New ML methods for ν_e interaction identification @T2K near detector

Anaëlle Chalumeau

Réunion du vendredi - LPNHE
11/04/2025

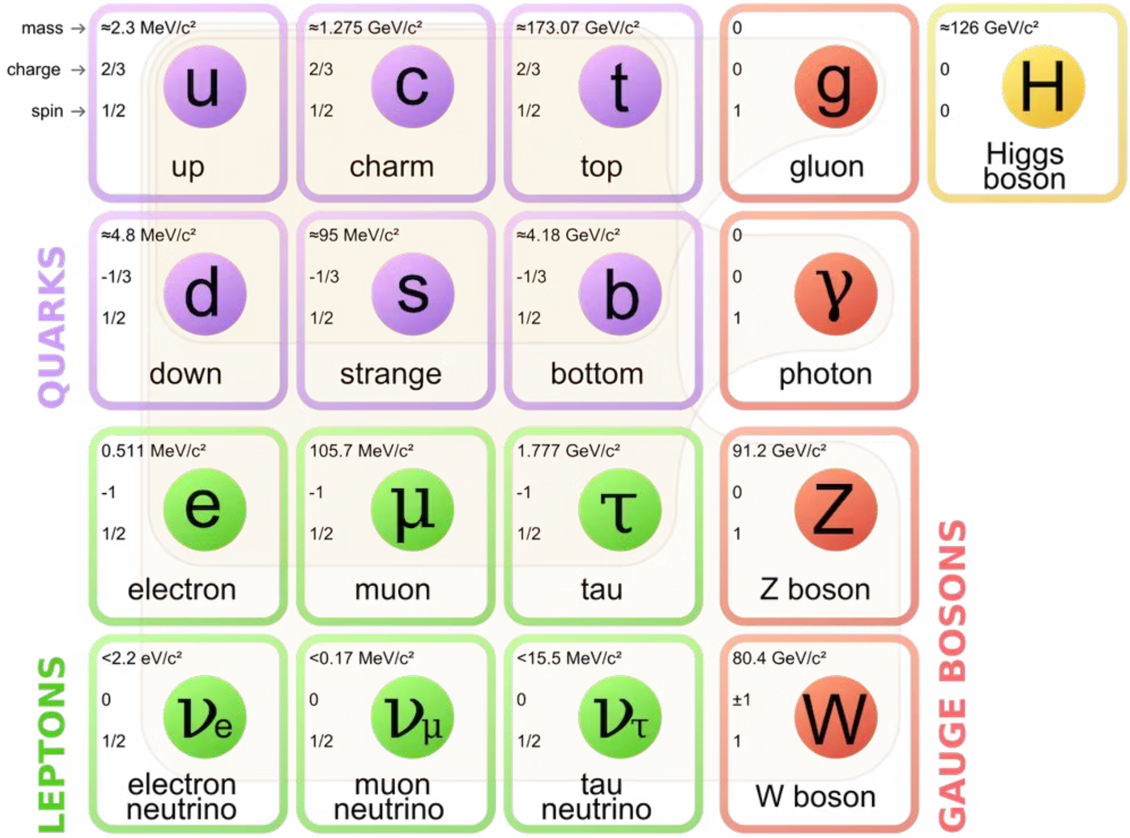
Overview

- Why neutrinos are interesting
- T2K and ν_e (electronic neutrinos)
- Why we need to improve the ν_e selection
- 2 new PID* methods with ML: Transformer & Variational Auto-Encoder
- The Transformer PID method
    - What is a Transformer
    - How we can apply it for the ν_e selection
    - Vision Transformer
    - First performances
- A glanced at the Variational Auto-Encoder method

