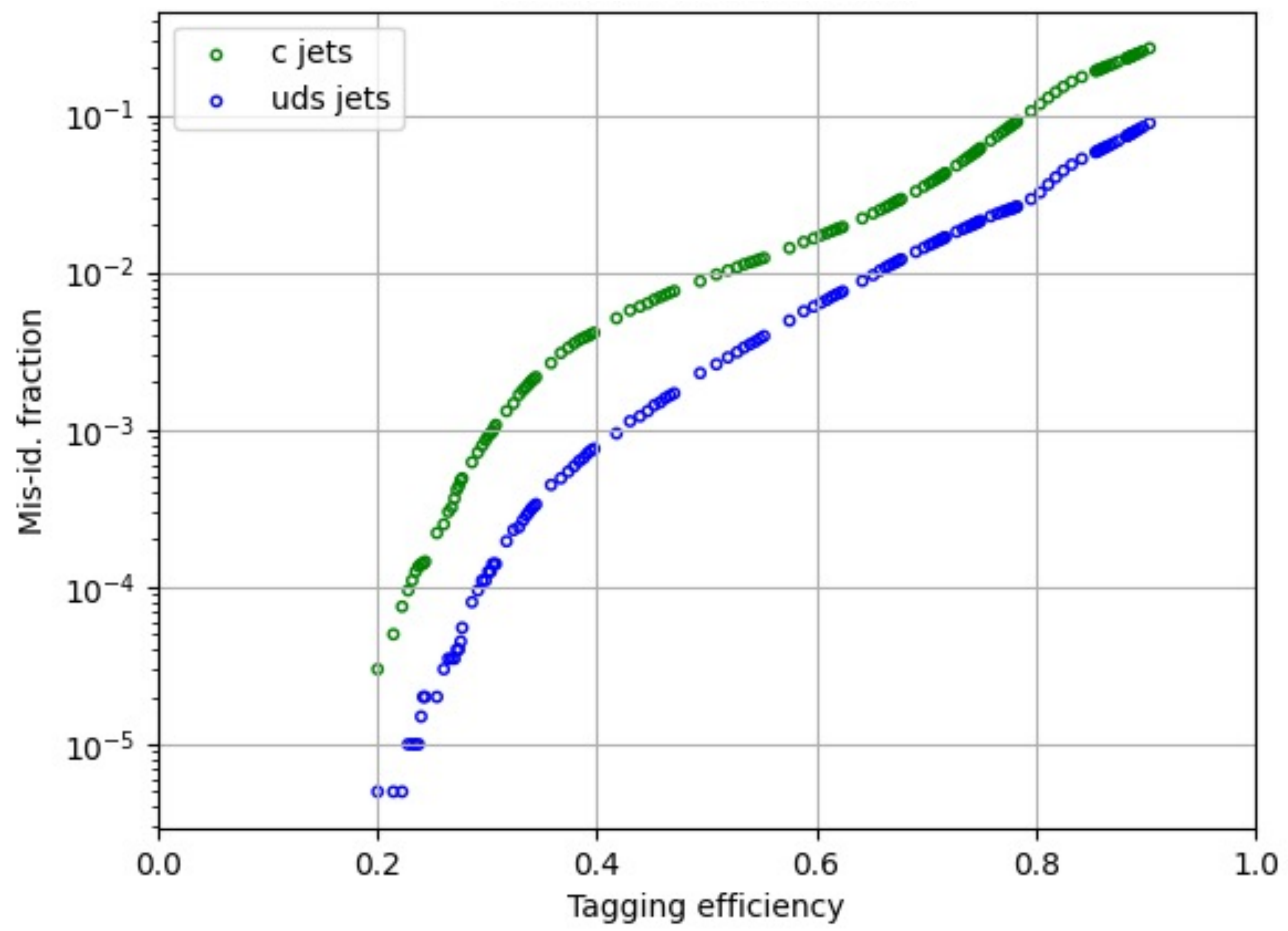
Evaluate

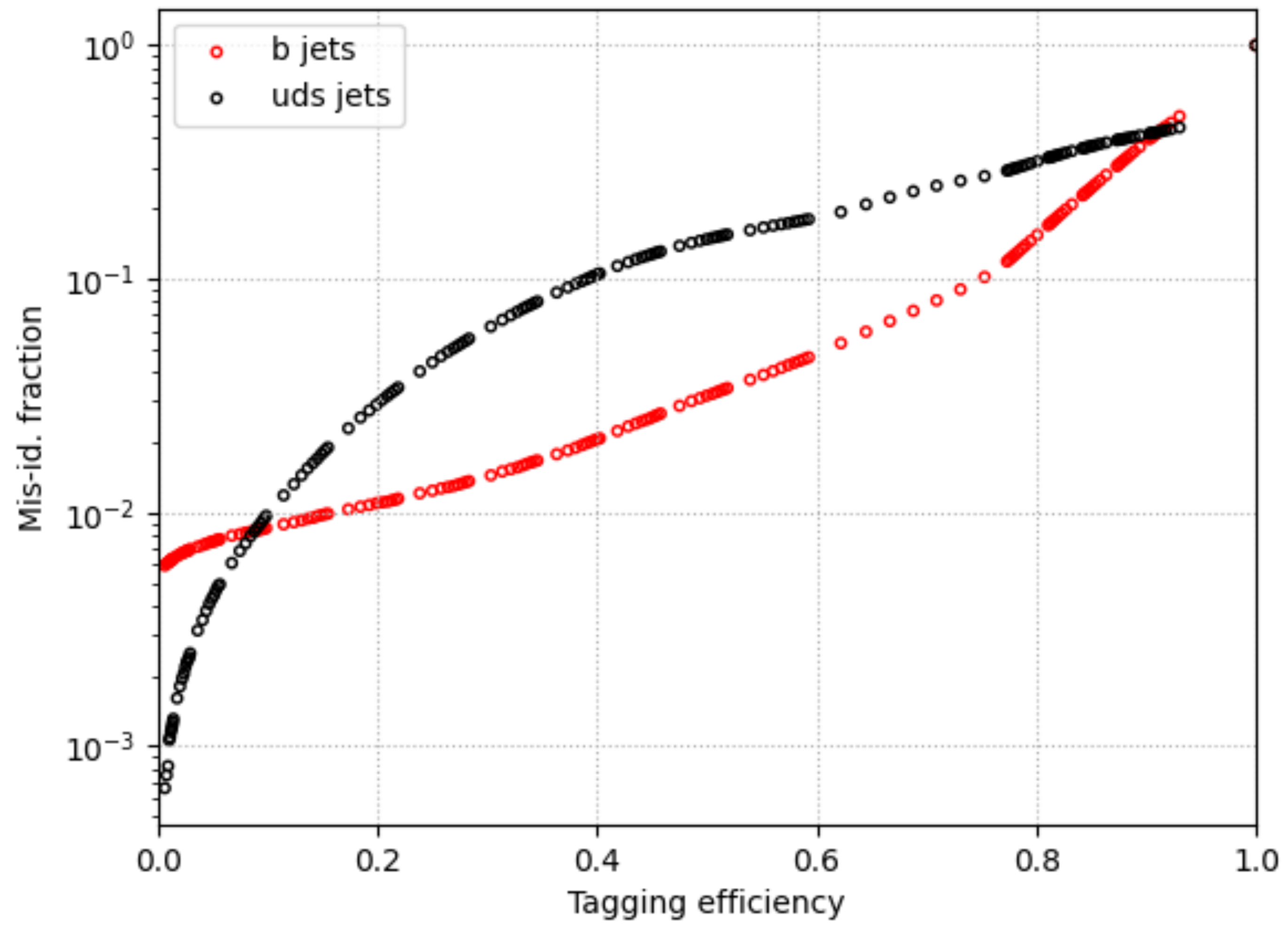
