THE LOTTERY TICKET HYPOTHESIS

Работу выполнили:
студенты НИУ ВШЭ ПМИ 182
Пак Ди Ун
Гольдман Артур
Голован Сергей
Котельников Аким

План работы

1. Презентация общего метода и обзор результатов экспериментов

План работы

1. Презентация общего метода и обзор результатов экспериментов
2. Рецензирование на рассматриваемую статью

План работы

1. Презентация общего метода и обзор результатов экспериментов
2. Рецензирование на рассматриваемую статью
3. Представление исследования контекста работы

План работы

1. Презентация общего метода и обзор результатов экспериментов
2. Рецензирование на рассматриваемую статью
3. Представление исследования контекста работы
4. Обзор применения данного метода на практике

Минусы моделей с большим количеством параметров

1. Тенденция таких моделей переобучаться

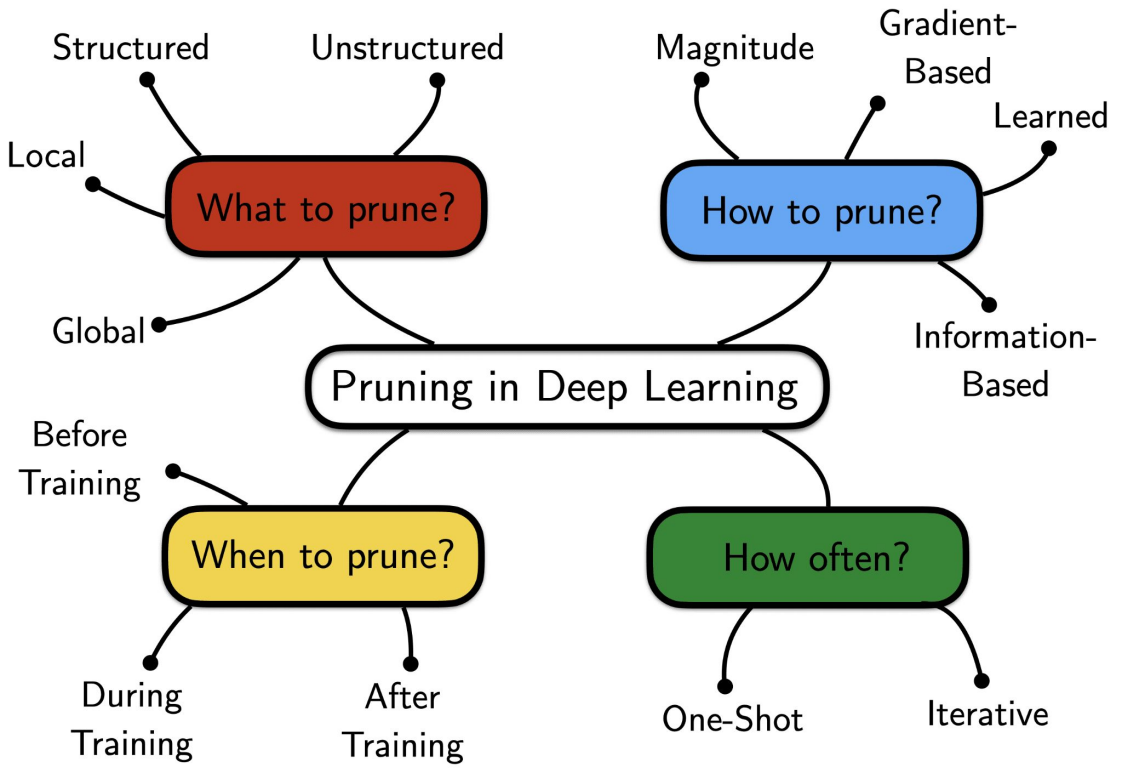
Минусы моделей с большим количеством параметров

1. Тенденция таких моделей переобучаться
2. Большой вес модели в памяти

Минусы моделей с большим количеством параметров

1. Тенденция таких моделей переобучаться
2. Большой вес модели в памяти
3. Повышенные требования на время исполнения, конфигурацию и объем памяти во время работы

