AlphaTensor

1) Алгоритм Штрассена

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}$$

$h_1 = a_{1,1}\, b_{1,1}$
$h_2 = a_{1,1}\, b_{1,2}$
$h_3 = a_{1,2}\, b_{2,1}$
$h_4 = a_{1,2}\, b_{2,2}$

$h_5 = a_{2,1}\, b_{1,1}$
$h_6 = a_{2,1}\, b_{1,2}$
$h_7 = a_{2,2}\, b_{2,1}$
$h_8 = a_{2,2}\, b_{2,2}$

$c_{1,1} = h_1 + h_3$
$c_{1,2} = h_2 + h_4$
$c_{2,1} = h_5 + h_7$
$c_{2,2} = h_6 + h_8$

– Наивный алгоритм
8 умножений, 4 сложения
Общая сложность: $O(N^3)$

$h_1 = (a_{1,1} + a_{2,2})\,(b_{1,1} + b_{2,2})$
$h_2 = (a_{2,1} + a_{2,2})\, b_{1,1}$
$h_3 = a_{1,1}\,(b_{1,2} - b_{2,2})$
$h_4 = a_{2,2}\,(-b_{1,1} + b_{2,1})$

$h_5 = (a_{1,1} + a_{1,2})\, b_{2,2}$
$h_6 = (-a_{1,1} + a_{2,1})\,(b_{1,1} + b_{1,2})$
$h_7 = (a_{1,2} - a_{2,2})(b_{2,1} + b_{2,2})$

$c_{1,1} = h_1 + h_4 - h_5 + h_7$
$c_{1,2} = h_3 + h_5$
$c_{2,1} = h_2 + h_4$
$c_{2,2} = h_1 - h_2 + h_3 + h_6$

Штрассен

7 умножений, 18 сложений
Общая сложность: $O(N^{\log_2 7}) \approx O(N^{2.8})$

Основная идея работы: Зафиксируем размеры матриц, и будем с помощью RL искать алгоритмы корректного умножения матриц с минимальным числом скалярных умножений.

Формулировка задачи в парадигме RL:

Агент пошагово генерирует действия над матрицами. В конце получает награду, пропорциональную числу действий.

$h_1 = (a_{1,1} + a_{2,2})(b_{1,1} + b_{2,2})$

$h_2 = (a_{2,1} + a_{2,2})\, b_{1,1}$

$h_3 = a_{1,1}(b_{1,2} - b_{2,2})$

$h_4 = a_{2,2}(-b_{1,1} + b_{2,1})$

$h_5 = (a_{1,1} + a_{1,2})\, b_{2,2}$

$h_6 = (-a_{1,1} + a_{2,1})(b_{1,1} + b_{1,2})$

$h_7 = (a_{1,2} - a_{2,2})(b_{2,1} + b_{2,2})$

— Каждая строчка - действие агента.

Проблемы формулировки:

1) Не понятно, в каком виде представлять действия;
2) Даже если придумаем вид, оно не будет учиться;

Параметризуем задачу:

<u>Утв.</u> Всякое билинейное отображение $\beta: V \times U \to W$, $\dim(V) = n$, $\dim(U) = m$, $\dim(W) = k$ можно представить в виде тензора размера $n \times m \times k$.

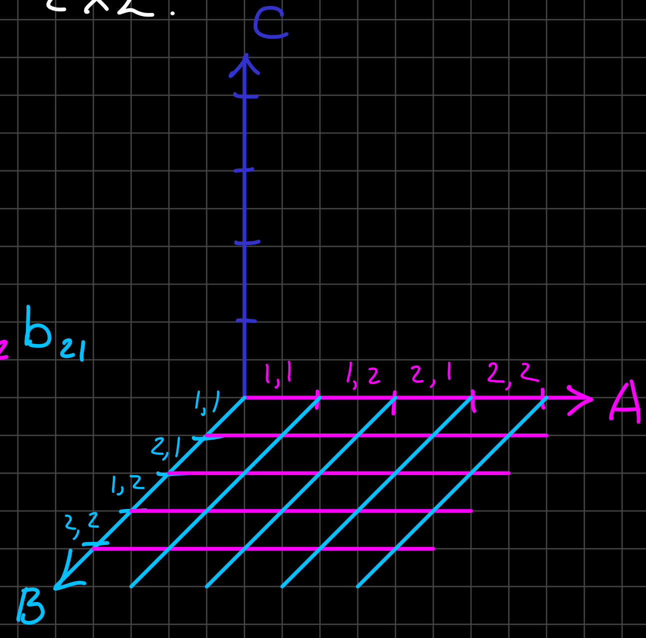
<u>Следствие</u>: Произведение матриц размера $N \times N$ можно представить в виде тензора размера $N^2 \times N^2 \times N^2$.

