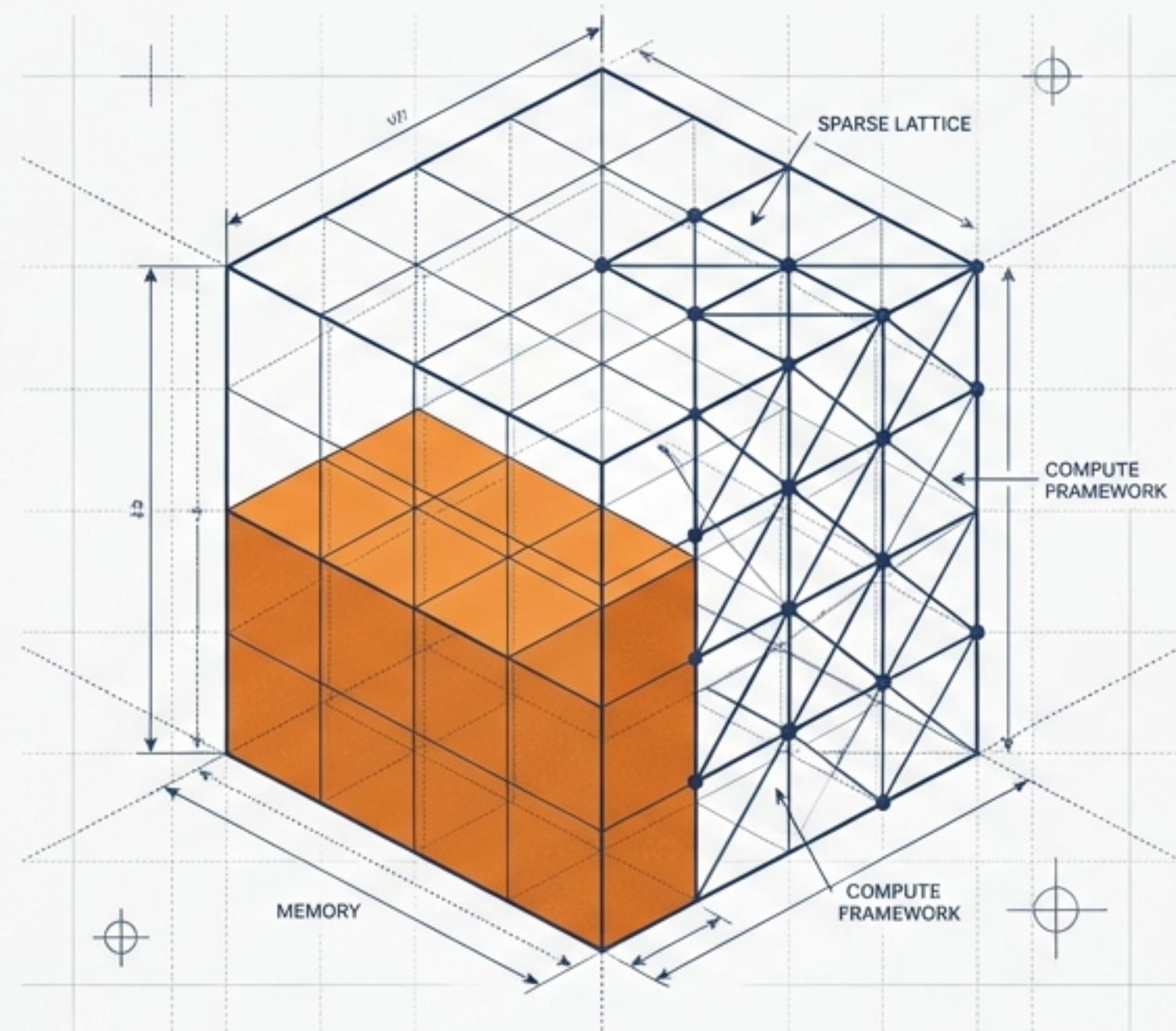


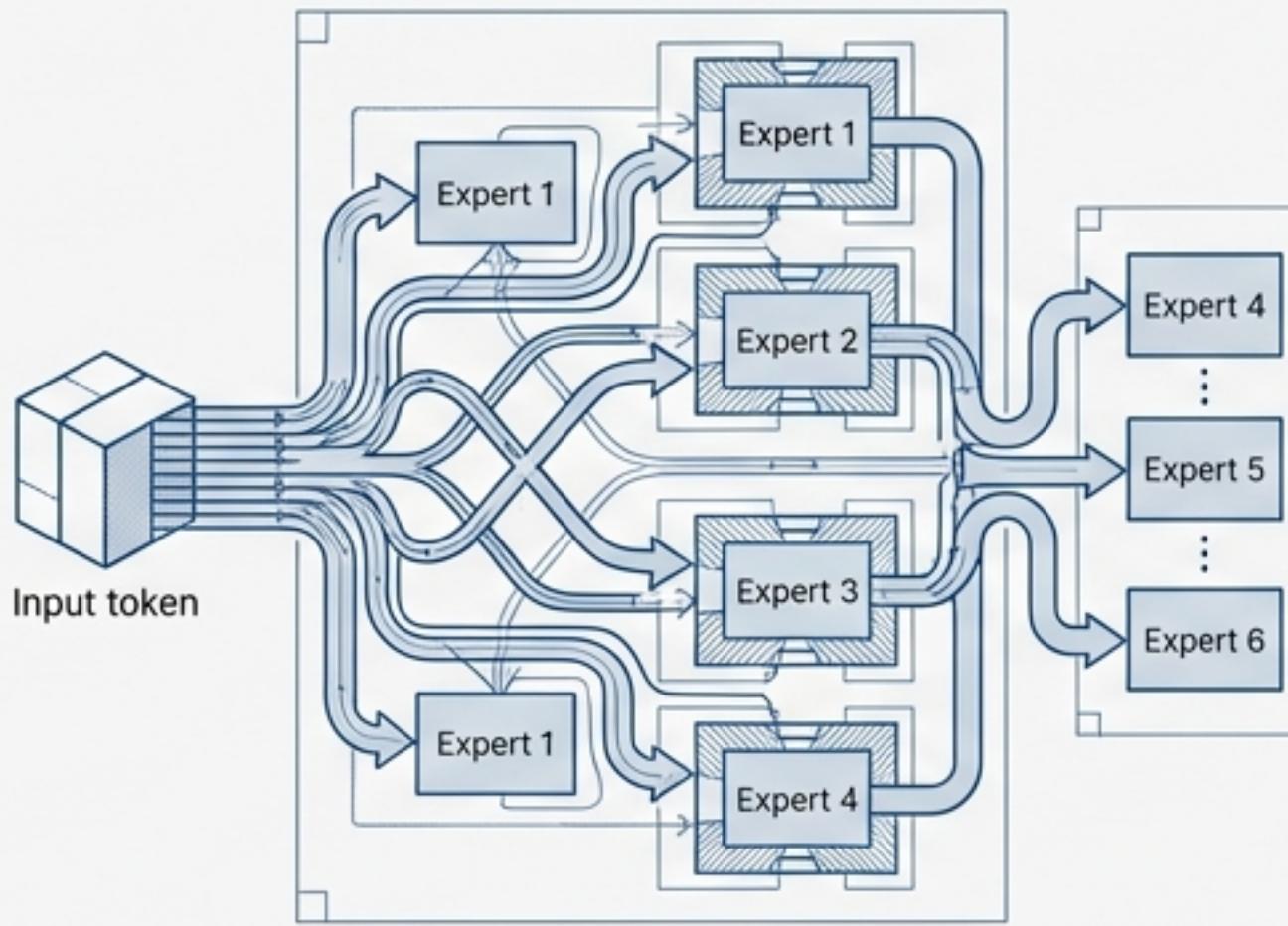
# Conditional Memory via Scalable Lookup: Новый вектор разреженности для LLM

