


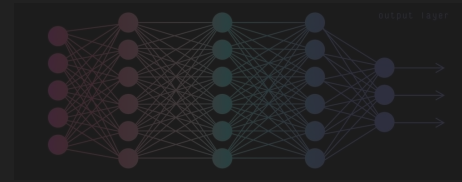
Тензорные методы для оптимизации эффективности нейронных сетей



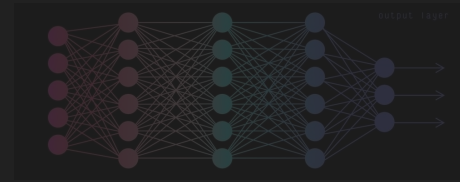
В. Кудревская + Я. Терещенко
+ К. Пугин + Л. Деда

=

InterPoLazio



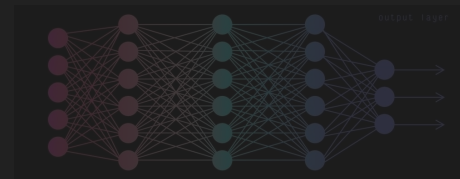
- Задача: применение тензорных декомпозиций для сжатия нейронных сетей без потерь в качестве
- Использованные тензорные декомпозиции:
 - Каноническое разложение
 - Разложение Таккера
 - Tensor Train
- Разложения применялись к полносвязным слоям и сверточным



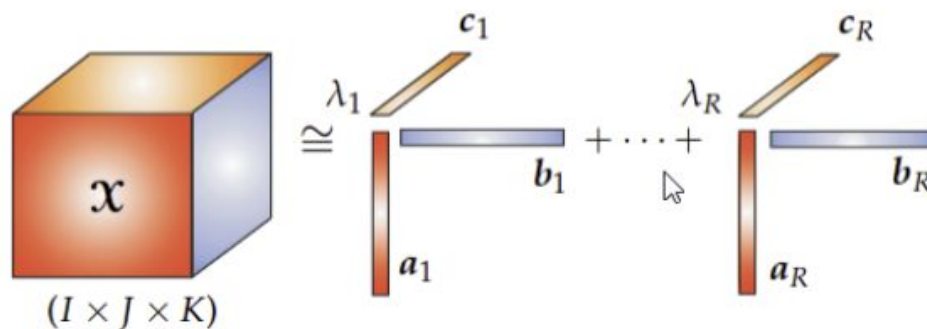
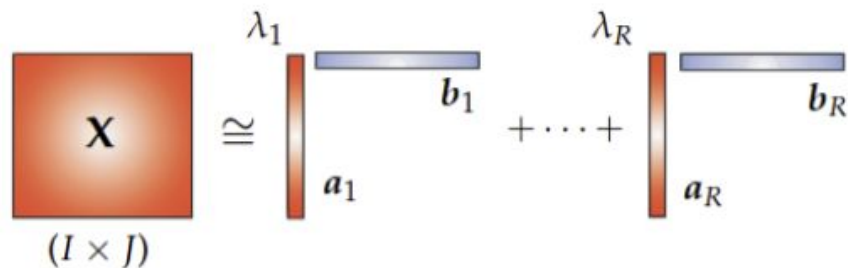
Для исследования были взяты библиотеки `tensorly` и `tltorch`. Помимо них были все три метода были реализованы и тоже добавлены к сравнению

- В качестве датасета была выбрана классификация медуз (6 классов), а метрика - accuracy
- Целью было сравнить реализации и готовые методы из библиотек `tensorly` и `tltorch`. Выявить, справляются ли они с поставленной задачей и что лучше использовать.