*PID = Particle IDentification

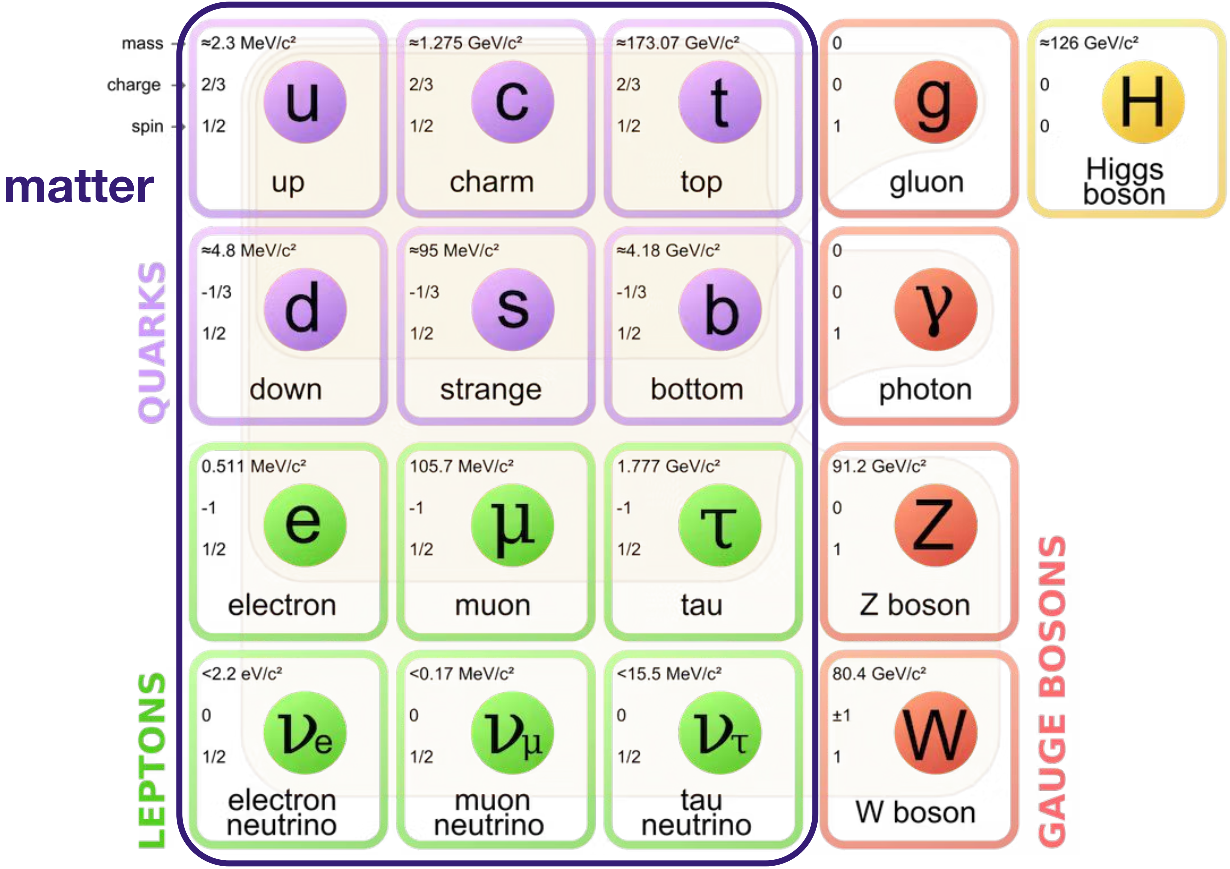
Why neutrinos are interesting

Standard model of particles



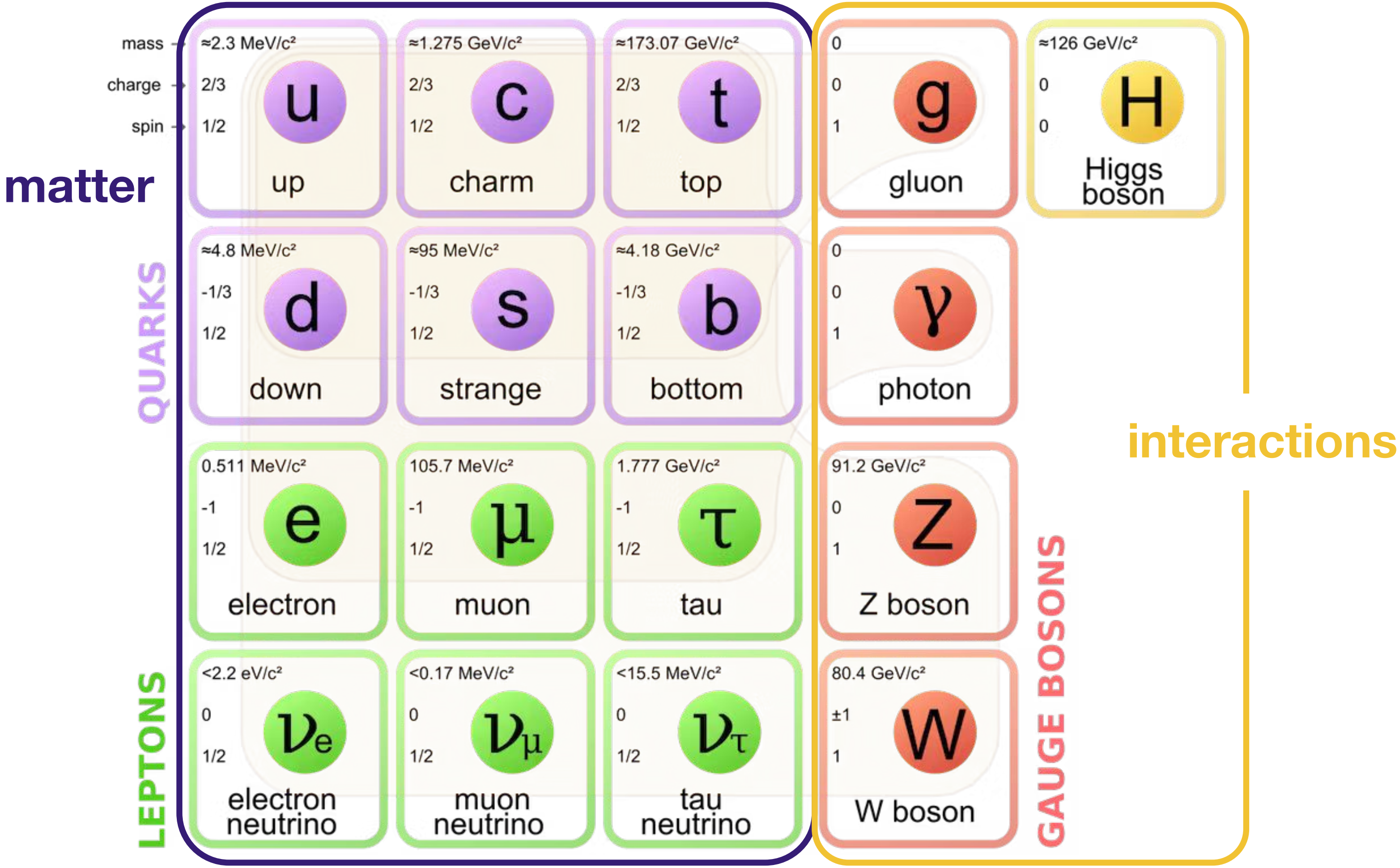
Why neutrinos are interesting

Standard model of particles



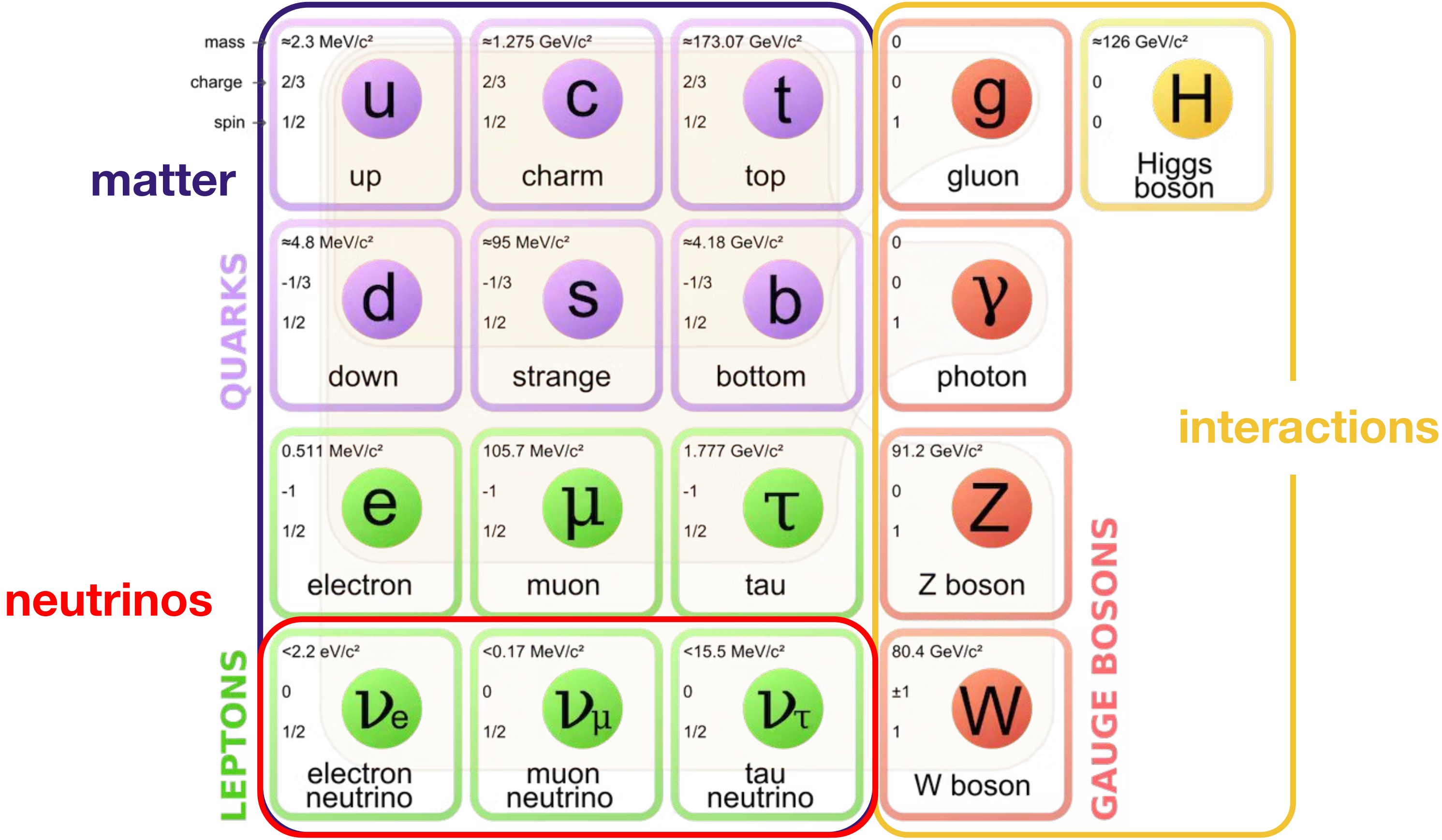
Why neutrinos are interesting

Standard model of particles



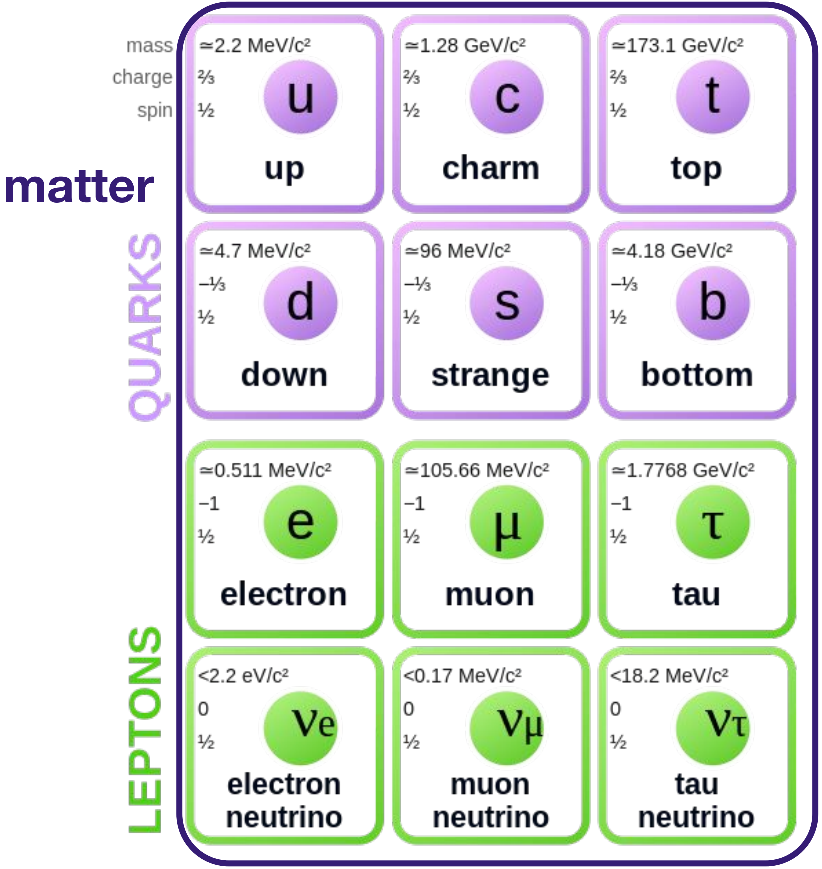
Why neutrinos are interesting

Standard model of particles



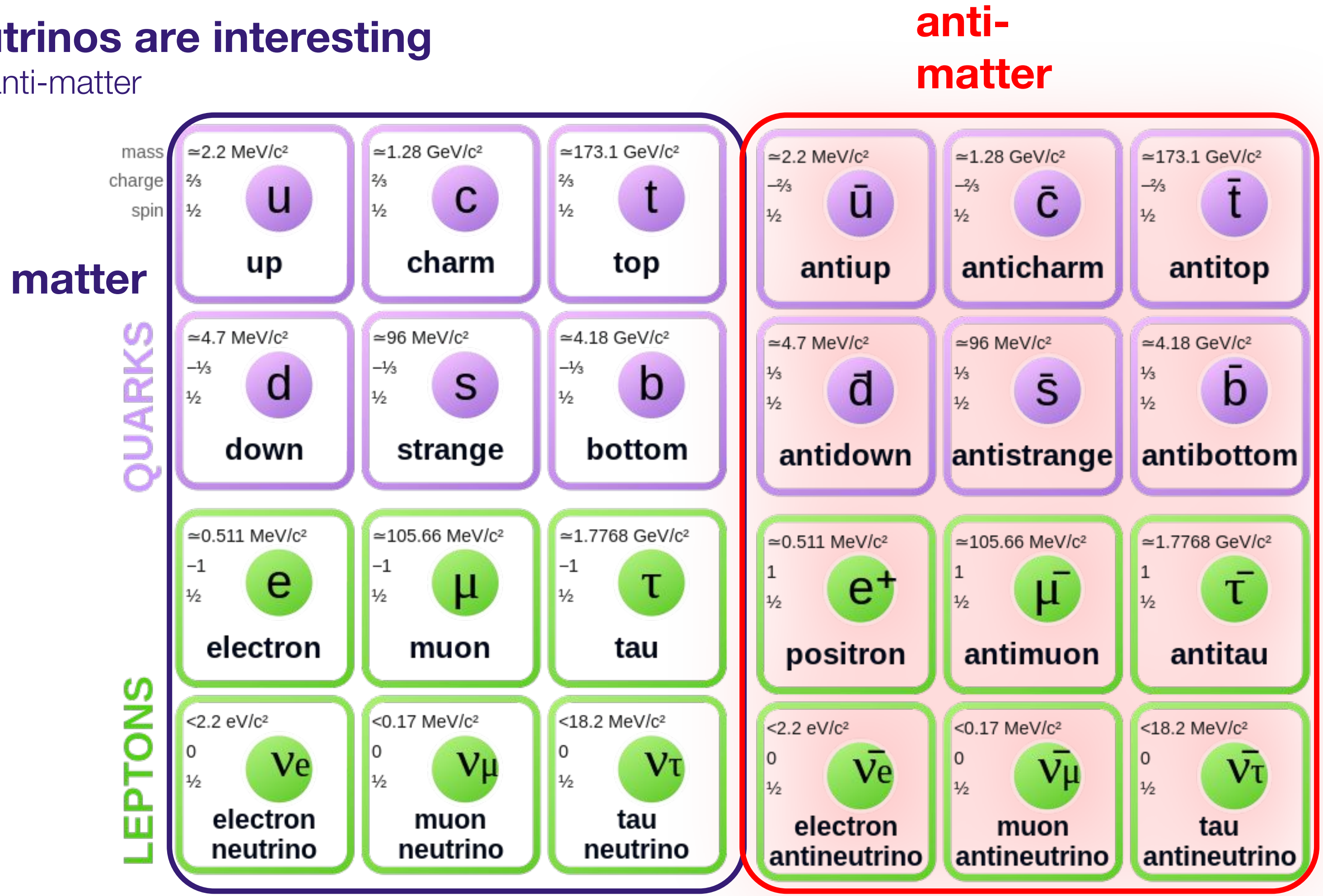
Why neutrinos are interesting

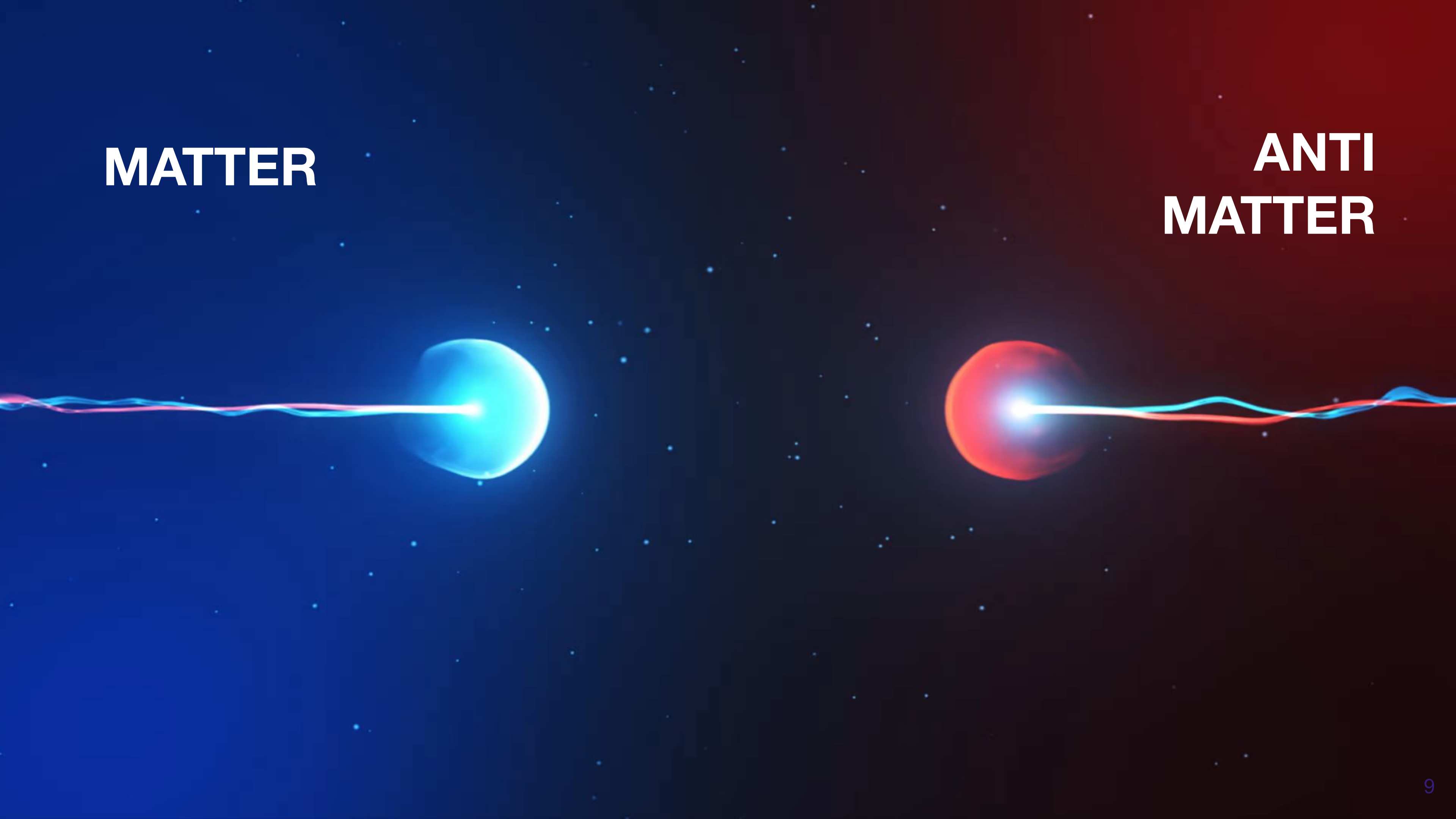
Matter



Why neutrinos are interesting

Matter vs anti-matter




MATTER
ANTI
MATTER

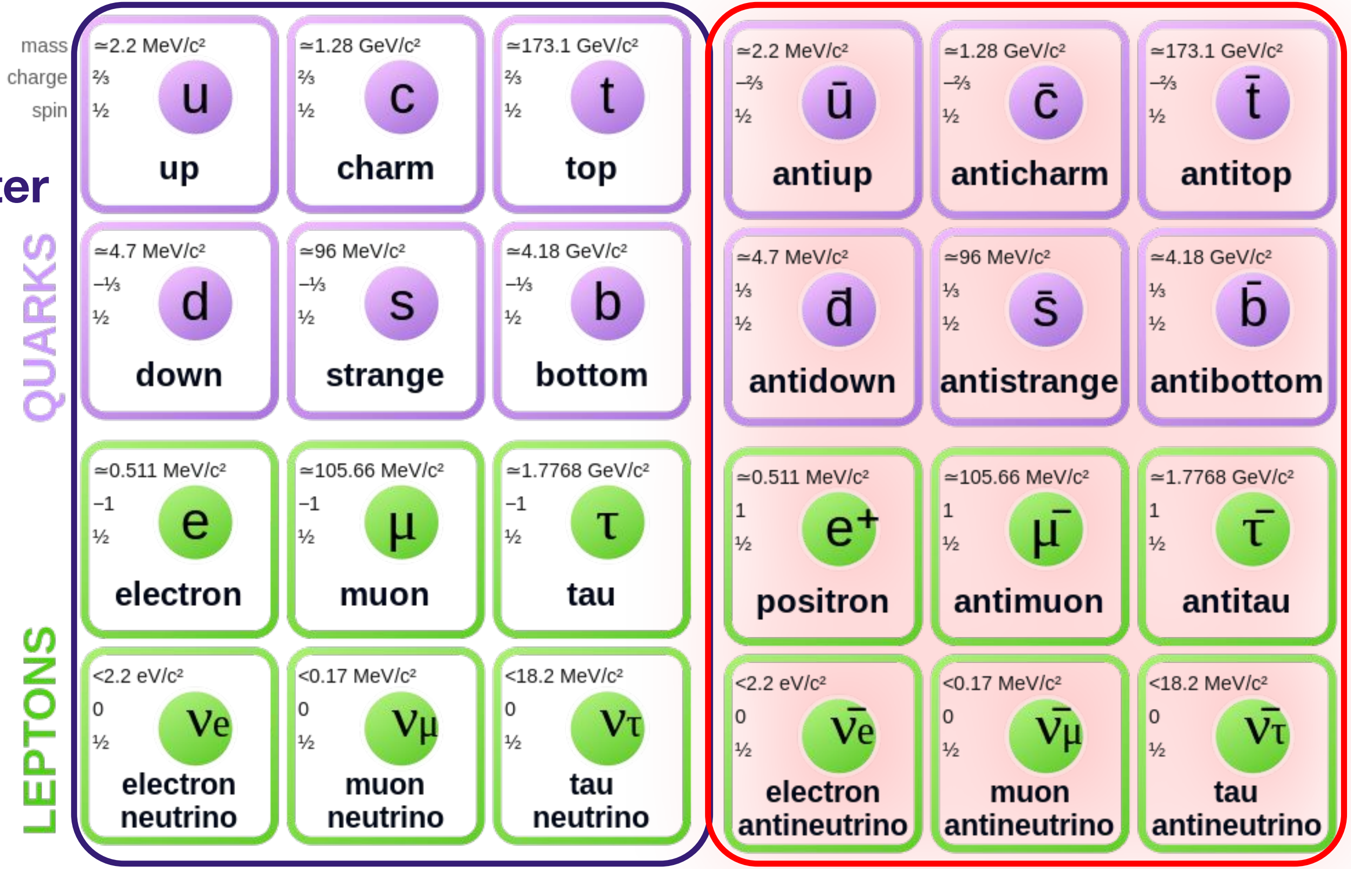
LIGHT

Why neutrinos are interesting

Matter vs anti-matter

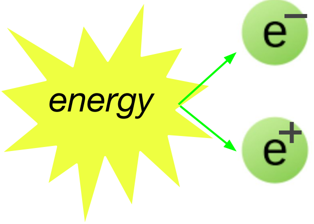
anti-matter

matter



Why neutrinos are interesting

Matter vs anti-matter

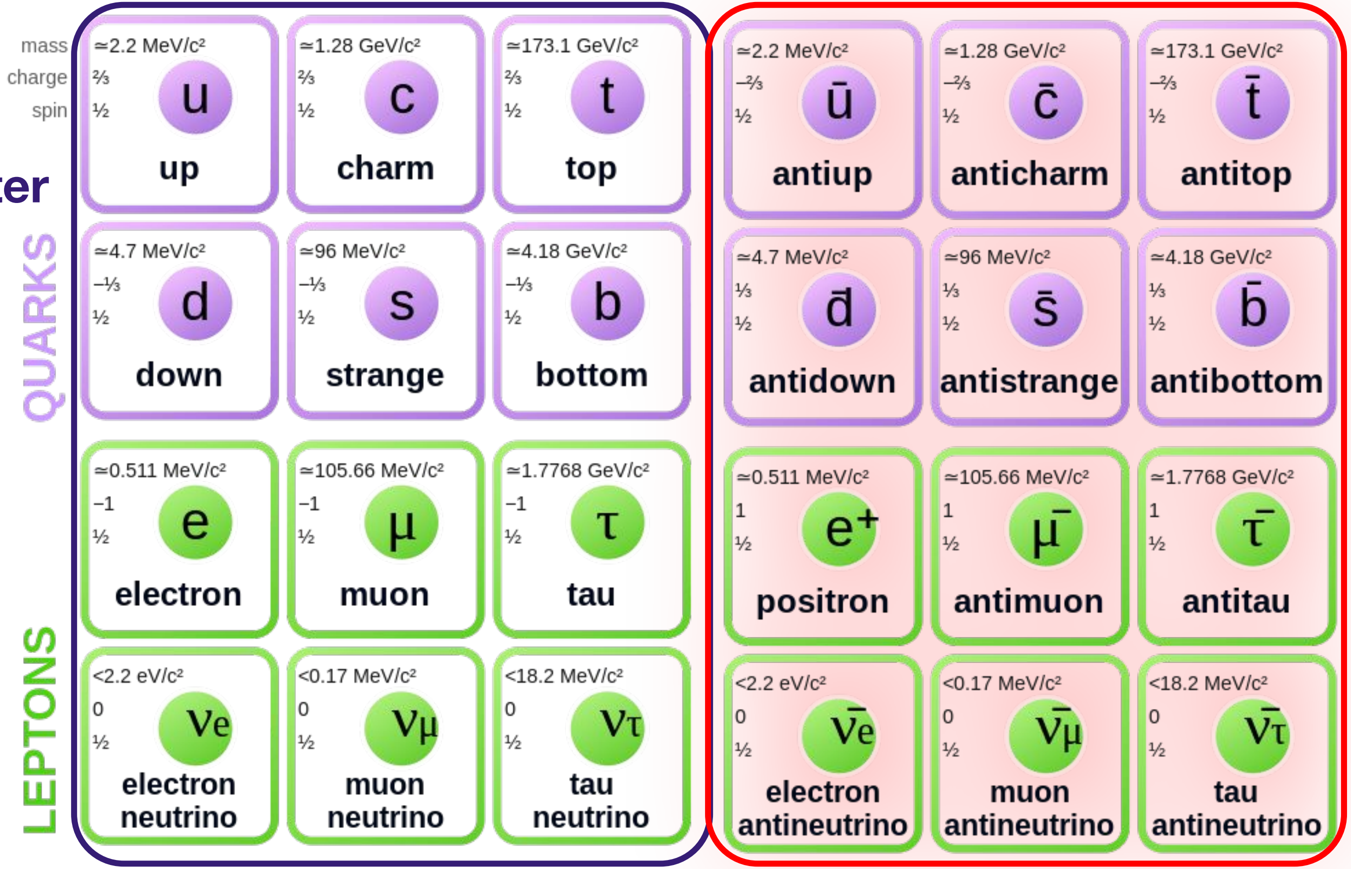


energy ↔ mass

$E = mc^2$

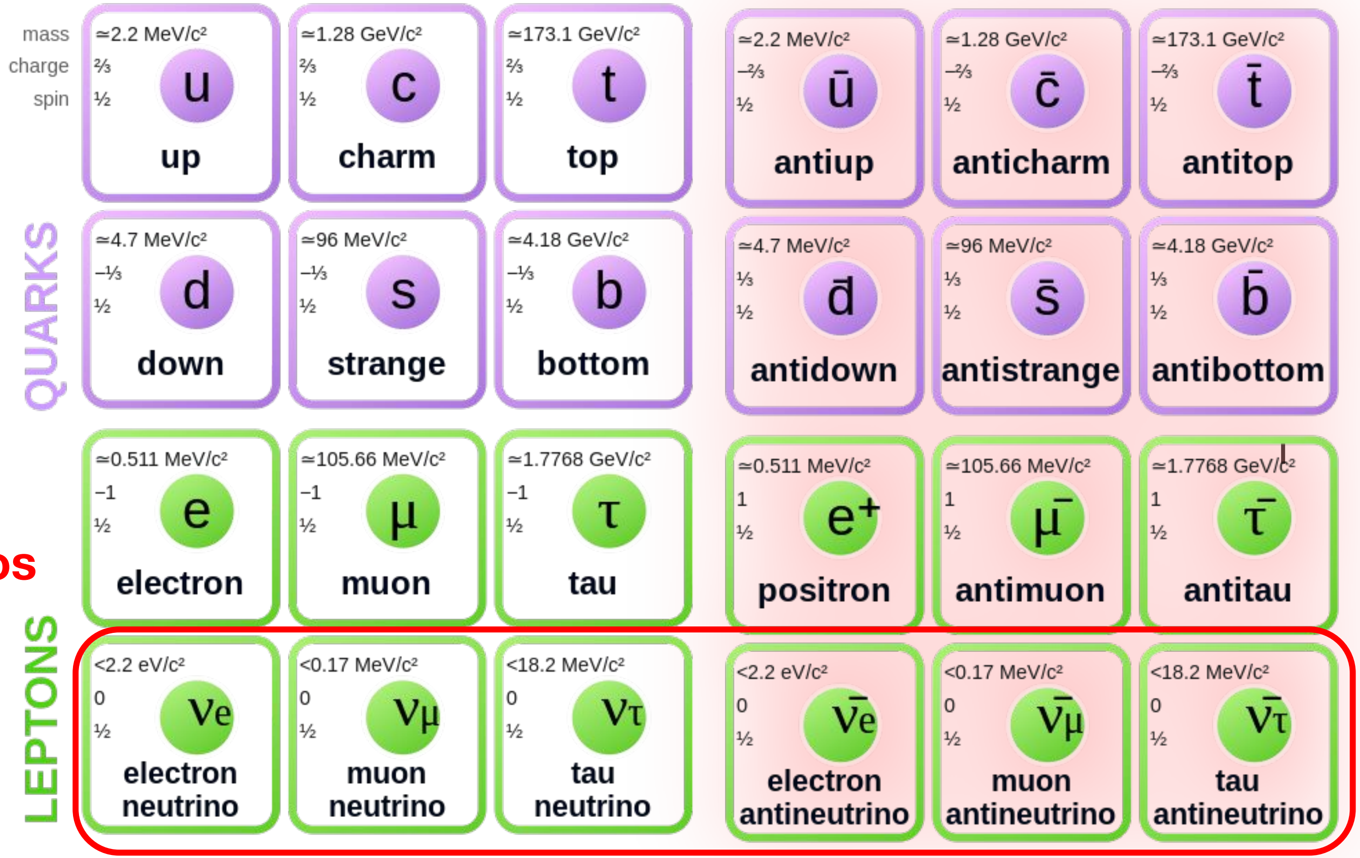
matter

anti-matter



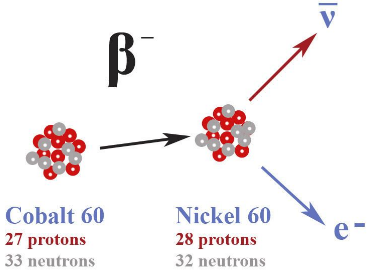
Why neutrinos are interesting

neutrinos



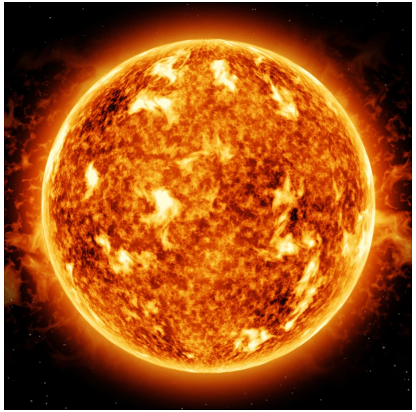
Why neutrinos are interesting

Where are they?

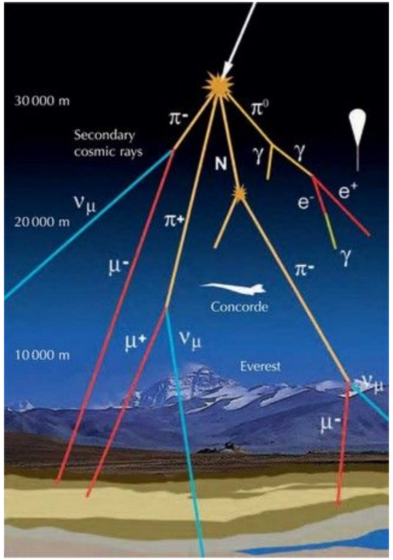


radioactivity

nuclear reactions in nuclear power plants



nuclear reaction in stars

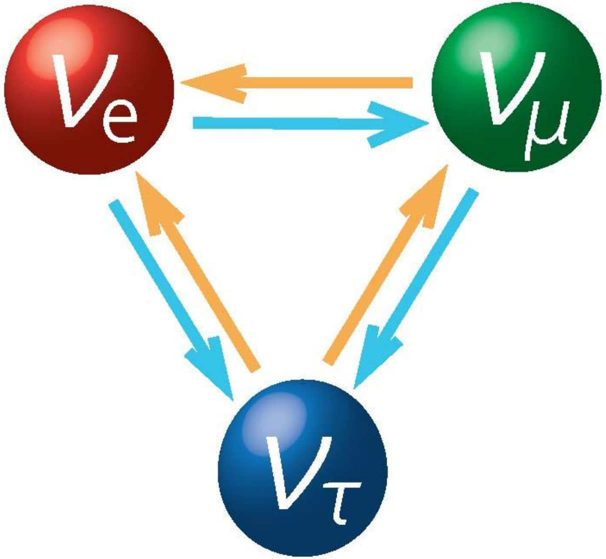


cosmic rays collision with the atmosphere

T2K (and similar experiments) create specifically a beam of neutrinos

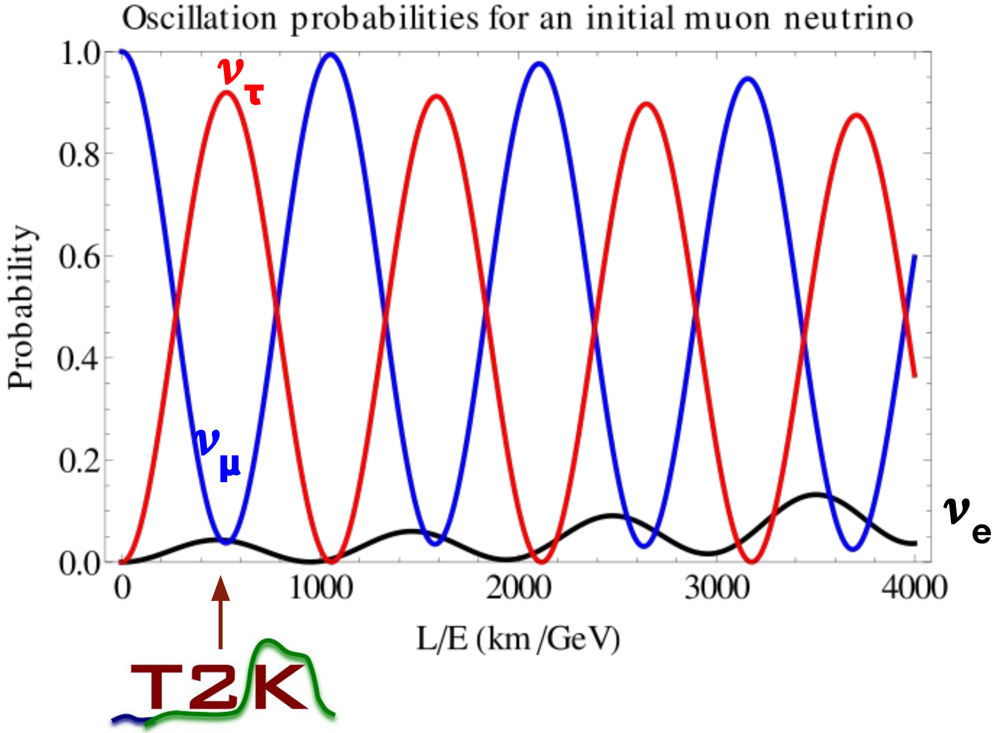
Why neutrinos are interesting

They oscillate!