LCFIPlus





DNN



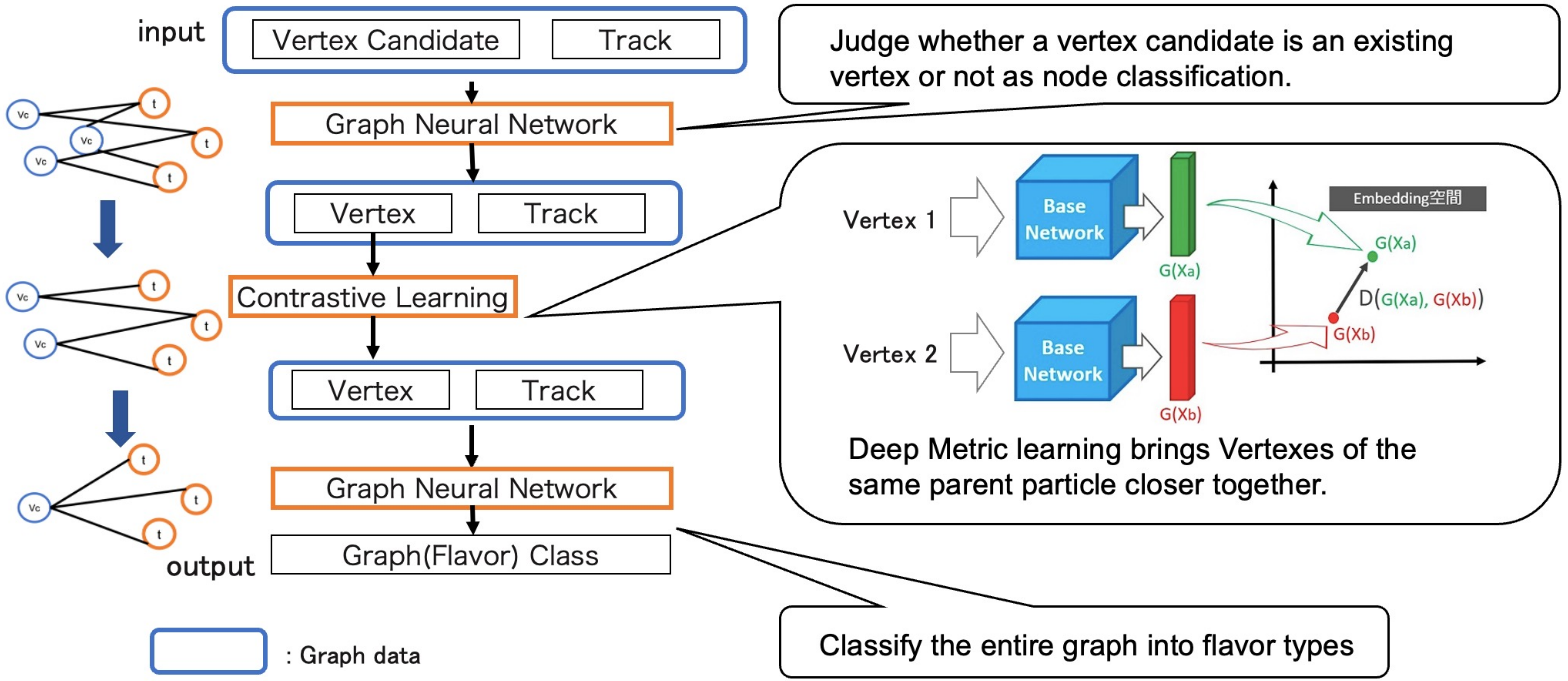


LCFIPlus is better identification now.