Представление архитектуры Engram: синергия вычислений и памяти



# Резюме: От симуляции памяти к нативному поиску

## MoE (Conditional Computation)



**Проблема:** Трансформеры “симулируют” память через дорогие вычисления (Attention/FFN).

## Ключевые достижения:

- **Производительность:** Превосходство над MoE при равных FLOPs.
- **Эффективность:** MMLU +3.4 (Знания), BBH +5.0 (Рассуждения).
- **Инфраструктура:** Оффлоадинг памяти на CPU (<3% overhead).

## Engram (Conditional Memory)

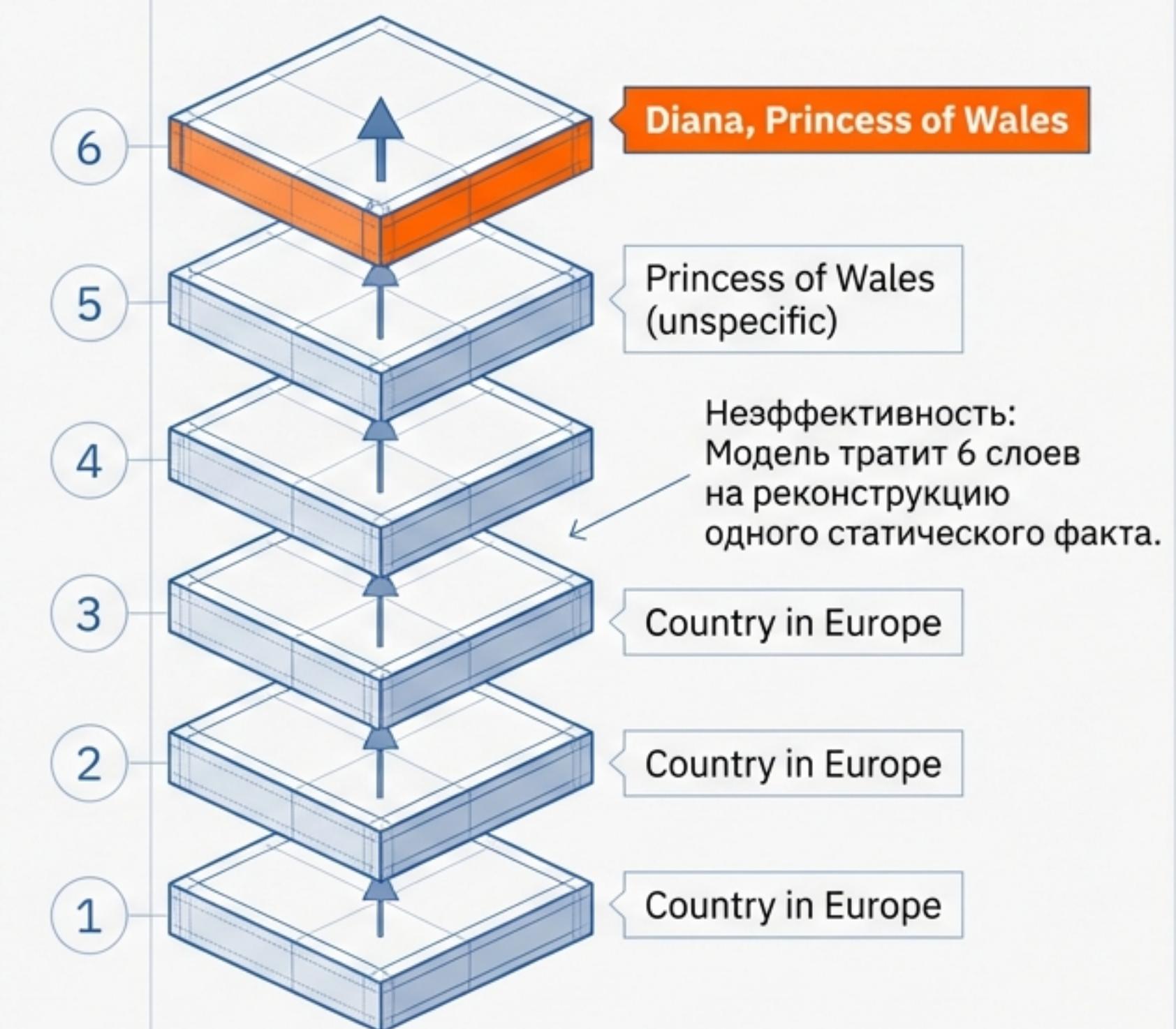


**Решение:** Engram модернизирует N-граммы для мгновенного поиска за  $O(1)$ .

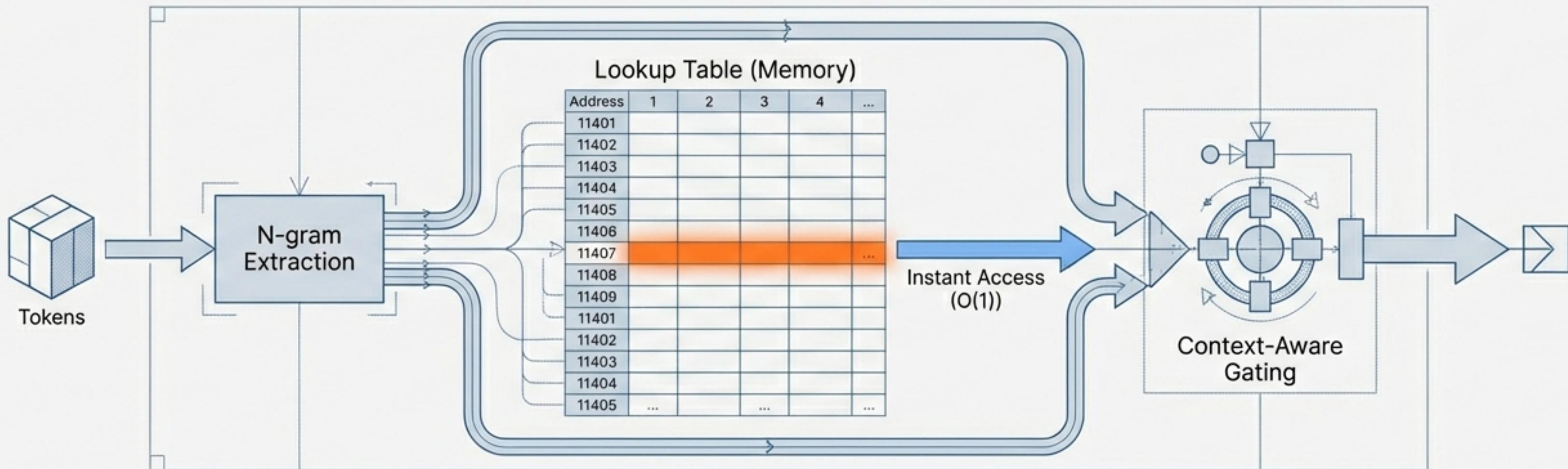