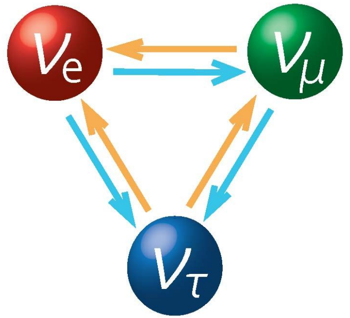


Why neutrinos are interesting

They oscillate!



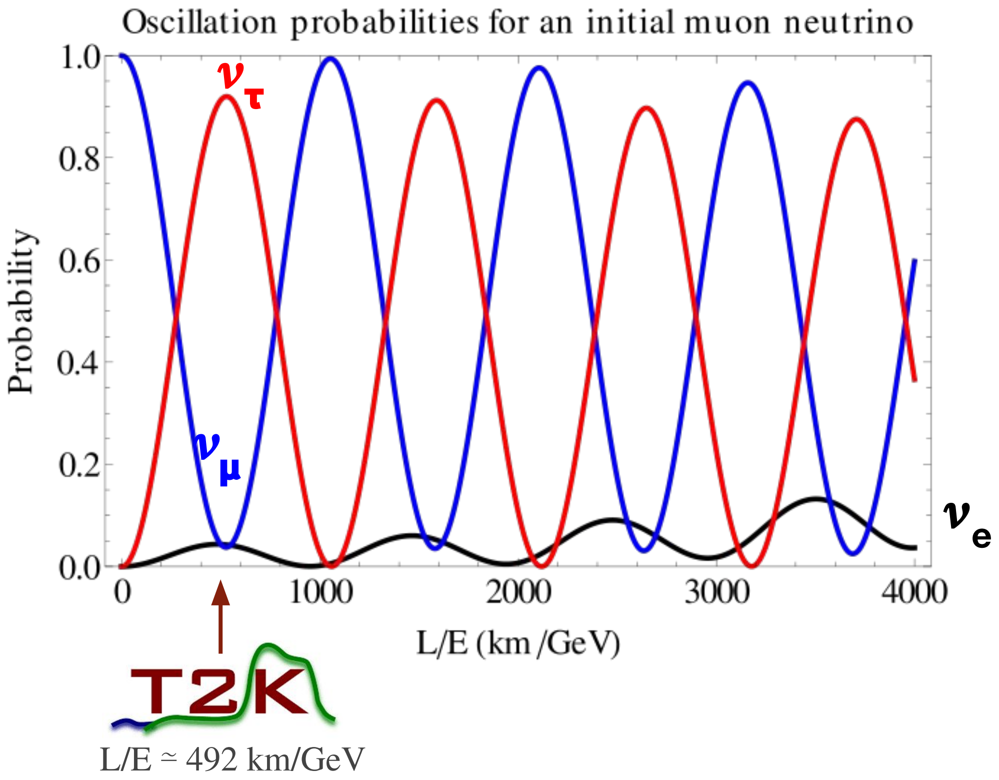
L/E ≃ 492 km/GeV

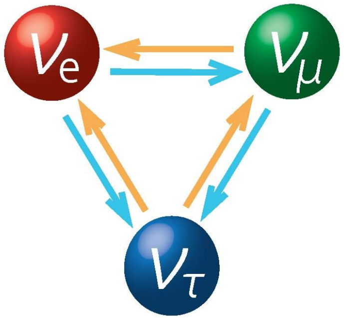


$$P(\nu_\mu \to \nu_e) = f(\theta, \delta)$$

Why neutrinos are interesting

They oscillate!



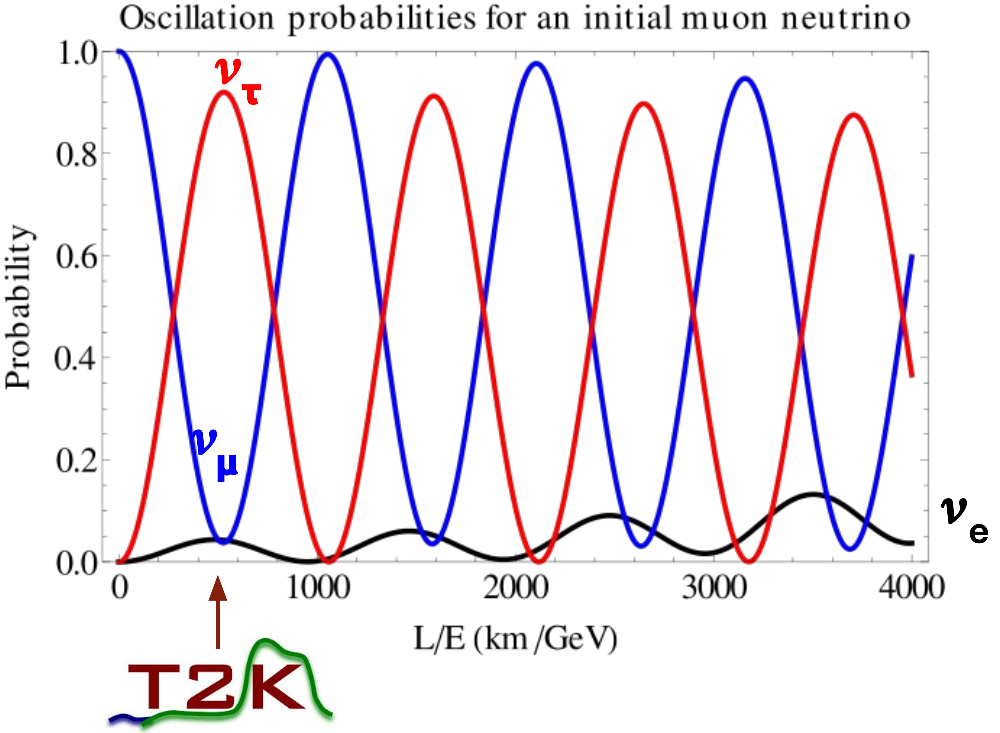


$$P(\nu_\mu \to \nu_e) = f(\theta, \delta)$$

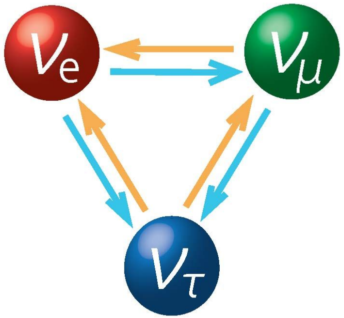
$$P(\overline{\nu}_\mu \to \overline{\nu}_e) = f'(\theta, \delta)$$

Why neutrinos are interesting

They oscillate!



L/E ≃ 492 km/GeV



$$P(\nu_\mu \rightarrow \nu_e) = f(\theta, \delta)$$

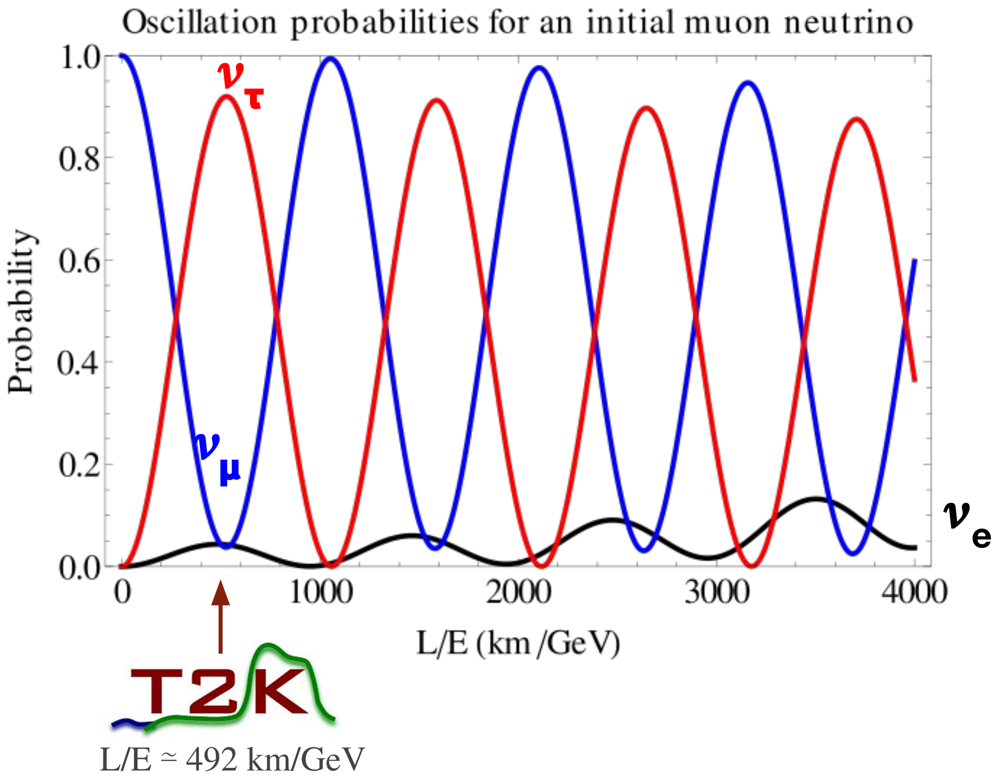
$$P(\overline{\nu}_\mu \rightarrow \overline{\nu}_e) = \boxed{f\prime(\theta, \delta)}$$

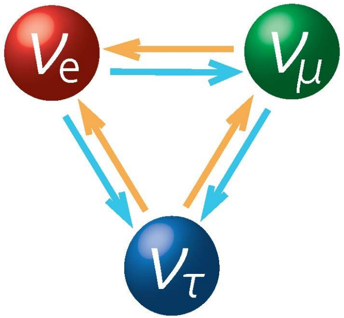
neutrinos & anti-neutrinos

don't oscillate the same way

Why neutrinos are interesting

They oscillate!





$$P(\nu_\mu \rightarrow \nu_e) = f(\theta, \delta)$$

$$P(\overline{\nu}_\mu \rightarrow \overline{\nu}_e) = f\prime(\theta, \delta)$$

neutrinos & anti-neutrinos **don't oscillate the same way**

could explain why there was more matter than anti-matter after the big-bang i.e. **why we exists**

T2K and ν_e (electronic neutrinos)

T2K and ν_e (electronic neutrinos)

particle accelerator to create neutrino beam

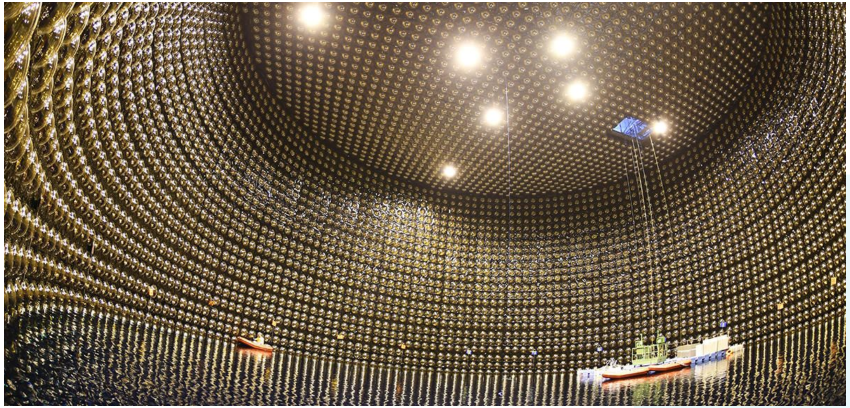
T2K and ν_e (electronic neutrinos)

near detector: ND280

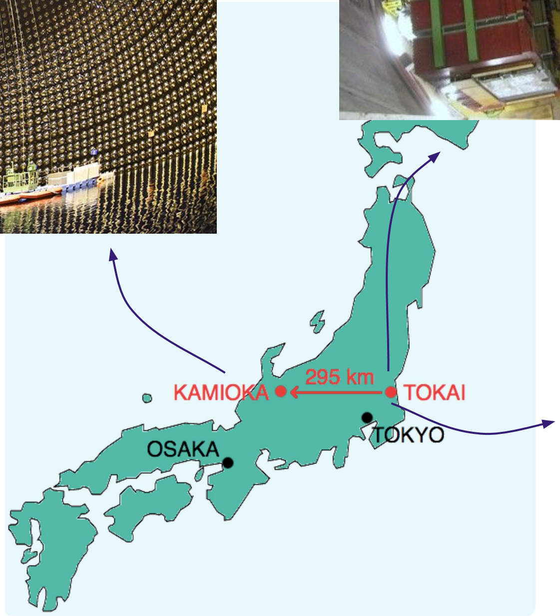
particle accelerator to create neutrino beam

T2K and ν_e (electronic neutrinos)

near detector: ND280

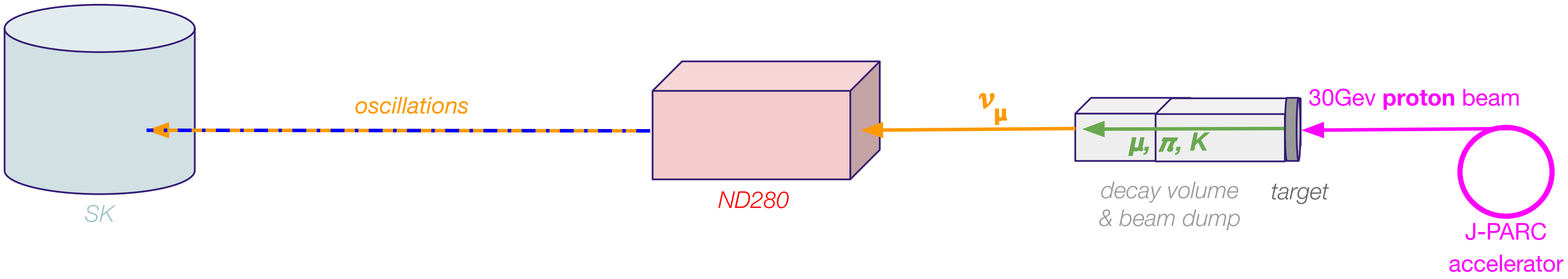


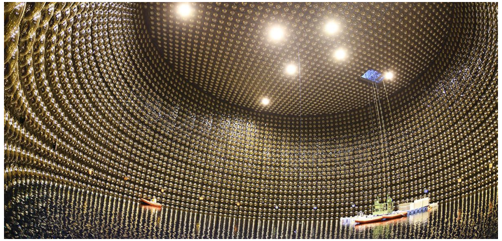
far detector: Super-Kamiokande



particle accelerator to create neutrino beam

T2K and ν_e (electronic neutrinos)





Super-Kamiokande

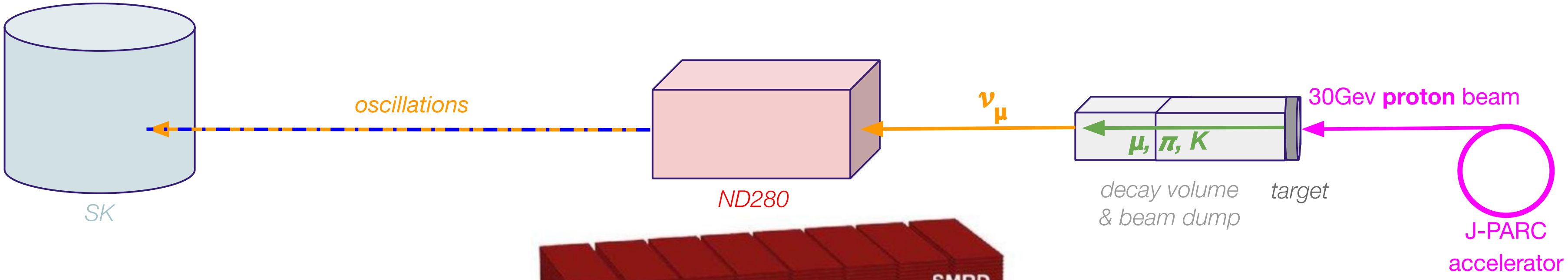
295km

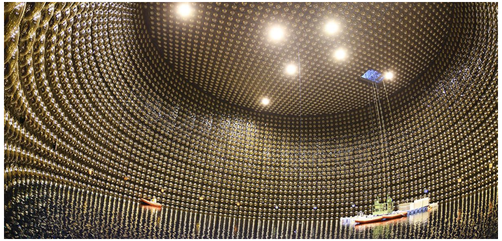
ND280

280m

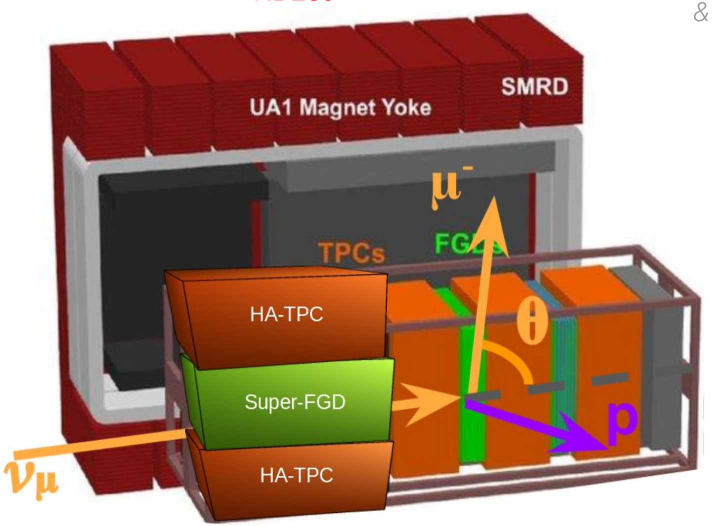
neutrino beam

T2K and ν_e (electronic neutrinos)



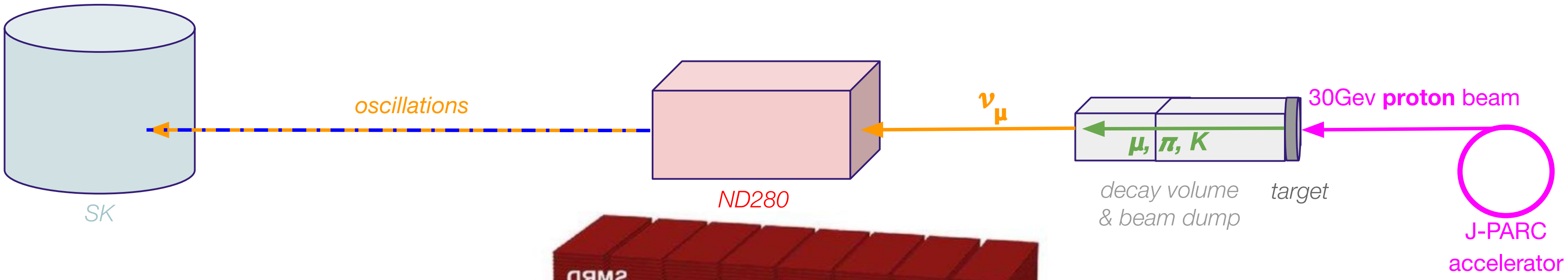


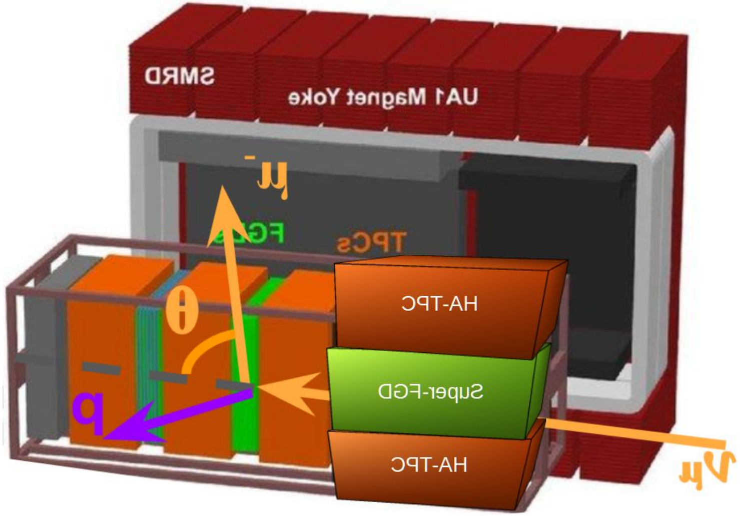
Super-Kamiokande

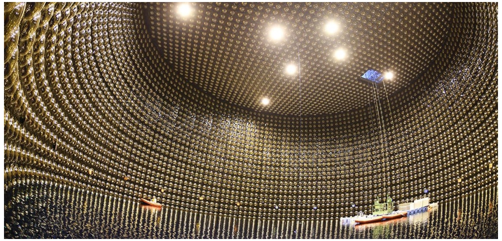


neutrino beam

T2K and ν_e (electronic neutrinos)