Approach by Graph structure data

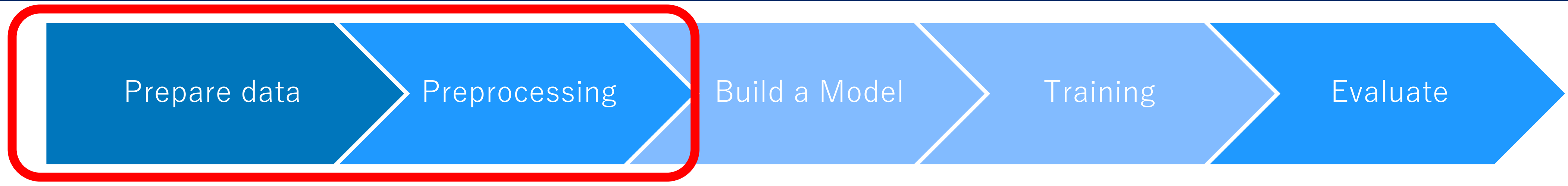
- As our strategy, Graph Neural Network (GNN) may be one solution.
- Graph can be represented with less information loss.

Overview of Graph approach



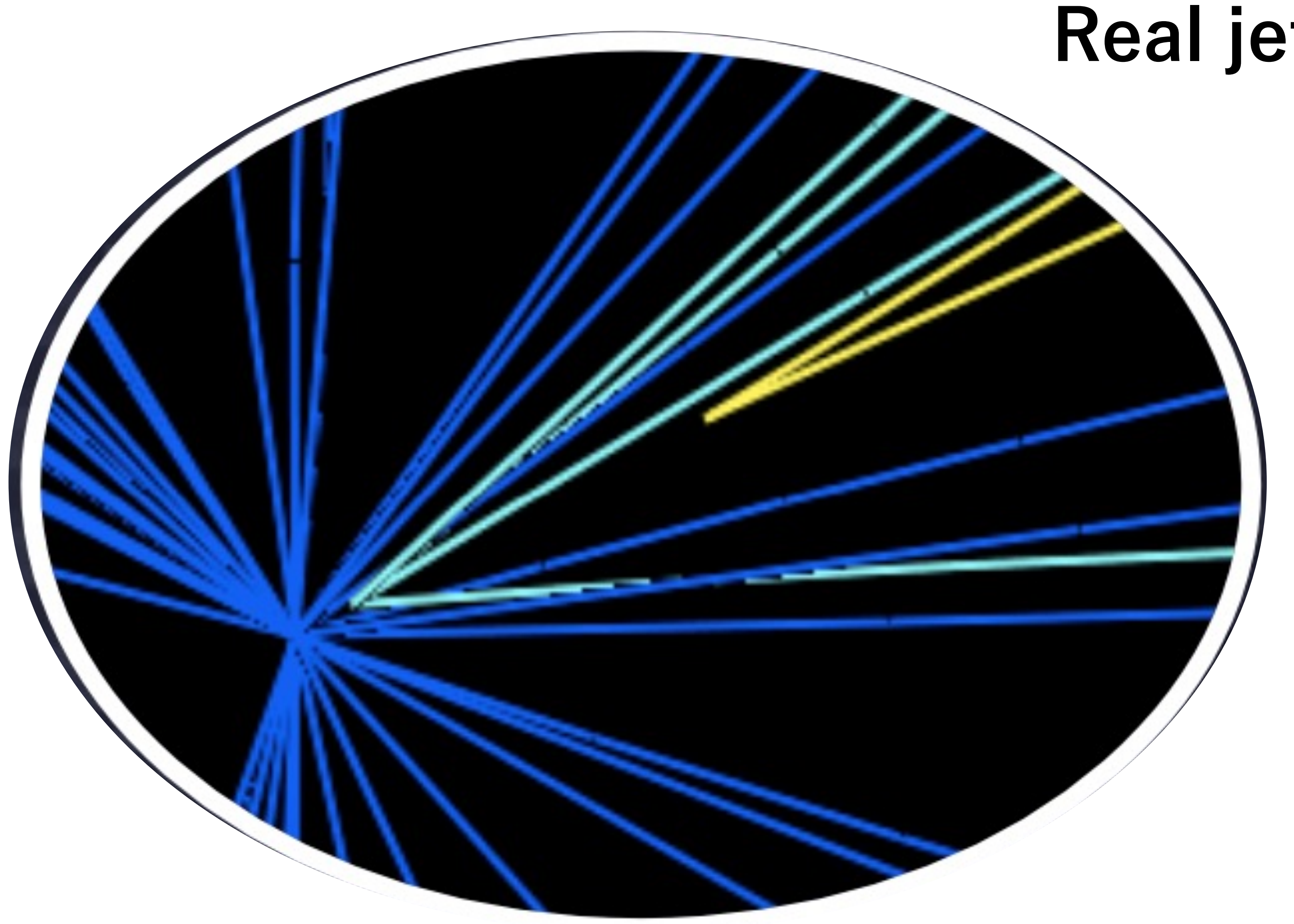
Data and Graph constructure

Prepare data and preprocessing

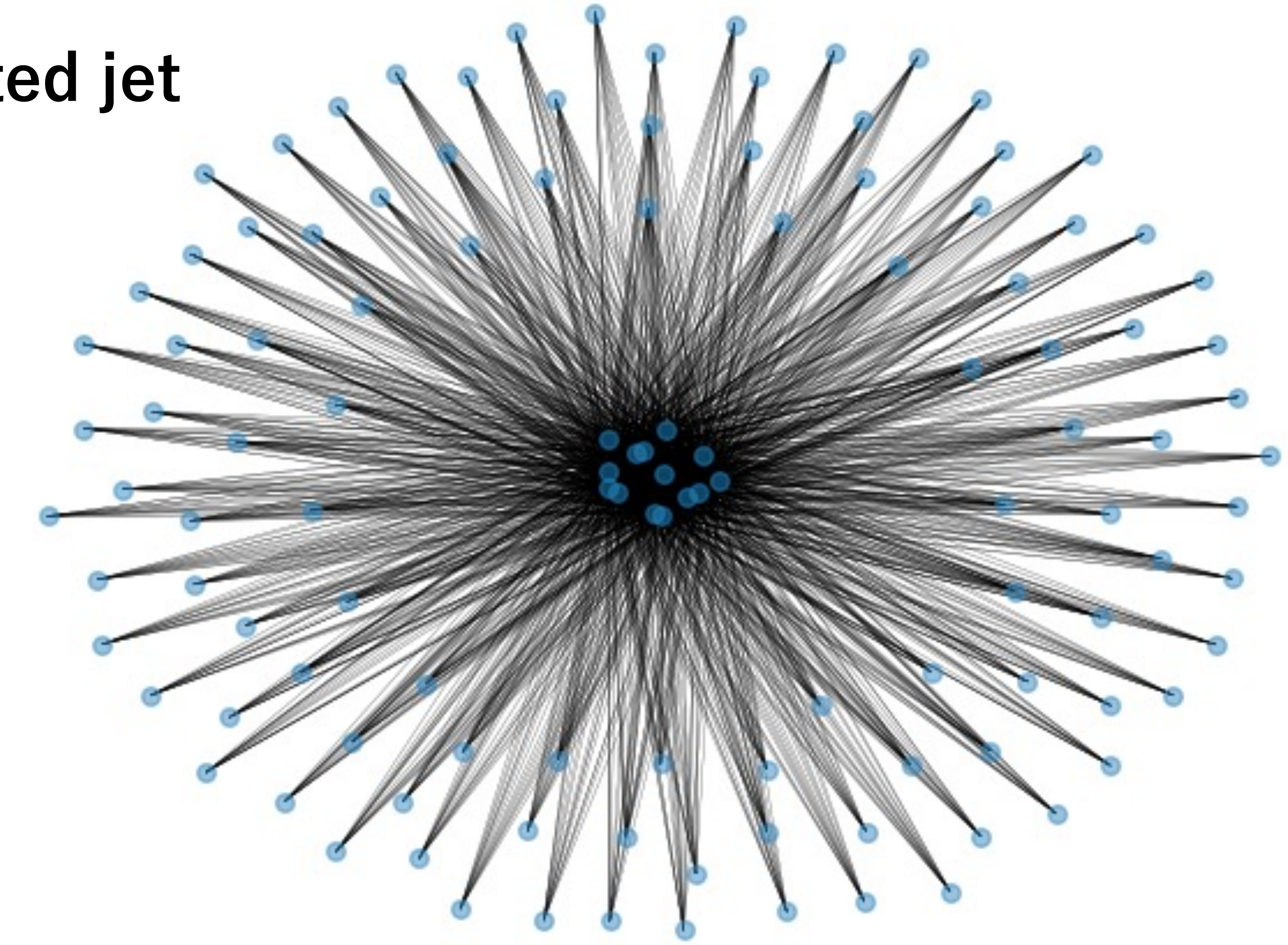


- 10k events data from ILD full simulation (250GeV, bb:cc:qq = 1:1:1)
- Features: Track ... 5 track fitting parameters $(d_0, z_0, \Omega, \phi_0, \tan\lambda)$
  Vertex Candidate (VC) ... Position(3d), probability (VC created by Vertex Fitter)
  ※ Use 3-D fitting for vertex position
- Pre-process each parameter for standardization (Mean=0, Variance=1)

