使用HepAI平台训练 ParticleNet
&
Particle Transforme

朱永峰

高能所计算中心

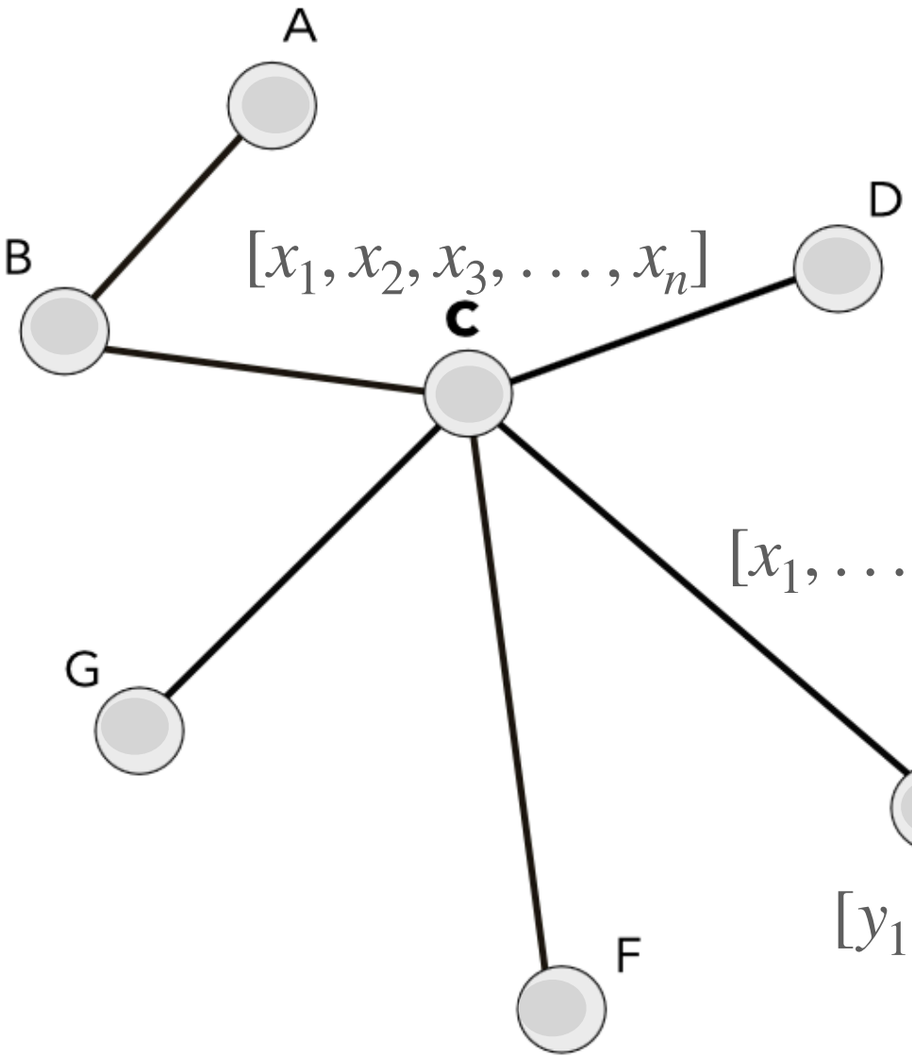
2025.07.18

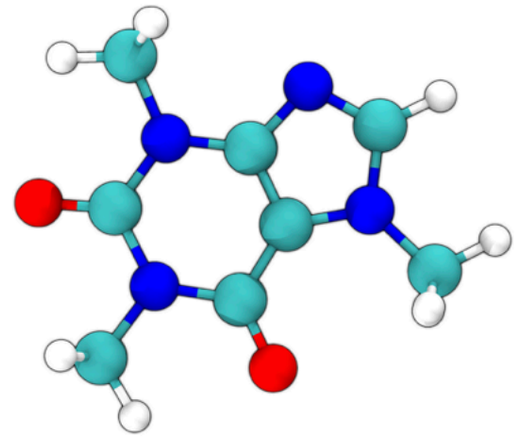
内容:

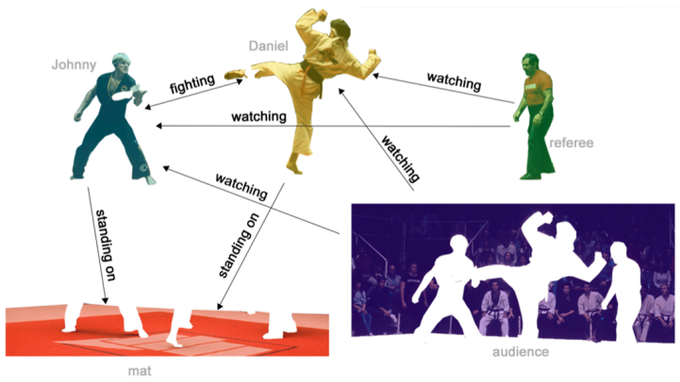
- ParticleNet
- Particle Transformer
- 喷注味道鉴别

代码路径：/aifs/user/home/zhuyf/PNParT

The idea of Graph Neural Network: update the features of nodes/edges based on the features of other nodes or edges.

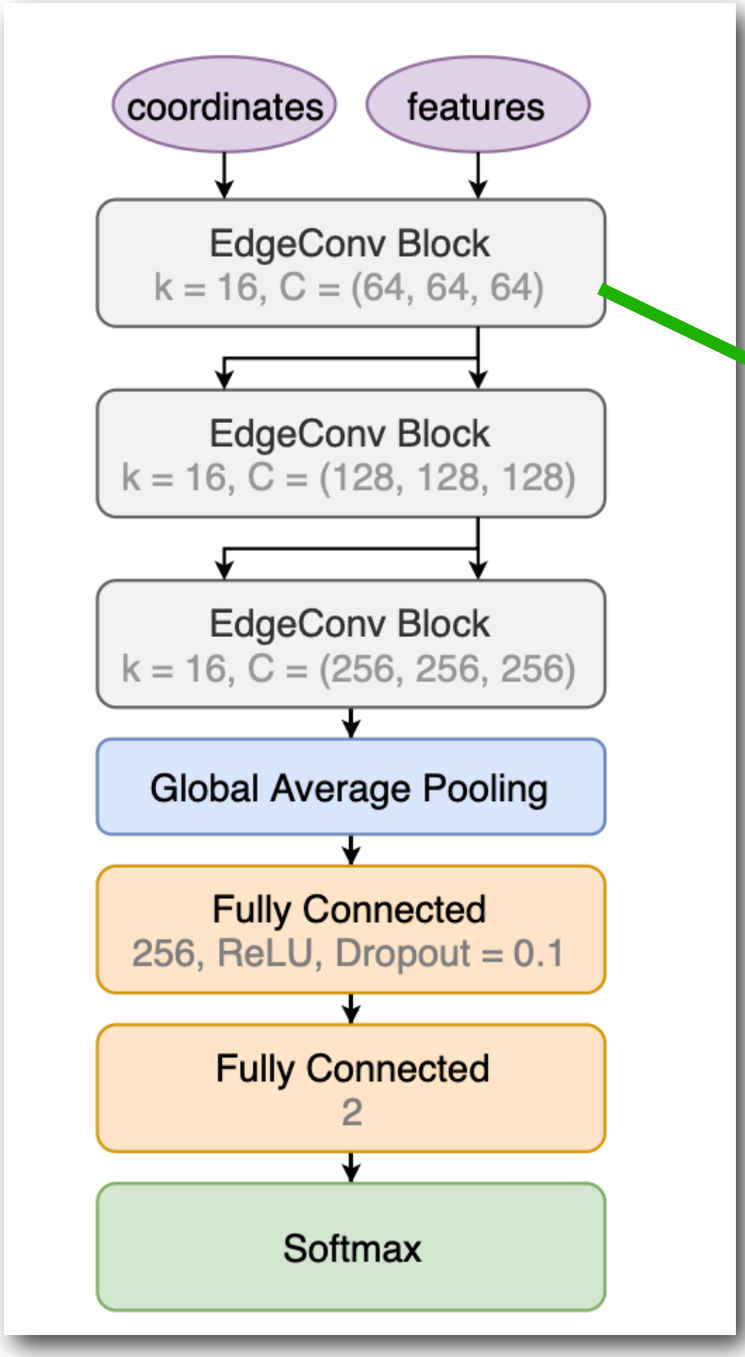


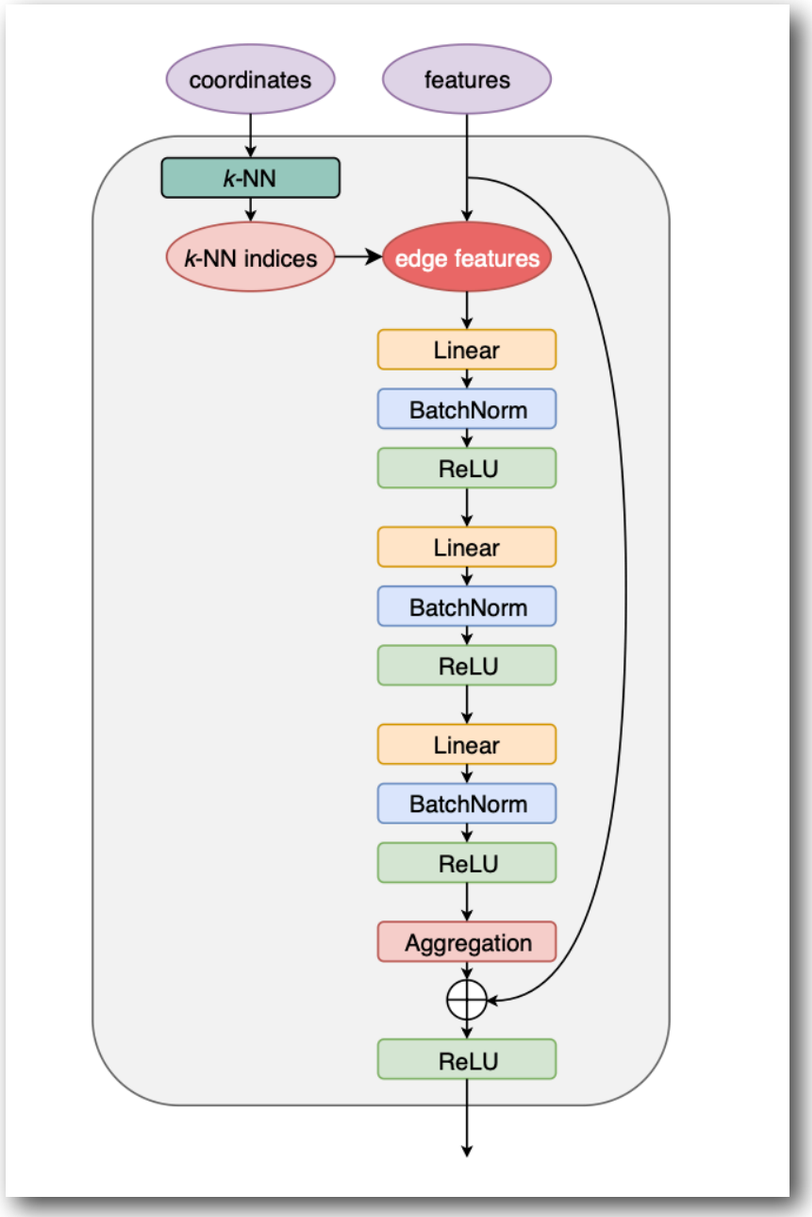




DOI: 10.23915/distill.00032

the ParticleNet architecture





[Phys. Rev. D 101, 056019]

提交作业脚本参数说明

```
python -u main.py \
  --batch_size 512 \
  --learning_rate 0.001 \
  --epochs 3 \
  --momentum 0.9 \
  --weight_decay 5.e-4 \
  --arch 'ParT' \
  --num_heads=4 \
  --num_layers=4 \
  --EdgeConv_config='[[16, [64, 64, 64]], [16, [128, 128, 128]], [16, [256, 256, 256]]]' \
  --fcs='[[256, 0.1]]' \
  --data_root "/aifs/user/home/zhuyf/LHAASO/dataset/L100/finetune/L100_FT_3" \
  --exp_dir "/aifs/user/home/zhuyf/Mine/save/test" \
  --mode "train" \
  --load_model "/aifs/user/home/zhuyf/Mine/save/test/best.pth" \
  --predict_output "/aifs/user/home/zhuyf/Mine/predict_output" \
  --Processing_conditions='[
      {"name": "vx",          "subtract": 0,    "multiply": 1,    "min": -10000, "max": 10000},
      {"name": "vy",          "subtract": 0,    "multiply": 1,    "min": -10000, "max": 10000},
      {"name": "loge",        "subtract": 0,    "multiply": 3.5,  "min": -5.0,   "max": 5.0},
      {"name": "logE",        "subtract": 1.5,  "multiply": 2.5,  "min": -5.0,   "max": 5.0},
      {"name": "deltaR",      "subtract": 0,    "multiply": 0.02, "min": -5.0,   "max": 5.0},
      {"name": "polar_angle", "subtract": 3.14, "multiply": 1.55, "min": -5.0,   "max": 5.0}
  ]' \
  --labels='["label_s","label_b"]' \
  --Observables='["vx", "vy", "vq", "xc", "yc", "energy", "label_b", "label_s"]' \
  --fixed_length 100 \
  --labelSmoothing 0. \
  --flip_rate 0. \
  --gpu 0
```