# Мотивация: Двойственная природа языка

Язык состоит из двух типов задач:

1. Compositional Reasoning  
(Рассуждения): Динамические вычисления. Сфера MoE.
2. Knowledge Retrieval  
(Извлечение знаний):  
Статические, локальные паттерны.



# Engram: Новый примитив моделирования



## Conditional Memory

Дополнительная ось разреженности, работающая в паре с MoE.

## Механизм

Встраивание модуля поиска (Lookup) напрямую в слои трансформера.

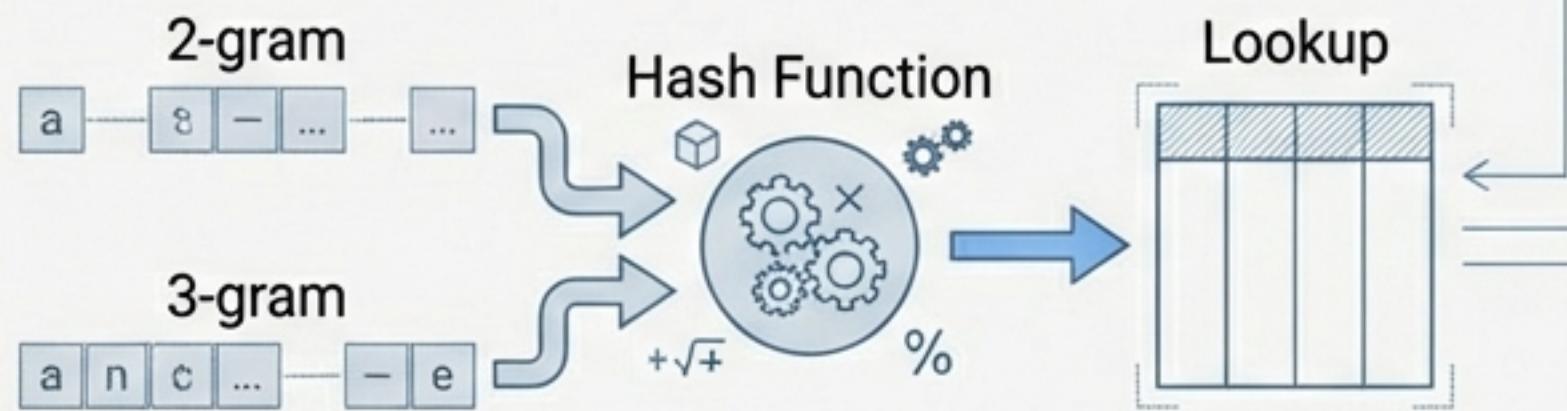
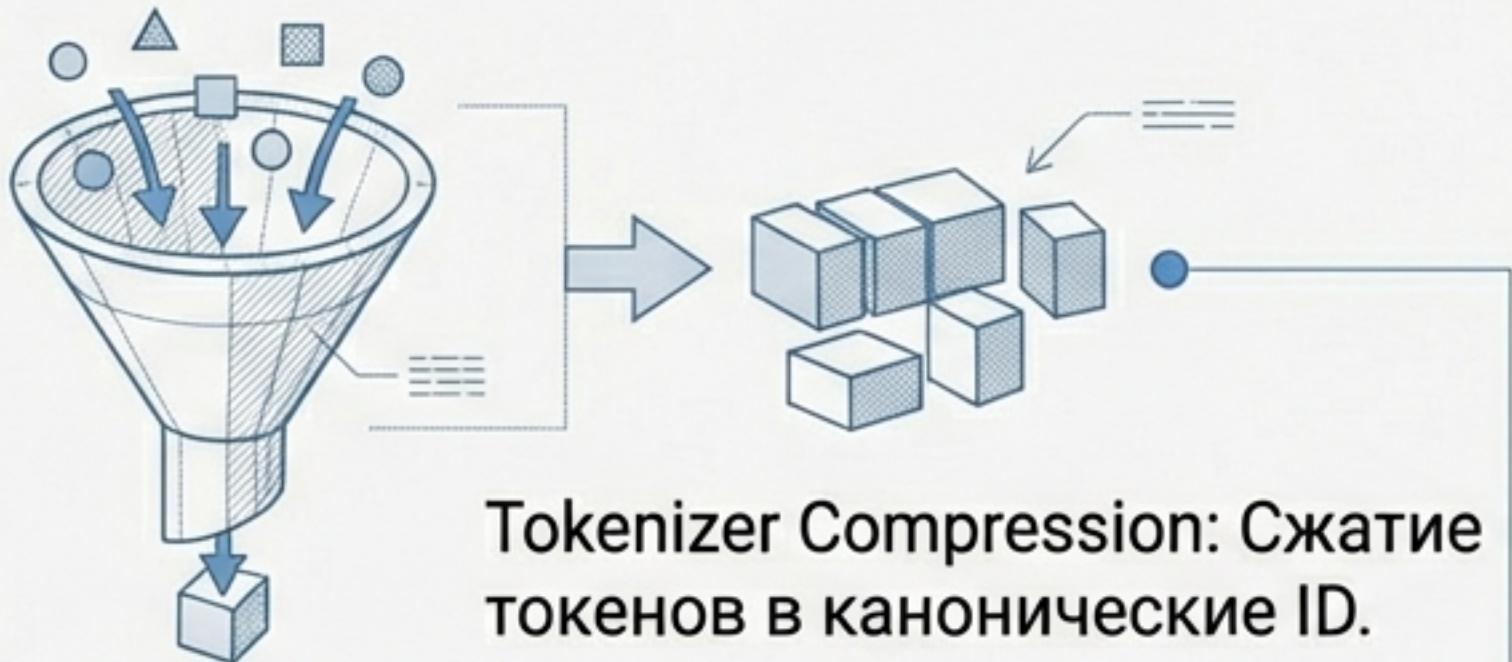
## Цель

Разгрузка основного "хребта" модели от хранения статических фактов.