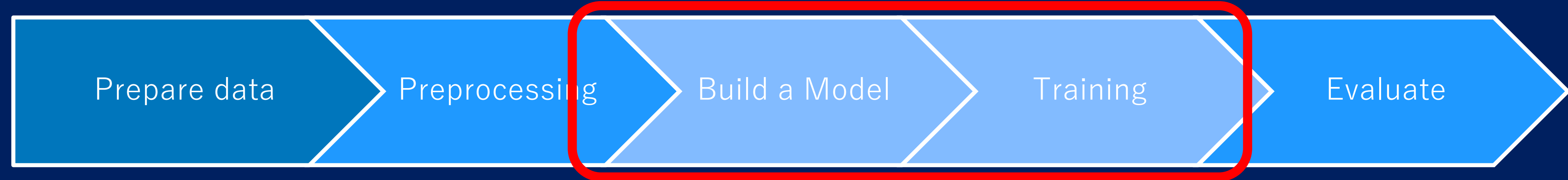
Real jet



Constructed jet



GNN Model & Training



- Network: GCN(Graph Convolution Network) 3 Layer

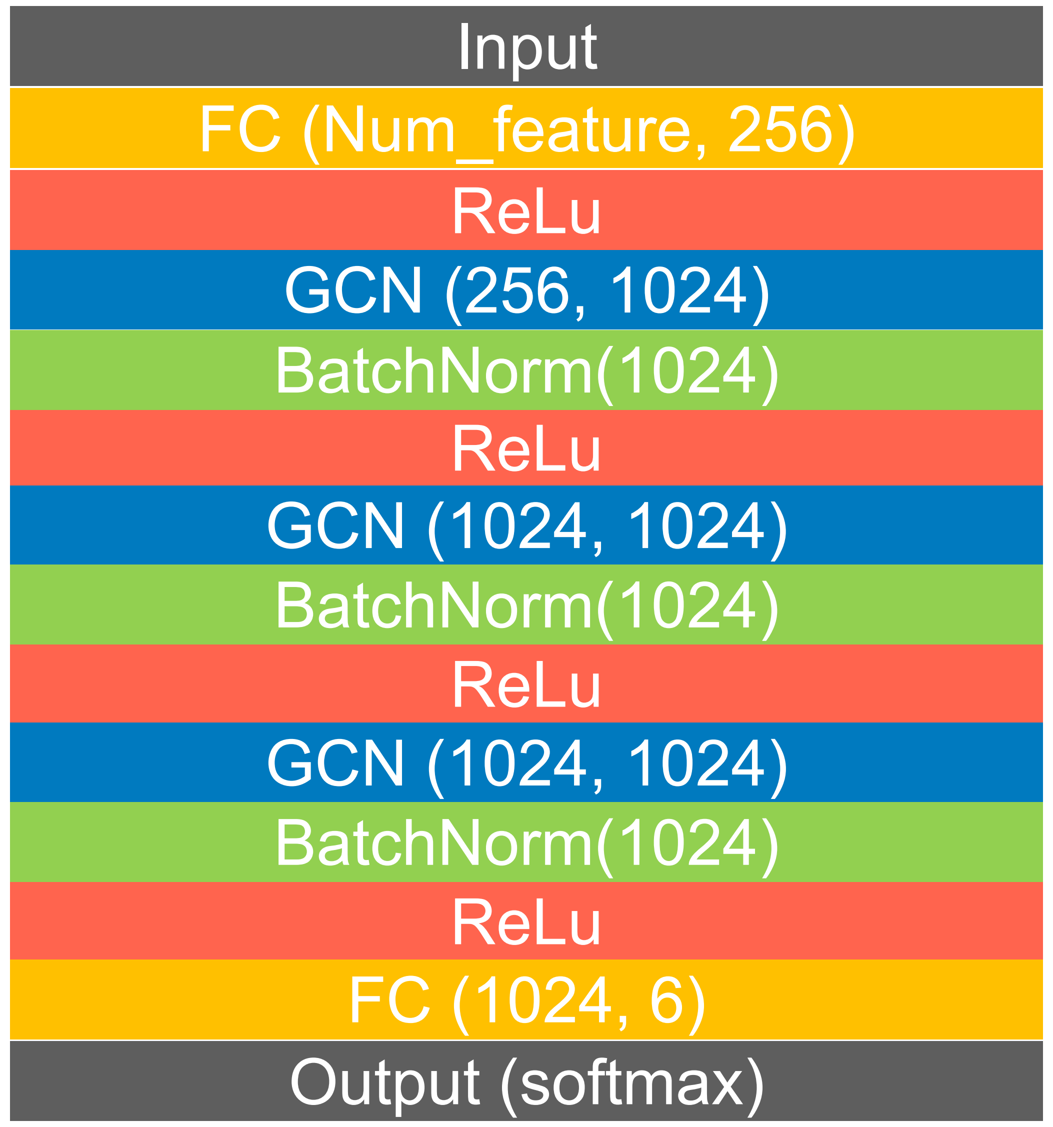
  <Semi-Supervised Classification with Graph Convolutional Networks: arXiv:1609.02907>

  - spectral convolution ... calculate by filter to graph Lagrangian. (contain entire graph info)
  - spatial convolution ... calculate at connected proximity nodes. (contain connected node info)