SGD 优化网络参数的方式

网络结构，可选ParT或者PN

控制ParT的网络结构

控制PN的网络结构

输入数据路径

训练模型存储路径

运行模式，train代表训练+推理，’infer’代表只进行推理

推理时用到的模型

模型推理结果输出路径

模型用到的特征变量，其中前两个是PN用来计算节点之间距离的变量

分类类别

观测量，原样输出

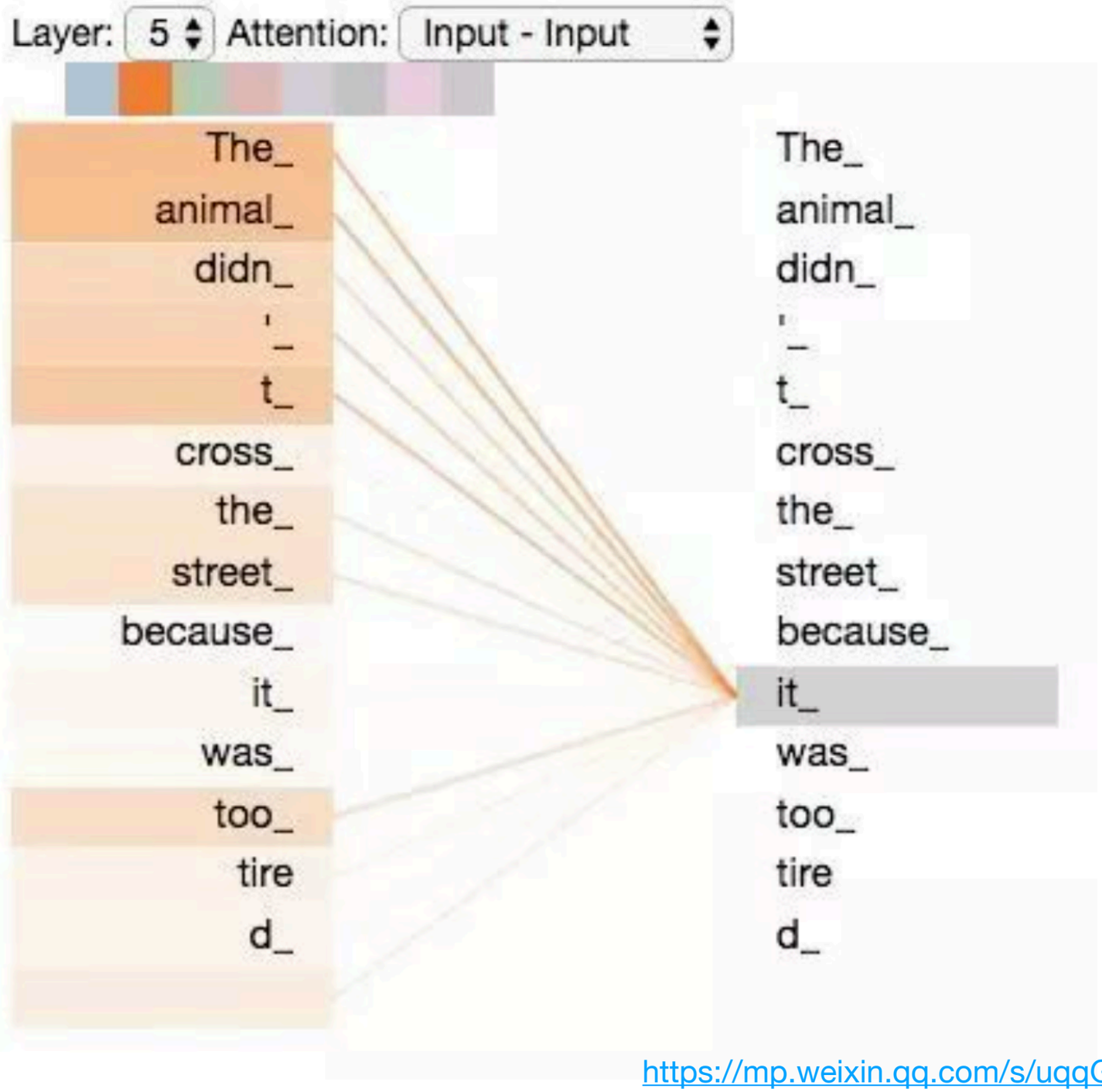
补齐/截断长度

目前用不到

使用申请的第0个gpu

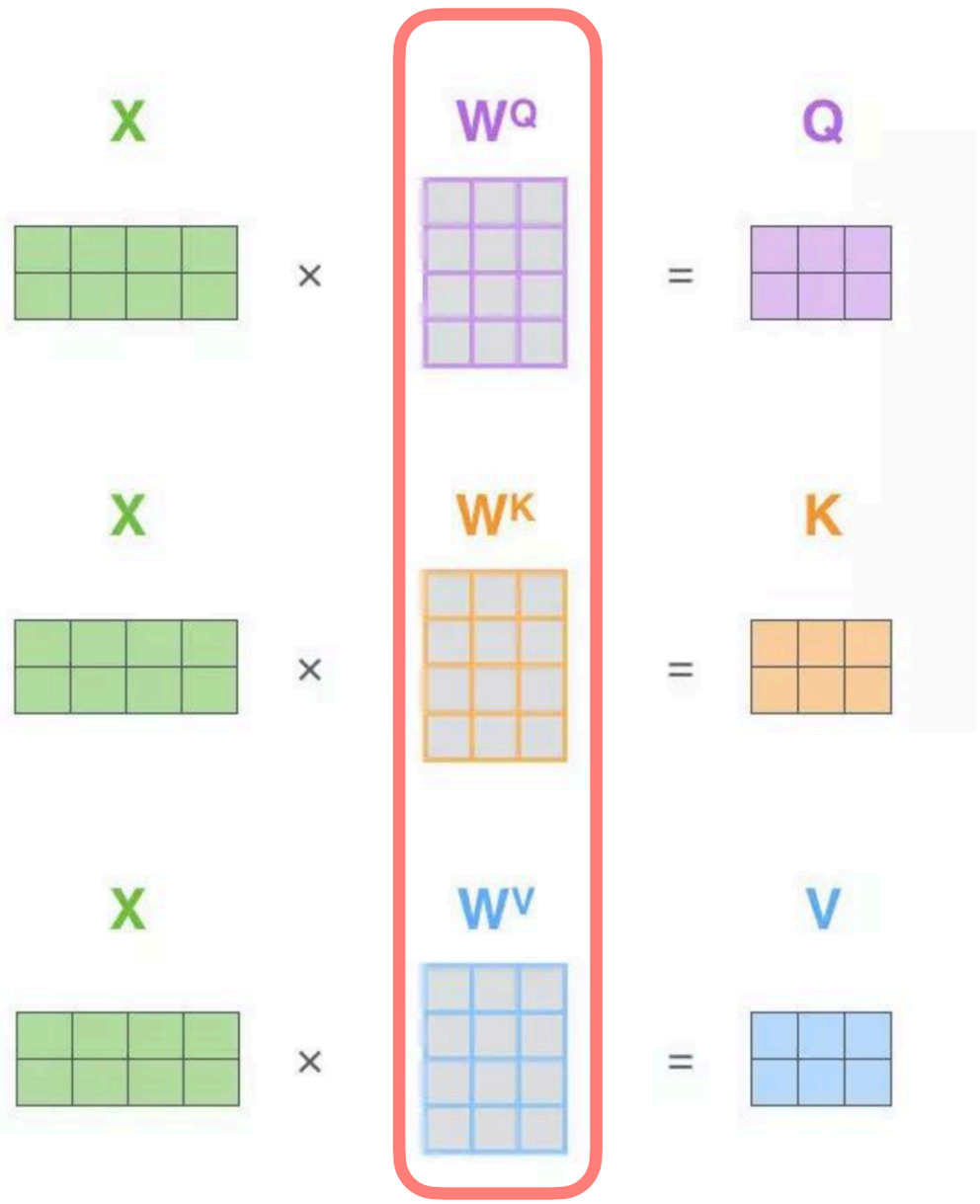
LHAASO-WCDA上的伽马和质子鉴别任务

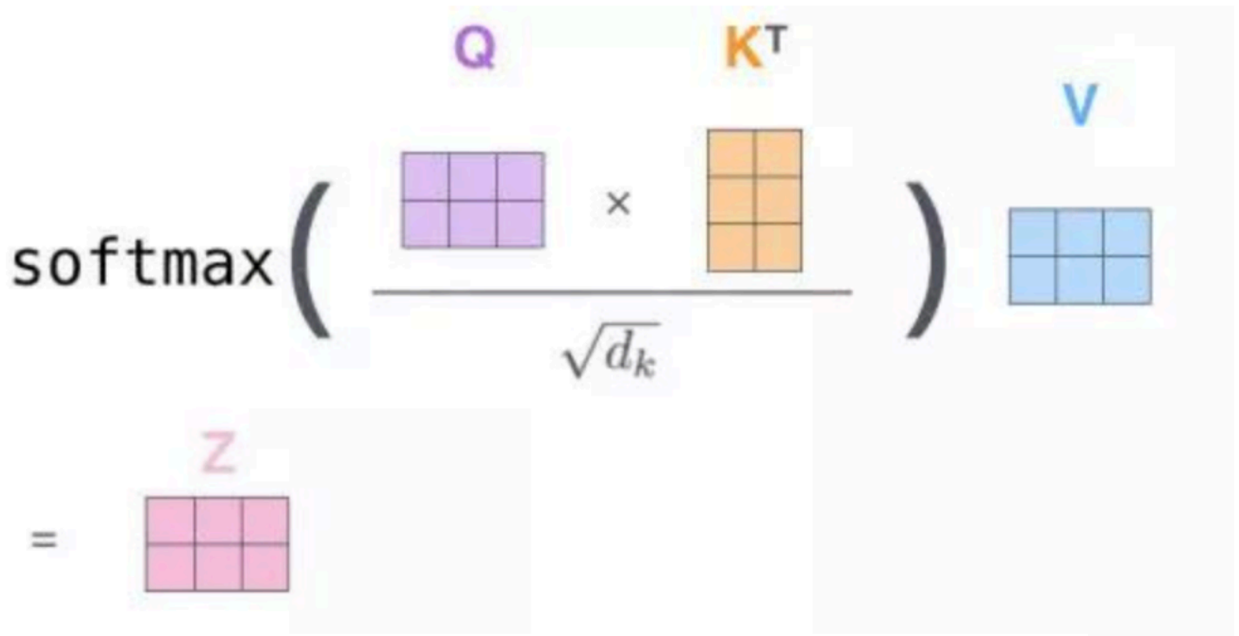
self-attention



https://mp.weixin.qq.com/s/uqqGI4GPXcSIrjE0jkh8vw

self-attention

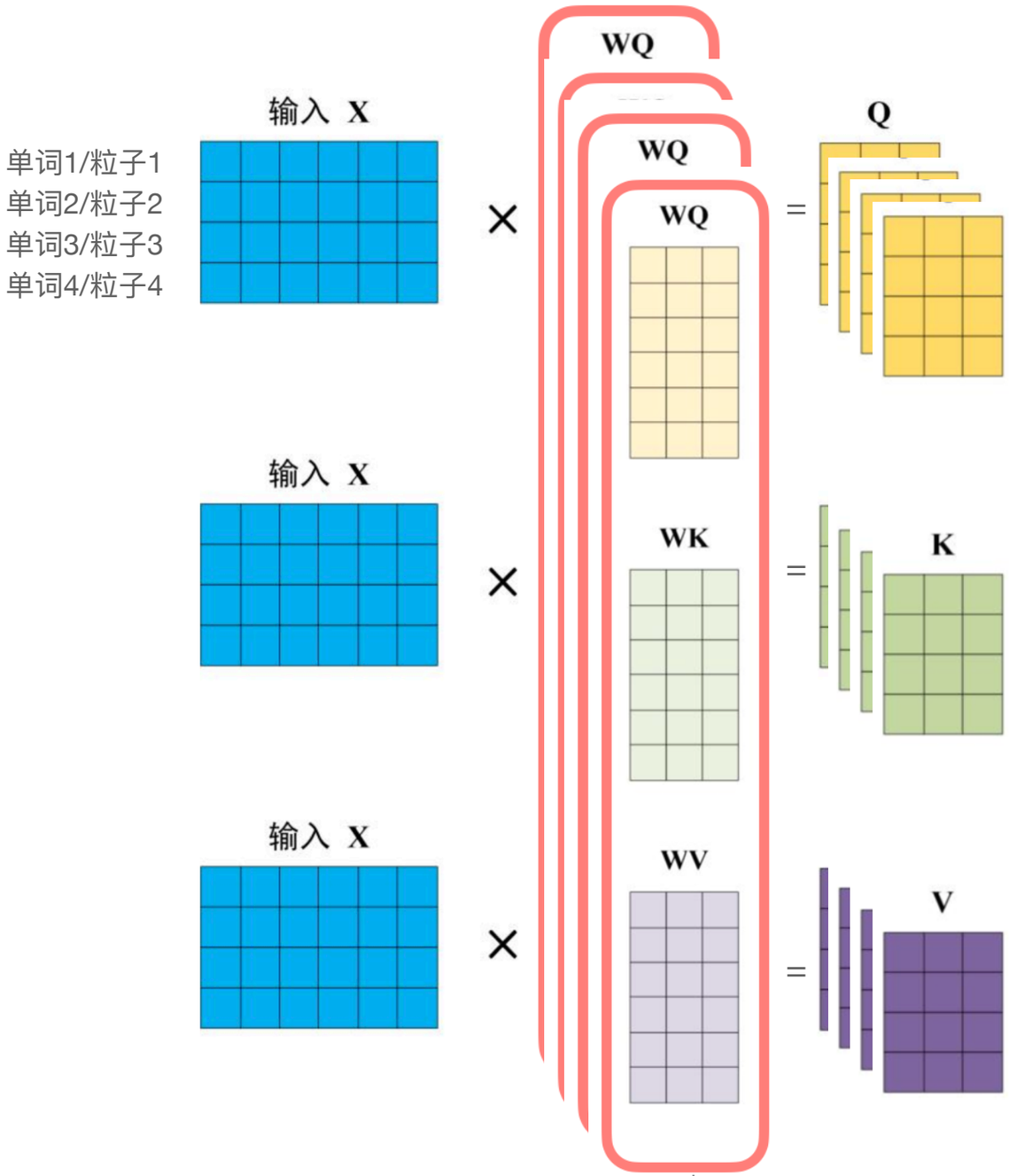


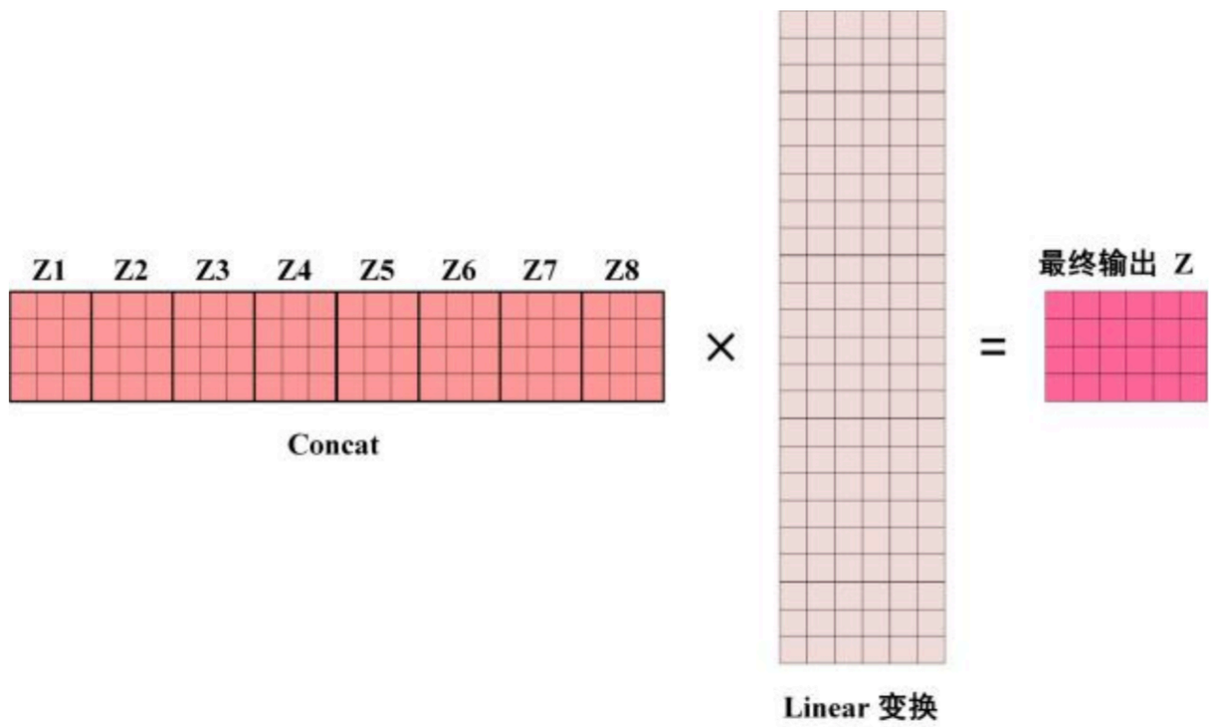


$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

https://mp.weixin.qq.com/s/uqqGI4GPXcSIrjE0jkh8vw

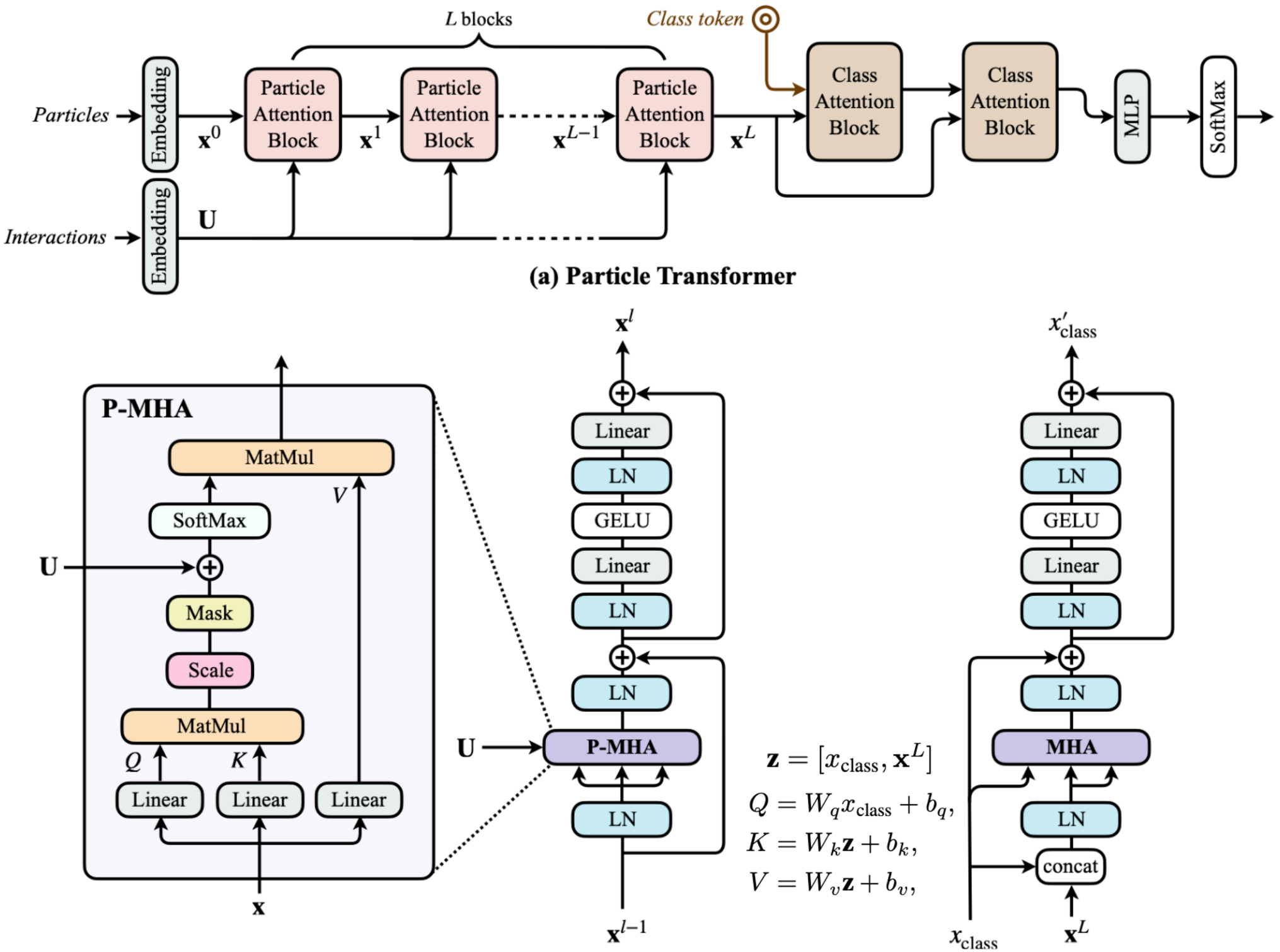
multi-head self-attention





$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_\text{h})W^O$$
$$\text{where head}_\text{i} = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

Particle Transformer



https://arxiv.org/abs/2202.03772

提交作业脚本参数说明

```
python -u main.py \
  --batch_size 512 \
  --learning_rate 0.001 \
  --epochs 3 \
  --momentum 0.9 \
  --weight_decay 5.e-4 \
  --arch 'ParT' \
  --num_heads=4 \
  --num_layers=4 \
  --EdgeConv_config='[[16, [64, 64, 64]], [16, [128, 128, 128]], [16, [256, 256, 256]]]' \
  --fcs='[[256, 0.1]]' \
  --data_root "/aifs/user/home/zhuyf/LHAASO/dataset/L100/finetune/L100_FT_3" \
  --exp_dir "/aifs/user/home/zhuyf/Mine/save/test" \
  --mode "train" \
  --load_model "/aifs/user/home/zhuyf/Mine/save/test/best.pth" \
  --predict_output "/aifs/user/home/zhuyf/Mine/predict_output" \
  --Processing_conditions='[
      {"name": "vx",          "subtract": 0,    "multiply": 1,    "min": -10000, "max": 10000},
      {"name": "vy",          "subtract": 0,    "multiply": 1,    "min": -10000, "max": 10000},
      {"name": "loge",        "subtract": 0,    "multiply": 3.5,  "min": -5.0,   "max": 5.0},
      {"name": "logE",        "subtract": 1.5,  "multiply": 2.5,  "min": -5.0,   "max": 5.0},
      {"name": "deltaR",      "subtract": 0,    "multiply": 0.02, "min": -5.0,   "max": 5.0},
      {"name": "polar_angle", "subtract": 3.14, "multiply": 1.55, "min": -5.0,   "max": 5.0}
  ]' \
  --labels='["label_s","label_b"]' \
  --Observables='["vx", "vy", "vq", "xc", "yc", "energy", "label_b", "label_s"]' \