Прунинг


Structured
Unstructured
Local
What to prune?
Global
Magnitude
Gradient-Based
Learned
How to prune?
Information-Based
Pruning in Deep Learning
Before Training
When to prune?
During Training
After Training
How often?
One-Shot
Iterative

Прунинг со старта

- Learning both Weights and Connections for Efficient Neural Networks (NIPS 2015)
  - Лучше дообучать после прунинга, чем переучивать

The Lottery Ticket Hypothesis

A randomly-initialized, dense neural network contains a subnetwork that is initialised such that — when trained in isolation — it can match the test accuracy of the original network after training for at most the same number of iterations. - Frankle & Carbin (2019, p.2)

Предлагаемый алгоритм

1. Случайным образом инициализируем модель $f(x; \theta_0)$

Предлагаемый алгоритм

1. Случайным образом инициализируем модель $f(x; \theta_0)$
2. Обучаем модель j итераций, получаем параметры модели θ_j

Предлагаемый алгоритм

1. Случайным образом инициализируем модель $f(x; \theta_0)$
2. Обучаем модель j итераций, получаем параметры модели θ_j
3. Используем прунинг для весов θ_j для p% параметров, получая маску m

Предлагаемый алгоритм

1. Случайным образом инициализируем модель $f(x; \theta_0)$
2. Обучаем модель j итераций, получаем параметры модели θ_j
3. Используем прунинг для весов θ_j для p% параметров, получая маску m
4. Возвращаем остальные параметры в исходное состояние θ_0, получив $f(x; m \odot \theta_0)$.

Предлагаемый алгоритм

1. Случайным образом инициализируем модель $f(x; \theta_0)$
2. Обучаем модель j итераций, получаем параметры модели θ_j
3. Используем прунинг для весов θ_j для p% параметров, получая маску m
4. Возвращаем остальные параметры в исходное состояние θ_0, получив $f(x; m \odot \theta_0)$.
5. Повторим 2-4 до нужной степени прунинга

Experiments

1. Наборы данных
    a. MNIST
    b. CIFAR10

Experiments

1. Наборы данных
2. Модели
    a. Lenet
    b. Conv-2
    c. Conv-4
    d. Conv-6
    e. Resnet-18
    f. VGG-19

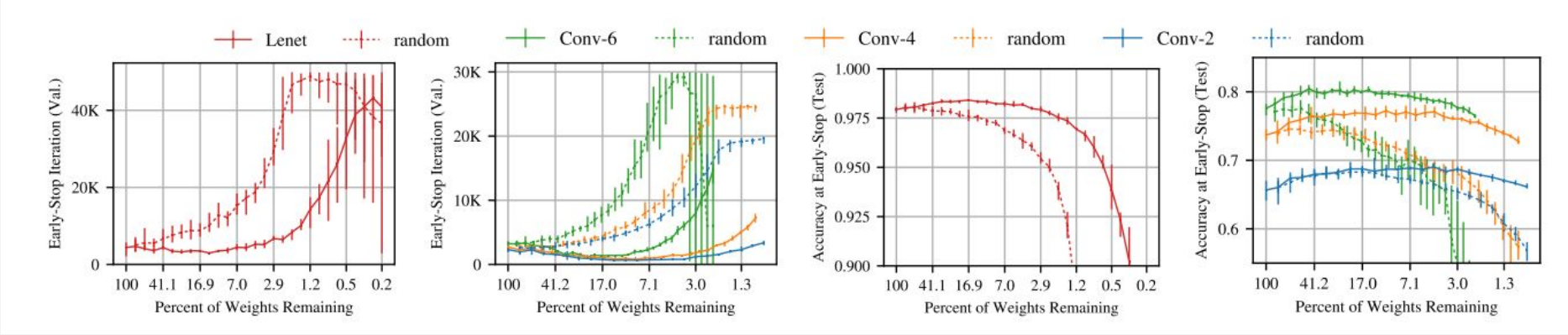
Experiments

1. Наборы данных
2. Модели
3. Оптимизаторы
    a. SGD
    b. momentum
    c. Adam