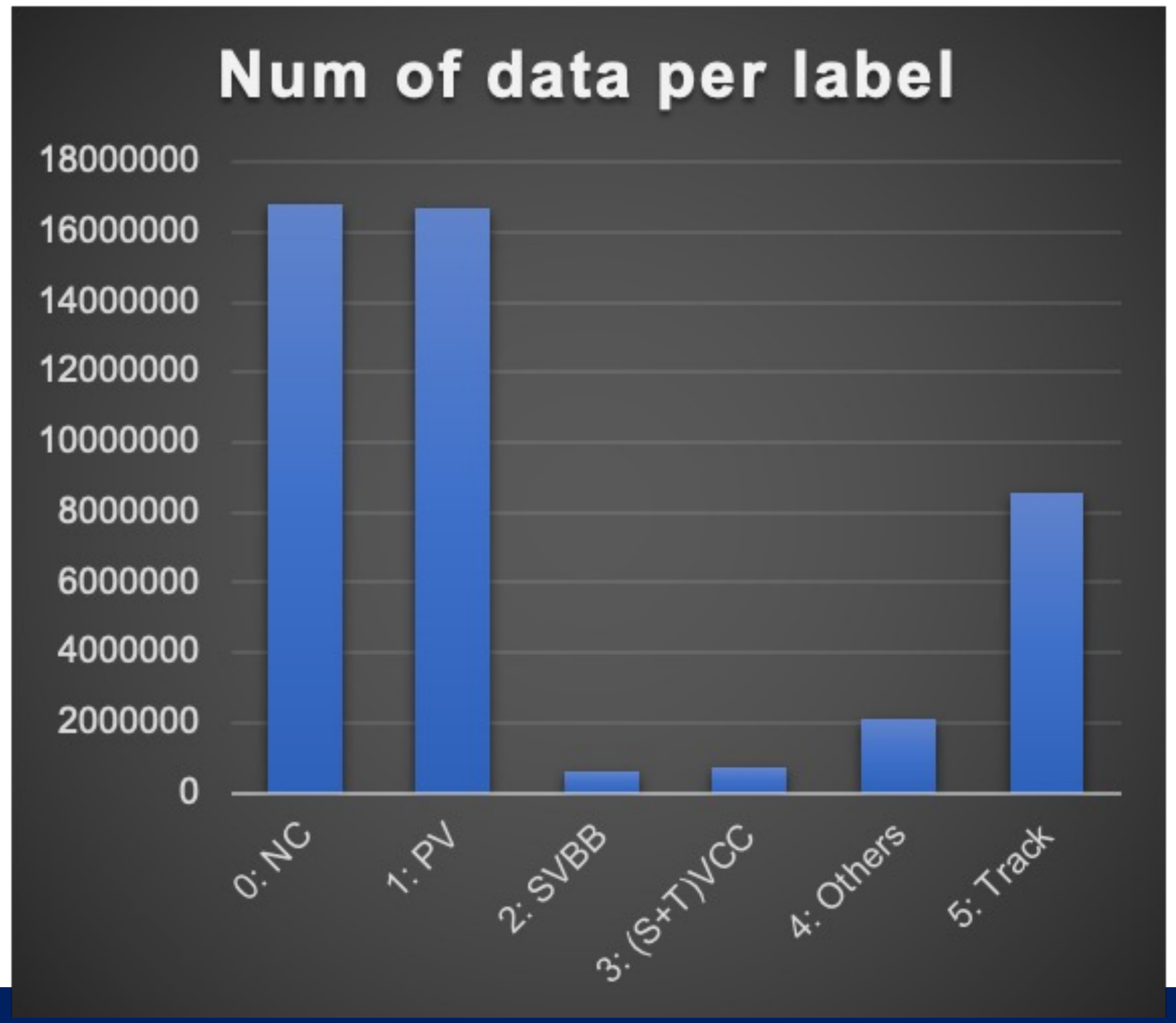
- In step1, we classify nodes to identify VC that do not exist.
- Answer label

  0 : NC ... Non-existent Vertex
  1 : PV ... Pair from primary vertex
  2 : SVBB ... Pair from secondary vertex of b-flavor
  3 : (S+T)VCC ... Pair from secondary or tertiary vertex of c-flavor
  4 : Others ... Other track pair
  5 : Track