  --fixed_length 100 \
  --labelSmoothing 0. \
  --flip_rate 0. \
  --gpu 0
```

SGD 优化网络参数的方式

网络结构，可选ParT或者PN

控制ParT的网络结构

控制PN的网络结构

输入数据路径

训练模型存储路径

运行模式，train代表训练+推理，’infer’代表只进行推理

推理时用到的模型

模型推理结果输出路径

模型用到的特征变量，其中前两个是PN用来计算节点之间距离的变量

分类类别

观测量，原样输出

补齐/截断长度

目前用不到

使用申请的第0个gpu

LHAASO-WCDA上的伽马和质子鉴别任务

Weaver 安装使用方式

https://github.com/jet-universe/particle_transformer

ParticleNet原文

Phys. Rev. D 101, 056019

Particle Transformer原文

https://arxiv.org/abs/2202.03772

HepAI平台训练ParticleNet 和 Particle Transformer

申请计算资源

```
#=================== part 2 : workload ======================
echo "slurm job ${SLURM_JOB_ID} start.."
date

echo "------------------"


source /aifs/user/home/zhuyf/env_conda.sh

conda activate weaver2

echo "args: $@"
```

激活深度学习环境

提交作业脚本参数说明

```
python -u main.py \
  --batch_size 512 \
  --learning_rate 0.001 \
  --epochs 3 \
  --momentum 0.9 \
  --weight_decay 5.e-4 \
  --arch 'ParT' \
  --num_heads=4 \
  --num_layers=4 \
  --EdgeConv_config='[[16, [64, 64, 64]], [16, [128, 128, 128]], [16, [256, 256, 256]]]' \
  --fcs='[[256, 0.1]]' \
  --data_root "/aifs/user/home/zhuyf/LHAASO/dataset/L100/finetune/L100_FT_3" \
  --exp_dir "/aifs/user/home/zhuyf/Mine/save/test" \
  --mode "train" \
  --load_model "/aifs/user/home/zhuyf/Mine/save/test/best.pth" \
  --predict_output "/aifs/user/home/zhuyf/Mine/predict_output" \
  --Processing_conditions='[
      {"name": "vx",          "subtract": 0,    "multiply": 1,    "min": -10000, "max": 10000},
      {"name": "vy",          "subtract": 0,    "multiply": 1,    "min": -10000, "max": 10000},
      {"name": "loge",        "subtract": 0,    "multiply": 3.5,  "min": -5.0,   "max": 5.0},
      {"name": "logeT",       "subtract": -3.5, "multiply": 2.5,  "min": -5.0,   "max": 5.0},
      {"name": "deltaR",      "subtract": 0,    "multiply": 0.02, "min": -5.0,   "max": 5.0},
      {"name": "polar_angle", "subtract": 3.14, "multiply": 1.55, "min": -5.0,   "max": 5.0}
  ]' \
  --labels='["label_s","label_b"]' \
  --Observables='["vx", "vy", "vq", "xc", "yc", "energy", "label_b", "label_s"]' \
  --fixed_length 100 \
  --labelSmoothing 0. \
  --flip_rate 0. \
  --gpu 0
```