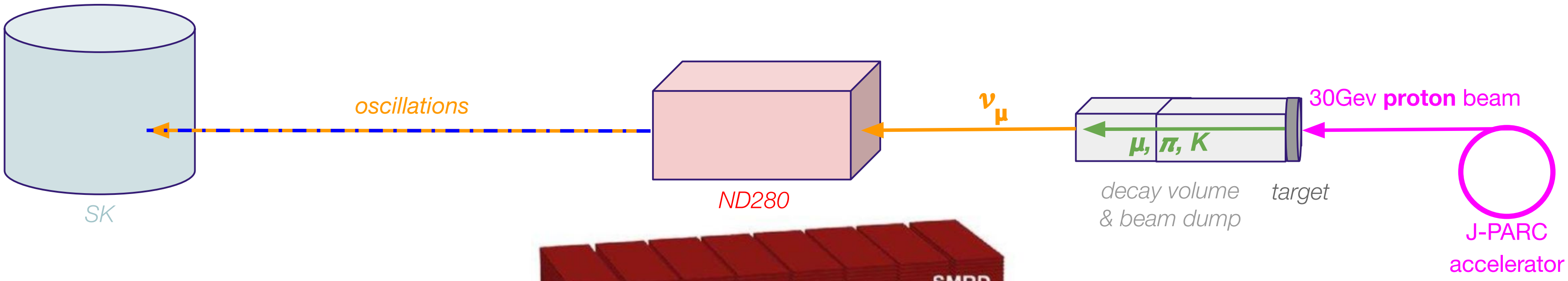


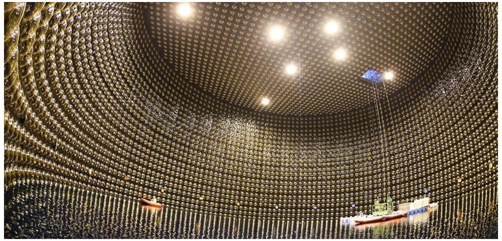



Super-Kamiokande

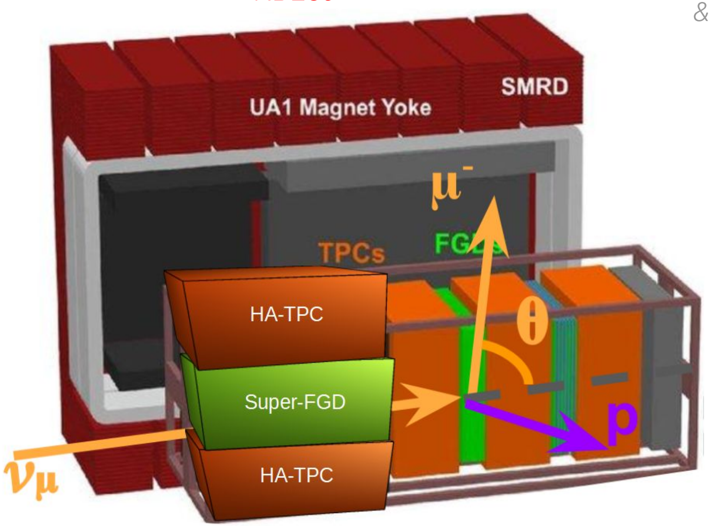

neutrino beam

T2K and ν_e (electronic neutrinos)



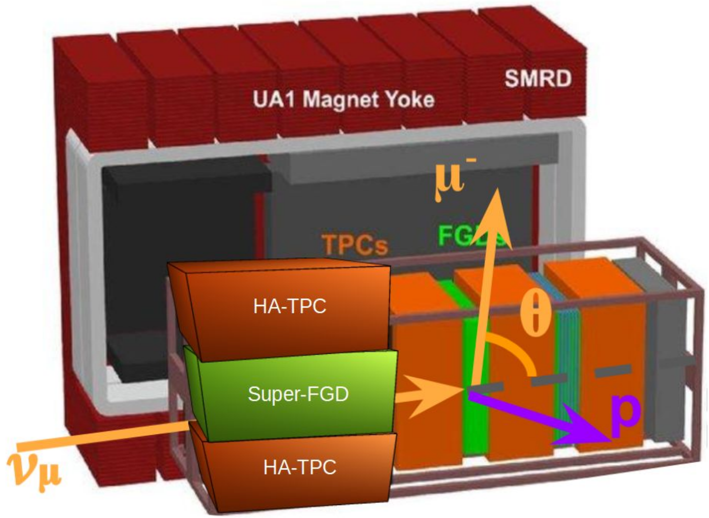


Super-Kamiokande

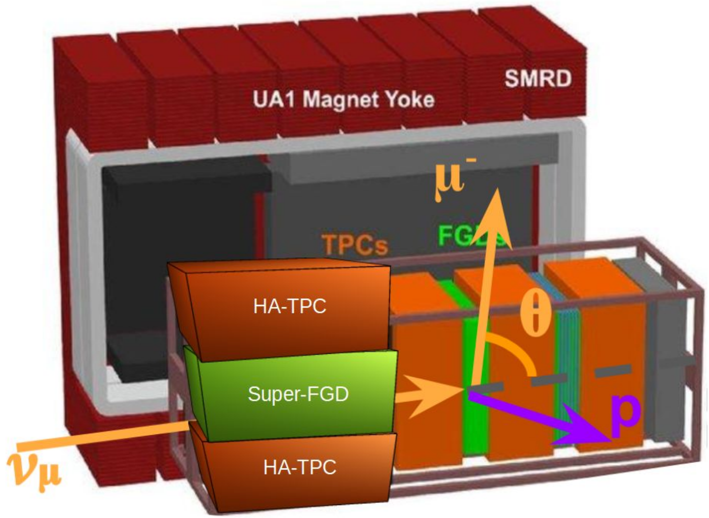


neutrino beam

T2K and ν_e (electronic neutrinos)



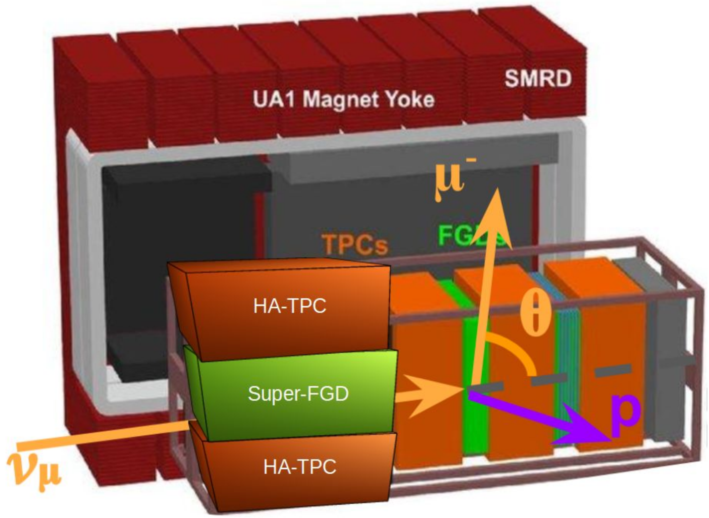
T2K and ν_e (electronic neutrinos)



ND280 Upgrade installed last year!

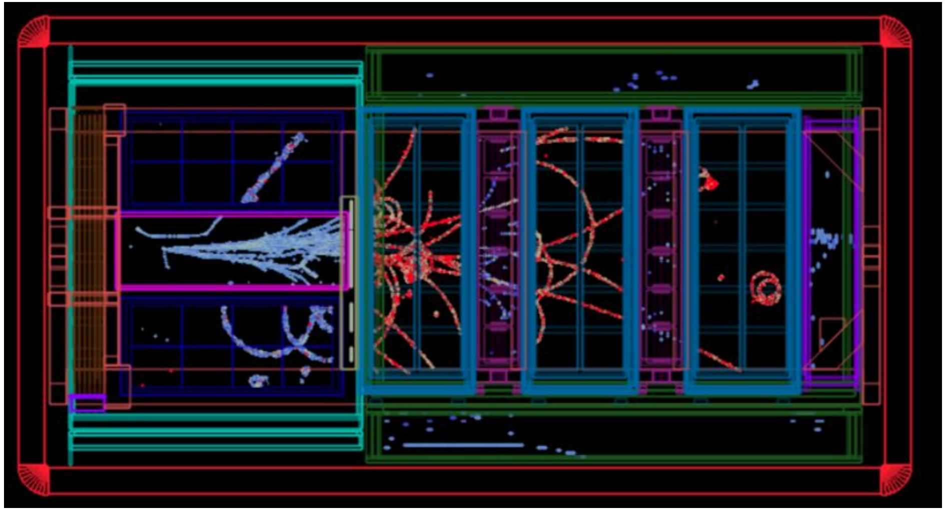
& data taking since end of 2024

T2K and ν_e (electronic neutrinos)

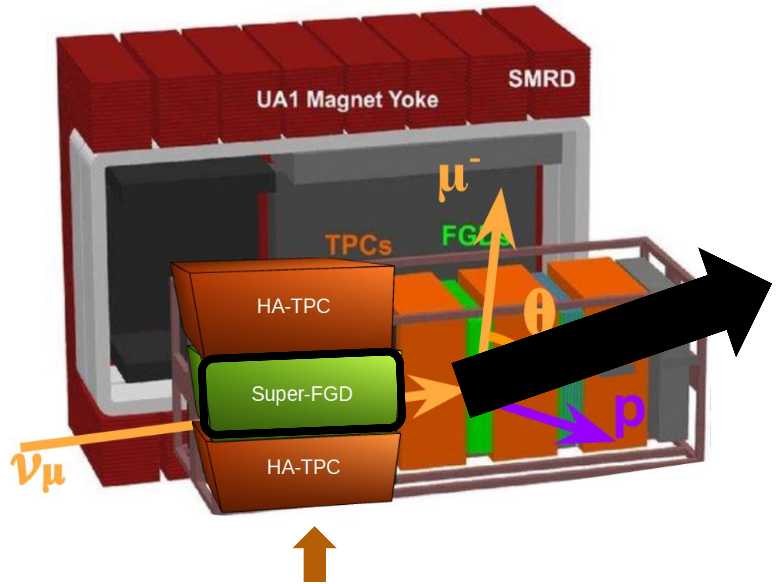


ND280 Upgrade installed last year!

& data taking since end of 2024



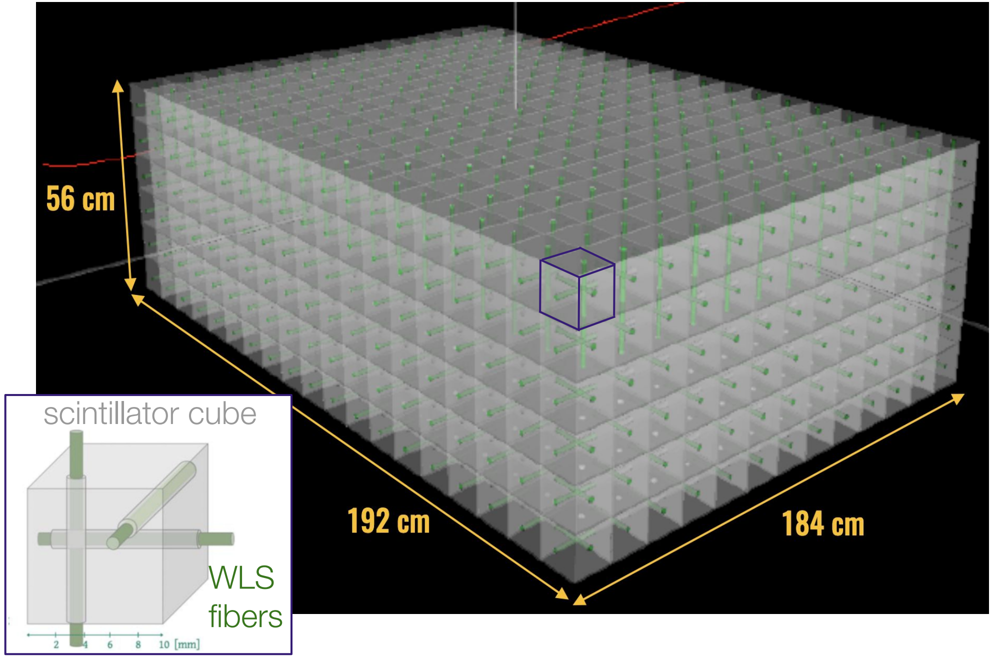
T2K and ν_e (electronic neutrinos)



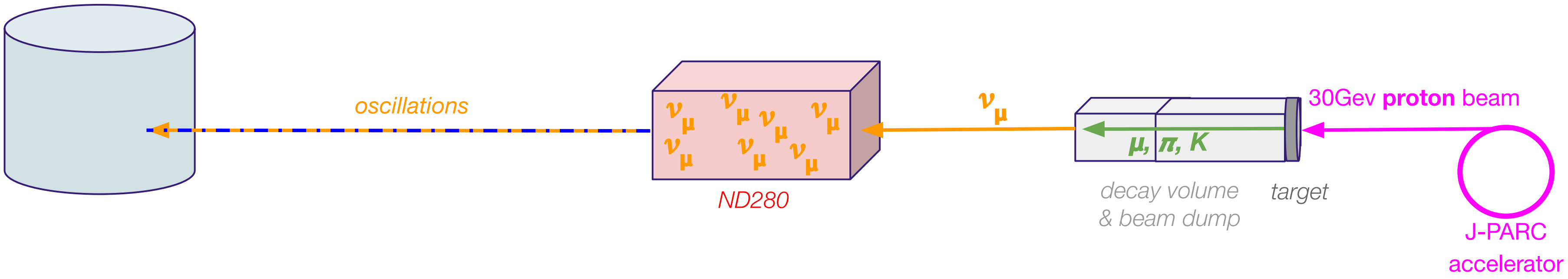
ND280 Upgrade installed last year!

& data taking since end of 2024

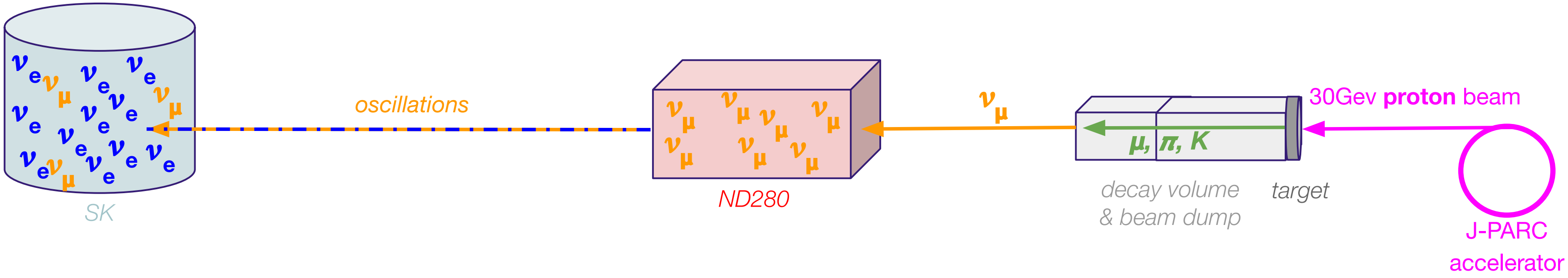
The Super-FGD



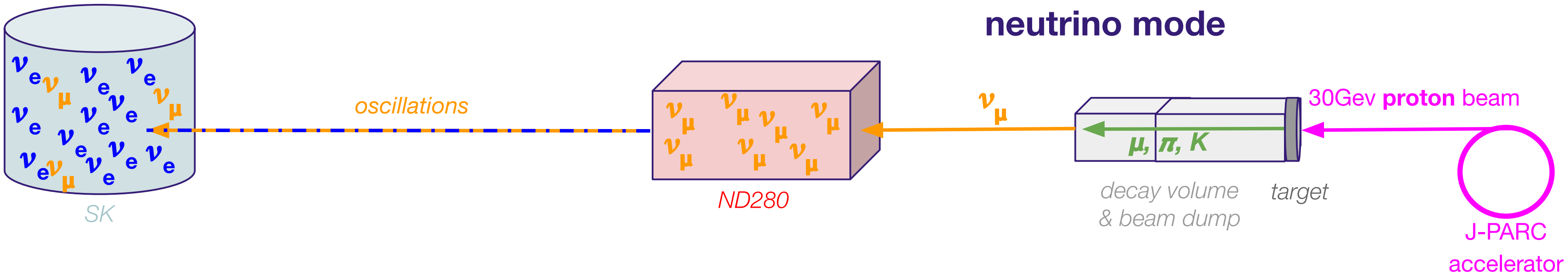
T2K and ν_e (electronic neutrinos)



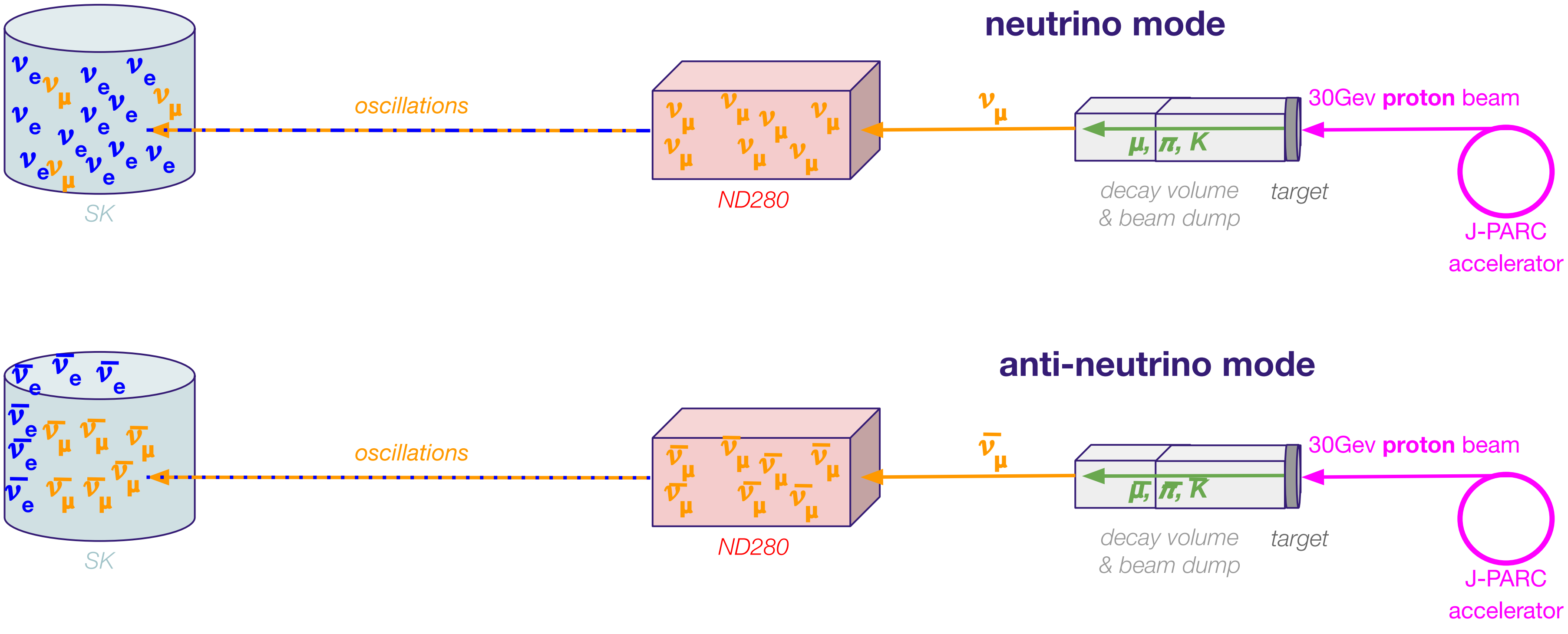
T2K and ν_e (electronic neutrinos)



