НИЯУ МИФИ · 2026

Реконструкция треков частиц в магнитном поле с помощью графовой нейронной сети (GNN) для эксперимента BM@N

Комплекс NICA (ОИЯИ)

Студент: Д.Н. Джавадов

Научный руководитель: к.т.н. К.В. Герценбергер

Основные задачи:

Реализовать работающий конвейер обработки данных:

- Реализовать физически мотивированное построение графа;
- Вычислить и максимизировать базовые метрики качества на уровне рёбер графа;
- Провести локальную оптимизацию разработанного решения;
- С использованием пакета Optuna выполнить подбор гипперпараметров модели;
- Подобрать лучшую функцию потерь;
- Оценить качество модели.

Цель работы:

разработка и экспериментальная оценка графовой нейронной сети с механизмом внимания (GATv2) для реконструкции треков заряженных частиц в магнитном поле детектора BM@N в условиях большого количества ложных треков.

ФИЗИЧЕСКАЯ ПОСТАНОВКА ЗАДАЧА

Эксперимент BM@N (NICA, ОИЯИ): столкновения тяжелых ионов с фиксированной мишенью, изучение свойств ядерной материи при экстемальных плотностях

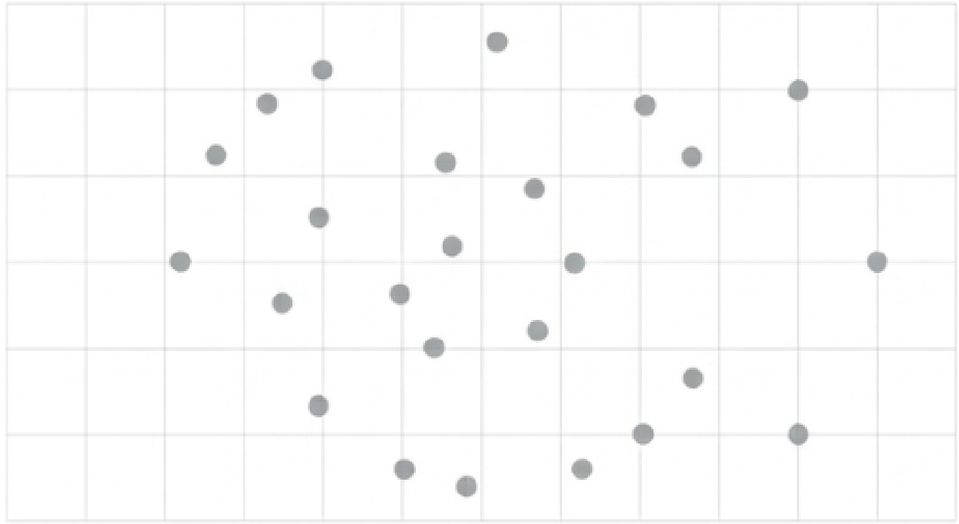
Заряженная частица в поле ***B*** движется по спирали - её радиус кривизны:

$$R = \frac{p_T}{0.3B\,|q|}$$

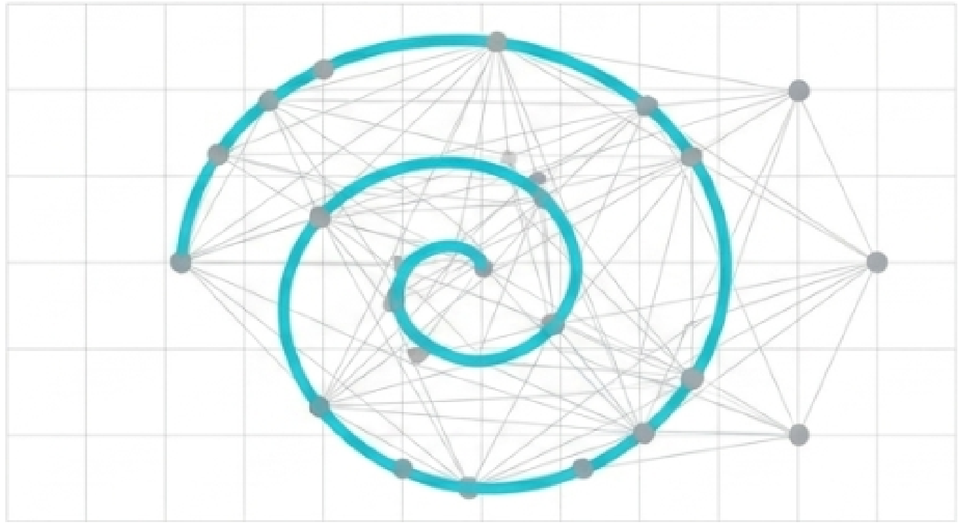
Задача: по облаку хитов (координатных сигналов в детекторах) восстановить исходные траектории частиц.