SGD 优化网络参数的方式

网络结构，可选ParT或者PN

控制ParT的网络结构

控制PN的网络结构

输入数据路径

训练模型存储路径

运行模式，train代表训练+推理，’infer’代表只进行推理

推理时用到的模型

模型推理结果输出路径

模型用到的特征变量，其中前两个是PN用来计算节点之间距离的变量

分类类别

观测量，原样输出

补齐/截断长度

目前用不到

使用申请的第0个gpu

正负电子对撞产生两个喷注，每个喷注中包含很多粒子

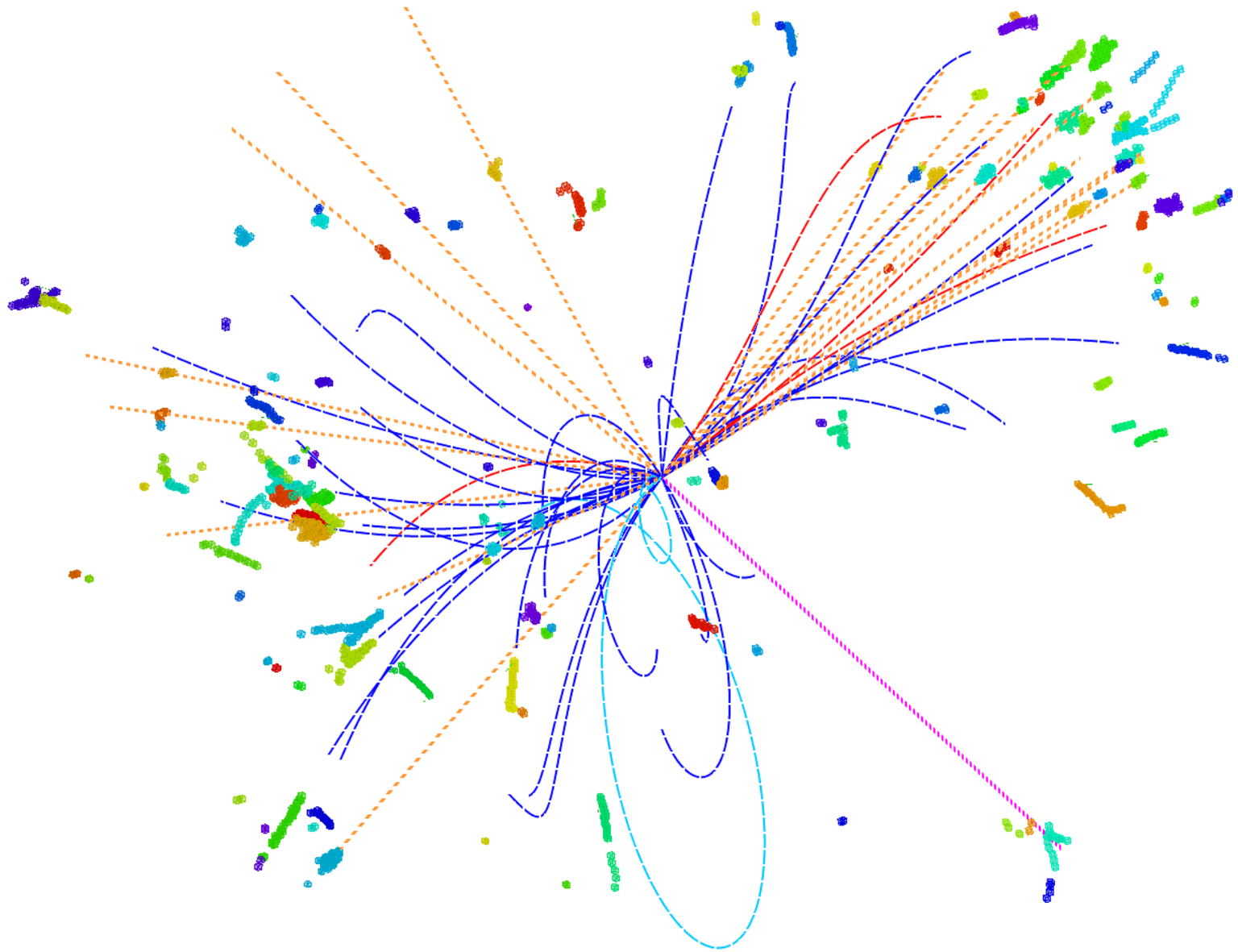
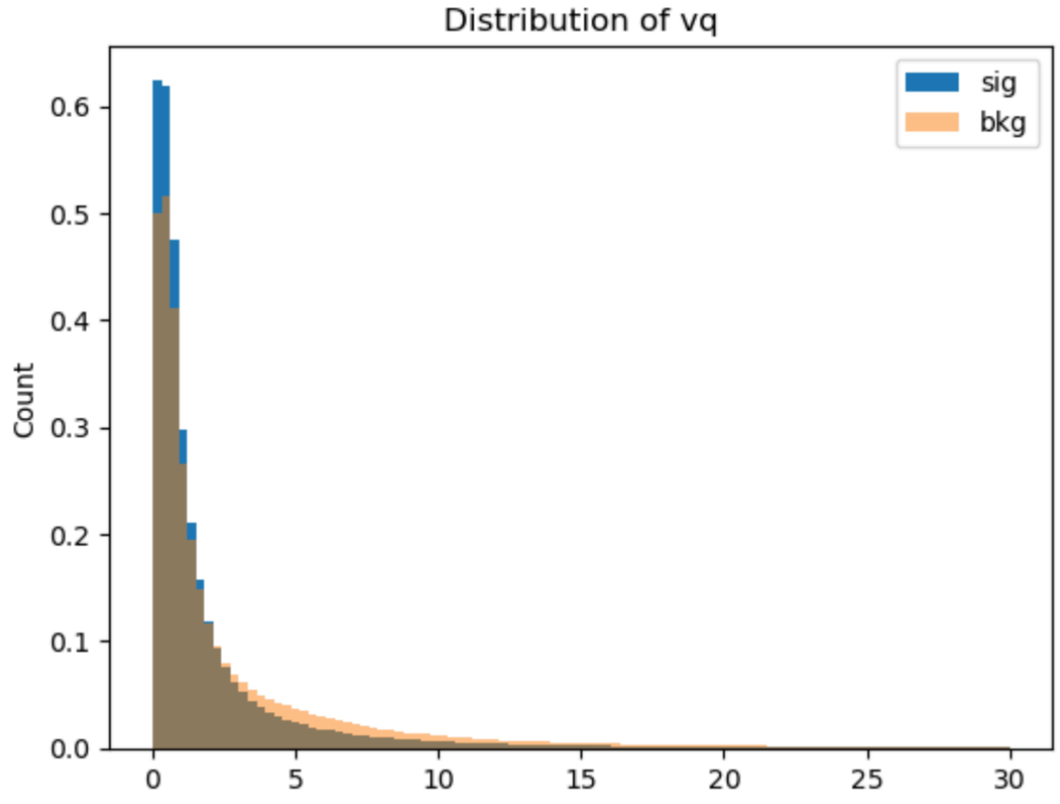
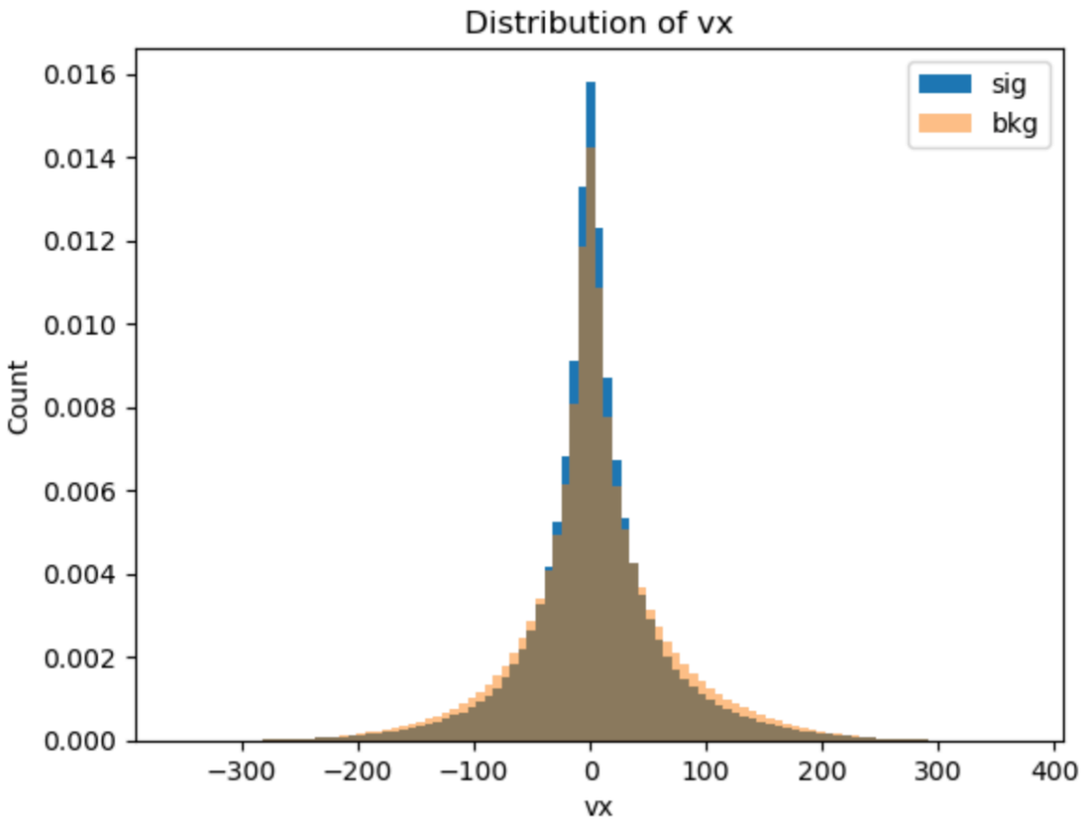


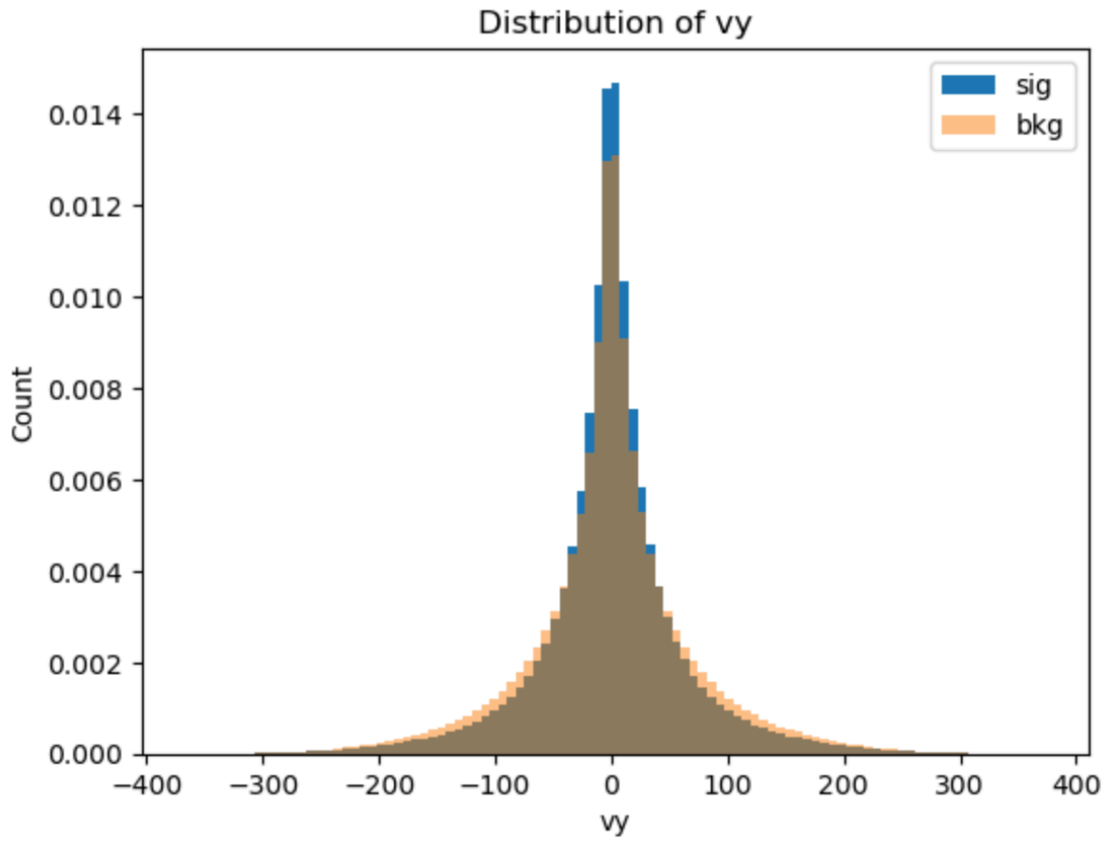
Table 3 The input variables used in ParticleNet for jet flavor tagging at the CEPC.

Variable	Definition
$\Delta\eta$	difference in pseudorapidity between the particle and the jet axis
$\Delta\phi$	difference in azimuthal angle between the particle and the jet axis
$\log \mathrm{P_t}$	logarithm of the particle's P_t
logE	logarithm of the particle's energy
$\log \frac{\mathrm{P_t}}{\mathrm{P_t(jet)}}$	logarithm of the particle's P_t relative to the jet P_t
$\log \frac{\mathrm{E}}{\mathrm{E(jet)}}$	logarithm of the particle's energy relative to the jet energy
ΔR	angular separation between the particle and the jet axis
d_0	transverse impact parameter of the track
$d_0$err	uncertainty associated with the measurement of the d_0
z_0	longitudinal impact parameter of the track
$z_0$err	uncertainty associated with the measurement of the z_0
charge	electric charge of the particle
isElectron	whether the particle is an electron
isMuon	whether the particle is a muon
isChargedKaon	whether the particle is a charged Kaon
isChargedPion	whether the particle is a charged Pion
isProton	whether the particle is a proton
isNeutralHadron	whether the particle is a neutral hadron
isPhoton	whether the particle is a photon

信号和本底的特征分布


Distribution of vq
sig
bkg
Count
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0
5
10
15
20
25
30


Distribution of vx
sig
bkg
Count
0.000
0.002
0.004
0.006
0.008
0.010
0.012
0.014
0.016
−300
−200
−100
0
100
200
300
400
vx


Distribution of vy
sig
bkg
Count
0.000
0.002
0.004
0.006
0.008
0.010
0.012
0.014
−400
−300
−200
−100
0
100
200
300
400
vy

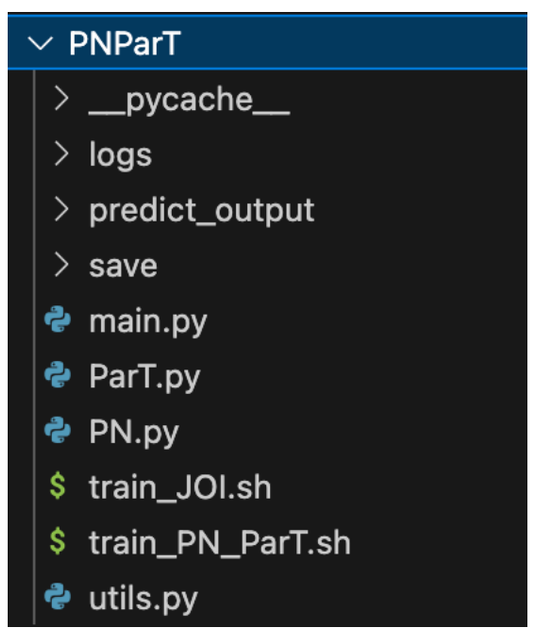
```
--Processing_conditions='[
    {"name": "part_deta",           "subtract": 0,      "multiply": 1.0,  "min": -1000.0, "max": 1000.0},
    {"name": "part_dphi",           "subtract": 0,      "multiply": 1.0,  "min": -1000.0, "max": 1000.0},
    {"name": "part_pt_log",         "subtract": -1.5,   "multiply": 1.0,  "min": -5.0,  "max": 5.0},
    {"name": "part_e_log",          "subtract": -0.687, "multiply": 1.0,  "min": -5.0, "max": 5.0},
    {"name": "part_logptrel",       "subtract": -4.7,   "multiply": 1.0,  "min": -5.0, "max": 5.0},
    {"name": "part_logerel",        "subtract": -4.473, "multiply": 1.0,  "min": -5.0, "max": 5.0},
    {"name": "part_deltaR",         "subtract": 2.1,    "multiply": 2.3,  "min": -5.0, "max": 5.0},
    {"name": "part_charge",         "subtract": 0,      "multiply": 1.0,  "min": -5.0, "max": 5.0},
    {"name": "part_isChargedKaon",  "subtract": 0,      "multiply": 1.0,  "min": -5.0, "max": 5.0},
    {"name": "part_isPion",         "subtract": 0,      "multiply": 1.0,  "min": -5.0, "max": 5.0},
    {"name": "part_isProton",       "subtract": 0,      "multiply": 1.0,  "min": -5.0, "max": 5.0},
    {"name": "part_isElectron",     "subtract": 0,      "multiply": 1.0,  "min": -5.0, "max": 5.0},
    {"name": "part_isMuon",         "subtract": 0,      "multiply": 1.0,  "min": -5.0, "max": 5.0},
    {"name": "part_isNeutralHadron", "subtract": 0,     "multiply": 1.0,  "min": -5.0, "max": 5.0},
    {"name": "part_isPhoton",       "subtract": 0,      "multiply": 1.0,  "min": -5.0, "max": 5.0},
    {"name": "part_d0",             "subtract": 0,      "multiply": 1.0,  "min": -5.0, "max": 5.0},
    {"name": "part_dz",             "subtract": 0,      "multiply": 1.0,  "min": -5.0, "max": 5.0},
    {"name": "part_deta",           "subtract": 0,      "multiply": 1.0,  "min": -5.0, "max": 5.0},
    {"name": "part_dphi",           "subtract": 0,      "multiply": 1.0,  "min": -5.0, "max": 5.0}
]' \
--labels='["label_b","label_c","label_u","label_d","label_s","label_g","label_bbar","label_cbar","label_ubar","label_dbar","label_sbar"]' \
--Observables='["jet_eta","jet_energy","jet_nparticles","label_b","label_c","label_u","label_d","label_s","label_g","label_bbar","label_cbar",'
```