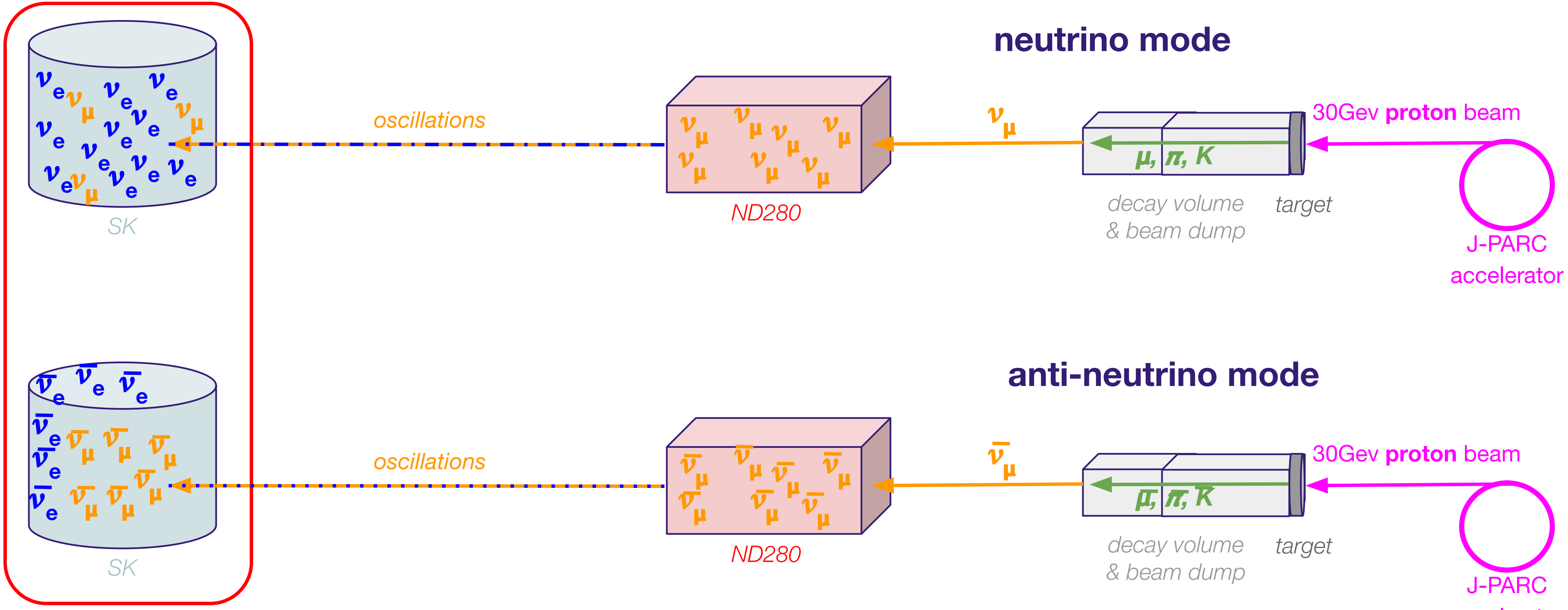
T2K and ν_e (electronic neutrinos)



T2K and ν_e (electronic neutrinos)

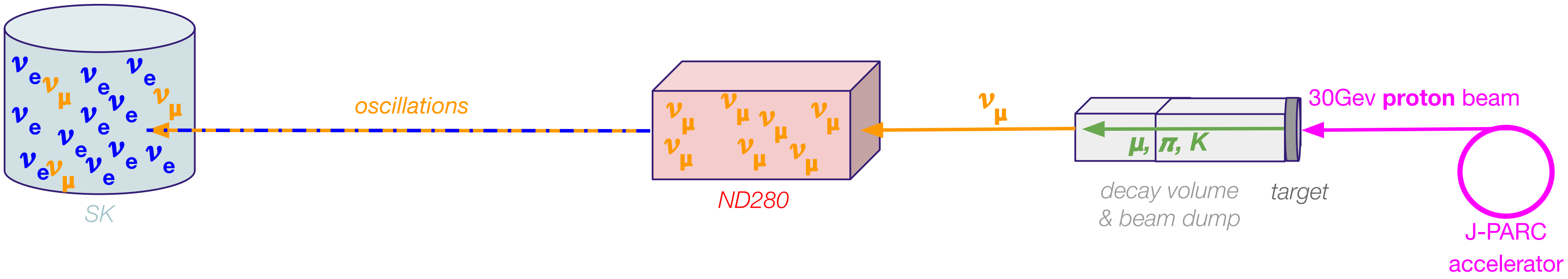


T2K and ν_e (electronic neutrinos)

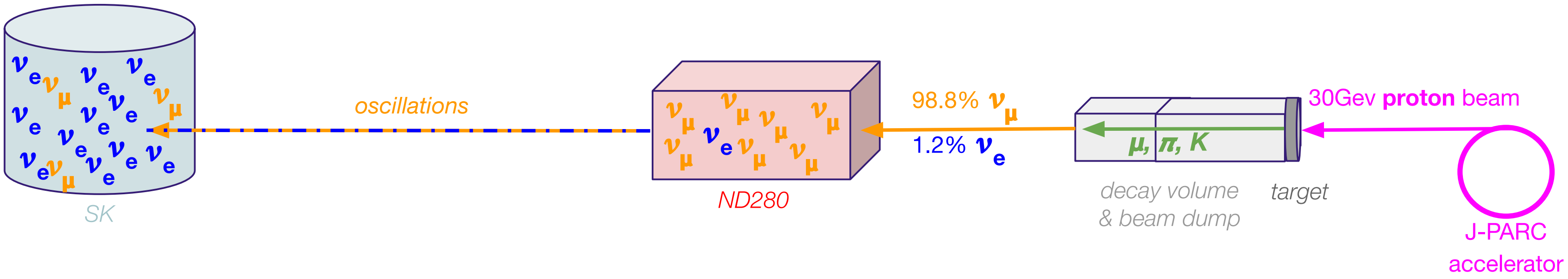


T2K goal: measure this difference!

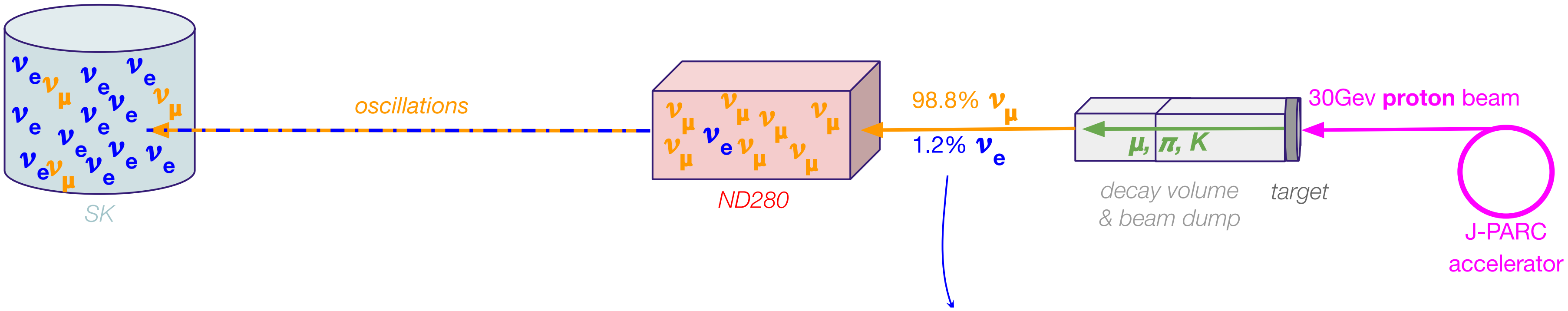
T2K and ν_e (electronic neutrinos)



T2K and ν_e (electronic neutrinos)

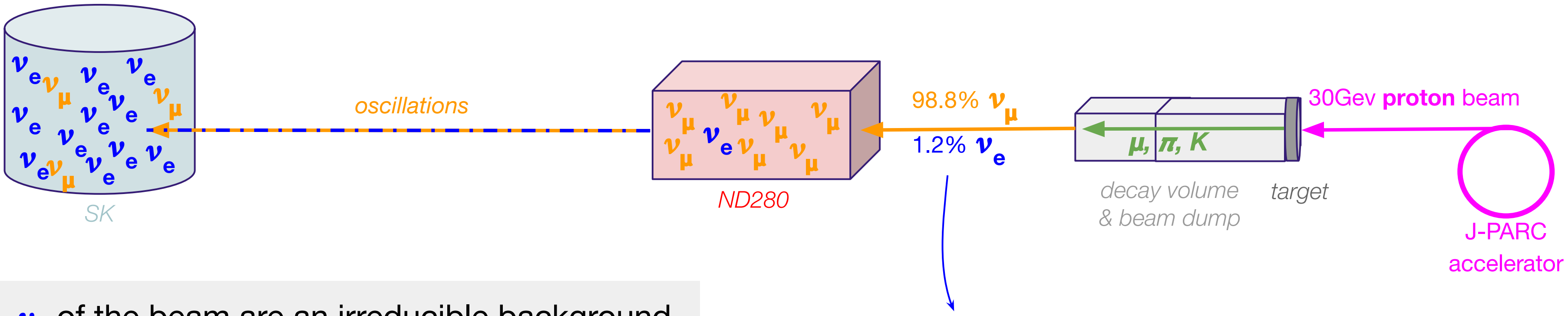


T2K and ν_e (electronic neutrinos)



Particle	Decay Products	Branching Fraction (%)
π^+	$\rightarrow \mu^+\nu_\mu$	99.9877
	$\rightarrow e^+\nu_e$	1.23×10^{-4}
K^+	$\rightarrow \mu^+\nu_\mu$	63.55
	$\rightarrow \pi^0\mu^+\nu_\mu$	3.353
	$\rightarrow \pi^0 e^+\nu_e$	5.07
K^0_L	$\rightarrow \pi^-\mu^+\nu_\mu$	27.04
	$\rightarrow \pi^- e^+\nu_e$	40.55
μ^+	$\rightarrow e^+\bar{\nu}_\mu\nu_e$	100

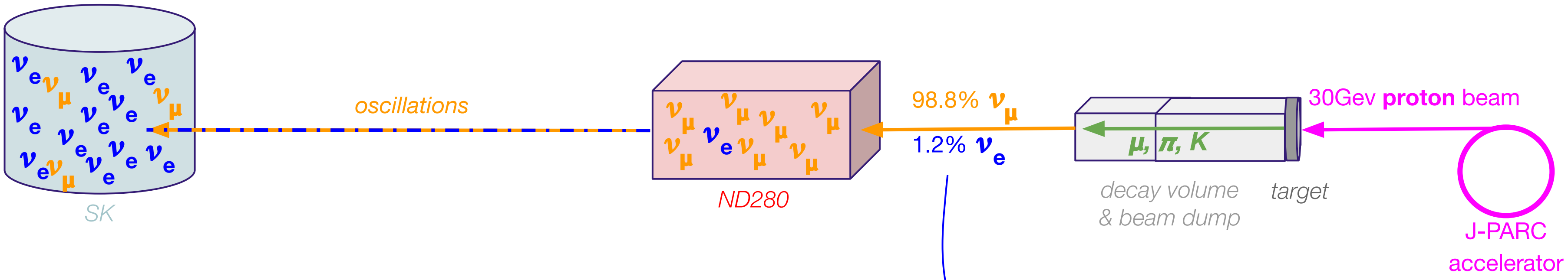
T2K and ν_e (electronic neutrinos)



- ν_e of the beam are an irreducible background

Particle	Decay Products	Branching Fraction (%)
π^+	$\rightarrow \mu^+\nu_\mu$	99.9877
	$\rightarrow e^+\nu_e$	1.23×10^{-4}
K^+	$\rightarrow \mu^+\nu_\mu$	63.55
	$\rightarrow \pi^0\mu^+\nu_\mu$	3.353
	$\rightarrow \pi^0 e^+\nu_e$	5.07
K^0_L	$\rightarrow \pi^-\mu^+\nu_\mu$	27.04
	$\rightarrow \pi^- e^+\nu_e$	40.55
μ^+	$\rightarrow e^+\bar{\nu}_\mu\nu_e$	100

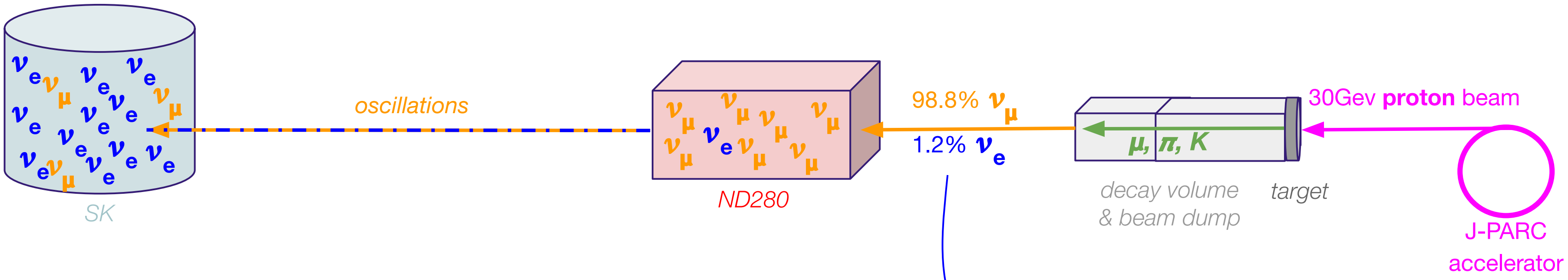
T2K and ν_e (electronic neutrinos)



- ν_e of the beam are an irreducible background
- ν_e are the signal at SK

Particle	Decay Products	Branching Fraction (%)
π^+	$\rightarrow \mu^+\nu_\mu$	99.9877
	$\rightarrow e^+\nu_e$	1.23×10^{-4}
K^+	$\rightarrow \mu^+\nu_\mu$	63.55
	$\rightarrow \pi^0\mu^+\nu_\mu$	3.353
	$\rightarrow \pi^0 e^+\nu_e$	5.07
K^0_L	$\rightarrow \pi^-\mu^+\nu_\mu$	27.04
	$\rightarrow \pi^- e^+\nu_e$	40.55
μ^+	$\rightarrow e^+\bar{\nu}_\mu\nu_e$	100

T2K and ν_e (electronic neutrinos)

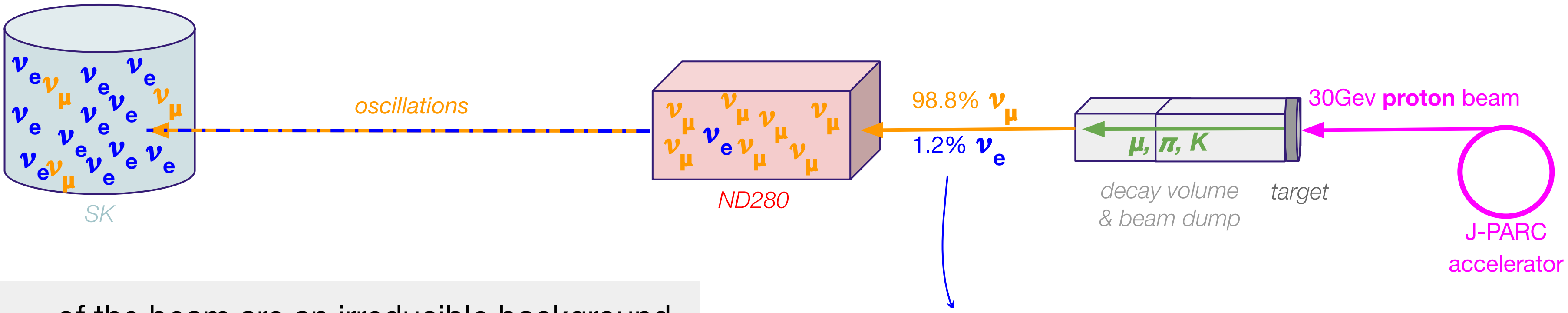


- ν_e of the beam are an irreducible background
- ν_e are the signal at SK

⇒ very important to characterize the ν_e component at ND280

Particle	Decay Products	Branching Fraction (%)
π^+	$\rightarrow \mu^+\nu_\mu$	99.9877
	$\rightarrow e^+\nu_e$	1.23×10^{-4}
K^+	$\rightarrow \mu^+\nu_\mu$	63.55
	$\rightarrow \pi^0\mu^+\nu_\mu$	3.353
	$\rightarrow \pi^0 e^+\nu_e$	5.07
K^0_L	$\rightarrow \pi^-\mu^+\nu_\mu$	27.04
	$\rightarrow \pi^- e^+\nu_e$	40.55
μ^+	$\rightarrow e^+\bar{\nu}_\mu\nu_e$	100

T2K and ν_e (electronic neutrinos)



- ν_e of the beam are an irreducible background
- ν_e are the signal at SK

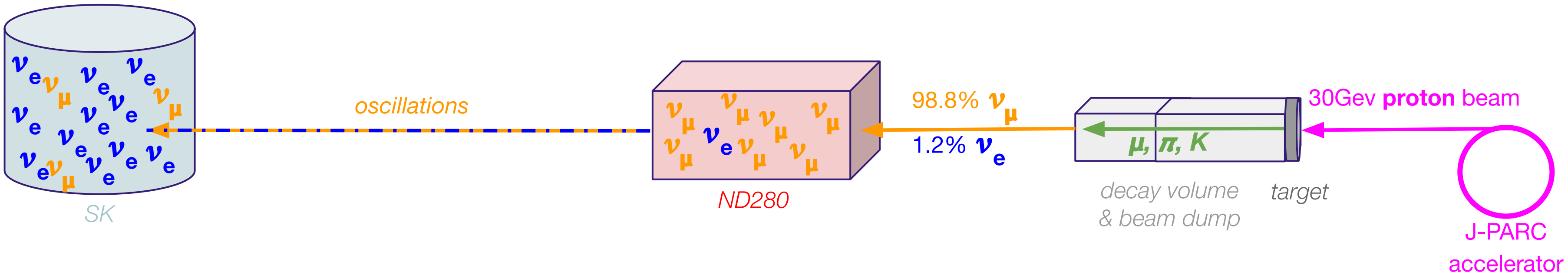
⇒ very important to characterize the ν_e component at ND280

⇒ need Particle Identification for ν_e selection

⇔ identifying ***e*** vs ***μ***, ***γ***, ***π***, ***p***

Particle	Decay Products	Branching Fraction (%)
π^+	$\rightarrow \mu^+\nu_\mu$	99.9877
	$\rightarrow e^+\nu_e$	1.23×10^{-4}
K^+	$\rightarrow \mu^+\nu_\mu$	63.55
	$\rightarrow \pi^0\mu^+\nu_\mu$	3.353
	$\rightarrow \pi^0 e^+\nu_e$	5.07
K^0_L	$\rightarrow \pi^-\mu^+\nu_\mu$	27.04
	$\rightarrow \pi^- e^+\nu_e$	40.55
μ^+	$\rightarrow e^+\bar{\nu}_\mu\nu_e$	100

T2K and ν_e (electronic neutrinos)



- ν_e of the beam are an irreducible background
- ν_e are the signal at SK

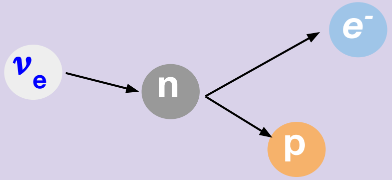
⇒ very important to characterize the ν_e component at ND280

⇒ need Particle Identification for ν_e selection

⇔ identifying ***e*** vs $\boldsymbol{\mu, \gamma, \pi, p}$

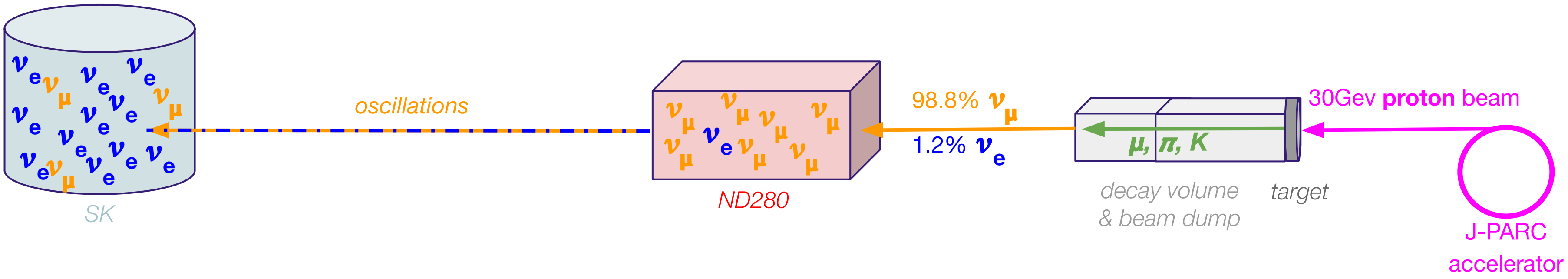
how to detect uncharged particles (like ν) in a detector that can only detect charged ones?

→ by detecting the charged particle from ν interaction* in the detector:



*called Charged Current (CC) interaction

T2K and ν_e (electronic neutrinos)



- ν_e of the beam are an irreducible background
- ν_e are the signal at SK

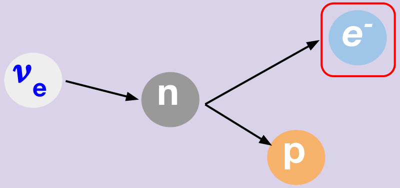
⇒ very important to characterize the ν_e component at ND280

⇒ need Particle Identification for ν_e selection

⇔ identifying ***e*** vs μ, γ, π, p

how to detect uncharged particles (like ν) in a detector that can only detect charged ones?