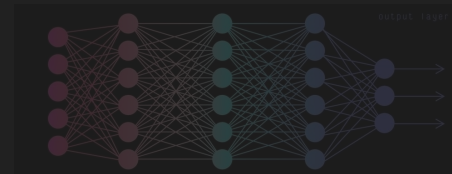
Каноническое разложение



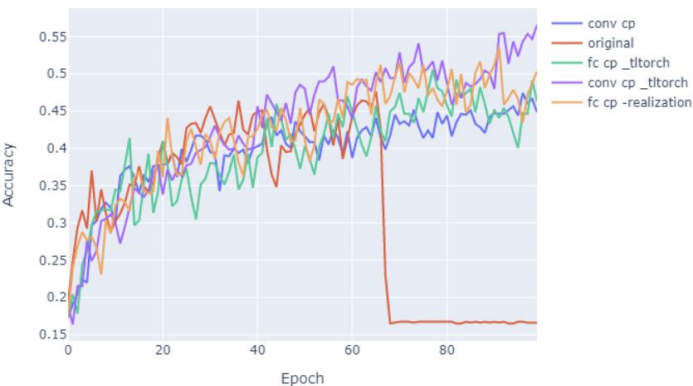
$$A(i_1, \dots, i_d) \approx \sum_{\alpha=1}^r U_1(i_1, \alpha) \dots U_d(i_d, \alpha)$$