喷注味道鉴别任务

每一行代表一种特征变量，存储了喷注中所有粒子的相关信息，是vector类型。

代码路径：/aifs/user/home/zhuyf/PNParT，内容：

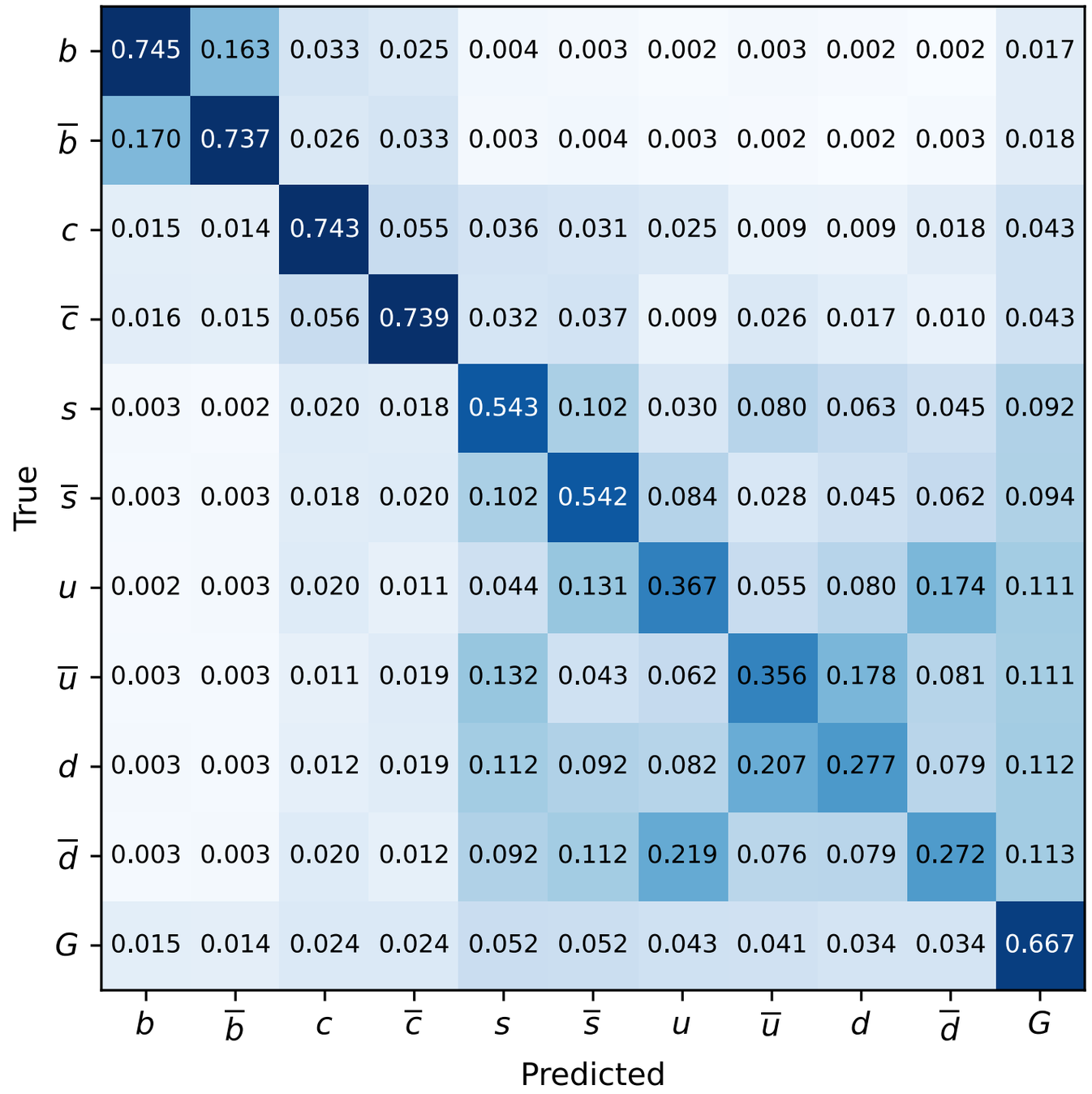
sbatch train_JOI.sh 提交作业



预测输出文件

```
(base) [zhuyf@ailogin001 PNParT]$ root -l predict_output/predictions.root
root [0]
Attaching file predict_output/predictions.root as _file0...
(TFile *) 0x560951975b30
root [1] .ls
TFile**         predict_output/predictions.root
 TFile*         predict_output/predictions.root
  KEY: TTree    tree;1
root [2] tree->Show(5)
======> EVENT:5
 score_label_b   = 0.427836
 score_label_c   = 0.000218556
 score_label_u   = 4.34324e-05
 score_label_d   = 4.27886e-05
 score_label_s   = 3.71376e-05
 score_label_g   = 0.0308735
score_label_bbar = 0.540575
score_label_cbar = 0.000220918
score_label_ubar = 3.9414e-05
score_label_dbar = 7.2794e-05
score_label_sbar = 4.13302e-05
jet_eta          = -0.318356
jet_energy       = 42.3029
jet_nparticles   = 22
label_b          = 0
label_c          = 0
label_u          = 0
label_d          = 0
label_s          = 0
label_g          = 0
label_bbar       = 1
label_cbar       = 0
label_ubar       = 0
label_dbar       = 0
label_sbar       = 0
```

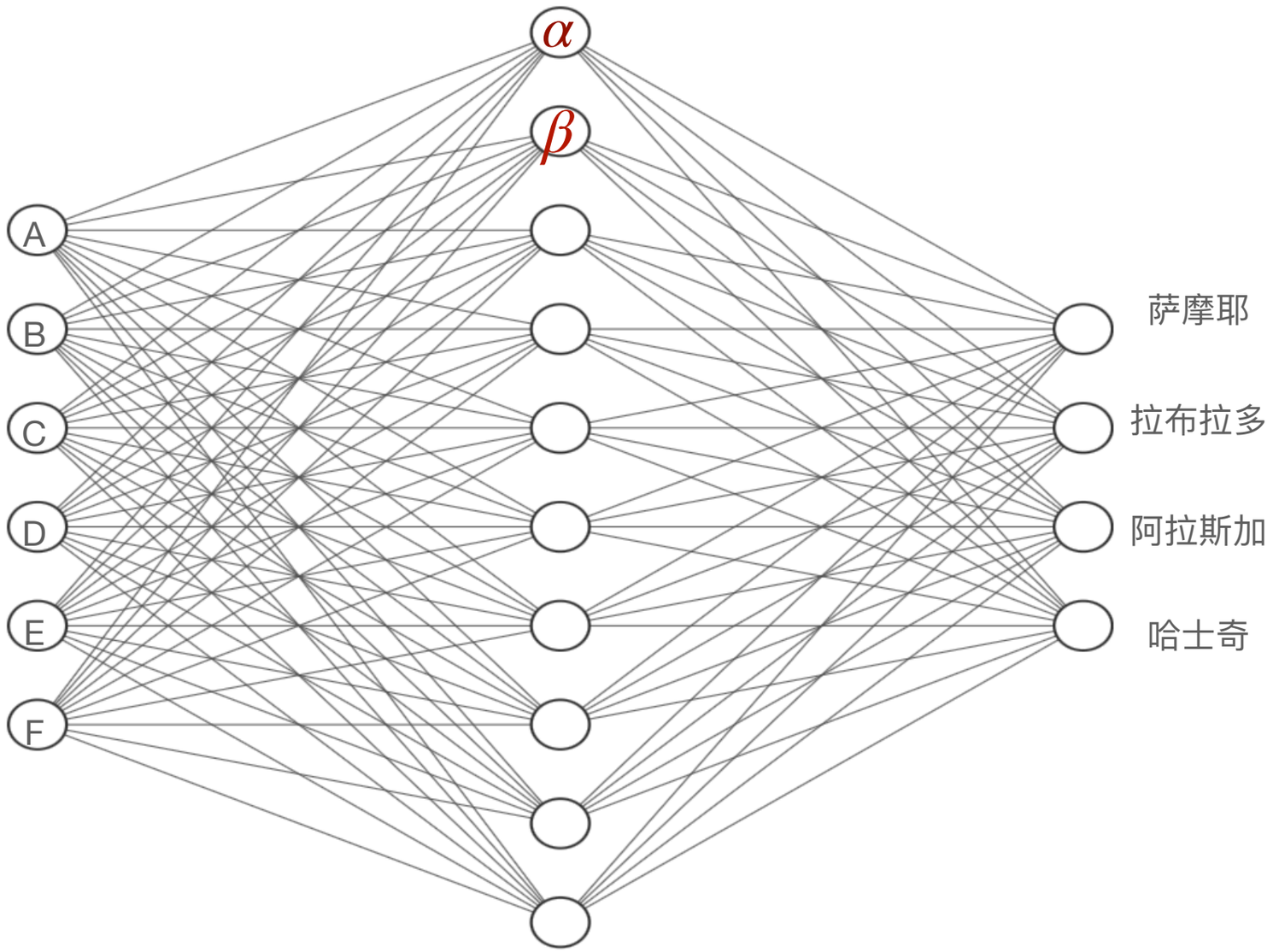
对输出文件进行分析得到喷注味道鉴别效果



谢谢

备注

Multi-Layer Perceptron (MLP)



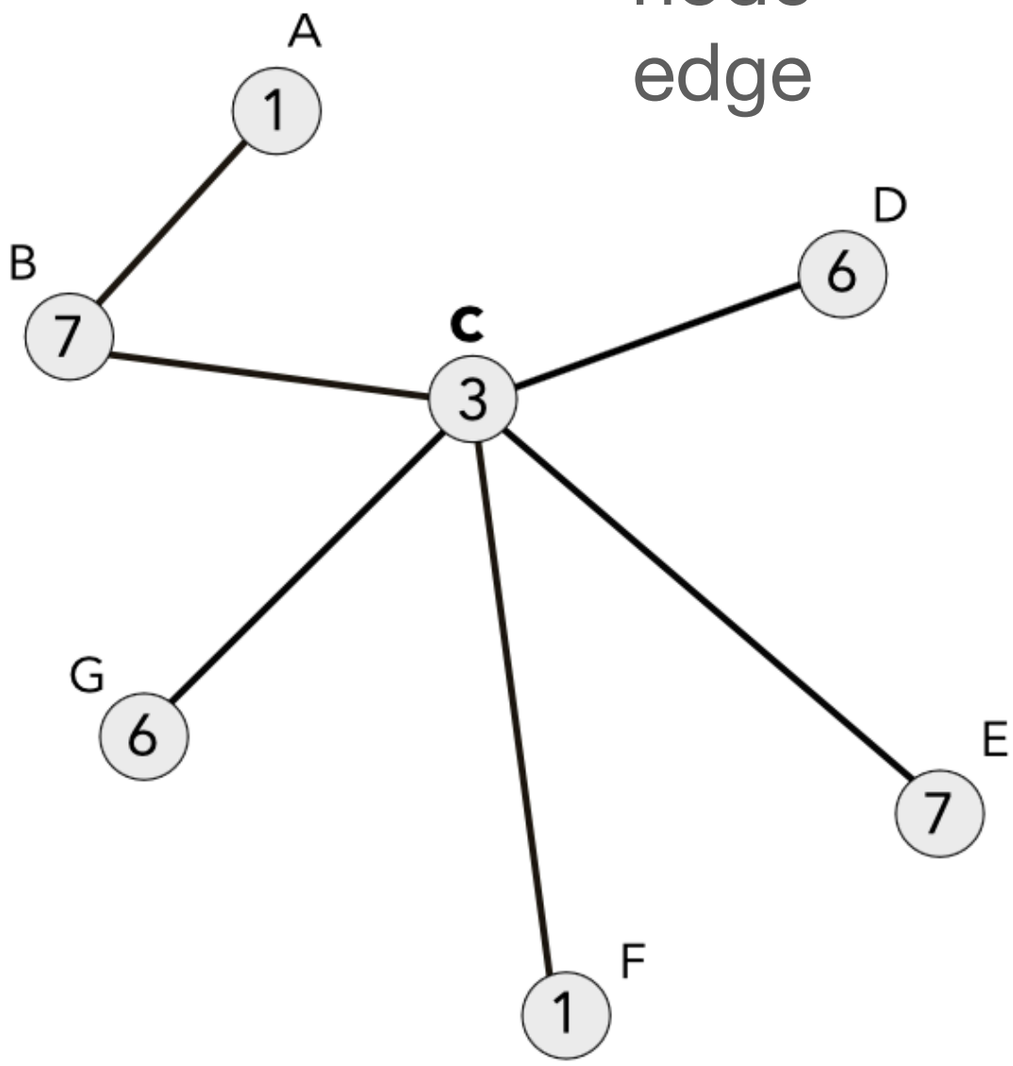
$$\alpha = \sigma(W_1 \cdot A + W_2 \cdot B + W_3 \cdot C + W_4 \cdot D + W_5 \cdot E + W_6 \cdot F)$$

