

REPORT

- [wandb репорт экспериментов](#)
- [wandb project](#)

pipeline

1. [paperswithcode on ImageNet](#)
2. sort_by(Number of params)
3. выбрал [SimpleNet](#) → почитал paper → анализ. [github](#)
4. обнаружил "successor" [SimpNet](#) → почитал paper → анализ. [github](#)
5. восстановил [сверточную архитектуру](#) с GPT (тк трансляция [caffemodel](#) → pytorch это АД, а в статье никакой подробной инфы нет)
p.s. использовал CIFAR100 model тк наиболее близка к задаче (не смотря на то что в [SimpleNet](#) есть IMAGENET model в репо, которая, конечно, не оптимальна из-за размера imgs)
6. трейн датасет 100k, модель 5M, так что эксперименты дорогие (не хочется брать подвыборку трейн, тк модель большая) ⇒
7. попытка сделать модель тоньше (салатовый график в wandb) → неудачно
p.s. стоило потестить сделать модель менее глубокой
8. так что предполагая архитектуру оптимальной (согласно статье) к CIFAR100, попробовал выжать больше из оптимизации
p.s. использовал простую аугментацию с RandomHorizontalFlip, RandomRotation(10), ColorJitter, and Normalize($\vec{\mu}$, $\vec{\sigma}$) (где $\vec{\mu}$ и $\vec{\sigma}$ взяты из распределений train/val/test TinyImageNet) как обычно делают авторы таких моделей

experiments

- batch 64 | 128
- lr 5e-4 | 3e-3
- рестарт с меньшим lr, MultiStepLR scheduler

лучший подход:

1. рестарт обученной (batch_64, lr_5e-4) с 29 эпохи с новым lr_5e-5 и MultiStepLR (с milestone 0.1 на 60 эпохе)
2. использование ансамбля из epoch_81, epoch_87, epoch_91 → average(probs) для улучшения acc 0.58 → 0.6
p.s. ансамбль prob1·prob2·prob3 и vote(pred1, pred2, pred3) работают хуже

ВЫВОД

- рестарты (сбрасывать стейт optimizer'а, менять сид и уменьшать lr) делать круто
- жаль что не сработал эксперимент [huge LR with MultiStepLR](#) (с идеей сначала поискать глобальный минимум, а потом залезть в него с помощью MultiStepLR)
- MultiStepLR в принципе не принес пользы
- все еще нет интуиции по устройству топологии train-loss'а
- не получилось объяснить очень странное поведение train_loss и train_аccuracy на [лучшем эксперименте](#) (которое не повлияло на поведение val статистик)

P.P.S. в google drive только epoch_81 чекпоинт, который дает accuracy около 0.58