*«Вместо реконструкции фактов через вычисления, модель просто 'вспоминает' их».*

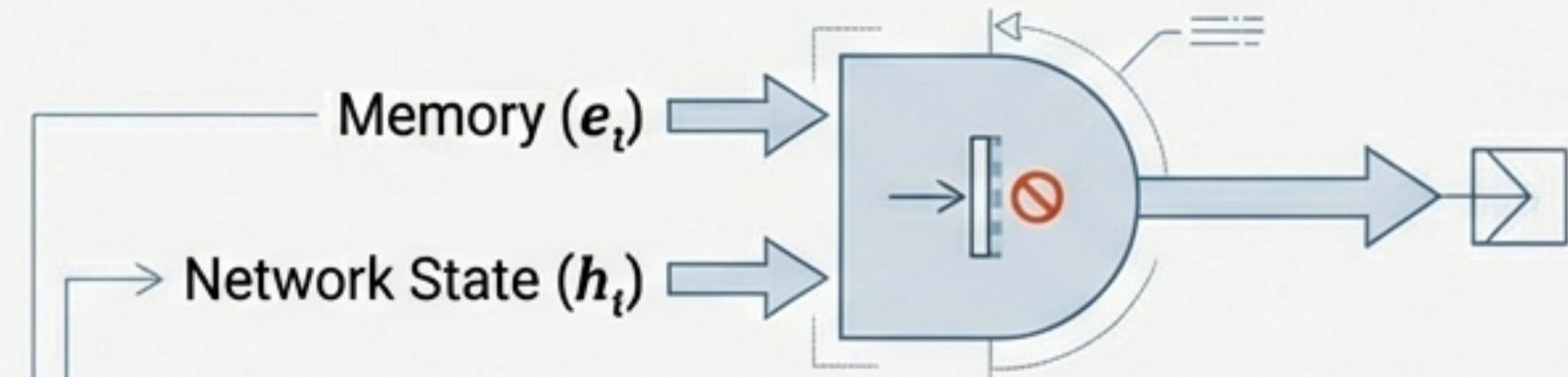
# Архитектура: От сжатия токенов до слияния контекста

## Sparse Retrieval (Разреженный поиск)



Multi-Head Hashing: Детерминированный поиск через хеширование.

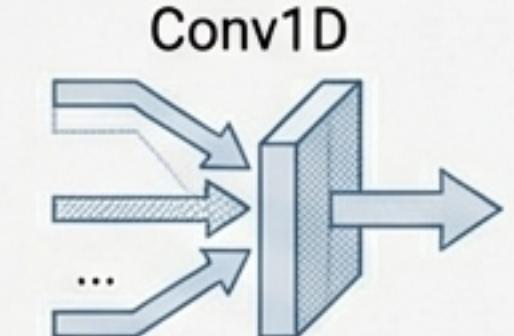
## Context-Aware Gating (Контекстное гейтирование)



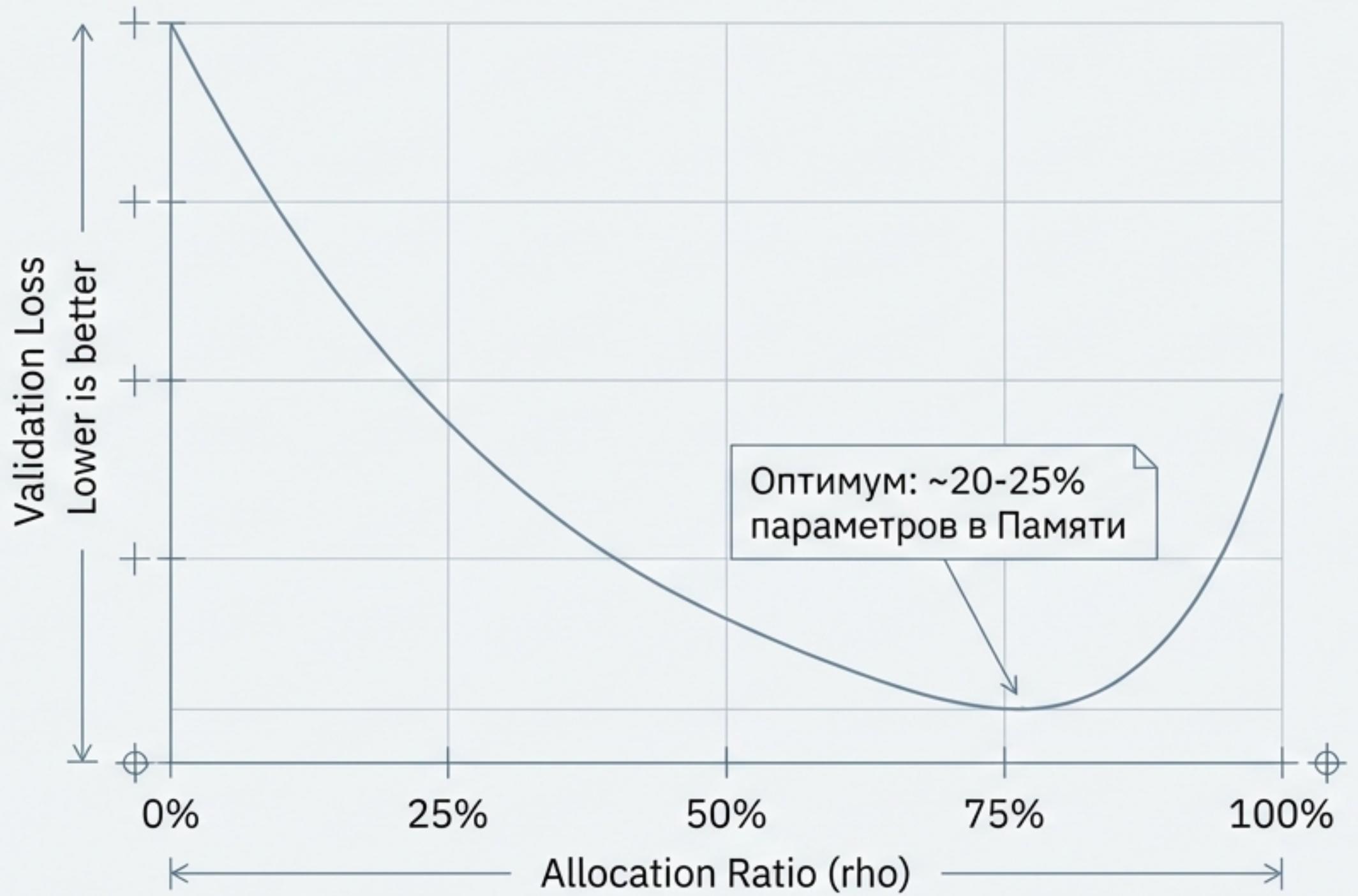
Фильтрация шума: Если память не релевантна контексту, гейт закрывается.

$$\alpha_t = \text{sigmoid} \left( \frac{\text{RMSNorm}(h_t)^T * \text{RMSNorm}(k_t)}{\sqrt{d}} \right)$$

Fusion: Слияние через легковесную свертку (Conv1D).