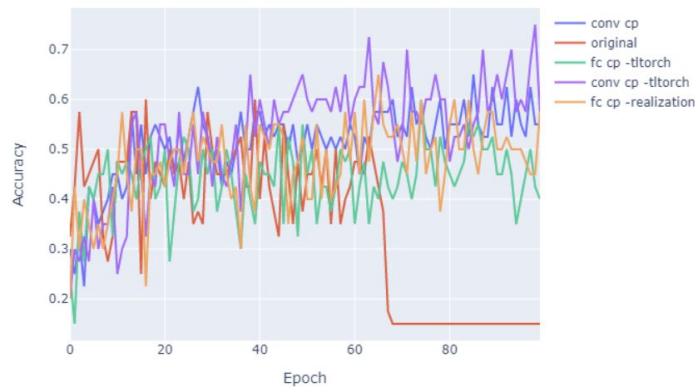
Метод 1: Каноническое разложение



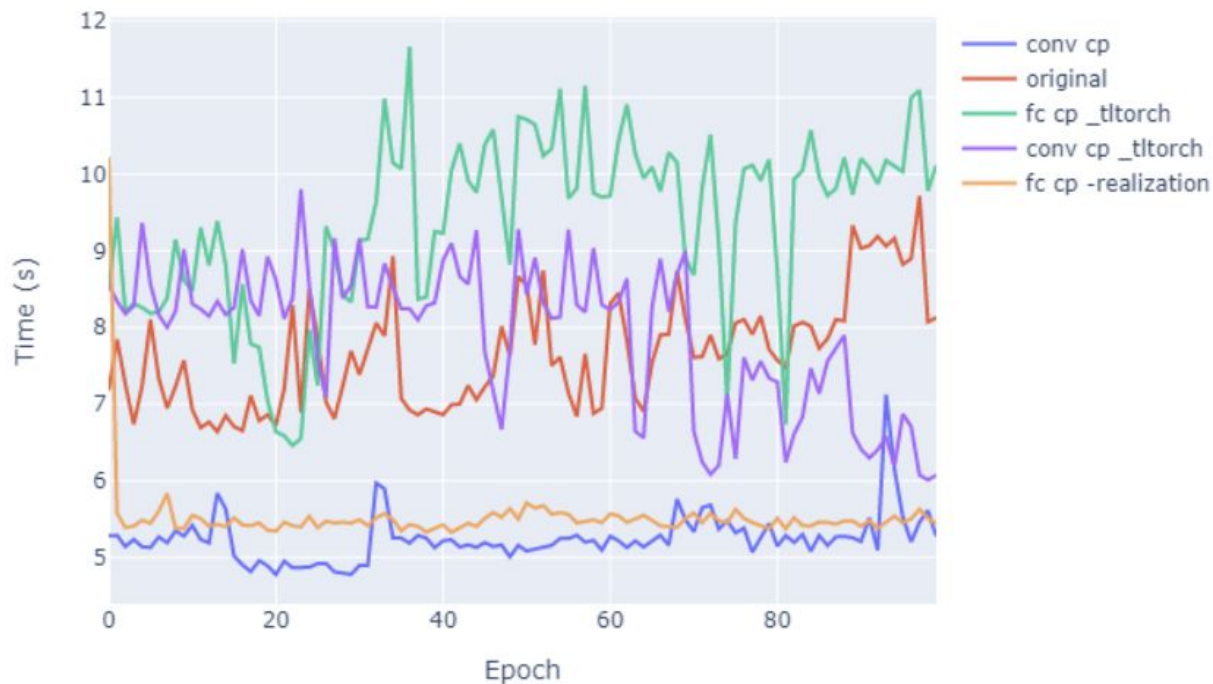
Training Accuracy



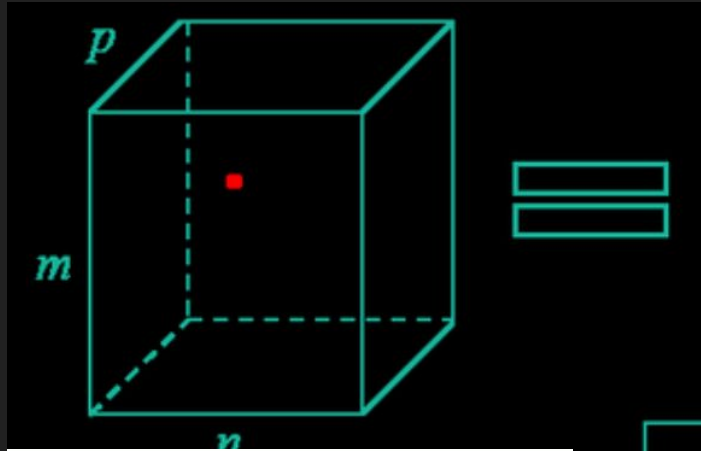
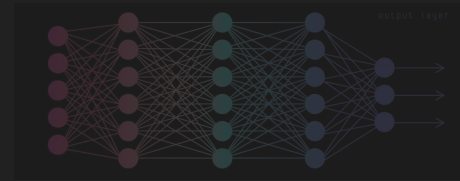
Testing Accuracy



Time per Epoch

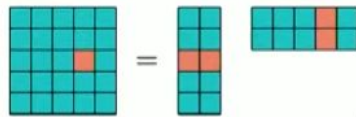


Tensor train decomposition



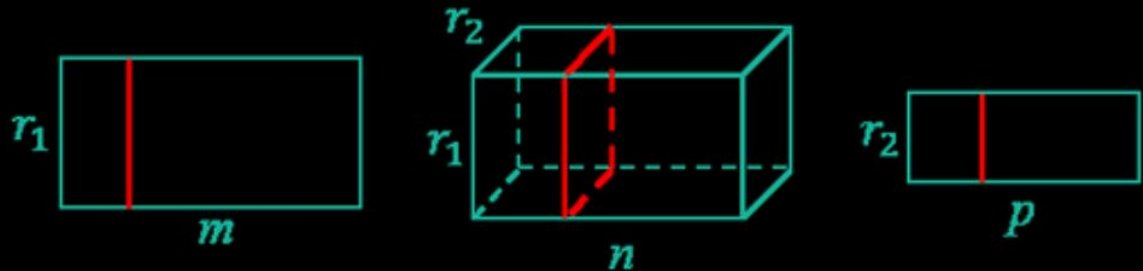
$$A(i_1, \dots, i_d) \approx G_1(i_1) \dots G_d(i_d)$$

$G_k(i_k)$ имеет размер $r_{k-1} \times r_k$, $r_0 = r_d = 1$.