Experiments

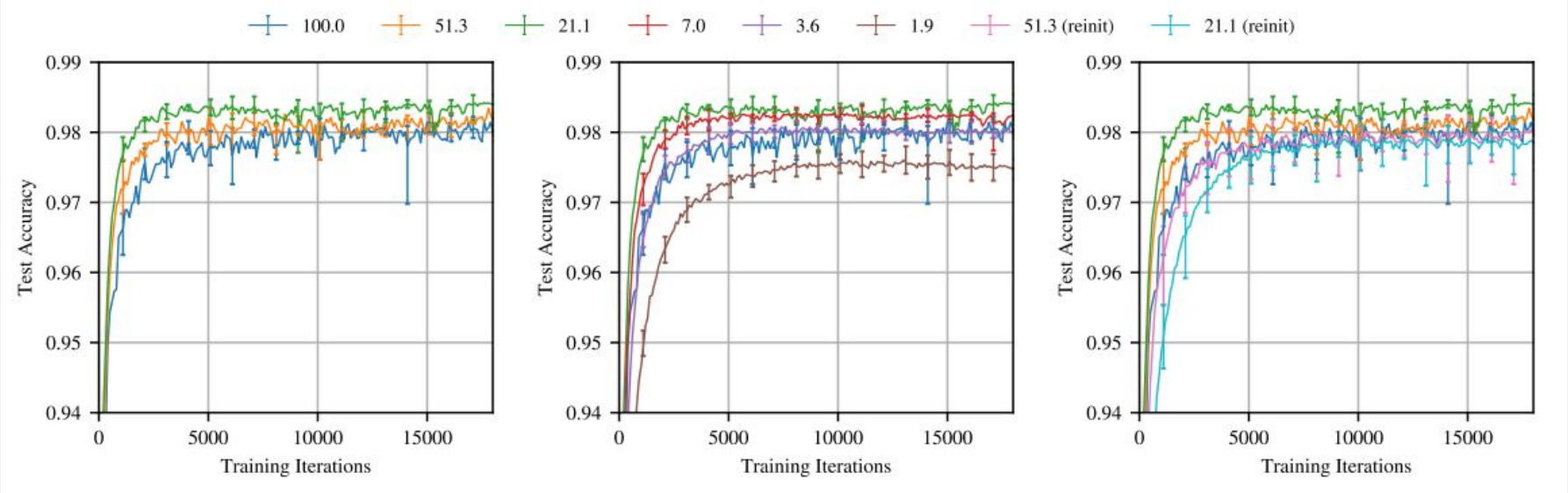
Network	Lenet	Conv-2	Conv-4	Conv-6	Resnet-18	VGG-19
Convolutions		64, 64, pool	64, 64, pool 128, 128, pool	64, 64, pool 128, 128, pool 256, 256, pool	16, 3x[16, 16] 3x[32, 32] 3x[64, 64]	2x64 pool 2x128 pool, 4x256, pool 4x512, pool, 4x512
FC Layers	300, 100, 10	256, 256, 10	256, 256, 10	256, 256, 10	avg-pool, 10	avg-pool, 10
All/Conv Weights	266K	4.3M / 38K	2.4M / 260K	1.7M / 1.1M	274K / 270K	20.0M
Iterations/Batch	50K / 60	20K / 60	25K / 60	30K / 60	30K / 128	112K / 64
Optimizer	Adam 1.2e-3	Adam 2e-4	Adam 3e-4	Adam 3e-4	← SGD 0.1-0.01-0.001 Momentum 0.9 →	
Pruning Rate	fc20%	conv10% fc20%	conv10% fc20%	conv15% fc20%	conv20% fc0%	conv20% fc0%