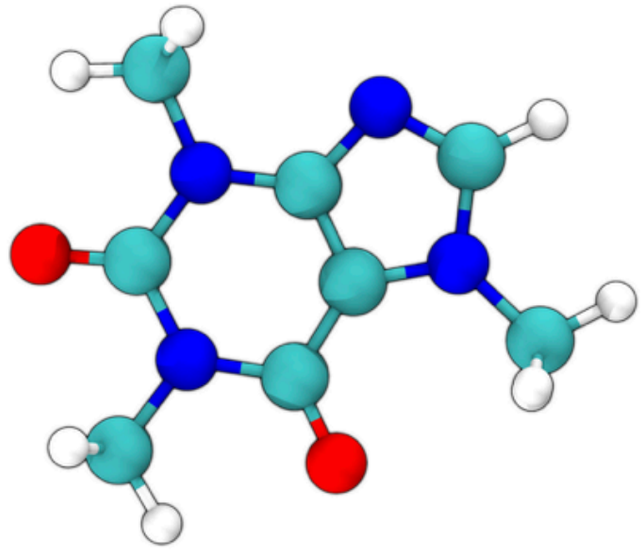
σ: activation function

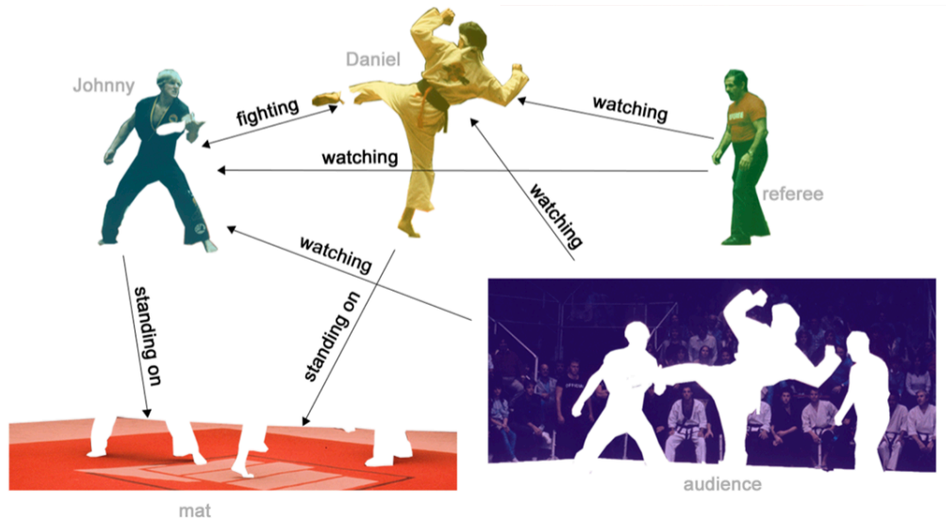
W: weight, learnable parameter

Graph

node
edge

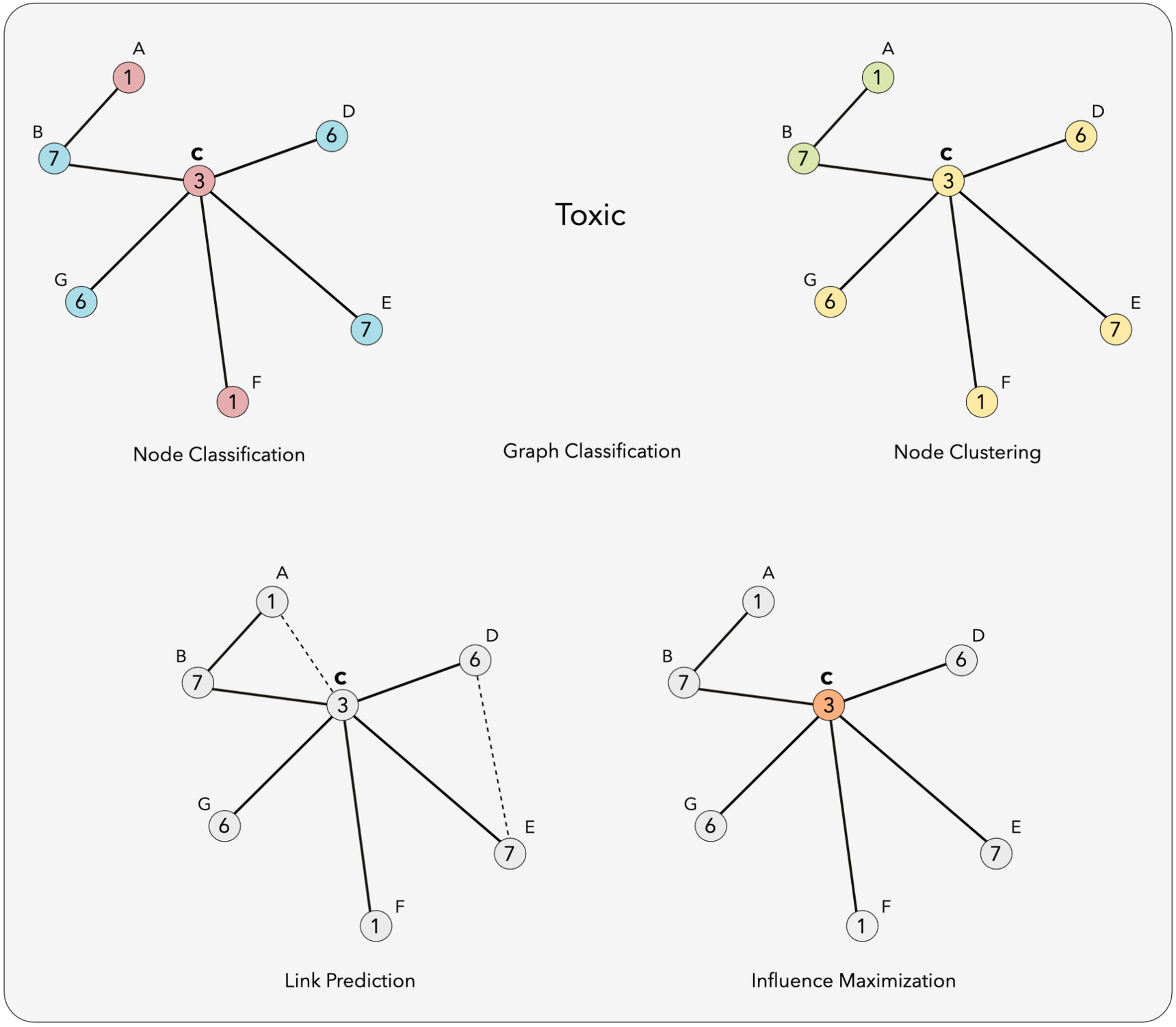






DOI: 10.23915/distill.00032

we can perform the following tasks:



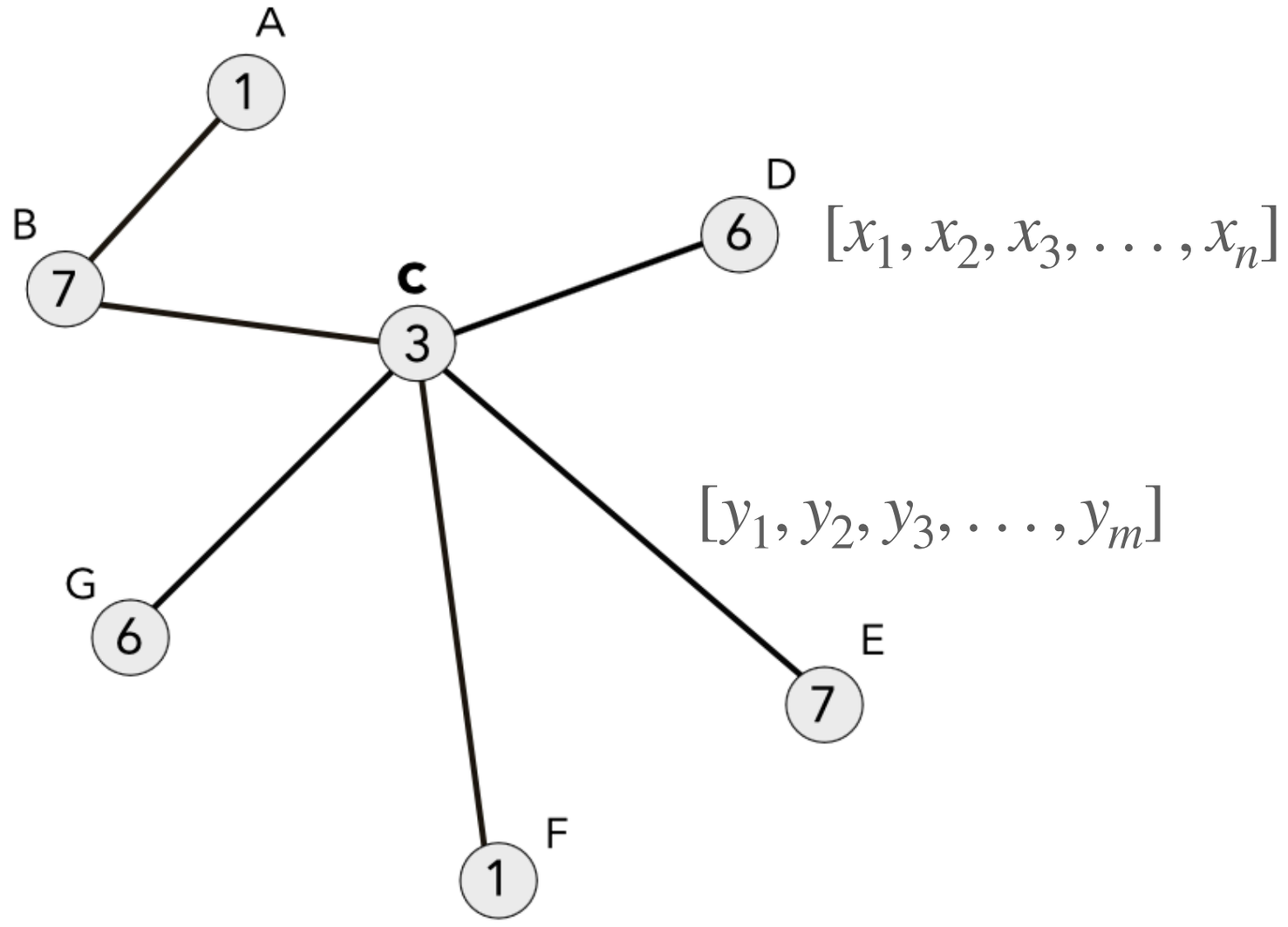
Examples of problems that can be defined over graphs. This list is not exhaustive!

DOI: 10.23915/distill.00032

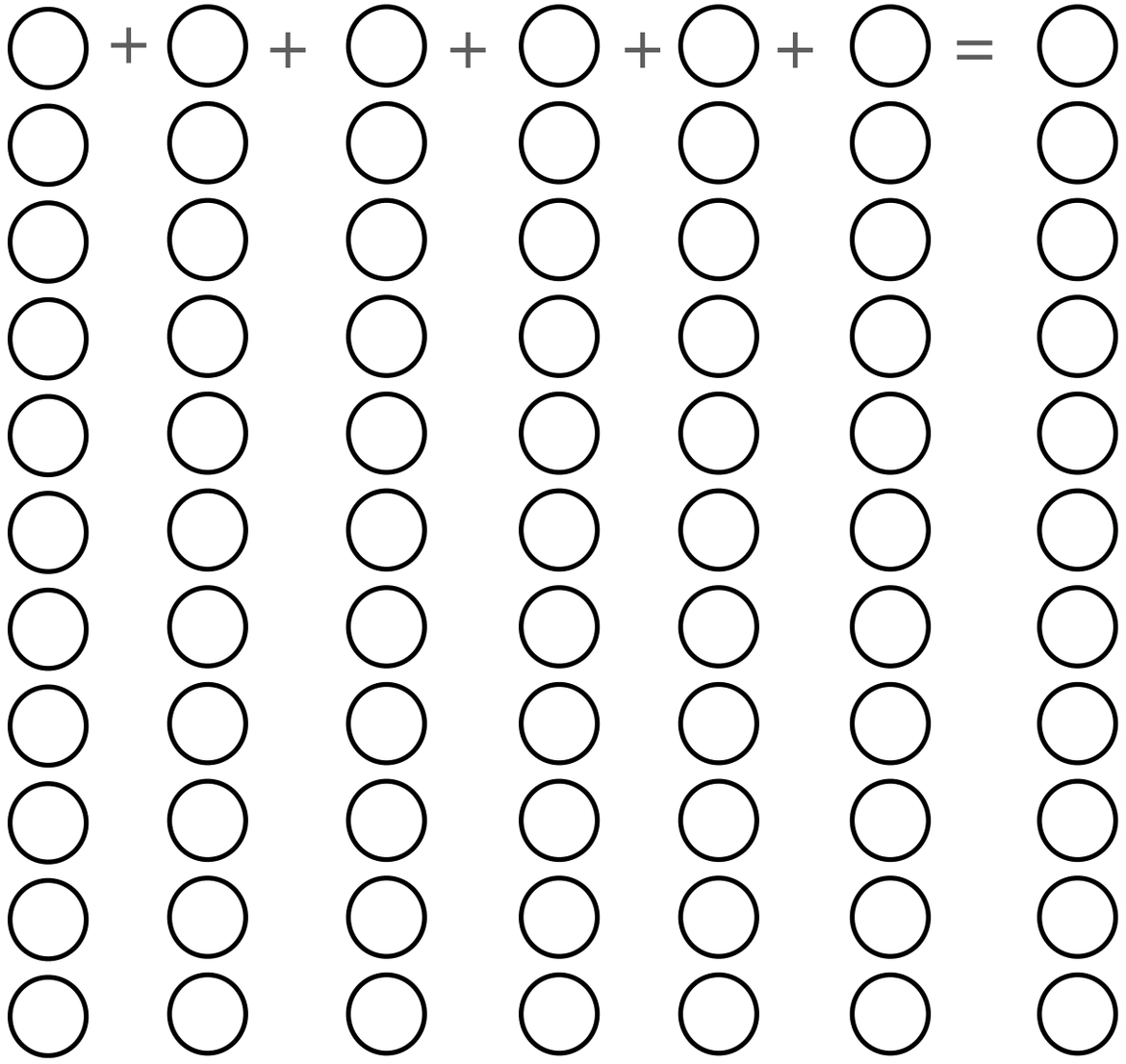
methods

1. If we apply an MLP on the node/ edge directly, the information would not flow among the nodes and edges.

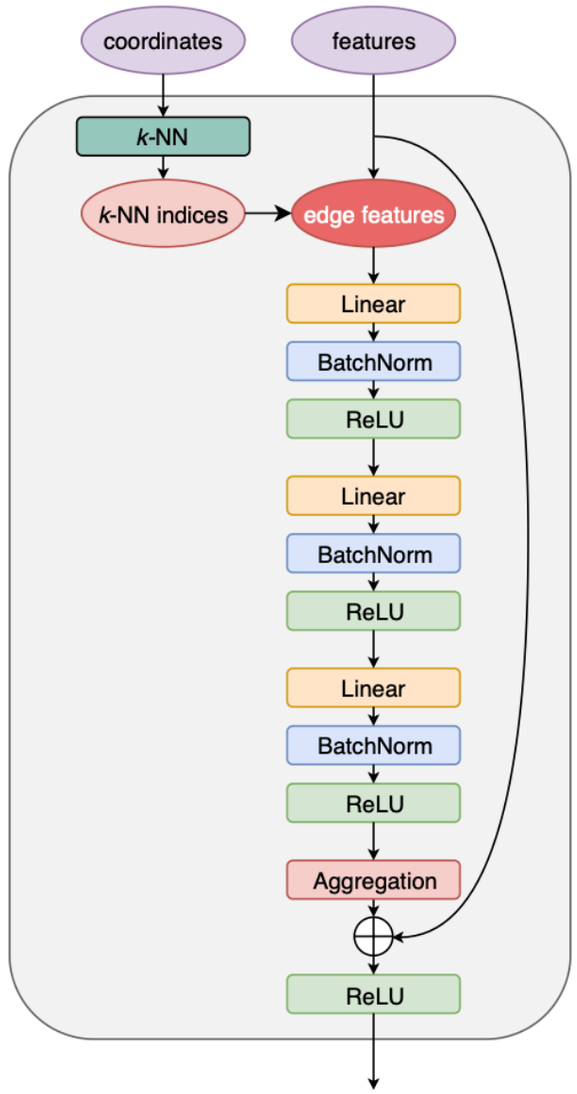
2， passing message: update the features of nodes/edges based on the features of other nodes or edges,