# Закон распределения разреженности

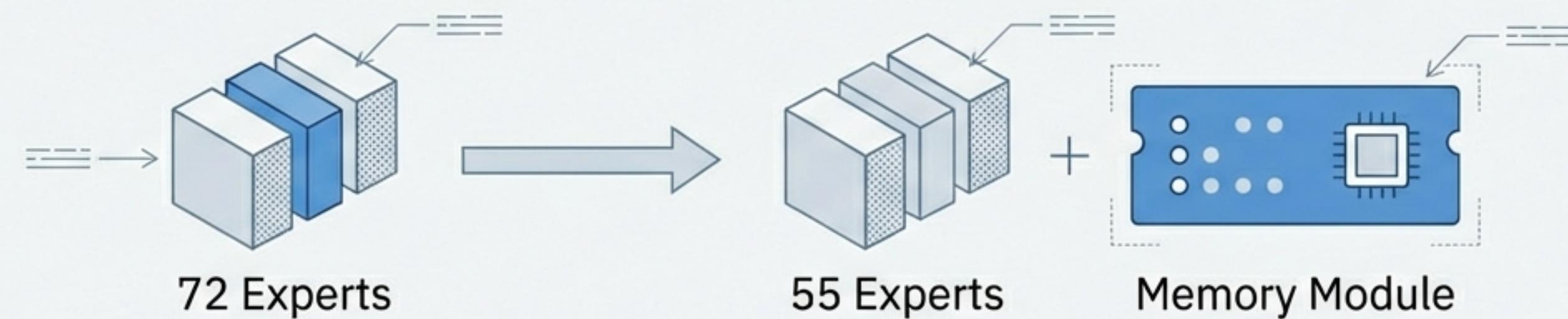


- PURE MoE ( $\rho=100\%$ ): Субоптимально. Не хватает выделенной памяти.
- Engram-Dominated ( $\rho>0\%$ ): Нехватка вычислительной мощности.
- Вывод: Гибридная архитектура (Compute + Memory) строго эффективнее.

# Масштабирование до 27В: Честное сравнение

Сравнение при условиях Iso-Parameters и Iso-FLOPs

Модель	Всего параметров	Routed Experts	Engram Memory
Dense-4B	4.1В	-	-
MoE-27В	26.7В	72	-
Engram-27В	26.7В	55	5.7В

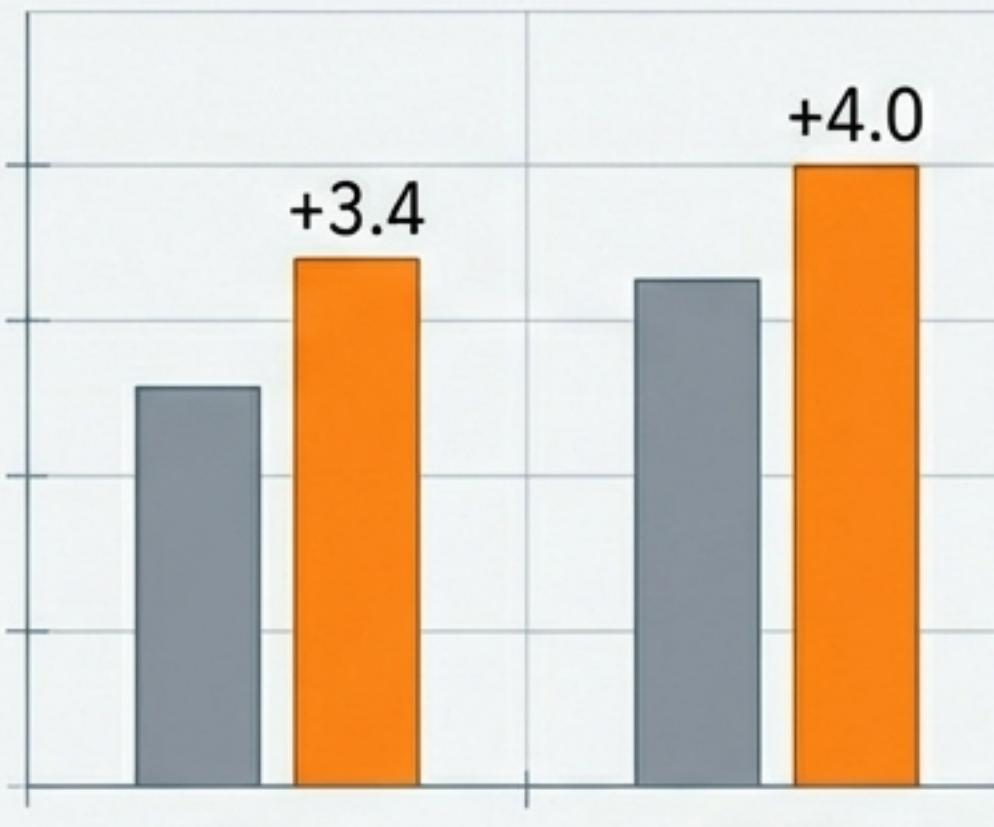


Мы уменьшили число экспертов, чтобы передать бюджет параметров в память (5.7В).