Experiments overall



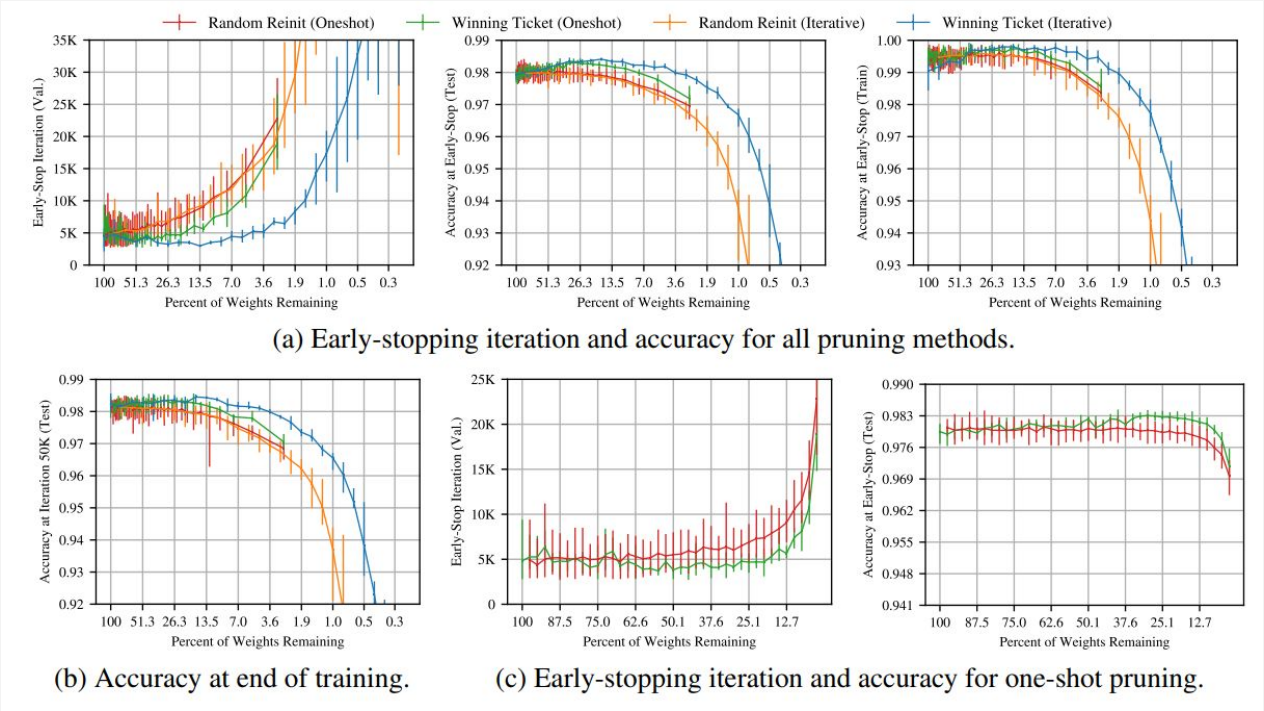
The iteration at which early-stopping would occur (left) and the test accuracy at that iteration (right) of the Lenet architecture for MNIST and the Conv-2, Conv-4, and Conv-6 architectures for CIFAR10 (see Figure 2) when trained starting at various sizes. Dashed lines are randomly sampled sparse networks (average of ten trials). Solid lines are winning tickets (average of five trials).