edge 1 edge 3

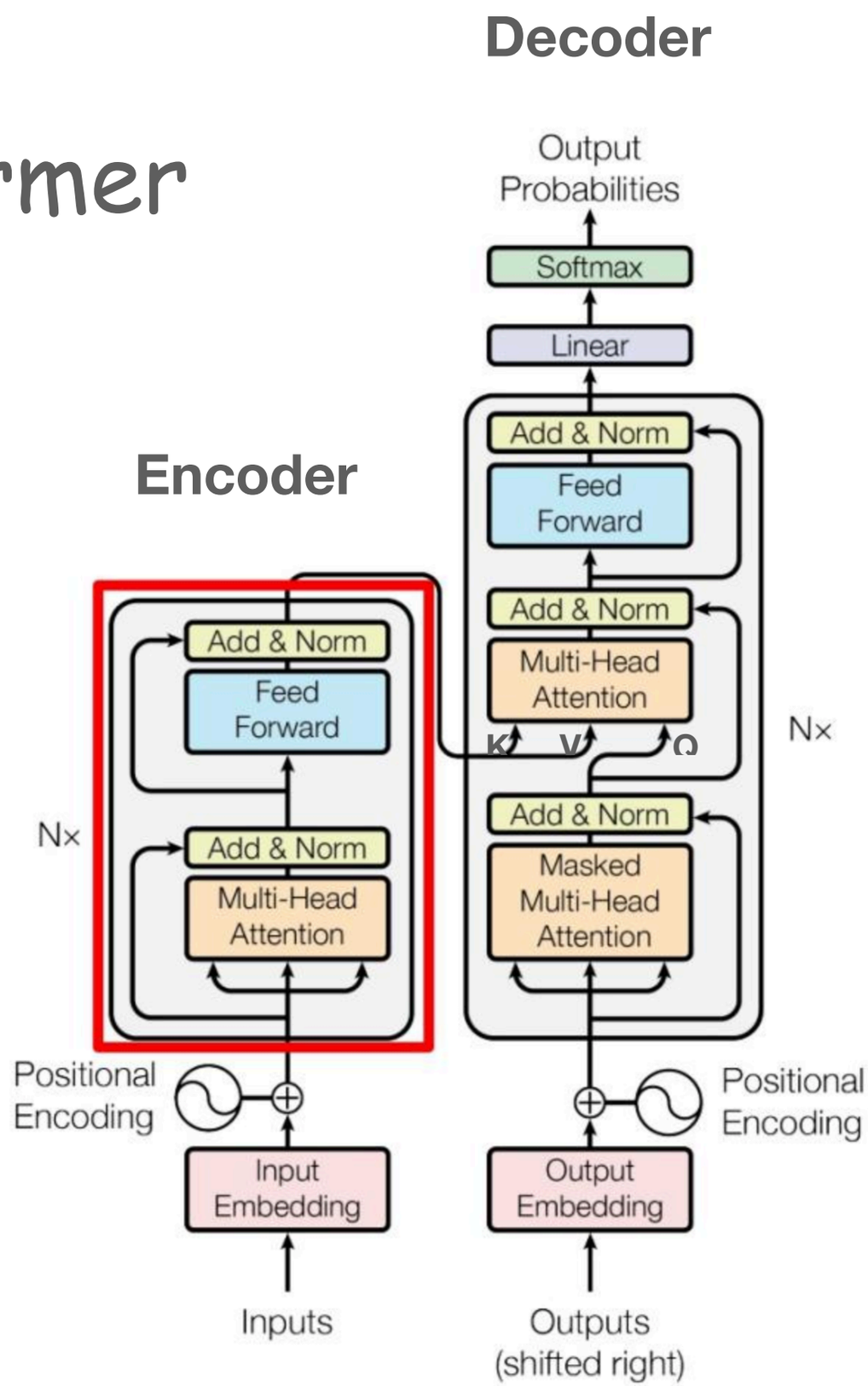


Jet composed of multiple particles



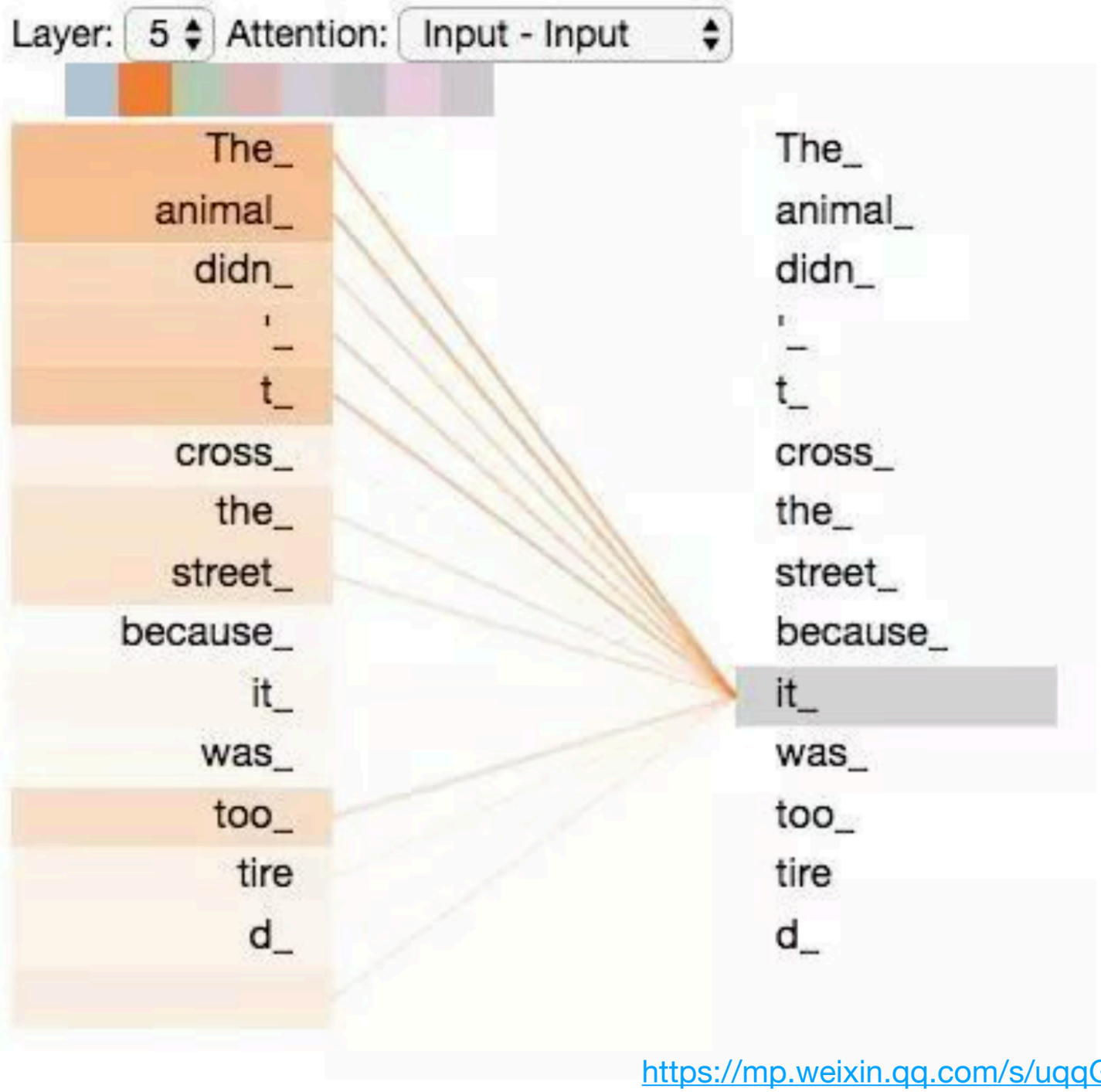
particle features : $[e, \mu, \pi^{\pm}, k^{\pm}, p^{\pm}, \gamma, charged\ hadron, neutral\ hadron, p_t, E, d_0, z_0]$