<u>Контрольный вопрос</u>: Операция умножения матриц размеров $N \times M$ и $M \times P$ представляется в виде тензора размера $N^2 \times M^2 \times P^2$?

Как выглядит такое представление:

Рассмотрим на примере матриц 2×2:

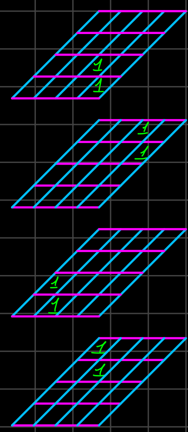
$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}\begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}$$



$$C_{11} = a_{11}b_{11} + a_{12}b_{21} = 1a_{11}b_{11} + 1a_{12}b_{21}$$

C_{11}:

	a_{11}	a_{12}	a_{21}	a_{22}
b_{11}	1			
b_{21}		1		
b_{12}				
b_{22}				



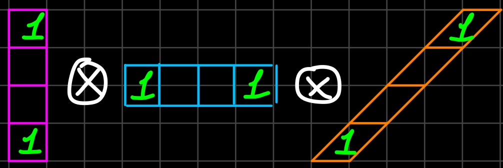
Теорема (CP-разложение): Для каждого трёхмерного тензора $T \in \mathbb{R}^{n\times m\times k}$ существует разложение в виде

$$T = \sum_{i=1}^{R} u_i \otimes v_i \otimes w_i, \quad u_i \in \mathbb{R}^n, v_i \in \mathbb{R}^m, w_i \in \mathbb{R}^k,$$

причём R определено неоднозначно.

Что нам это даёт?

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}\begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}$$

$h_1 = (a_{11} + a_{22})(b_{11} + b_{22}) \longleftrightarrow$ 

$h_2 = (a_{11} + a_{22})\, b_{11}$

$h_3 = a_{11}(b_{12} - b_{22})$

$h_4 = a_{22}(-b_{11} + b_{12})$

$h_5 = (a_{11} + a_{12})\, b_{22}$

$h_6 = (-a_{11} + a_{21})(b_{11} + b_{12})$

$h_7 = (a_{12} - a_{22})(b_{21} + b_{22})$

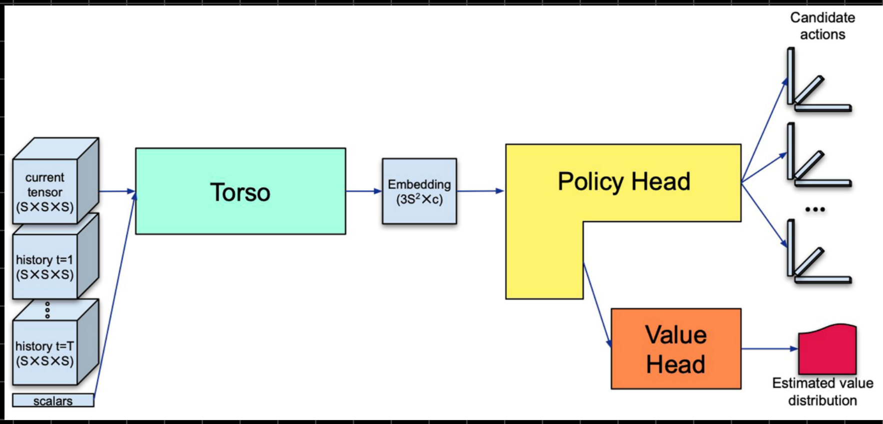
$$\begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix} = \begin{pmatrix} h_1 + h_4 - h_5 + h_7 & h_3 + h_5 \\ h_2 + h_4 & h_1 + h_3 - h_2 + h_6 \end{pmatrix}$$

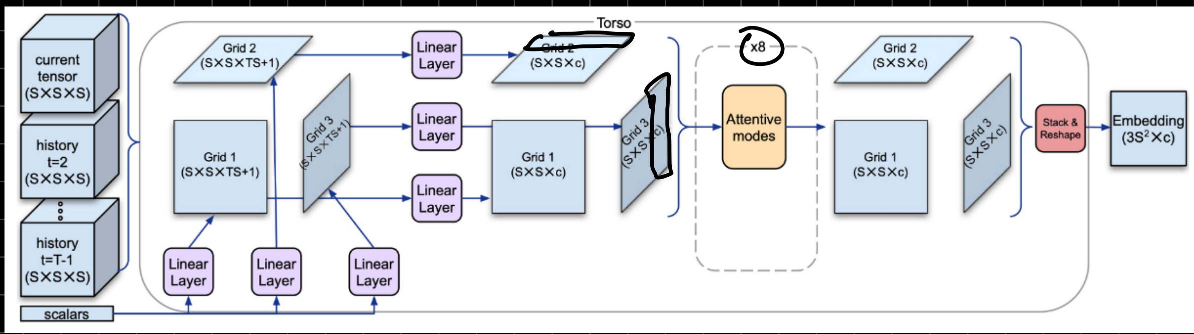
Таким образом мы параметризовали задачу и действия агента. Однако, учиться он не будет.