Experiments LeNet



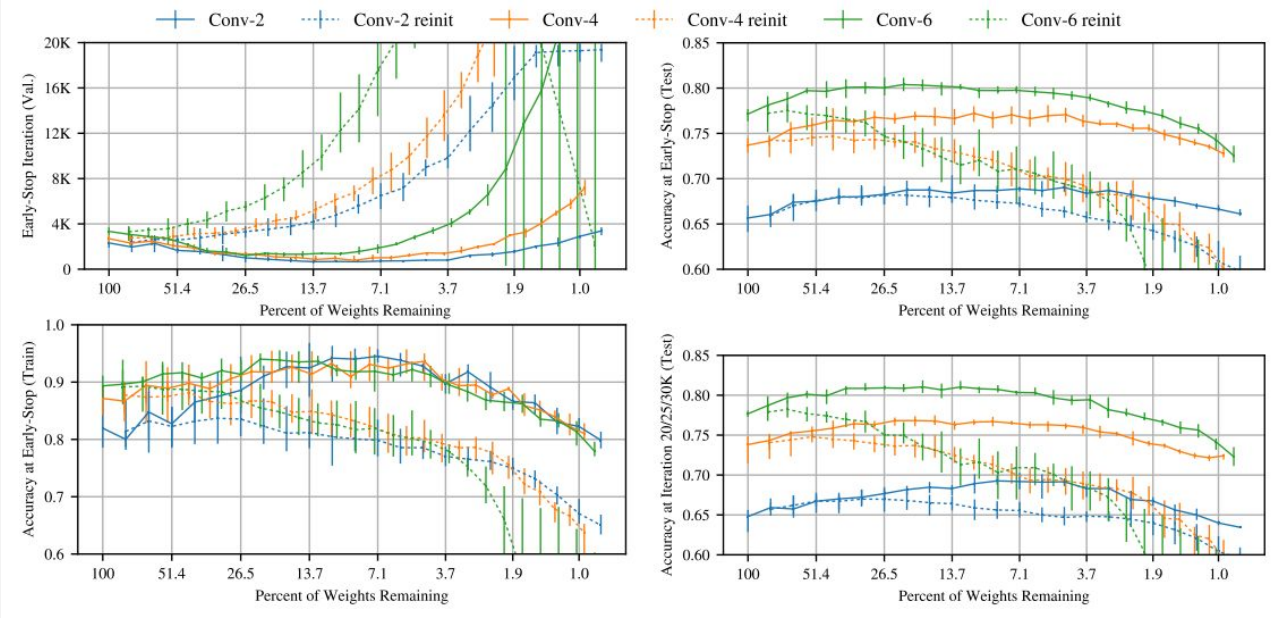
Test accuracy on Lenet (iterative pruning) as training proceeds. Each curve is the average of five trials. Labels are Pm — the fraction of weights remaining in the network after pruning. Error bars are the minimum and maximum of any trial.

Experiments OneShot vs Iterative



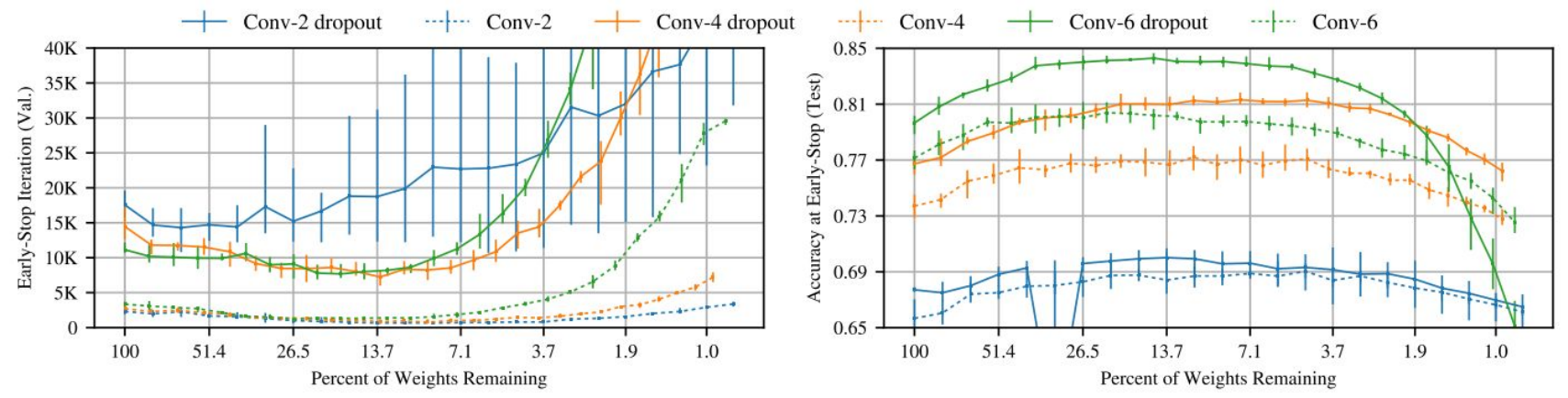
Early-stopping iteration and accuracy of Lenet under one-shot and iterative pruning. Average of five trials; error bars for the minimum and maximum values. At iteration 50,000, training accuracy ≈ 100% for Pm ≥ 2% for iterative winning tickets