Трек частицы – последовательность хитов, оставленных в слоях детекторной системы

Исходное облако хитов



Восстановление траекторий заряженных частиц



Почему это задача на графе и почему GNN

Хит – вершина графа

Каждый сигнал в детекторе – узел графа с набором признаков

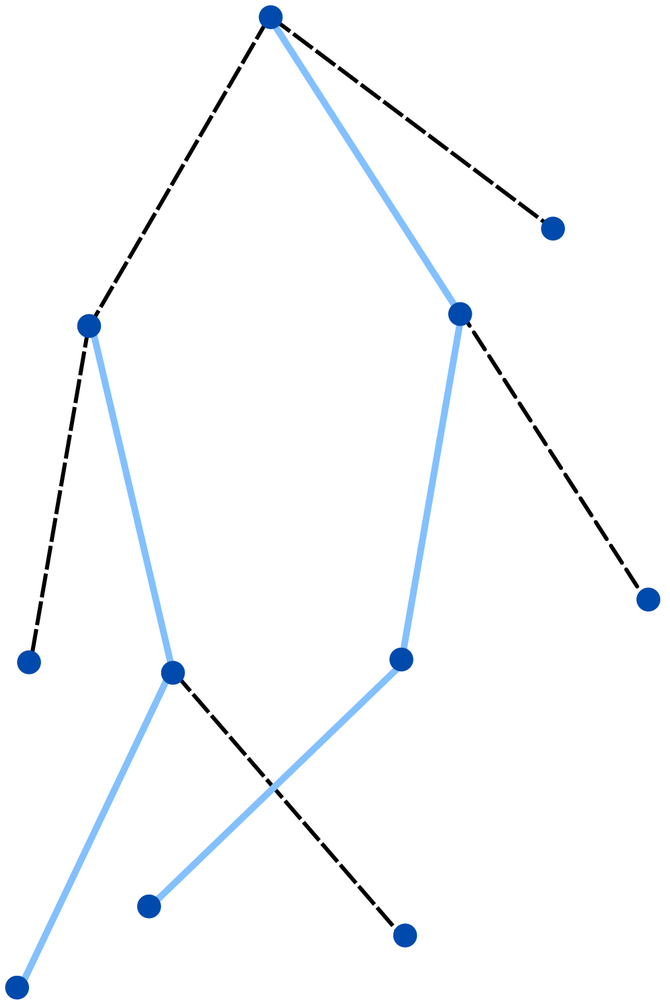
Возможная связь – рёбро-кандидат

Пару хитов, которая физически может принадлежать одному треку

GNN классифицирует рёбра

Сеть учитывает не только сам хит, но и контекст соседних хитов в графе

$$y_{ij} = \begin{cases} 1, & \text{если хиты } i \text{ и } j \text{ – один трек} \\ 0, & \text{иначе} \end{cases}$$



истинное ребро $y_{ij} = 1$ ложное ребро $y_{ij} = 0$

Смоделированный датасет BmnRoot

10 000

событий

0.7 Тл

магнитное поле

3.0 Гэв

энергия/нуклон

XeCsI

пучок + мишень

Параметры хита

x, y, z	координаты хита частицы в детекторе [см]
px, py, pz	импульс частицы в данном хите [ГэВ/см]
station, module	топология детектора
trackID	метка трека (для обучения)
particle_charge	заряд частицы

Обучение vs инференс:

на инференсе импульс и заряд неизвестны – используются только геометрические признаки (x, y, z, station, module), как в реальном эксперименте

EdgeBuilder: двухэтапная фильтрация рёбер-кандидатов

1

Быстрая фильтрация

порядок станций,
Δstation ≤ 2,
предел по расстоянию

→

2

Физическая проверка

совместимость
кривизны траектории
с (pT, q)