Transformer



arXiv:1706.03762

self-attention



https://mp.weixin.qq.com/s/uqqGI4GPXcSIrjE0jkh8vw

self-attention

输入

词嵌入

查询向量

键向量

值向量

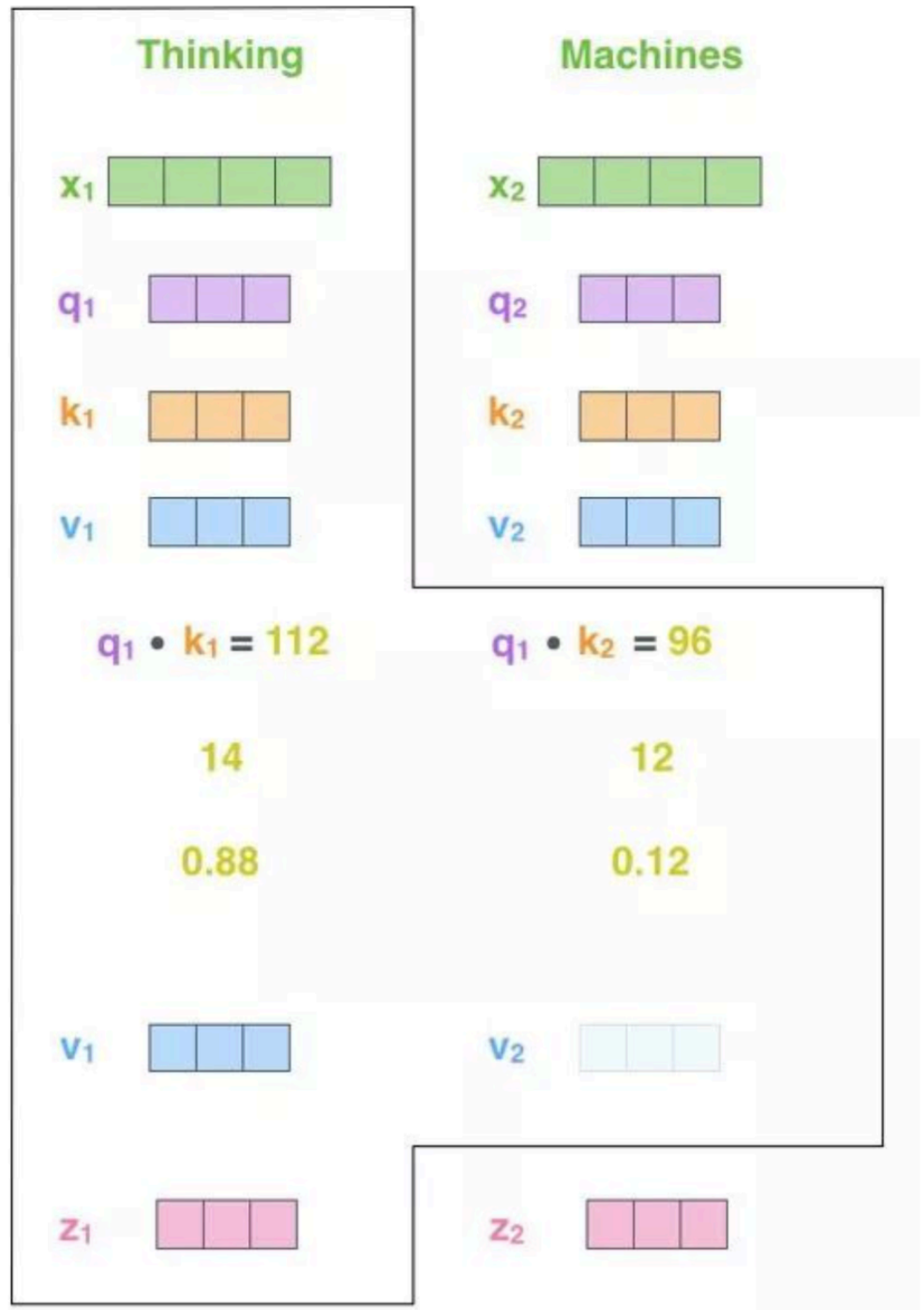
打分

除以8（$\sqrt{d_k}$）

Softmax

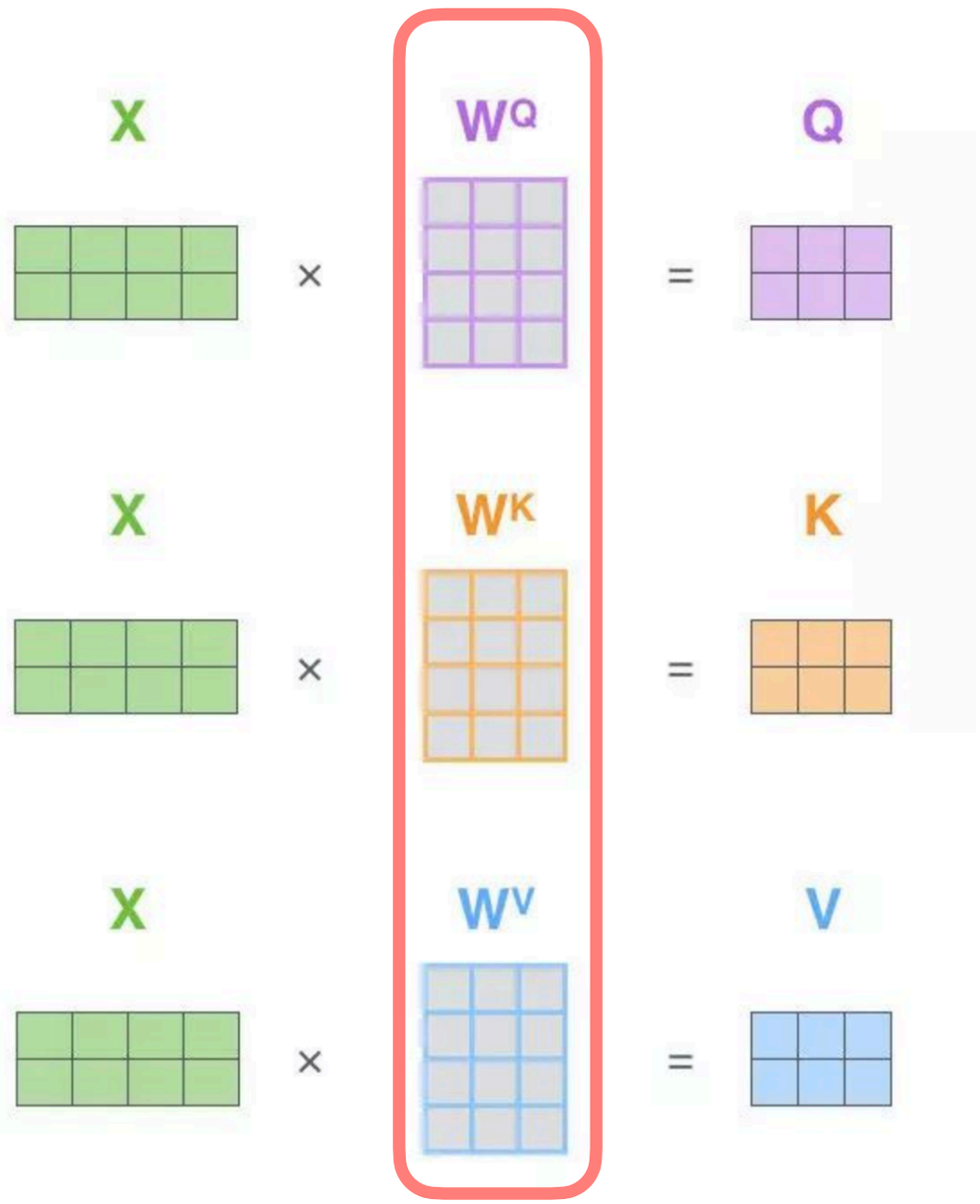
softmax
乘以
值向量

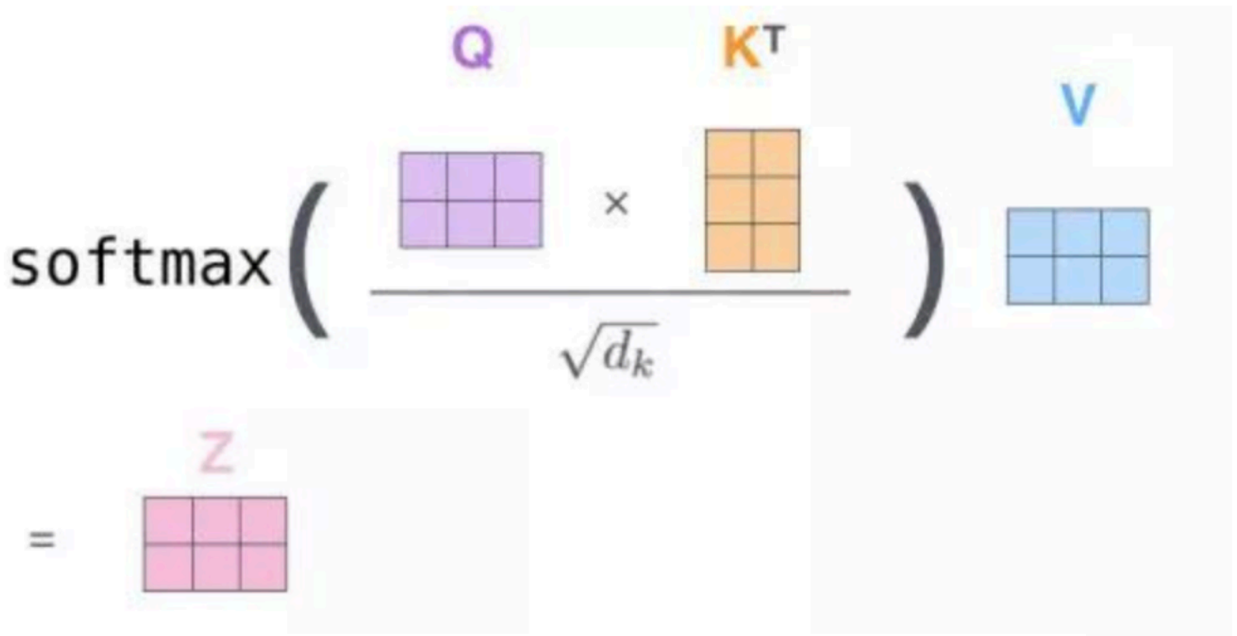
求和



https://mp.weixin.qq.com/s/uqqGI4GPXcSIrjE0jkh8vw

self-attention

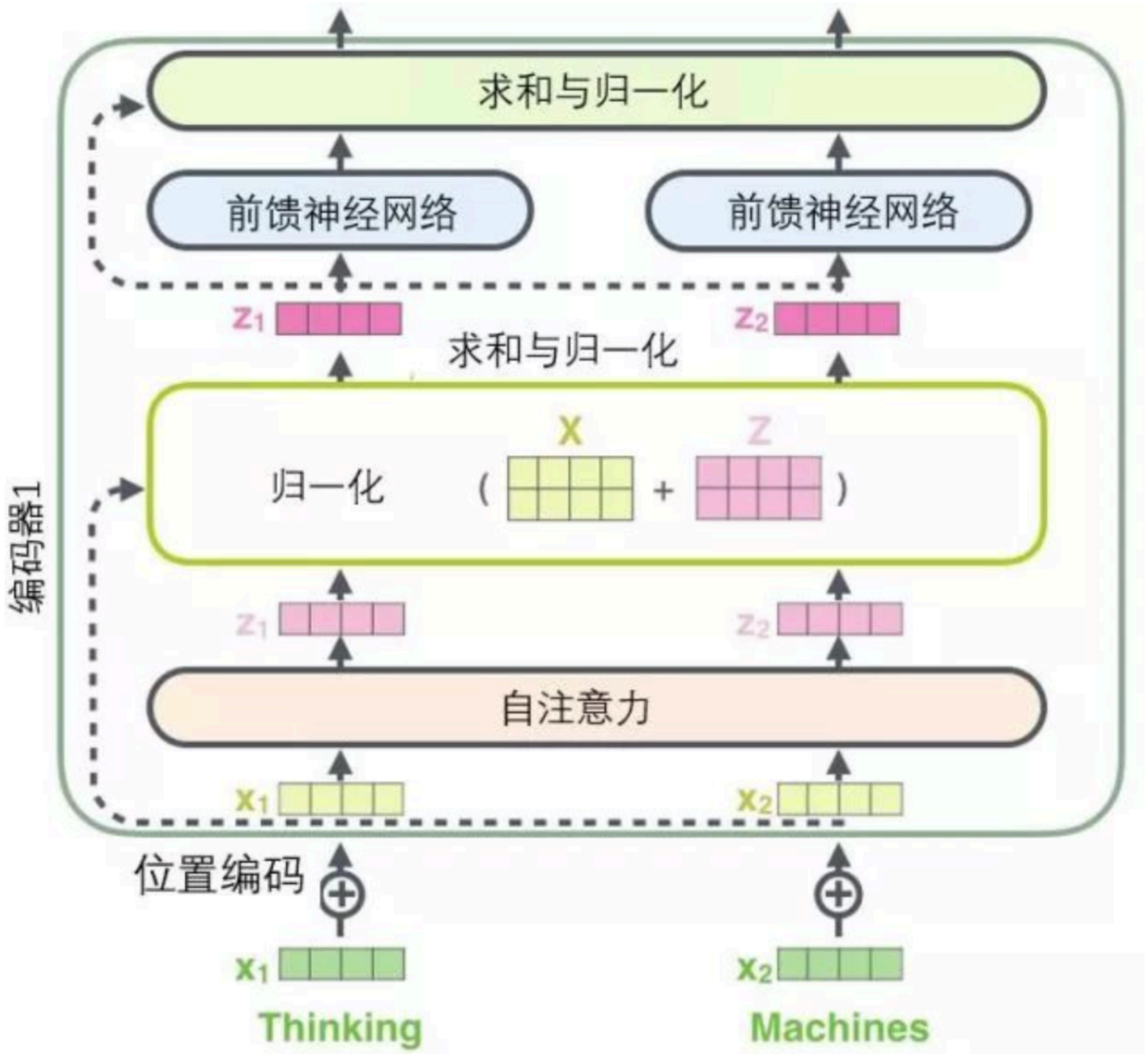




$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

https://mp.weixin.qq.com/s/uqqGI4GPXcSIrjE0jkh8vw

Encoder



https://mp.weixin.qq.com/s/uqqGI4GPXcSIrjE0jkh8vw

Graph Convolution Network:

$$h_v^{(k)} = f^{(k)}(W^{(k)} \cdot \frac{\sum_{u \in N(v)} h_u^{(k-1)}}{|N(v)|} + B^k \cdot h_v^{(k-1)})$$

$h_v^{(k)}$: embedding of node v

$|N(v)|$: the number of node v neighbours

For each step k, the function $f^{(k)}$, matrices $W^{(k)}$ and $B^{(k)}$ are shared across all nodes.

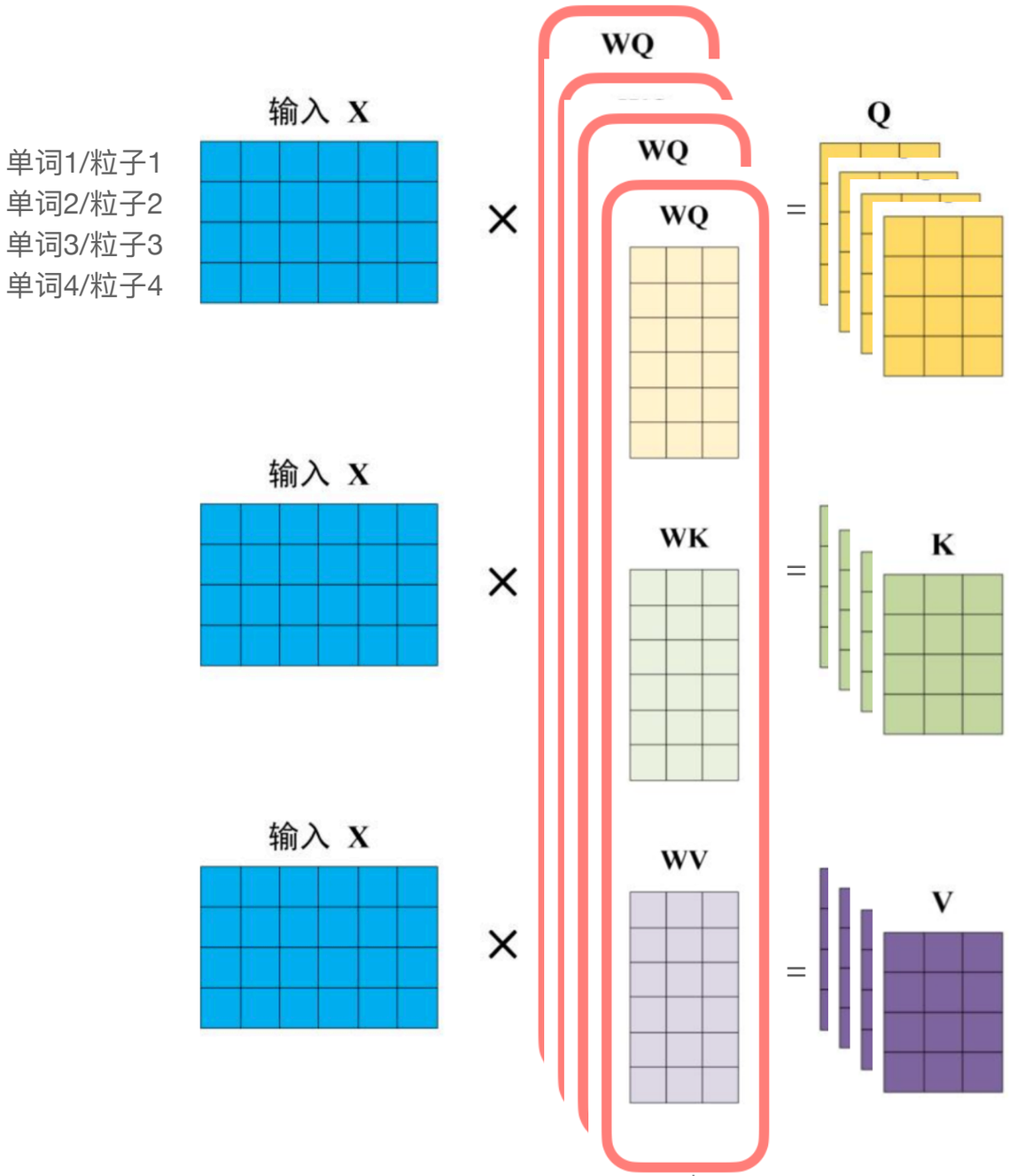
Graph Attention Network

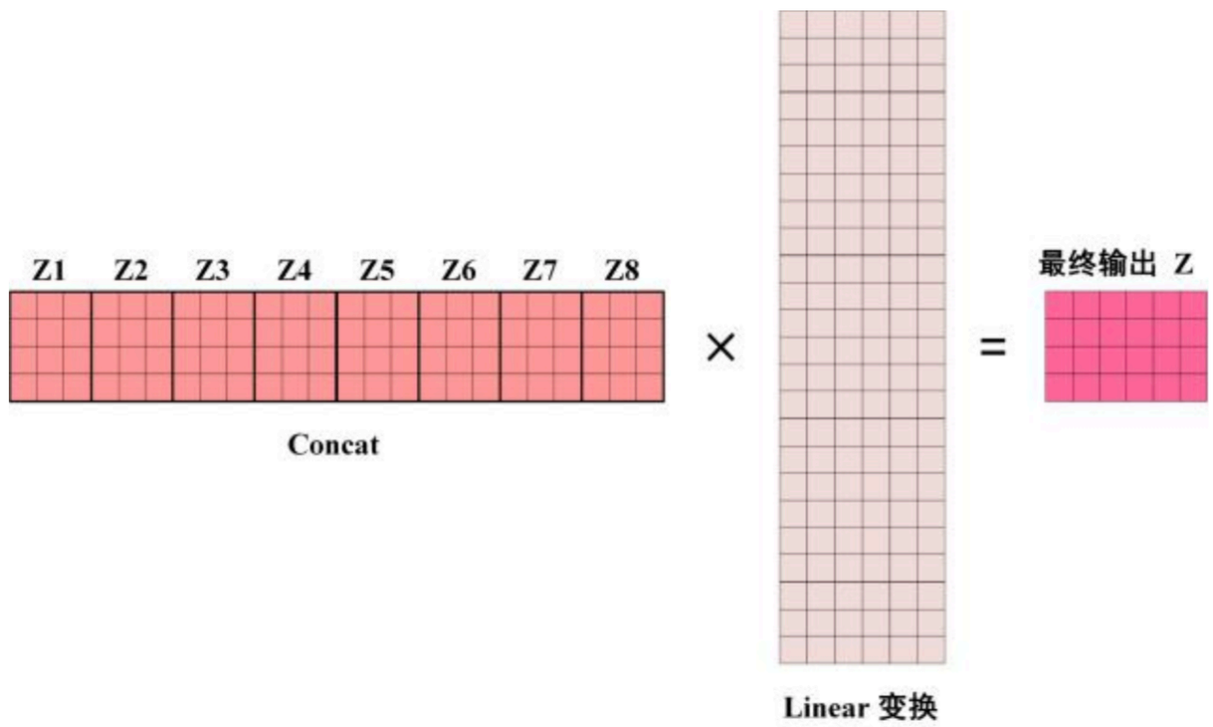
$$h_v^{(k)} = f^{(k)}(W^{(k)} \cdot [\sum_{u \in N(v)} \alpha_{vu}^{(k-1)} h_u^{(k-1)} + \alpha_{vv}^{(k-1)} h_v^{(k-1)}])$$

$$\alpha_{vu}^{(k)} = \frac{A^{(k)}(h_v^{(k)}, h_u^{(k)})}{\sum_{w \in N(v)} A^{(k)}(h_v^{(k)}, h_w^{(k)})}$$

For each step k, the function $f^{(k)}$, matrices $W^{(k)}$ and attention mechanism $A^{(k)}$ (generally, another neural network) are shared across all nodes.

multi-head self-attention





$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_\text{h})W^O$$
$$\text{where head}_\text{i} = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$