Experiments Conv architecture



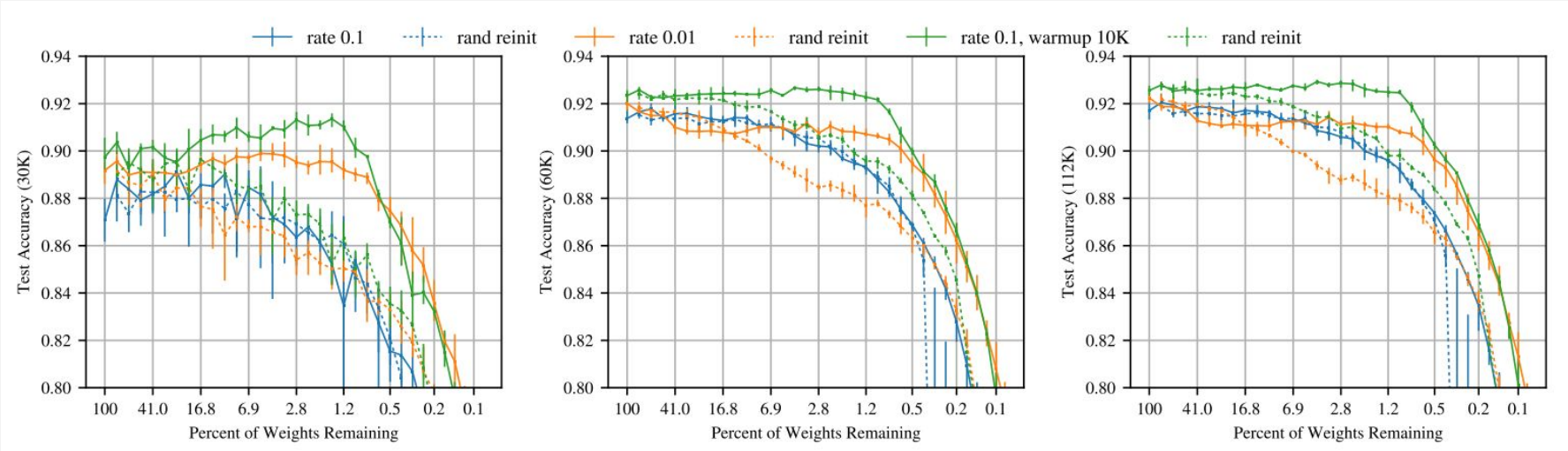
Early-stopping iteration and test and training accuracy of the Conv-2/4/6 architectures when iteratively pruned and when randomly reinitialized. Each solid line is the average of five trials; each dashed line is the average of fifteen reinitializations (three per trial). The bottom right graph plots test accuracy of winning tickets at iterations corresponding to the last iteration of training for the original network (20,000 for Conv-2, 25,000 for Conv-4, and 30,000 for Conv-6); at this iteration, training accuracy ≈ 100% for Pm ≥ 2% for winning tickets.

Experiments with dropout



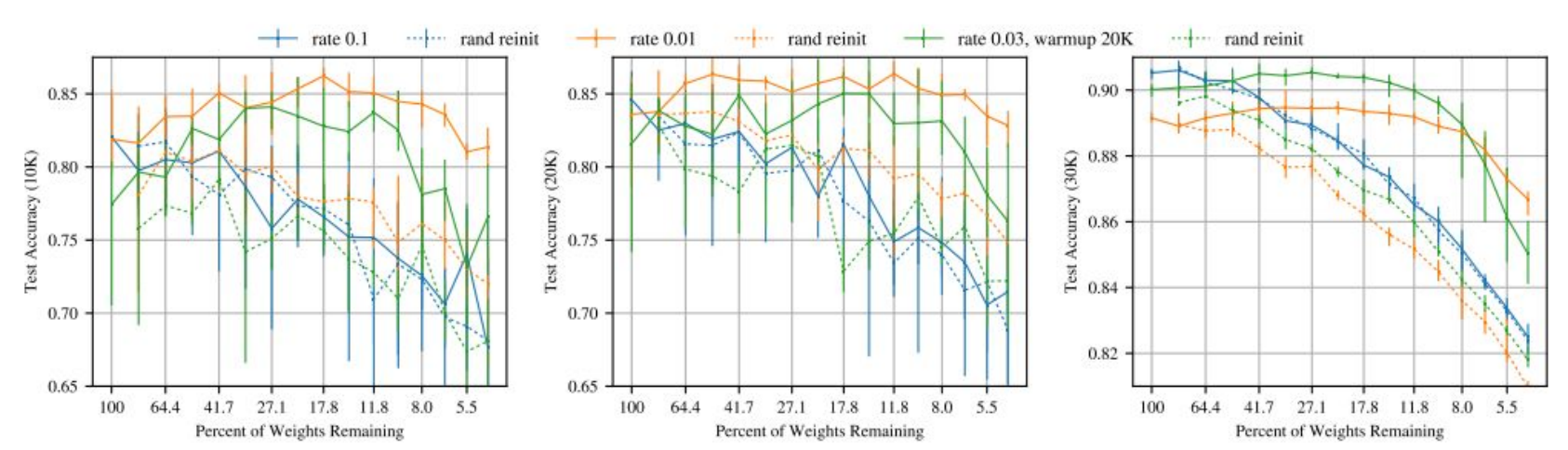
Early-stopping iteration and test accuracy at early-stopping of Conv-2/4/6 when iteratively pruned and trained with dropout. The dashed lines are the same networks trained without dropout (the solid lines in Figure 5). Learning rates are 0.0003 for Conv-2 and 0.0002 for Conv-4 and Conv-6.