→ by detecting the charged particle from ν interaction* in the detector:



*called Charged Current (CC) interaction

Why we need to improve the ν_e selection

Selection of a particle =

Particle **ID**entification

1) Define a **PID** method to identify your particle (***e***)

Using simulations of particle:

2) count how many true ***e*** are selected by the method
3) count how many *other particles* are selected by the method

This allows to define:

- **efficiency** = how much true ***e*** we selected
- **purity** = how much background is kept in the selected particles

Important! goes into the T2K uncertainties
→ need method which gives best eff & pur

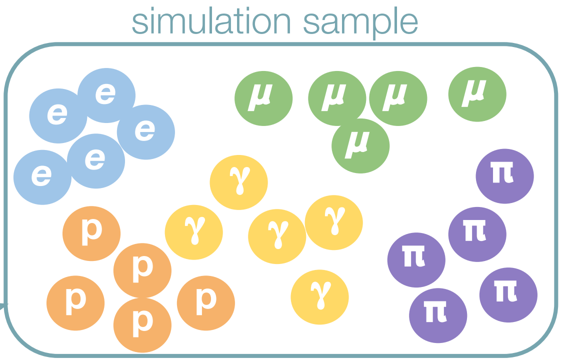
Why we need to improve the ν_e selection

Selection of a particle =

Particle IDentification

1) Define a **PID** method to identify your particle (***e***)

Using simulations of particle:



2) count how many true ***e*** are selected by the method

3) count how many *other particles* are selected by the method

This allows to define:

- **efficiency** = how much true ***e*** we selected
- **purity** = how much background is kept in the selected particles

Important! goes into the T2K uncertainties

→ need method which gives best eff & pur

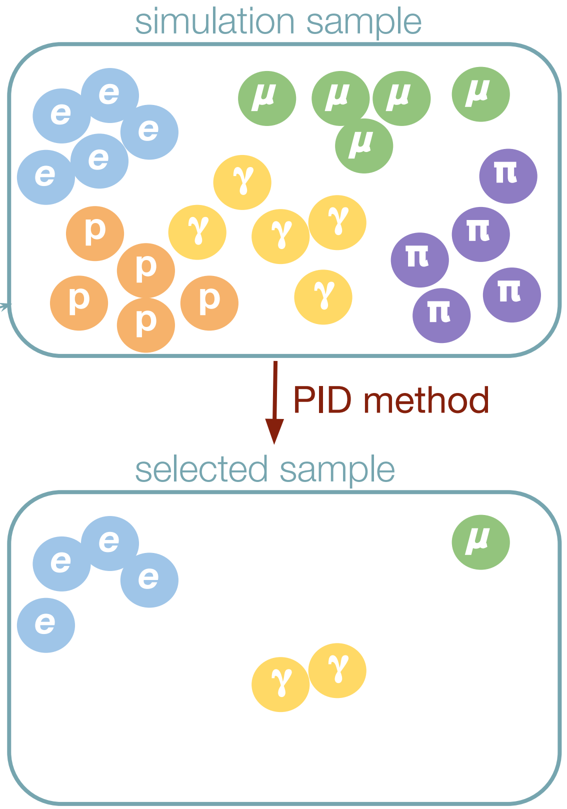
Why we need to improve the ν_e selection

Selection of a particle =

Particle **ID**entification

1) Define a **PID** method to identify your particle (***e***)

Using simulations of particle:



2) count how many true ***e*** are selected by the method

3) count how many *other particles* are selected by the method

This allows to define:

- **efficiency** = how much true ***e*** we selected
- **purity** = how much background is kept in the selected particles

Important! goes into the T2K uncertainties
→ need method which gives best eff & pur

Why we need to improve the ν_e selection

Selection of a particle =

Particle IDentification

1) Define a **PID** method to identify your particle (***e***)

Using simulations of particle:

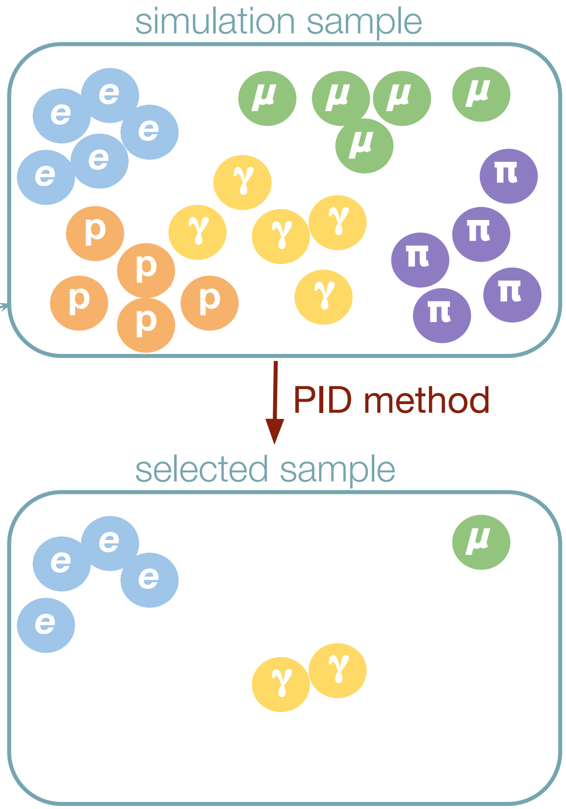
2) count how many true ***e*** are selected by the method

3) count how many *other particles* are selected by the method

This allows to define:

- **efficiency** = how much true ***e*** we selected
- **purity** = how much background is kept in the selected particles

Important! goes into the T2K uncertainties
→ need method which gives best eff & pur



$$\text{efficiency} = \frac{4 \text{ selected } e^-}{5 \text{ simulated } e^-} = 80\%$$

$$\text{purity} = \frac{4 \text{ selected } e^-}{7 \text{ selected in tot}} = 57\%$$

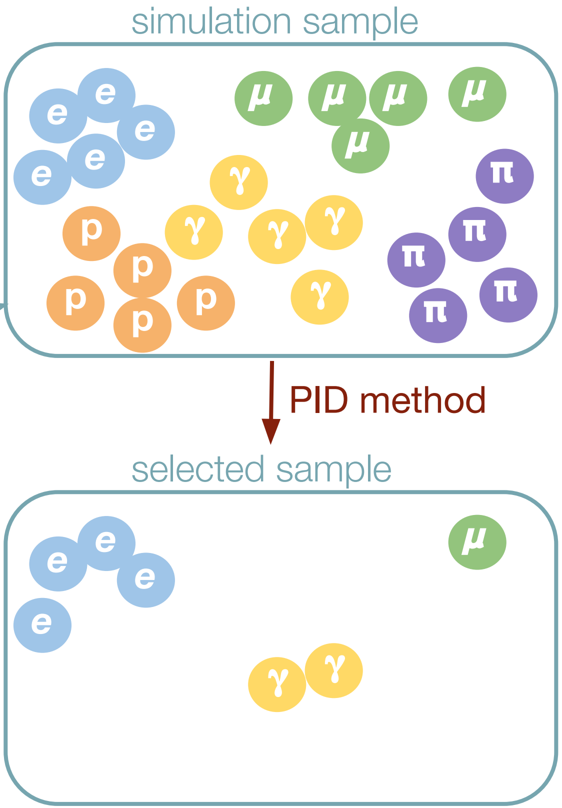
Why we need to improve the ν_e selection

Selection of a particle =

Particle IDentification

1) Define a **PID** method to identify your particle (***e***)

Using simulations of particle:



2) count how many true ***e*** are selected by the method

3) count how many *other particles* are selected by the method

This allows to define:

- **efficiency** = how much true ***e*** we selected
- **purity** = how much background is kept in the selected particles

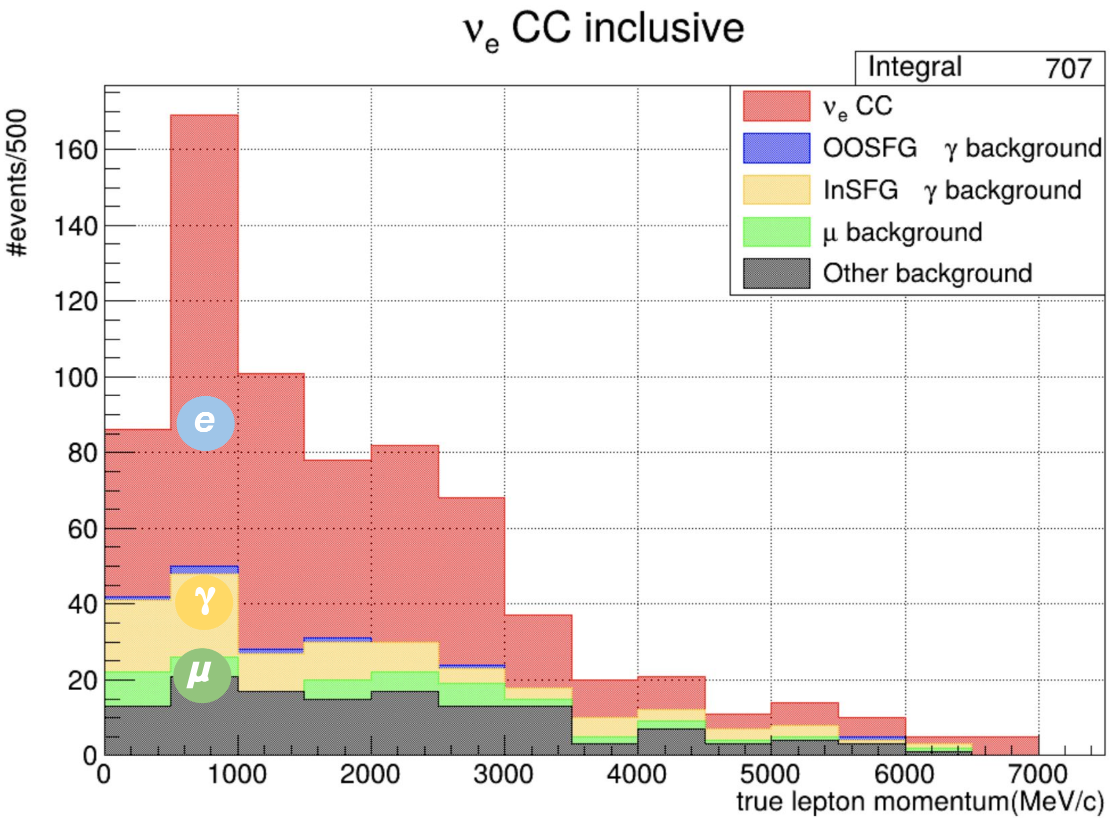
Important! goes into the T2K uncertainties
→ need method which gives best eff & pur

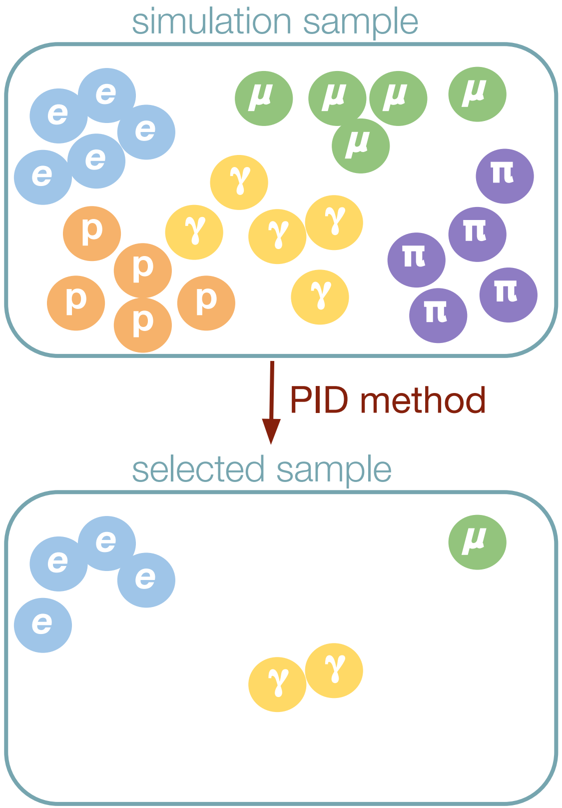
with current method:

efficiency = 18.6%

purity = 68%

Why we need to improve the ν_e selection



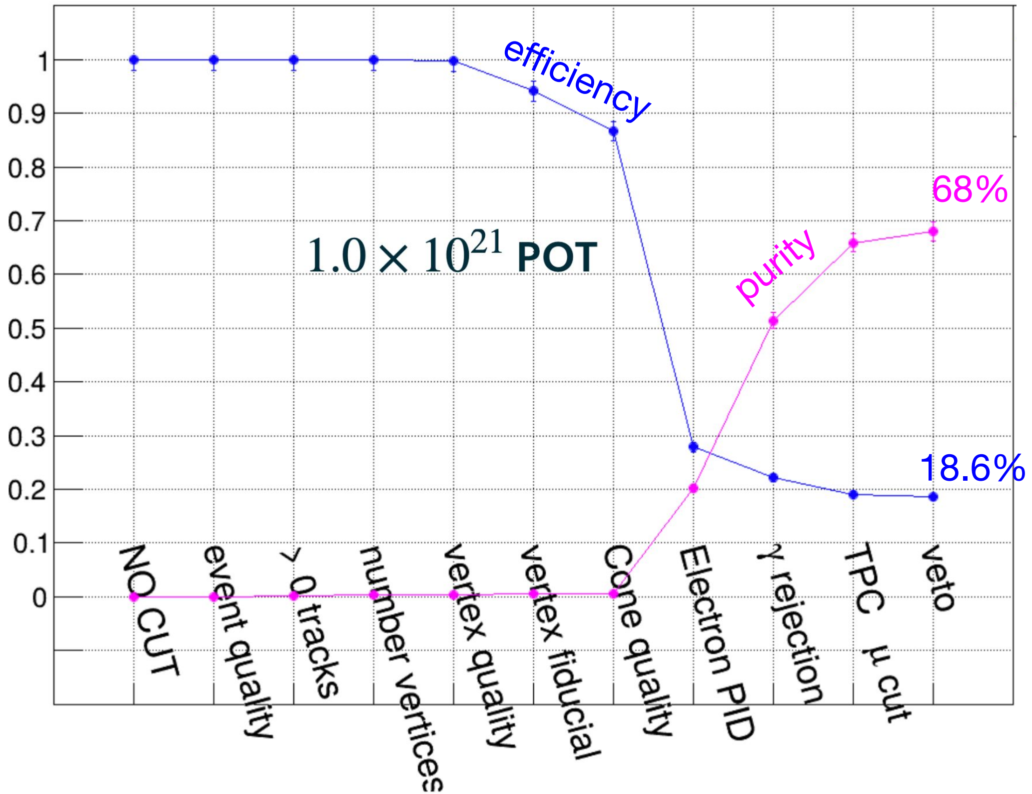


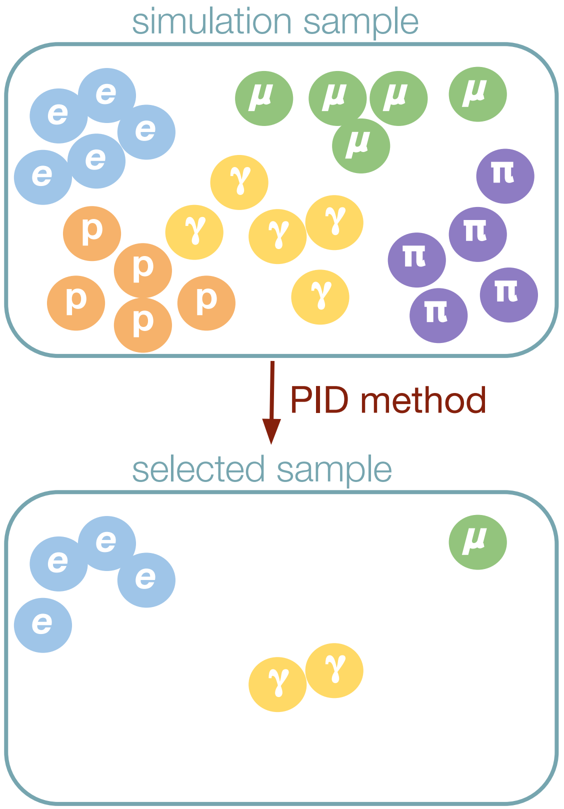
with current method:

efficiency = 18.6%

purity = 68%

Why we need to improve the ν_e selection





with current method:

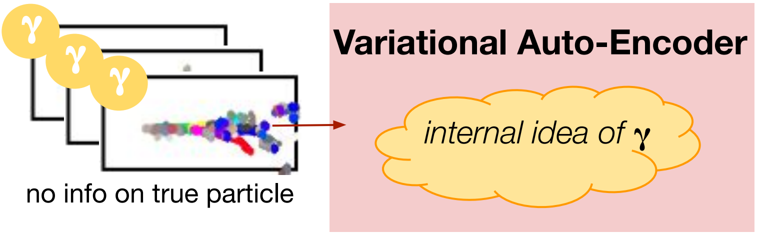
efficiency = 18.6%

purity = 68%

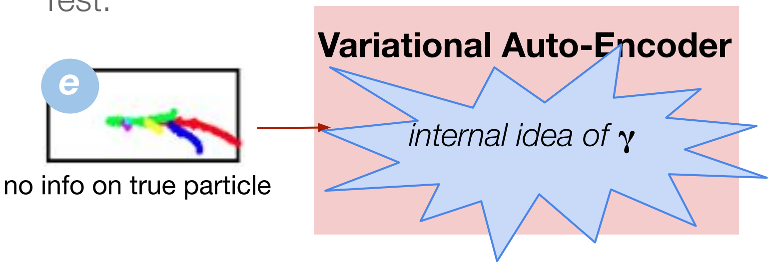
2 new PID methods with ML: Transformer & Variational Auto-Encoder

VAE: un-supervised ML

Training:



Test:

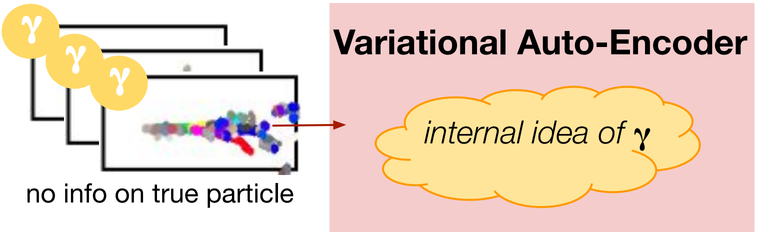


performs worse in controlled, labeled settings but should be better to deals with real data variability

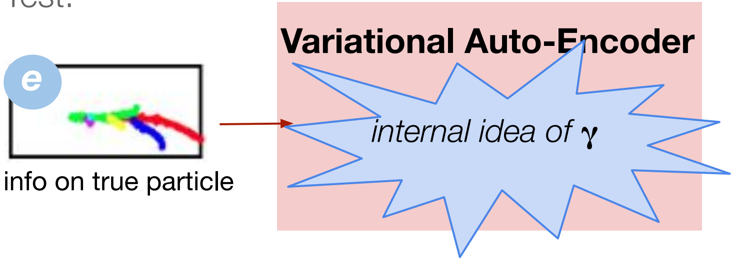
2 new PID methods with ML: Transformer & Variational Auto-Encoder

VAE: un-supervised ML

Training:



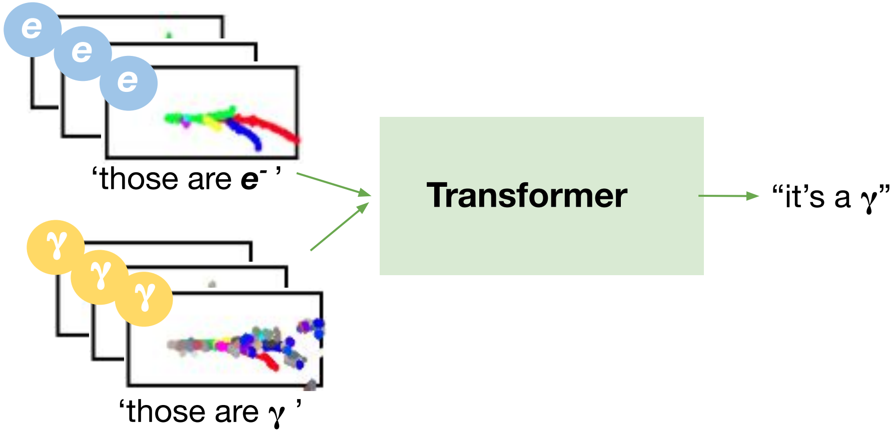
Test:



performs worse in controlled, labeled settings but should be better to deals with real data variability

Transformer: supervised ML

Training & test:



performs well in controlled, labeled settings but will probably struggle more with real data variability