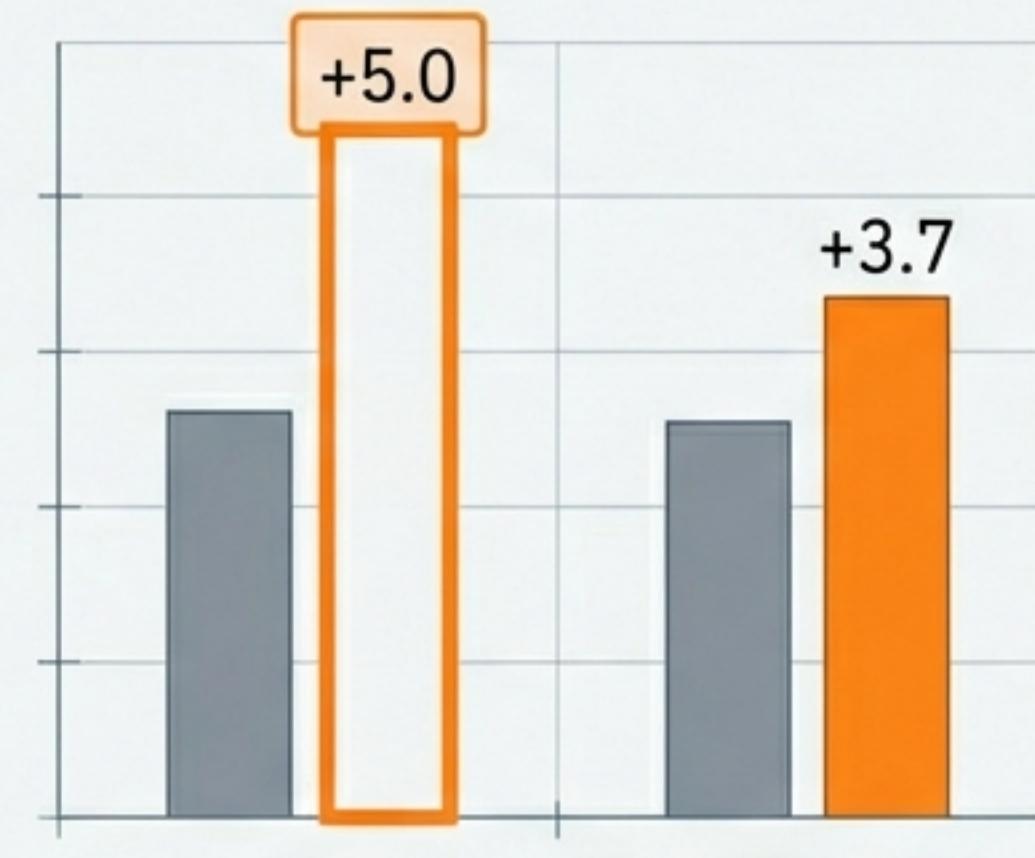
# Результаты: Улучшение рассуждений через разгрузку памяти

■ MoE-27B ■ Engram-27B

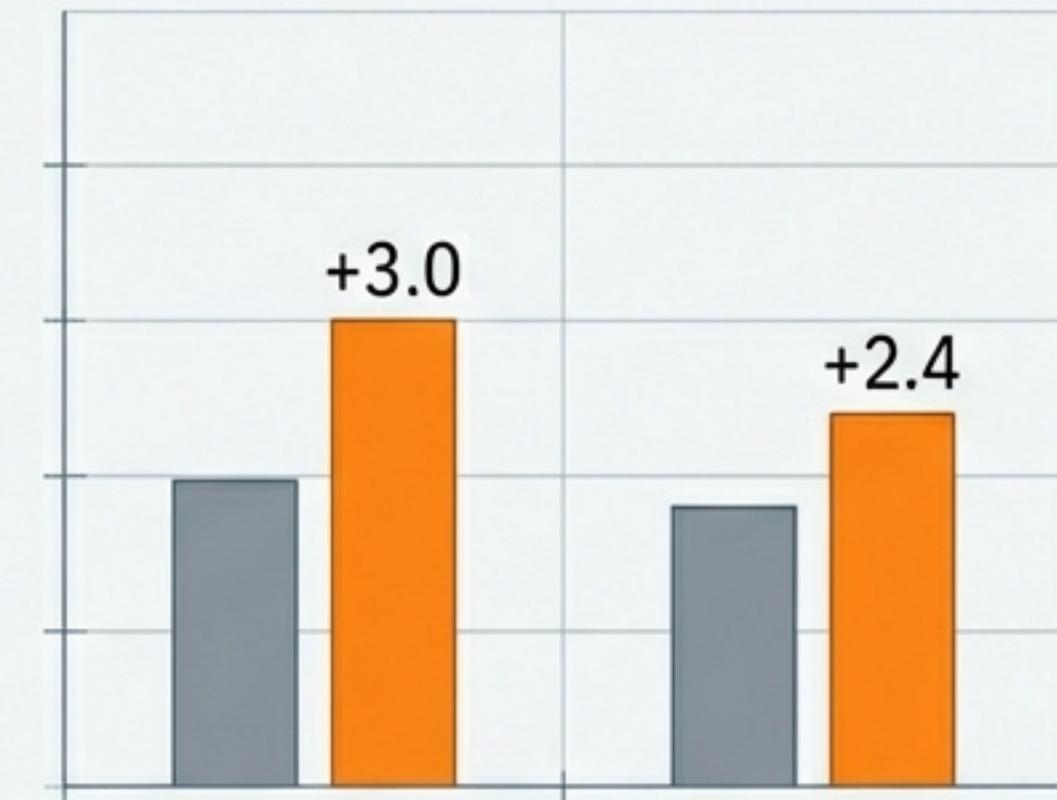
Знания (Knowledge)



Рассуждения (Reasoning)

