Документация на реализацию нейросети **Network3D**

## Общая идея

Network3D — это трёхмерная симуляция нейросети с адаптивными связями и рецепторами разных типов входных данных. Каждый нейрон располагается в координатах (depth, height, width), имеет определённое количество связей и адаптируемые пороги активации. Сигналы распространяются от рецепторов через сеть, и со временем сеть обучается усиливать или ослаблять связи в зависимости от совместной активации нейронов.

## Структура классов

### Neuron

**Поля:**

* position: Tuple[int, int, int] — координаты нейрона в 3D-сетке.
* connections: List[Neuron] — соседи, с которыми установлены связи.
* barriers: List[float] — пороги активации для каждой связи.
* accumulated\_signals: Dict[int, float] — накопленные значения по id мыслеформы.
* activation\_count: int — общее число активаций нейрона.
* last\_activation\_step: int — последний шаг, когда нейрон был активирован.
* coactivation\_counts: Dict[Neuron, float] — счётчики совместных активаций с каждым соседом.

**Методы:**

* add\_connection(neighbor) — добавить соседа, если не превышен лимит.
* process\_signal(signal, incoming\_connection\_index, step) — накопить сигнал и при превышении барьера передать соседям.
* update\_coactivations(step, window=10, decay\_rate=0.01) — оценка временной корреляции активаций.
* adapt\_barriers(rate=0.01) — адаптация порогов по результатам со-активации.

### Signal

* ampl: float — амплитуда сигнала [-1, 1]
* id: int — уникальный ID сигнала
* id\_thought: int — ID мыслеформы/рецептора

### Receptor (и подклассы)

* RGBReceptor — принимает (R,G,B), кодирует в сигнал.
* TextReceptor — принимает строку, хеширует.
* SoundReceptor — принимает амплитуду звука.
* MemoryReceptor — пустой сигнал (можно расширять под память).

### Network3D

* grid — 3D-массив нейронов
* step — шаг симуляции
* update() — основной цикл: генерация сигналов, обработка, обновление коактиваций, адаптация барьеров
* run\_network() — бесконечный цикл вызова update()

## Математическая модель

### 1. Накопление сигналов:

Где (A) — амплитуда сигнала, (S\_i) — накопленный сигнал нейрона (i).

### 2. Активация:

Где (B\_i) — порог активации (barrier).

### 3. Обновление совместных активаций:

### 4. Адаптация порогов:

## Потенциальные расширения:

* Поддержка Hebbian Learning
* Динамическое формирование связей (растущая структура)
* Сохранение весов и состояния сети между сессиями
* Поддержка пластичности на основе reward-сигналов (RL)
* Сжатие / свертывание сети для inference

## 