The Transformer PID method

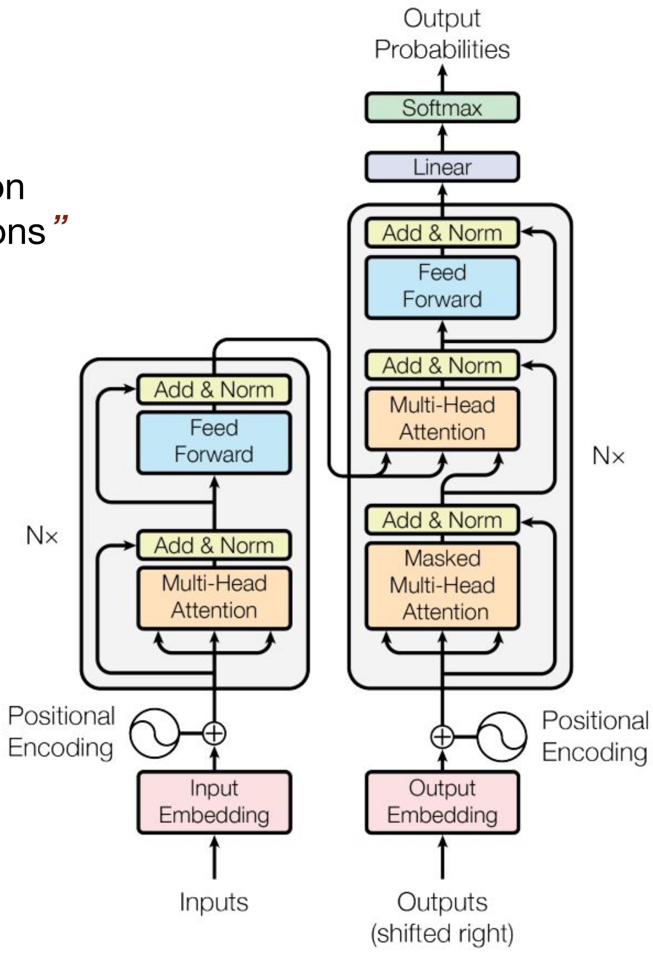
What is a Transformer

"**Transformer** = a new simple network architecture, based solely on attention mechanisms, dispensing with recurrence and convolutions"

[*Attention Is All You Need*]

Key ideas:

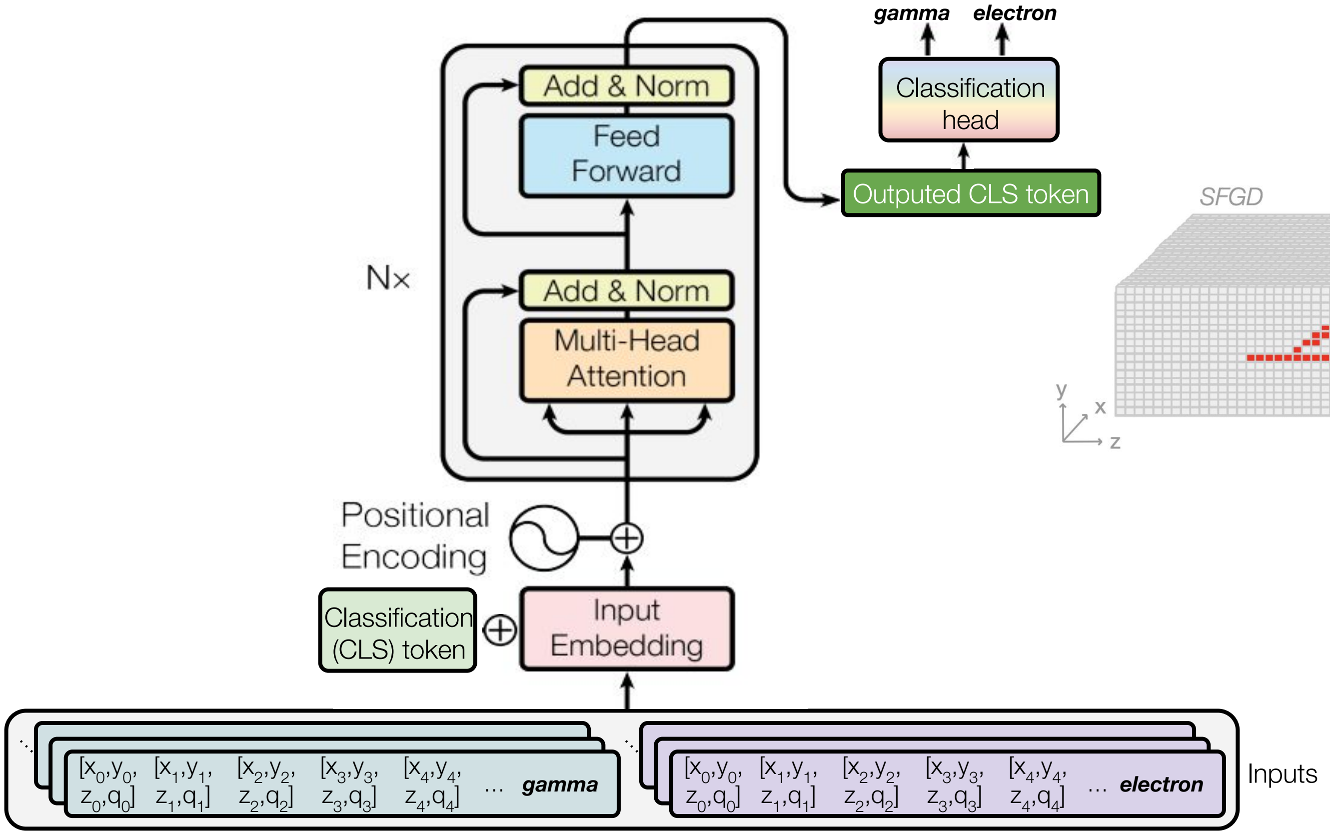
- **takes sequences as inputs**
- **attention mechanisms**: allows the model to focus on the most relevant parts of the input sequence when producing an output
- **multi-head attention:** attending to different parts of the input in parallel allows for the model to capture multiple types of relationship
- **parallelizable architecture**

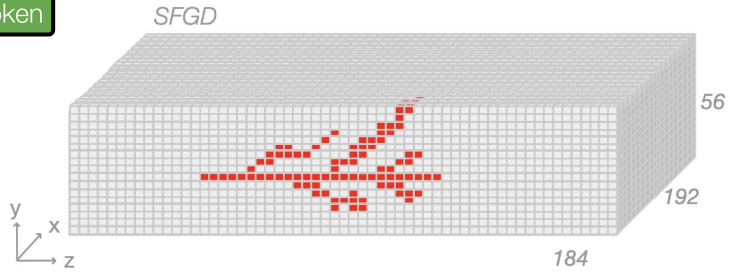


standard BERT architecture

The Transformer PID method

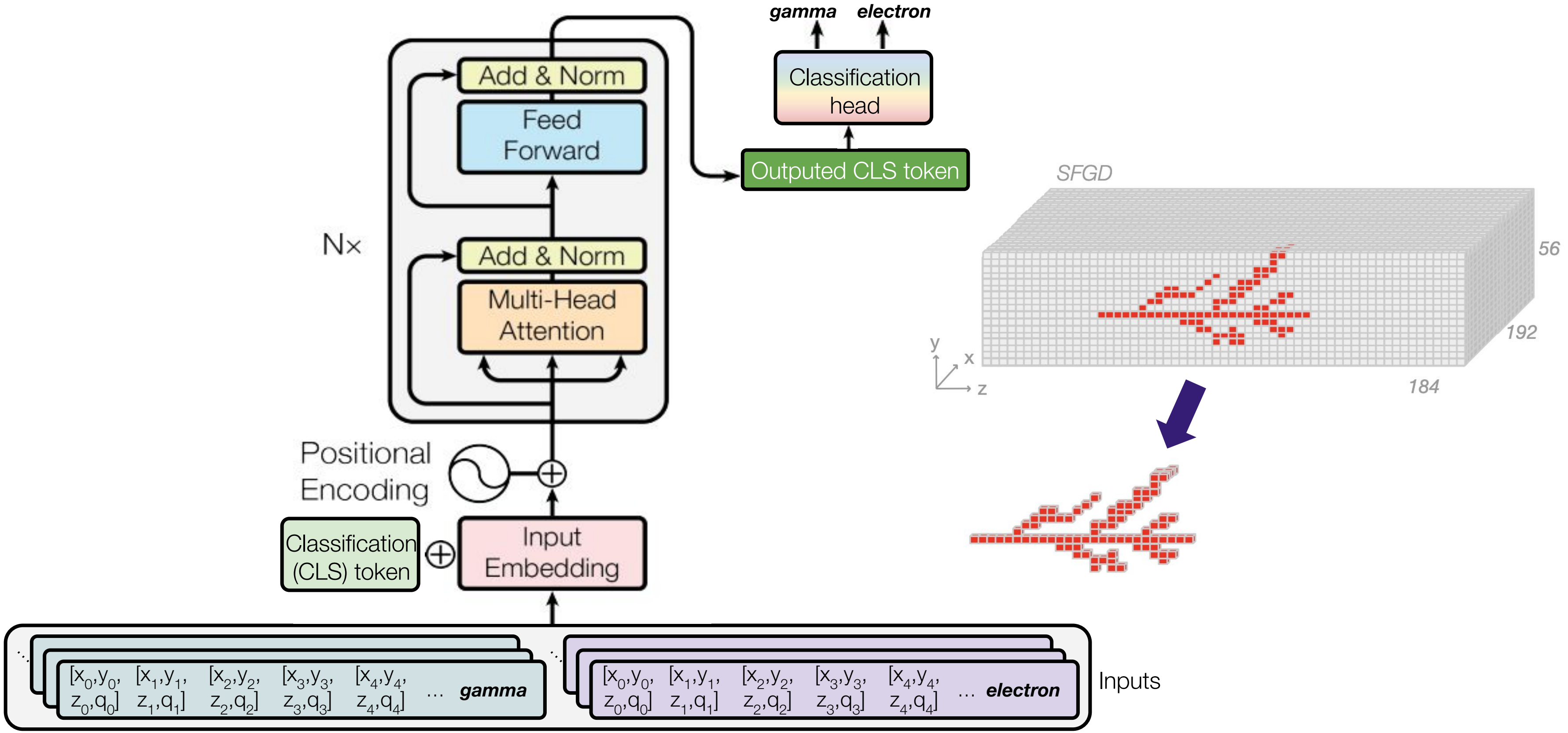
How we can apply it for the ν_e selection





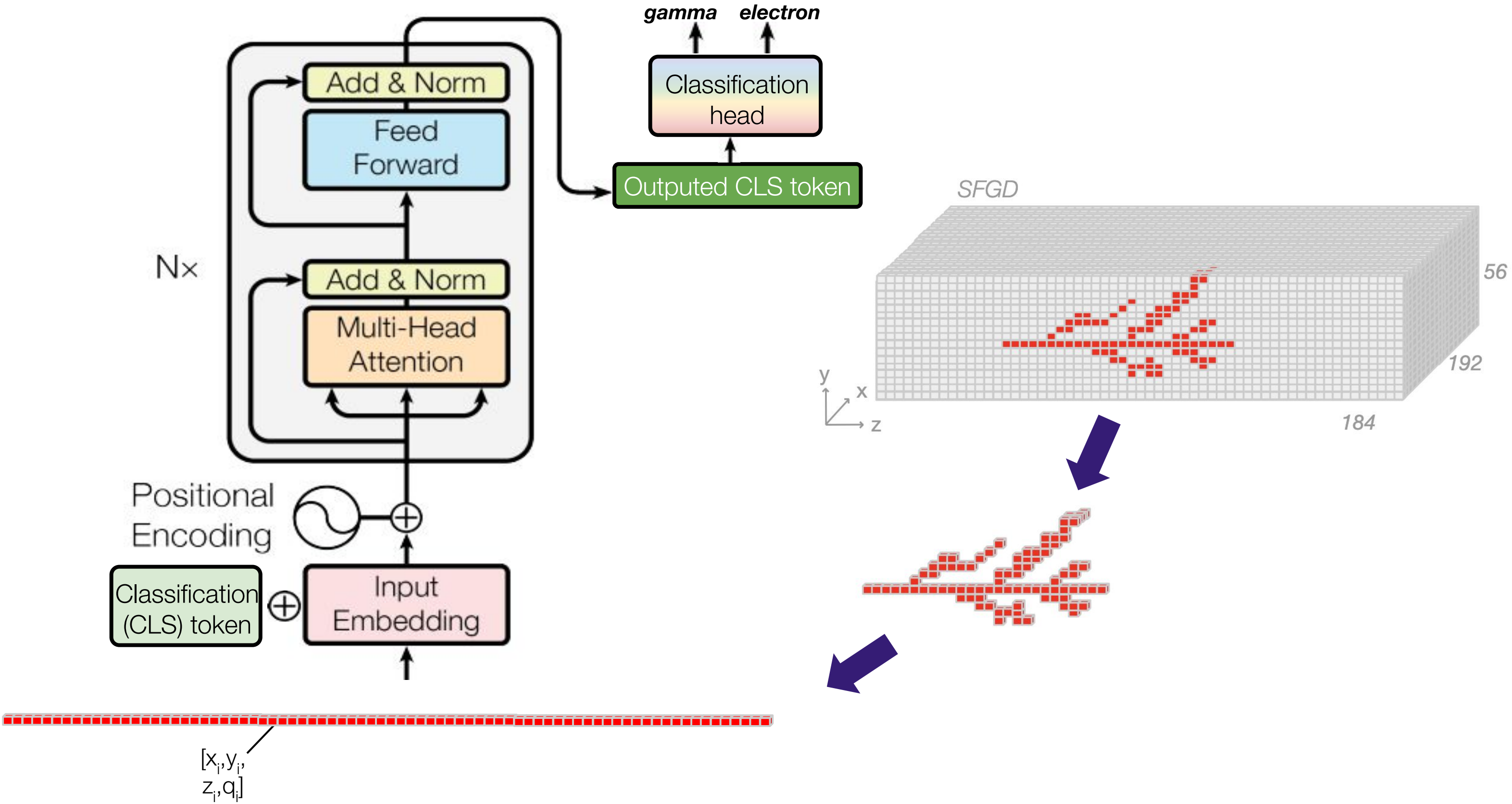
The Transformer PID method

How we can apply it for the ν_e selection



The Transformer PID method

How we can apply it for the ν_e selection



The Transformer PID method
Vision Transformer

- Problem: some sequences of hits are **too long**:

⇒ not computationally efficient as complexity of transformer ~ $O(\text{sequence length}^2)$

⇒ training would take too long

- Solution: use the principle behind **vision transformers**

The Transformer PID method

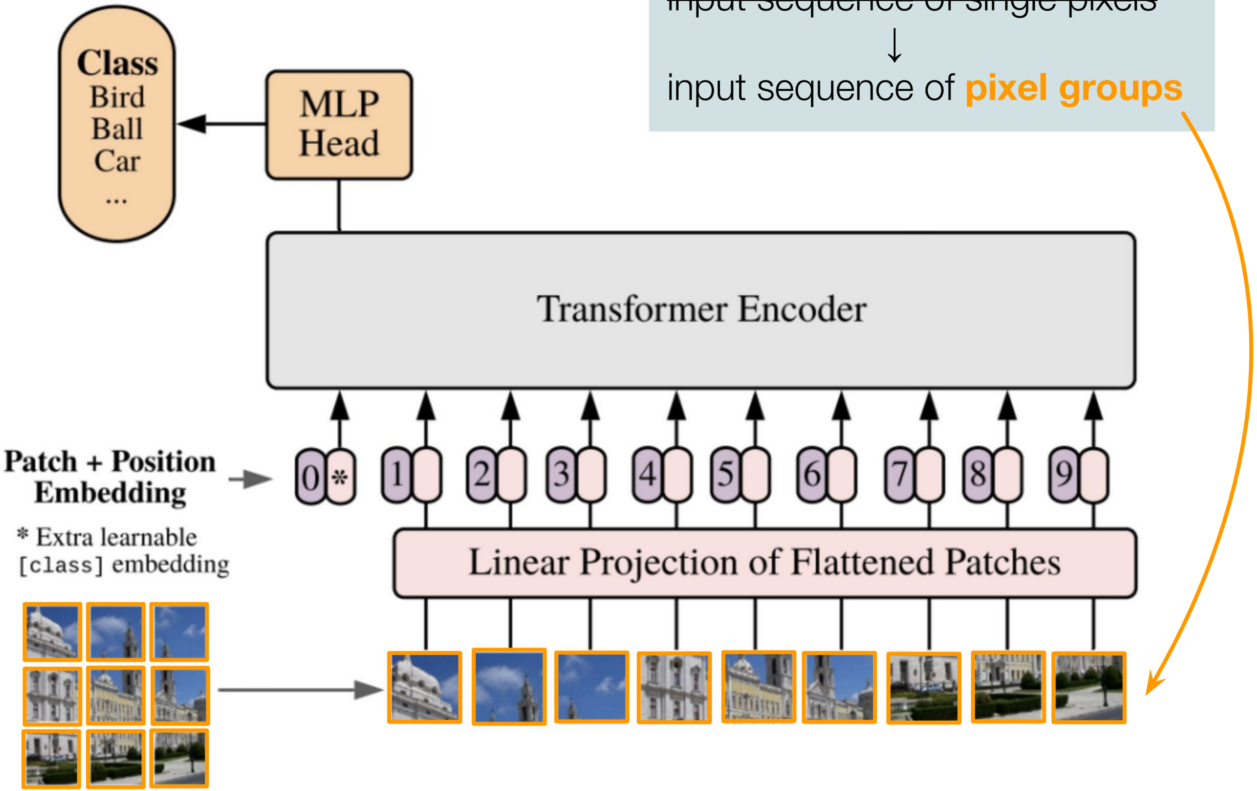
Vision Transformer

- Problem: some sequences of hits are **too long**:
⇒ not computationally efficient as complexity of transformer ~ O(sequence length^2)

⇒ training would take too long

- Solution: use the principle behind **vision transformers**

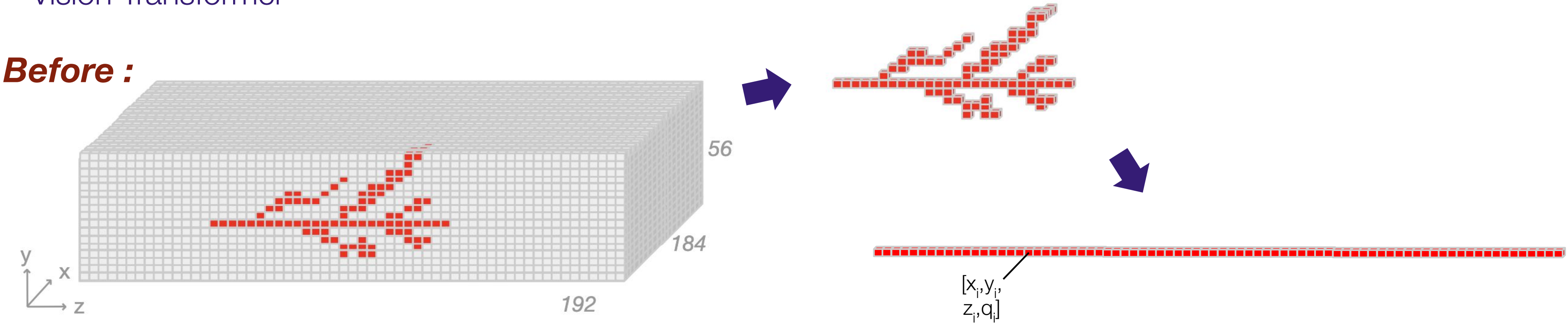
~~input sequence of single pixels~~
↓
input sequence of **pixel groups**



The Transformer PID method

Vision Transformer

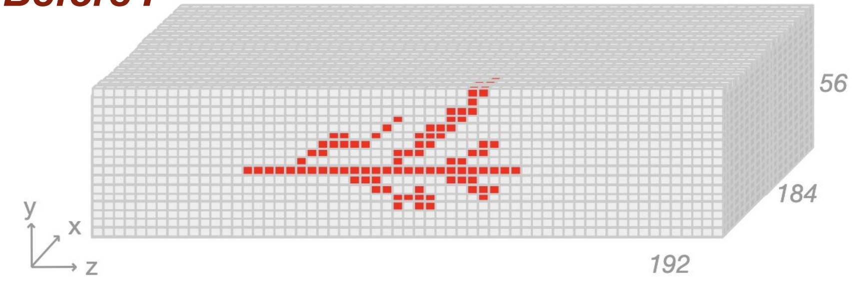
Before :

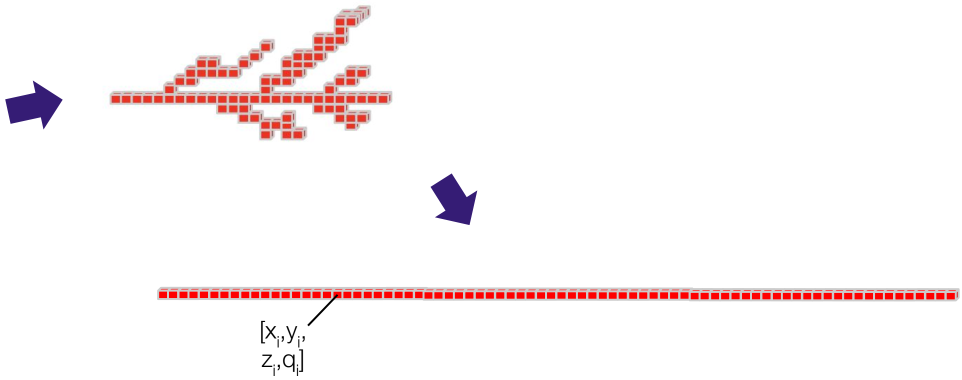


The Transformer PID method

Vision Transformer

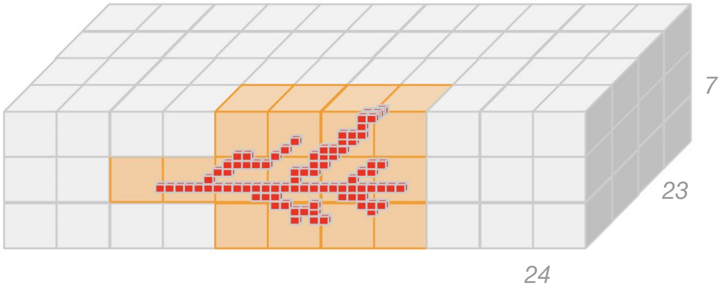
Before :