→

3

Топ-K кандидатов

ранжирование
по score
совместимости

→

4

Граф готов к GNN

18 признаков
на каждое
ребро

Momentum dropout: в 30% случаев на обучении импульс искусственно скрывается – модель учится строить граф по голой геометрии , как на инференсе

GATv2: сеть сама учится, каким соседям доверять

dataset | edge_builder | **gat_model** | trainer | predictor | track_builder

Идея механизма внимания:

Каждый хит связан с несколькими хитами-кандидатами, среди которых большинство ложные рёбра

Сеть назначает каждой связи вес внимания αij – насколько ей доверять

Модель сама подавляет ложные связи и усиливает физически совместимые

Ключевое преимущество GATv2: коэффициент внимания вычисляется по обоим концам ребра одновременно — это устраняет проблему статического внимания классического GAT, где ранжирование соседей не зависело от запрашивающей вершины.

Обучение – решение проблемы дисбаланса классов

Дисбаланс классов: количество истинных рёбер в ~6 раз меньше ложных

BCE (Binary Cross-Entropy) с весом
val_loss = 0.0257

$$L_{BCE} = -\frac{1}{N}\sum_{i=1}^{N}[w_+ \cdot y_i \log \hat{p}_i + (1-y_i)\log(1-\hat{p}_i)]$$

- Взвешивание положительного класса (pos_weight (w+) компенсирует дисбаланс
- Грубый подход: одинаковый штраф для всех ложных срабатываний

Focal Loss
val_loss = 0.00028

$$L_{FL} = -\frac{\alpha}{N}\sum_{i=1}^{N}(1-\hat{p}_i)^{\gamma} \cdot y_i \log \hat{p}_i$$

- Снижает вклад «лёгких» примеров, фокусируется на сложных граничных рёбрах
- Автоматически адаптируется к уверенности модели

Подбор гиперпараметров: Optuna (TPE)

Двухэтапная стратегия: (1) Быстрый поиск на подвыборке (2000 событий, 85 испытаний) → нахождение перспективной области. (2) Точный поиск на полном датасете из лучших параметров.
MedianPruner для досрочного прерывания неперспективных испытаний.

Качество классификации рёбер

Метрика	Промежуточные результаты	Данный момент	Δ
Accuracy	0.940	**0.996**	+0.056
Precision	0.743	**0.989**	+0.246
Recall	0.852	**0.998**	+0.146
F1- score	0.794	**0.987**	+0.193
AUC-ROC	0.974	**0.9998**	+0.026

Track Purity

0.9975

доля корректно собранных треков среди найденных

Главный вклад в улучшение:

1 **Добавление псевдобыстроты η**
в признаки рёбер

2 **Переход на Focal Loss**
вместо взвешенной BCE

3 **Точный подбор гиперпараметров**
Optuna, двухэтапная стратегия

Хорошие рёбра - ещё не значит хорошие треки

Проблема

Несмотря на отличные edge-level метрики, **Track Efficiency** = 0.267 — низкая. Хорошая классификация рёбер ≠ хорошая сборка треков.

Оптимизация производительности

C++/CuPy гибрид

Пользовательские CUDA-ядра, pybind11 для критических модулей

Оффлайн-препроцессинг

графы сохраняются в .pt заранее, исключая повторные вычисления

KD-tree / ball tree

Замена полного перебора пар $O(N^2)$ на пространственный поиск

Numba / NumPy векторизация

Перевод горячих функций с Python-циклов на векторизованные операции

Улучшение track-level качества

Анализ причин низкой Track Efficiency

Детальное исследование влияния факторов на восстановление треков

Доработка постобработки

Улучшение алгоритмов сборки треков из классифицированных рёбер

Расширение признаков и данных

Увеличение объёма обучающей выборки, новые физические признаки

Конвейер реконструкции треков на основе GATv2 реализован и работает

1. Реализован полный конвейер от сырых данных до сборки треков, включая физически мотивированное построение графа

2. Систематический подбор гиперпараметров (Optuna) и сравнение функций потерь: BCE vs Focal Loss

3. Достигнуты высокие edge-level метрики, track-level требует доработки постобработки — ключевое направление дальнейшей работы наряду с оптимизацией производительности

Спасибо за внимание!

Готов ответить на вопросы

Д. Н. Джавадов · Научный руководитель: к.т.н. К. В. Герценбергер
НИЯУ МИФИ · 2026