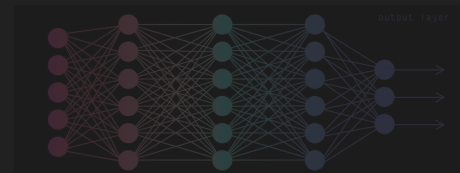


$$A_{3,4} = u_{3,:}^\top v_{4,:}$$

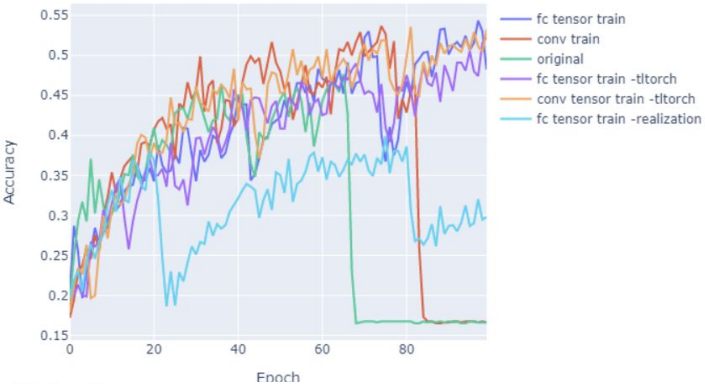
$$B_{2,3,1} = u_{2,:}^\top v_{3,:,:} w_{1,:}$$