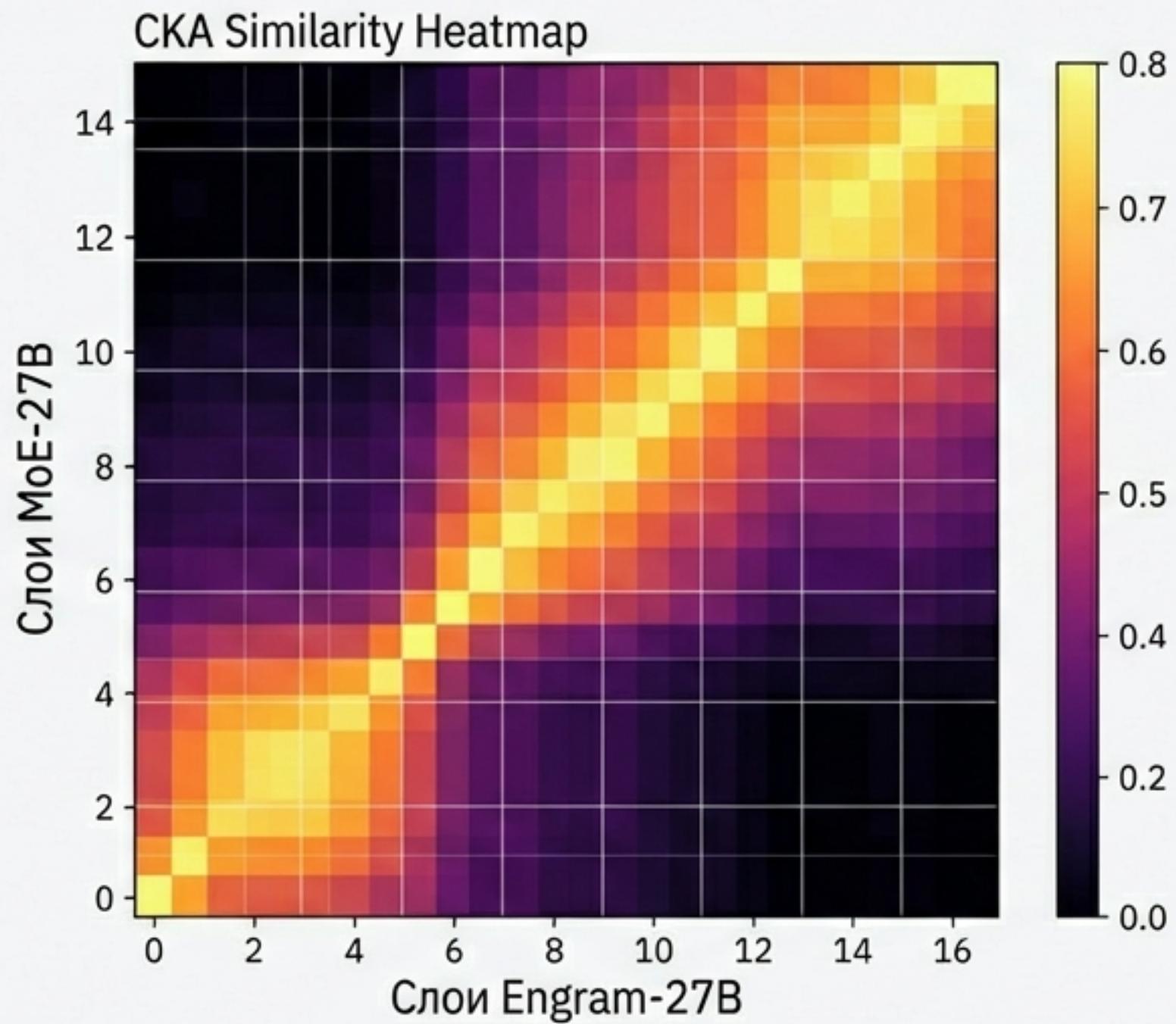


Code & Math



**Инсайт:** Освобождая нейросеть от запоминания, мы улучшаем её способность мыслить.

# Механизм действия: Эффективная глубина



Сравнение слоев Engram-27B и MoE-27B (СКА Analysis).

- » Тепловая карта показывает сдвиг диагонали сходства вверх.
- Интерпретация: 5-й слой Engram семантически соответствует 12-му слою MoE.
- Вывод: Модуль памяти функционально "углубляет" сеть, позволяя пропускать этап рутинной реконструкции признаков.

# Разгрузка внимания: Длинный контекст

## Needle In A Haystack (НИАН)

Multi-Query Accuracy

MoE-27B: 84.2

Engram-27B: 97.0

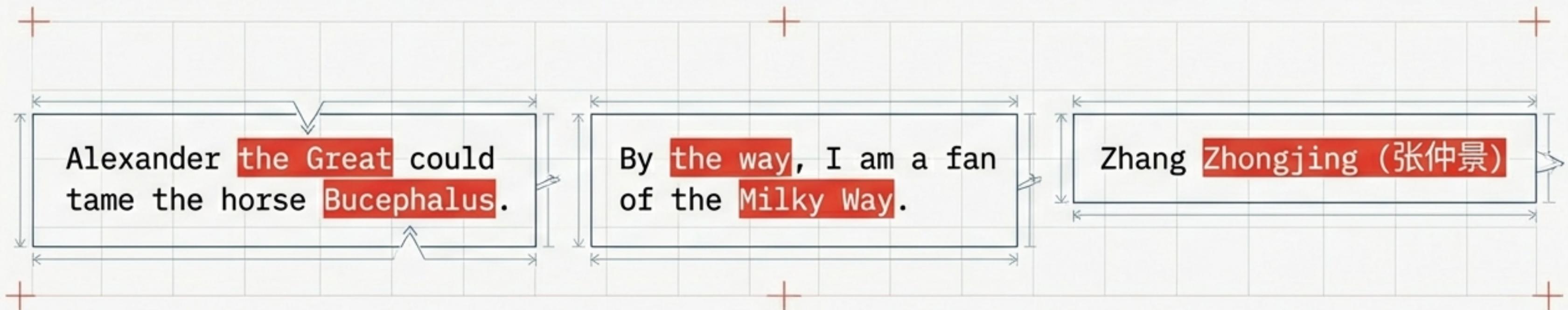
## Variable Tracking

MoE-27B: 77.0

Engram-27B: 87.2

Механизм: Делегирование локальных зависимостей (N-грамм) в Lookup освобождает Self-Attention для работы с глобальным макро-контекстом.