- Loss weighted by inverse of the number of data for balanced data training

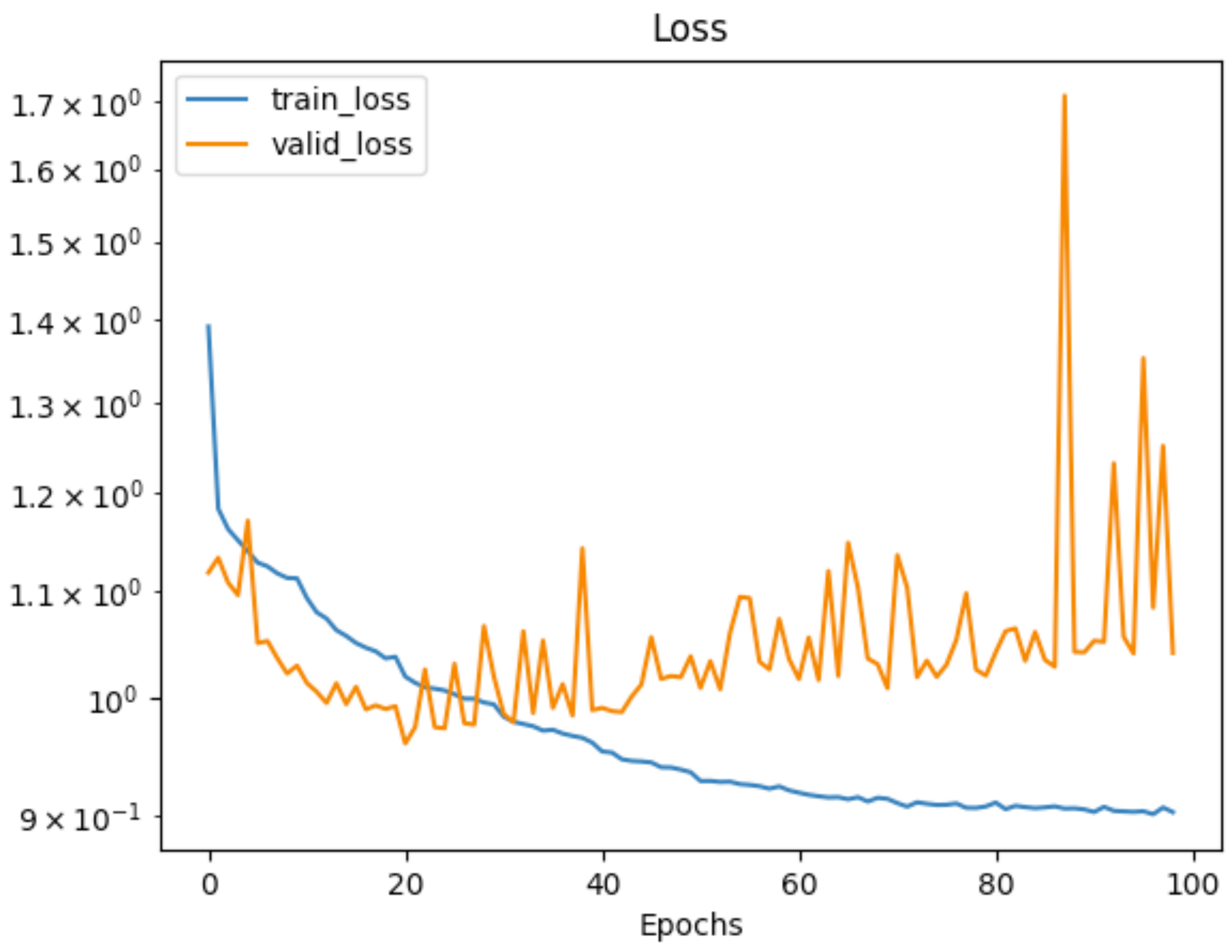


Network structure

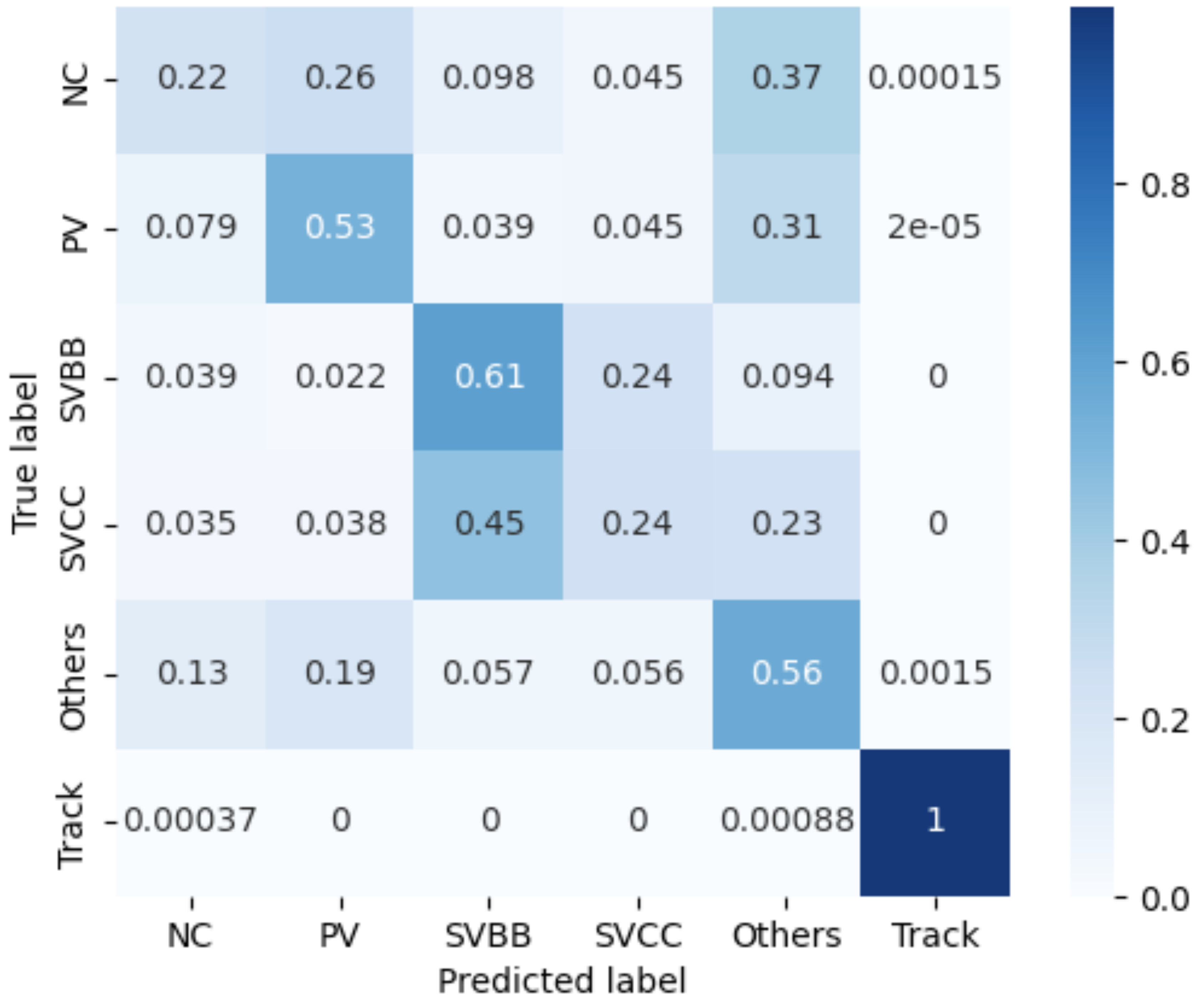


Results by GNN

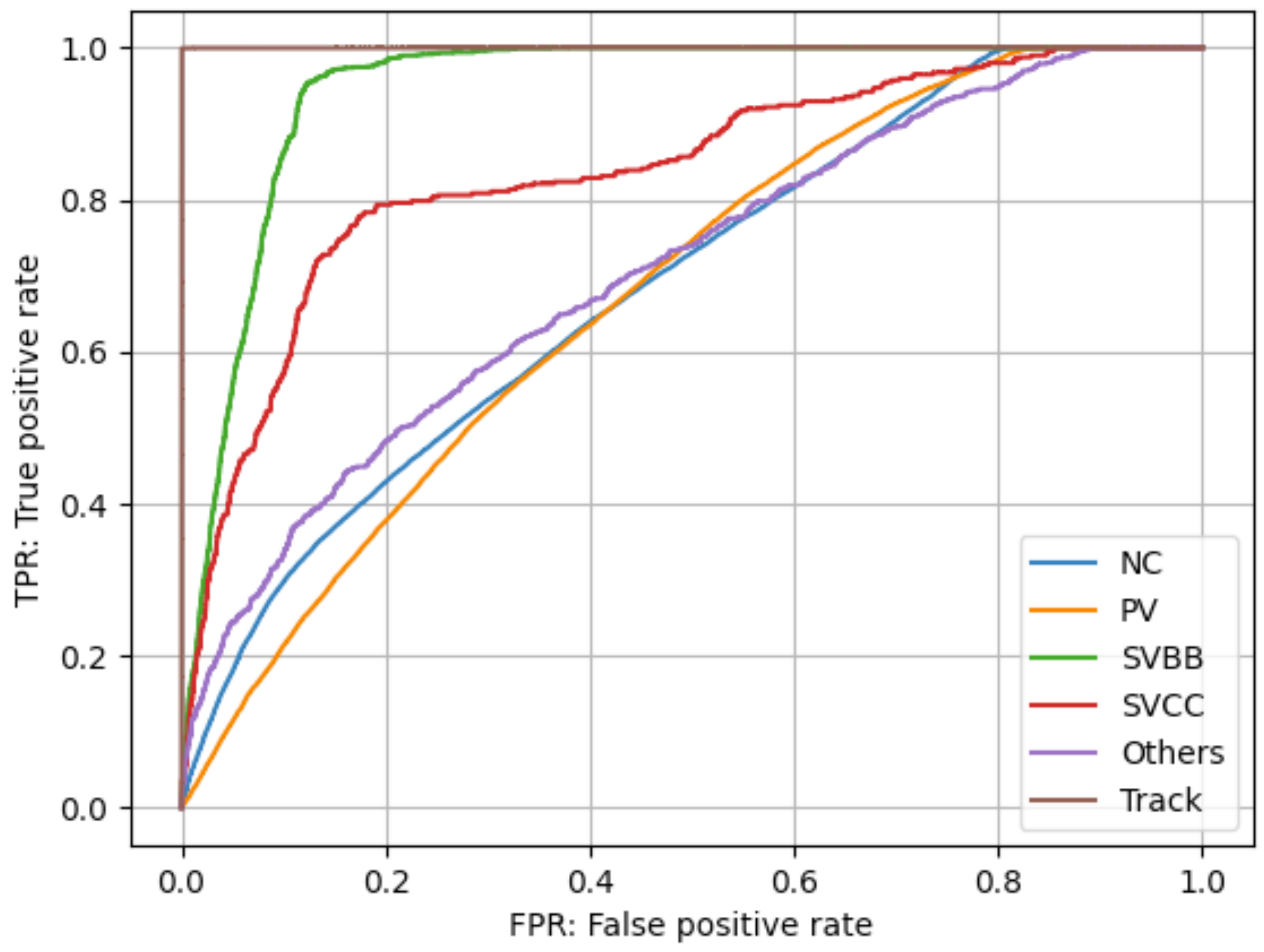
Loss function



Confusion matrix



ROC curve



- Now we cannot classify VCs with sufficient accuracy.
- Over-training (Divergence between validation and training) is also a problem.
- These are expected to be improved by adjusting the network structure, training parameters, and also processing input data.

Summary

Summary

- In the process of constructing the network for flavor tagging.
- The accuracy of DNN was about 82%.

- Flavor Tagging by GNN is expected to improve the performance.
- Spatial Convolution Graph Neural Network is better than the spectral approach.
- Optimizing learning parameter can improve the performance of GCN

Back up