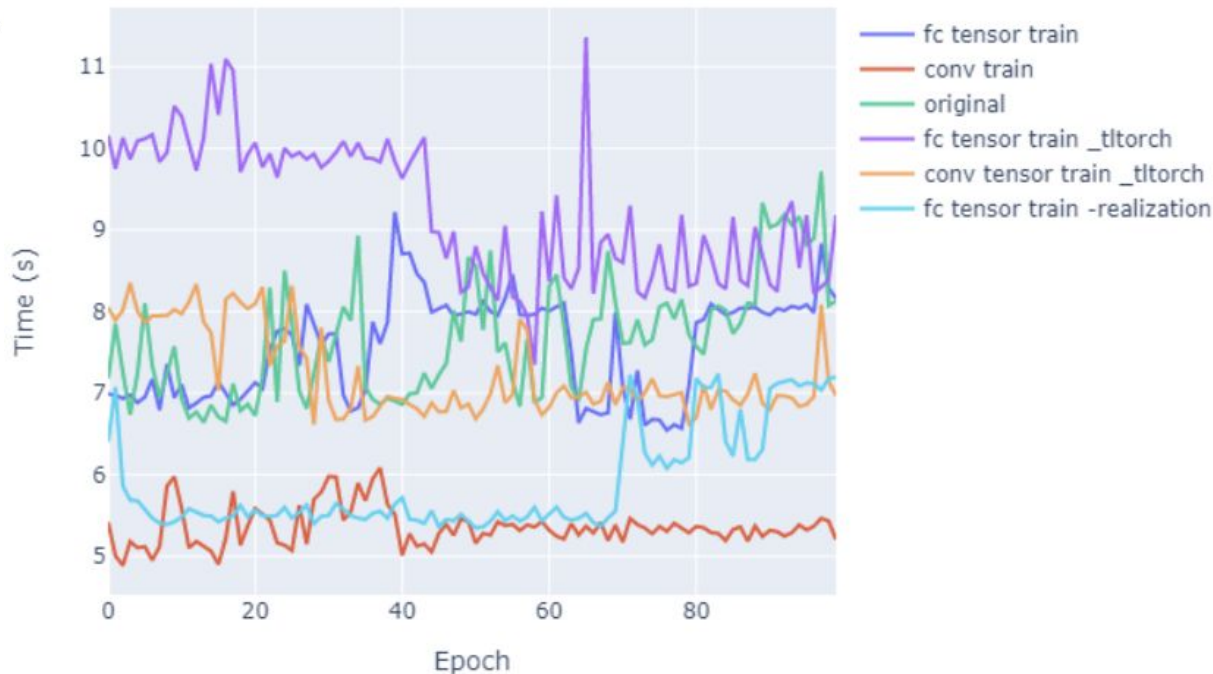
Метод 2: tensor train decomposition



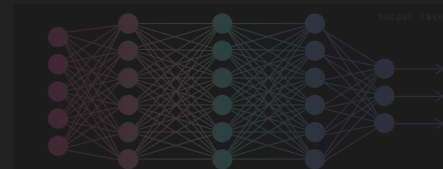
Training Accuracy



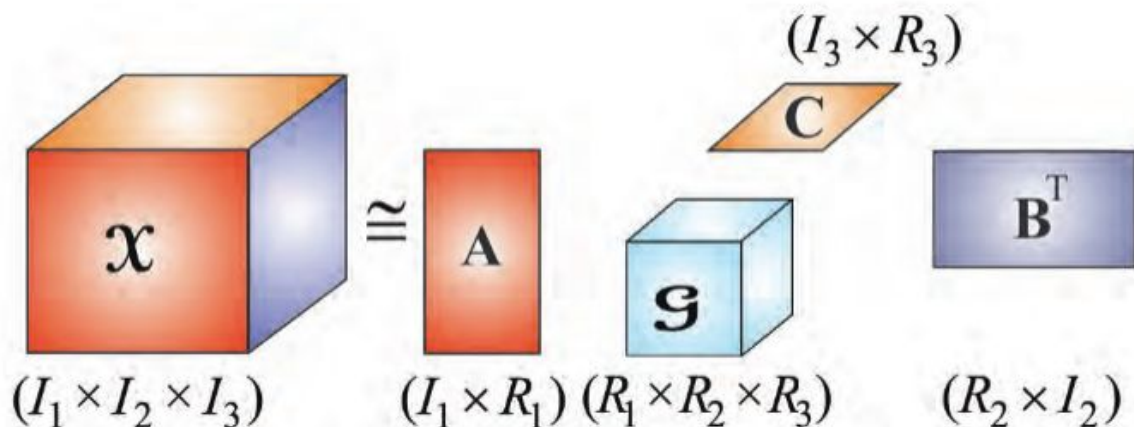
Time per Epoch



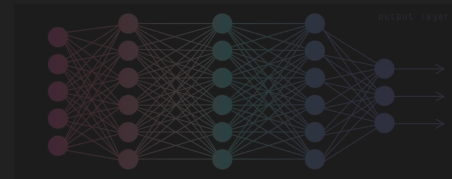
Разложение Такера (Tucker)



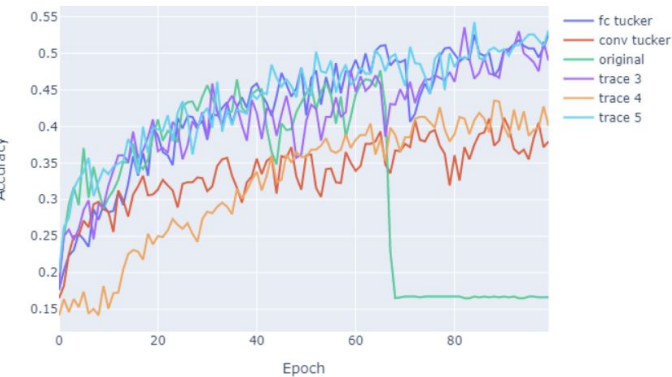
$$A(i, j, k) \approx \sum_{\alpha, \beta, \gamma} G(\alpha, \beta, \gamma) U_1(i_1, \alpha) U_2(i_2, \beta) U_3(i_3, \gamma)$$