# Визуализация работы: Что запоминает Engram?



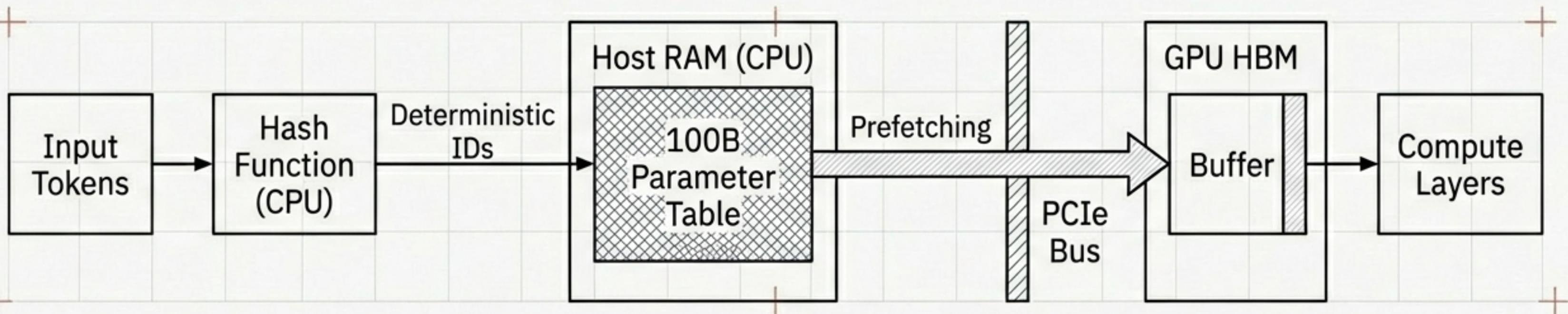
Именованные сущности

Идиомы

Редкие токены

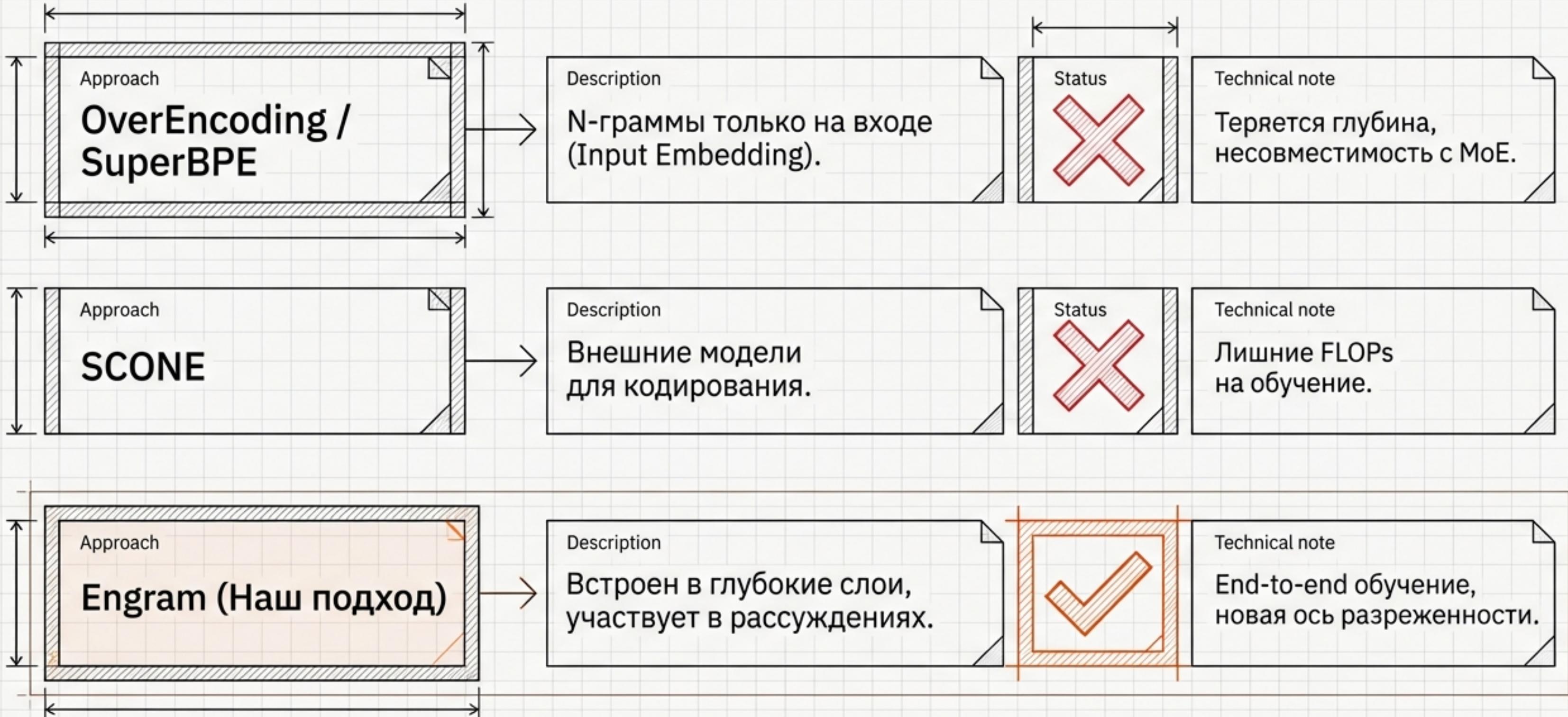
Engram работает как специализированный словарь для стереотипных паттернов.

# Системная эффективность: Преодоление Memory Wall

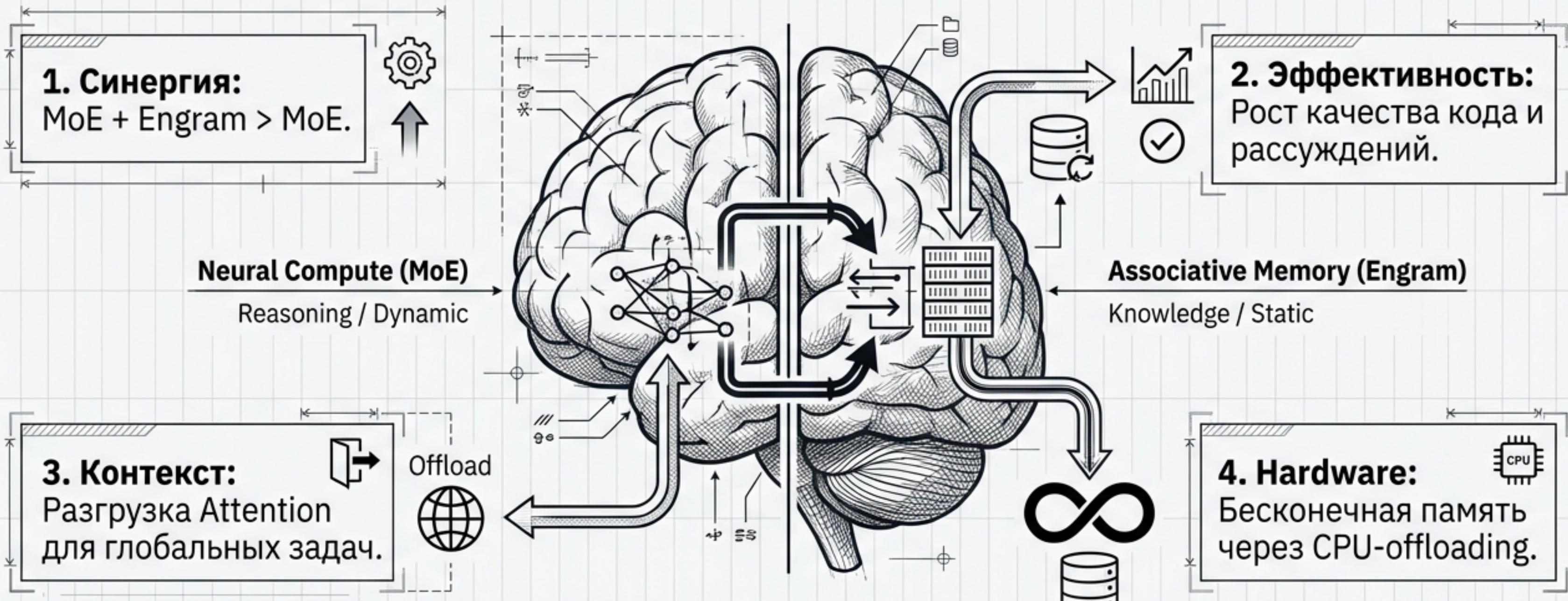


- 1. Deterministic Addressing:** Адреса известны до начала вычислений.
  - 2. Prefetching:** Асинхронная подгрузка данных из CPU RAM.
- Результат: <3% overhead при оффлоадинге 100B параметров.

# Сравнение с альтернативными подходами



# Заключение: Будущее разреженных архитектур



Следующее поколение LLM: Явное разделение труда между вычислениями и памятью.



# Ресурсы

Paper: Conditional Memory via Scalable Lookup: A New Axis of Sparsity for Large Language Models

- Authors: Xin Cheng, Wangding Zeng, et al.  
Affiliation: DeepSeek-AI & Peking University

Code: <https://github.com/deepseek-ai/Engram>

Contact: [chengxin@deepseek.com](mailto:chengxin@deepseek.com), [zengwangding@deepseek.com](mailto:zengwangding@deepseek.com)



GitHub Repository