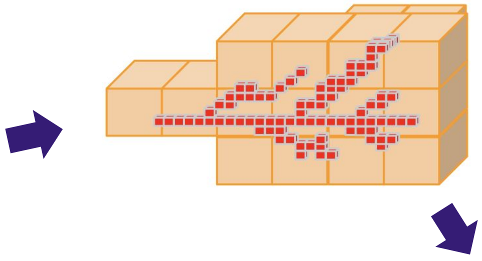




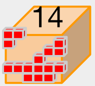
Now (ViT) :

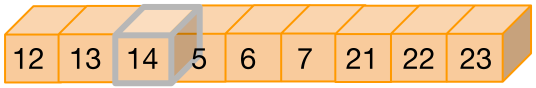
e.g: create parent cubes of size 8x8x8





each parent cube holds the information of its child voxels: relative positions & charge



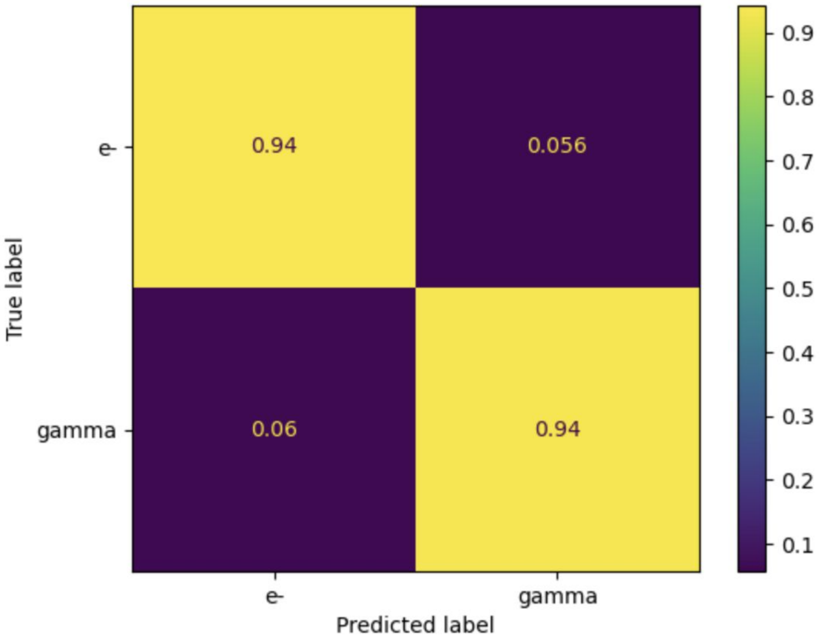


positional encoding of the parent cubes in the parent grid

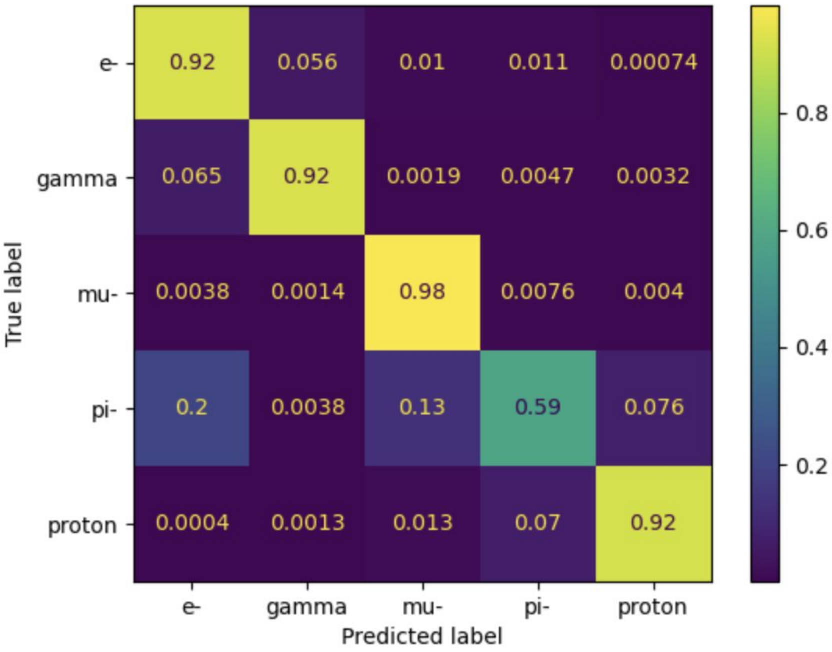
The Transformer PID method

First performances

*model trained for **e**/γ separation*



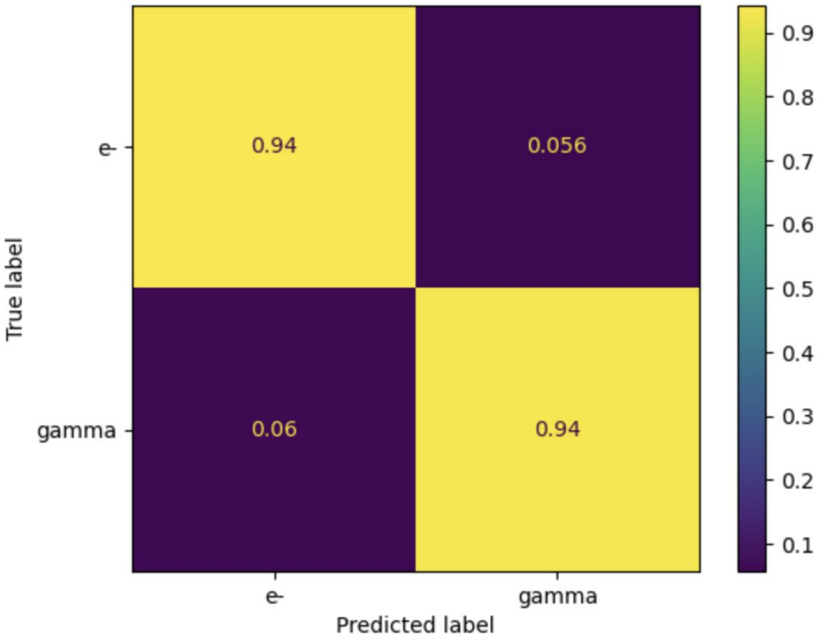
*model trained for **e**/γ/μ/π/**p** separation*

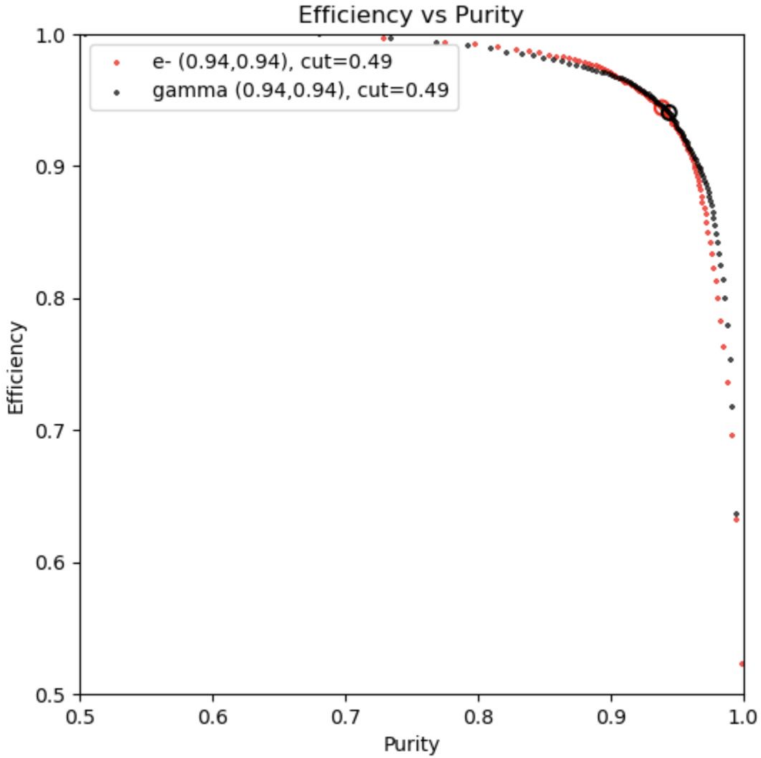


The Transformer PID method

First performances

*model trained for **e**/γ separation*

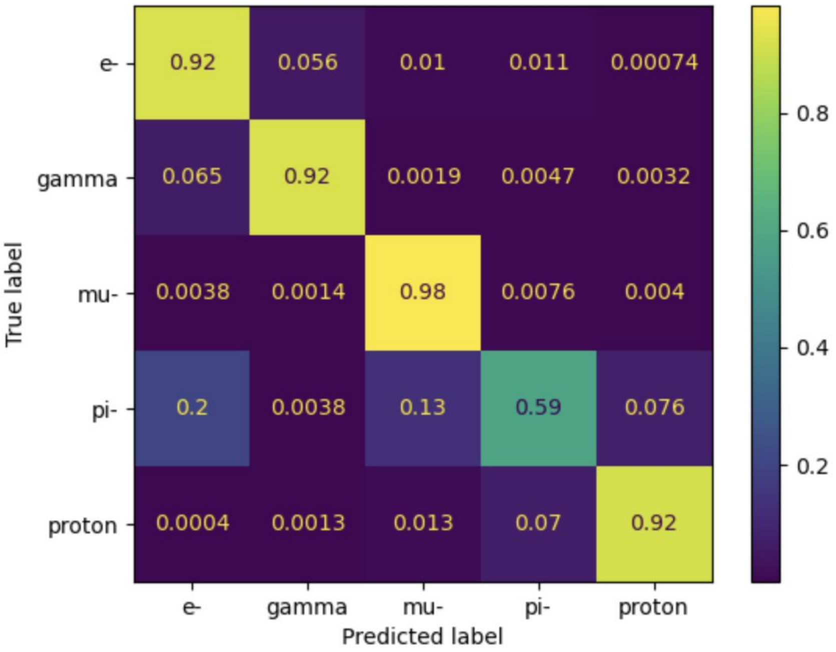


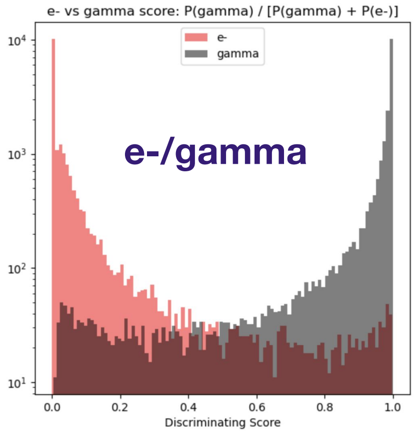


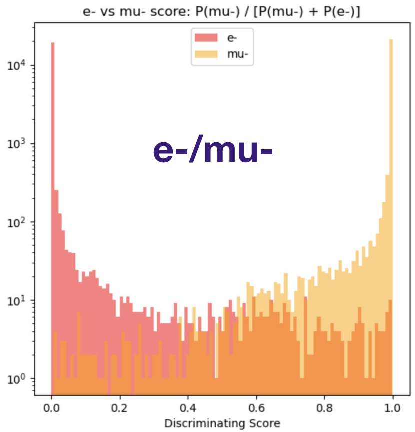
The Transformer PID method

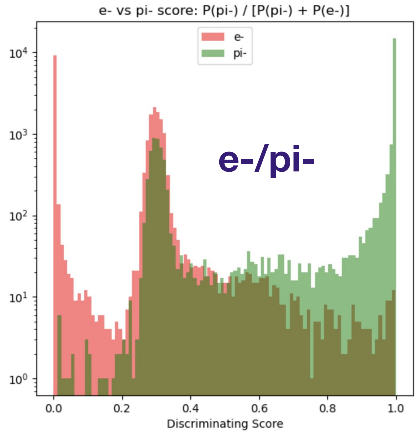
First performances

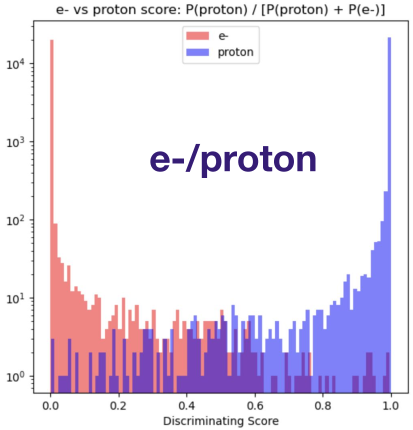
*model trained for **e**/**γ**/**μ**/**π**/**p** separation*







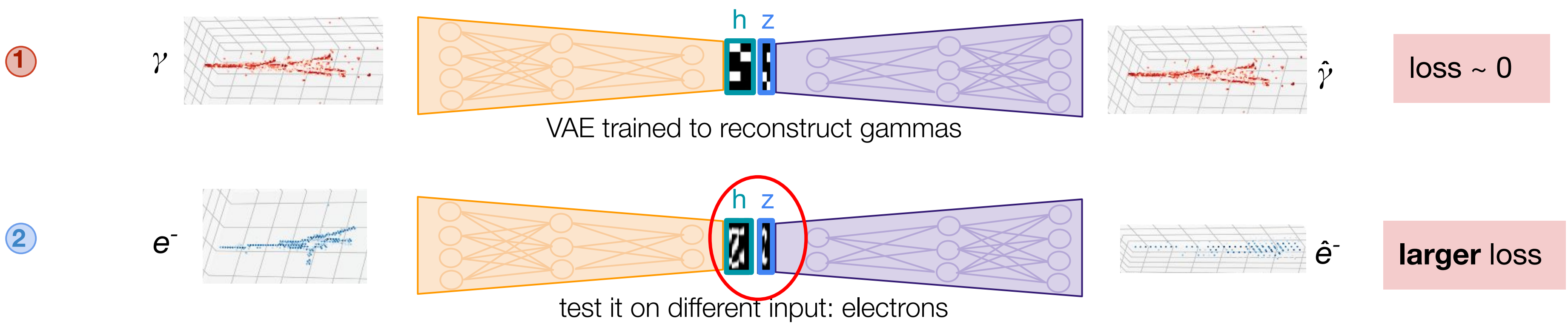




A glanced at the Variational Auto-Encoder method

Key idea: anomaly detection to distinguish *e-* (anomalies) out of γ (trained on)

1. Train the VAE with many gamma conversion in the SFGD: it will learn to reconstruct the gamma events
2. Test it on e- events: the reconstruction should not be as good → anomaly in the loss function



A glanced at the Variational Auto-Encoder method

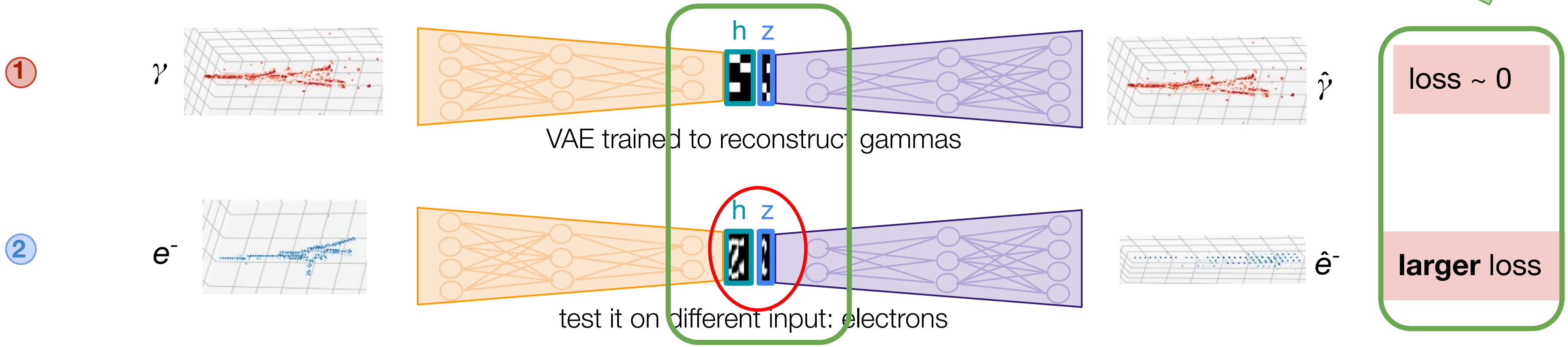
2 distinction metrics to analyse

latent space

regression loss

Key idea: anomaly detection to distinguish *e-* (anomalies) out of γ (trained on)

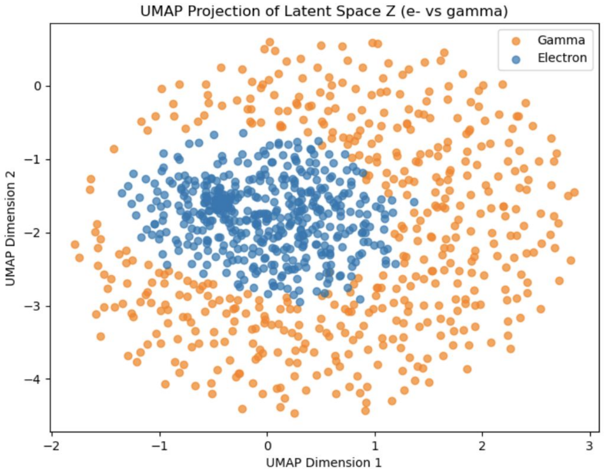
1. Train the VAE with many gamma conversion in the SFGD: it will learn to reconstruct the gamma events
2. Test it on e- events: the reconstruction should not be as good → anomaly in the loss function



A glanced at the Variational Auto-Encoder method

Separation of e and γ

latent space: dimensionality reduction with UMAP



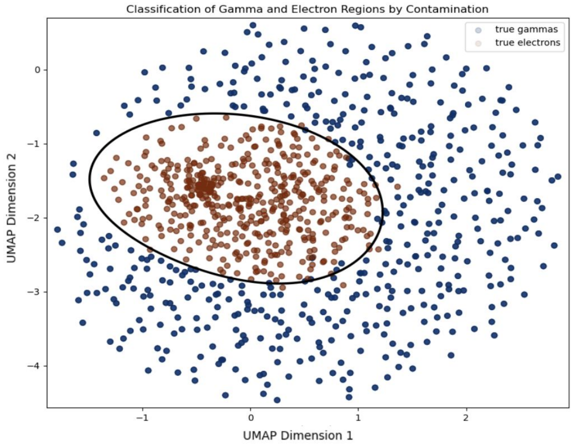
z-distrib for one event: vector of size (1,1024)

→ reduce it to (1,2) using UMAP (dimensionality reduction algo)

→ plot all tested event on this 2D map

Separation of e and γ

separation in the 2D latent space for electron selection



1) compute the **contamination** of each point
2) low-contamination points are fed into a **SVC with an RBF** kernel
3) the SVC creates a **delimitation in this 2D space**
4) **efficiencies** and **purities** using the electron/gamma regions can be computed:

e- efficiency: 98.73%
e- purity: 97.91%
γ efficiency: 97.89%
γ purity: 98.72%

BUT: this 2D umap was *transformed* on the same sample used for *fitting* the 2D umap: those eff,pur are only **upper limits**

Summary

- It's necessary to c**haractize precisely the ν_e background** at the near detector
- New **methods are being developed** for this matter: a **Transformer** and a **VAE**
- The developpement of the **Transformer in the T2K analysis software** is done!
  (first complex ML algo to be implemented there)
- The **Transformer** training shows **promising PID results**: other improvements are needed
- Some **VAE upgrades are on-going**, then new analyses are too come

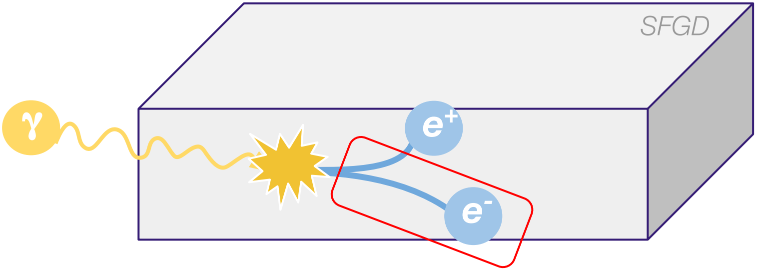
Next steps:

- compare those 2 methods with the old ones on the same samples
- control sample

Back-ups

The gamma background

gamma event



CC electronic neutrino event