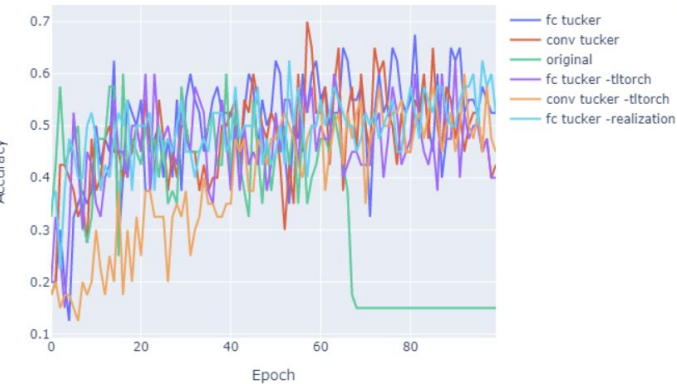
Метод 3: разложение Такера



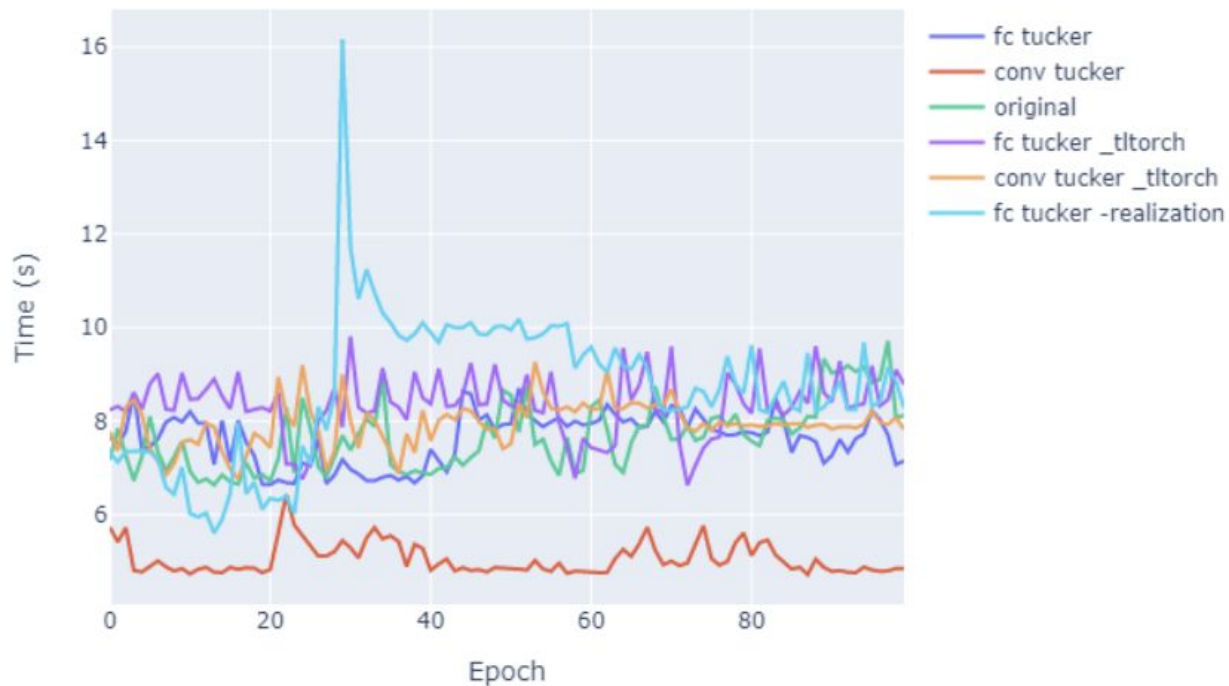
Training Accuracy



Testing Accuracy

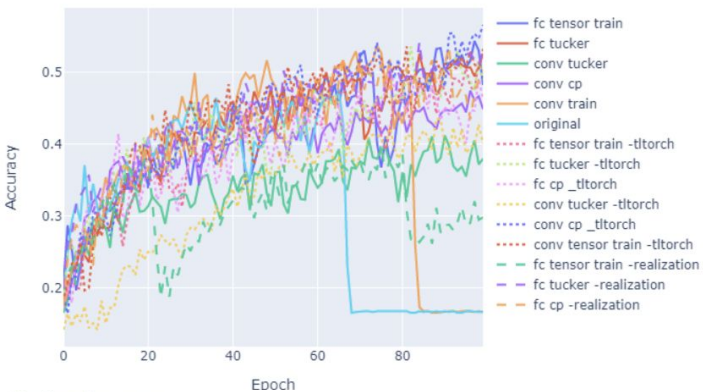


Time per Epoch

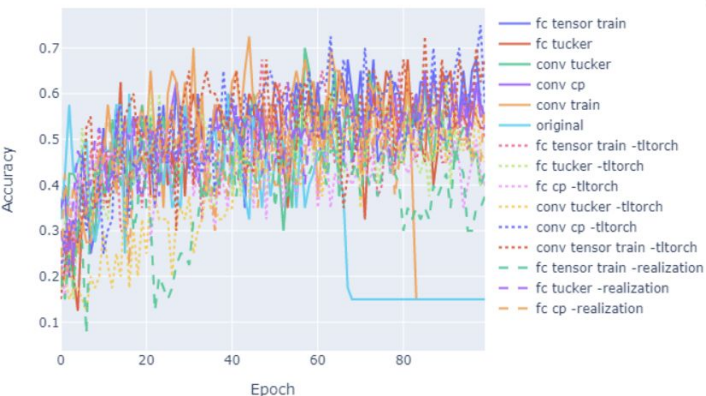


Общие графики

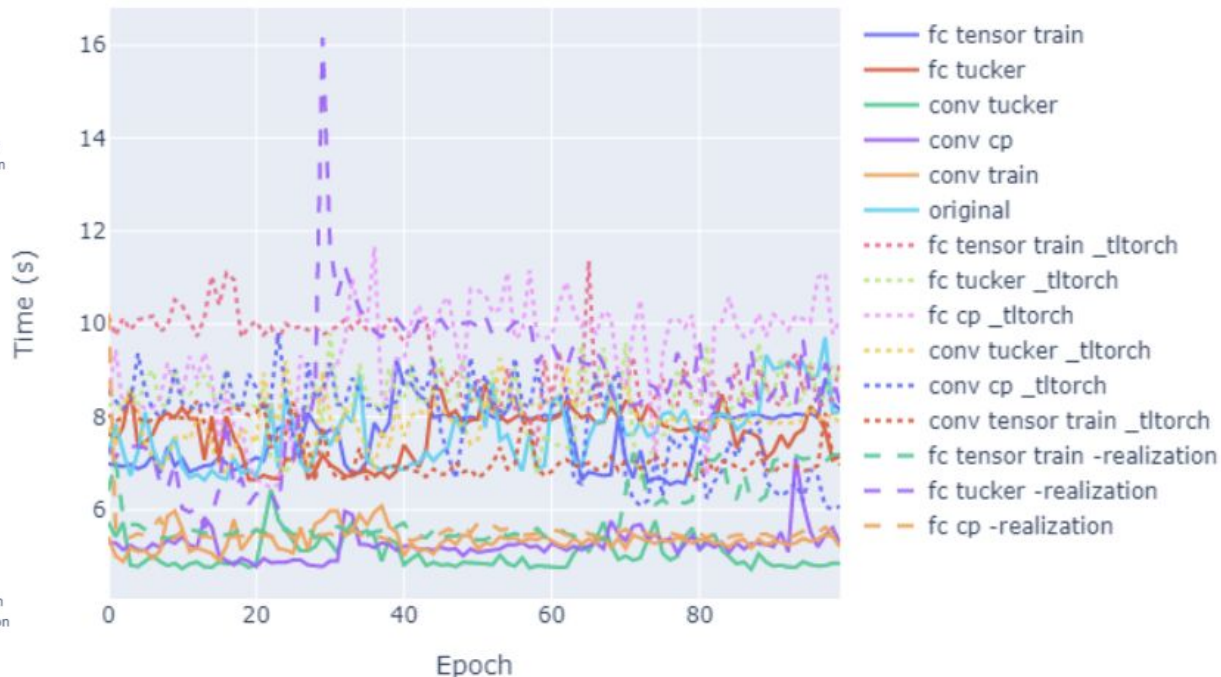
Training Accuracy



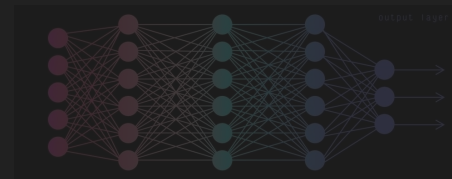
Testing Accuracy



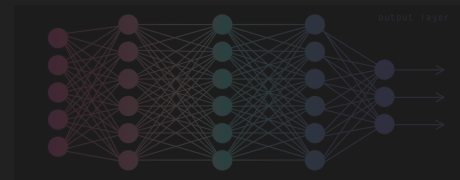
Time per Epoch



Еще один график



Полезные ссылки