Как заставить это учиться?

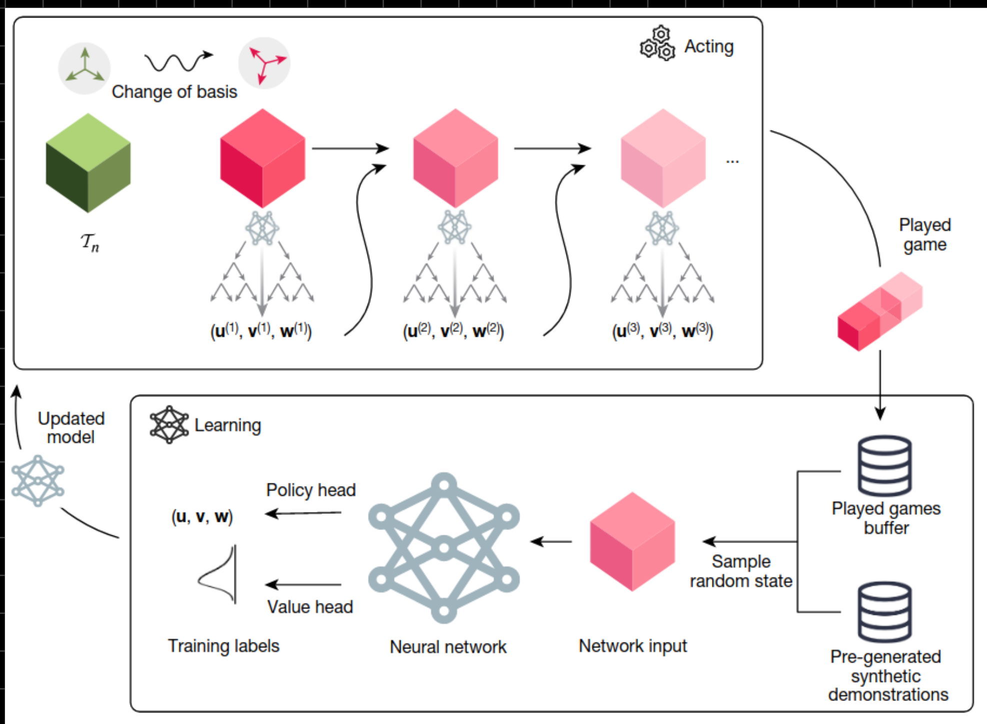
1) Перевернём задачу;
2) Используем Alpha Zero для генерации действия;
3) Обучаем на синтетических данных;
4) Смена базиса;
5) Использование одного агента для выучивание алгоритмов для матриц разных размеров;

Архитектура модели и итоговый алгоритм





Алгоритм:



Результаты

Size (n, m, p)	Best method known	Best rank known	AlphaTensor rank Modular	AlphaTensor rank Standard
(2, 2, 2)	(Strassen, 1969)[2]	7	7	7
(3, 3, 3)	(Laderman, 1976)[15]	23	23	23
(4, 4, 4)	(Strassen, 1969)[2] $(2,2,2) \otimes (2,2,2)$	49	47	49
(5, 5, 5)	(3, 5, 5) + (2, 5, 5)	98	96	98
(2, 2, 3)	(2, 2, 2) + (2, 2, 1)	11	11	11
(2, 2, 4)	(2, 2, 2) + (2, 2, 2)	14	14	14
(2, 2, 5)	(2, 2, 2) + (2, 2, 3)	18	18	18
(2, 3, 3)	(Hopcroft and Kerr, 1971)[16]	15	15	15
(2, 3, 4)	(Hopcroft and Kerr, 1971)[16]	20	20	20
(2, 3, 5)	(Hopcroft and Kerr, 1971)[16]	25	25	25
(2, 4, 4)	(Hopcroft and Kerr, 1971)[16]	26	26	26
(2, 4, 5)	(Hopcroft and Kerr, 1971)[16]	33	33	33
(2, 5, 5)	(Hopcroft and Kerr, 1971)[16]	40	40	40
(3, 3, 4)	(Smirnov, 2013)[18]	29	29	29
(3, 3, 5)	(Smirnov, 2013)[18]	36	36	36
(3, 4, 4)	(Smirnov, 2013)[18]	38	38	38
(3, 4, 5)	(Smirnov, 2013)[18]	48	47	47
(3, 5, 5)	(Sedoglavic and Smirnov, 2021)[19]	58	58	58
(4, 4, 5)	(4, 4, 2) + (4, 4, 3)	64	63	63
(4, 5, 5)	$(2,5,5) \otimes (2,1,1)$	80	76	76