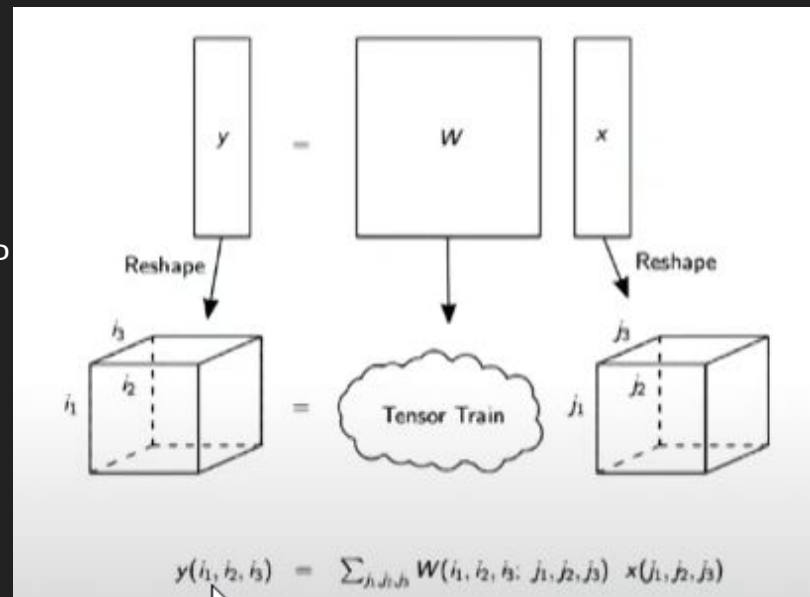
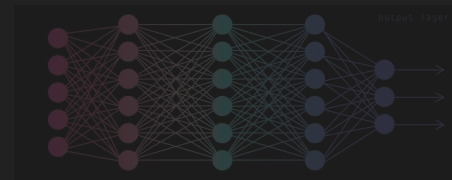
Ссылка на рабочую версию программы: <https://github.com/dmsy4/aim-nla-tensor-decompositions>

Другие полезные ссылки:

- V. Lebedev, Y. Ganin, M. Rakhuba, I. Oseledets, V. Lempitsky, *Speeding-up convolutional neural networks using fine-tuned cp-decomposition*, URL: <https://arxiv.org/pdf/1412.6553.pdf>, 2015.
- Yu Pan, Maolin Wang, Zenglin Xu, *TedNet: A Pytorch Toolkit for Tensor Decomposition Networks*, URL: <https://arxiv.org/pdf/2104.05018.pdf>, 2021.
- <https://jacobgil.github.io/deeplearning/tensor-decompositions-deep-learning>
- <https://tensorly.org/stable/index.html>

Нереализовано

- Изменение размеров векторов ответов и признаков, с целью получить тензорный оператор из матрицы Весов, который в последствии можно было бы декомпозировать
- Не до конца изучены зависимости, когда декомпозиции Дают реальный прирост в скорости, а когда нет



Спасибо за внимание!