Experiments VGG-19



Test accuracy (at 30K, 60K, and 112K iterations) of VGG-19 when iteratively pruned.

Experiments ResNet-18



Test accuracy (at 10K, 20K, and 30K iterations) of Resnet-18 when iteratively pruned.

Возможные вопросы

1. Почему сохранение инициализации так важно?

Возможные вопросы

1. Почему сохранение инициализации так важно?
2. Почему мы получаем лучшее качество?

Проблемы и направления для исследований

1. Использованы только маленькие наборы данных

Проблемы и направления для исследований

1. Использованы только маленькие наборы данных
2. Поиск выигрышных билетов осуществляется только прунингом

Проблемы и направления для исследований

1. Использованы только маленькие наборы данных
2. Поиск выигрышных билетов осуществляется только прунингом
3. Не исследованы свойства полученных моделей, которые позволяют сходится к оптимуму лучше, чем оригинал

Проблемы и направления для исследований

1. Использованы только маленькие наборы данных
2. Поиск выигрышных билетов осуществляется только прунингом
3. Не исследованы свойства полученных моделей, которые позволяют сходится к оптимуму лучше, чем оригинал
4. Поиск лотерейных билетов в больших сетях невозможен без warmup lr

Выводы

С помощью итеративного прунинга можно находить подсети, достигающие сравнимый или превосходящий результат на тестовой выборке

Полученные выигрышные билеты обучаются быстрее и лучше, а их генерализирующая способность растет

Источники

1. Frankle J., Carbin M. The lottery ticket hypothesis: Finding sparse, trainable neural networks //arXiv preprint arXiv:1803.03635. – 2018.

Рецензия

В статье проведено исследование гипотезы о наличии т.н. Lottery ticket: подсети небольшого размера, которая добивается качества не хуже, чем исходная. Данная гипотеза дает надежду на использование моделей меньшего размера с более ранним временем остановки обучения.

Рецензия

Плюсы:

- Качественная работа с литературой. Эксперименты при необходимости воспроизведены и результаты сравниваются с гипотезой
- Много промежуточных итогов, выводов и обсуждений. Статью легко читать, за мыслью легко следить
- Есть имплементация от авторов (в статье не указано, но есть github репозиторий под их авторством с реализацией). Легко воспроизводима

Рецензия

Минусы:

- Строго эмпирическое исследование. Проведено огромное количество качественных эмпирических экспериментов, большинство из которых либо без попытки понять причины итогов, либо с небольшими догадками
- Исследовано только на моделях и задачах компьютерного зрения на небольших наборах данных (MNIST, CIFAR10)

Рецензия

Дальнейшие пожелания:

- Эта гипотеза предполагает наличие сетей специальной структуры, которая небольшая по размеру, но добивается хорошей точности. Как можно пытаться получать ее изначально: задавать эту структуру и подбирать правильную инициализацию? Возможно ли это вообще?

Рецензия

NIPS-like mark: 8/10

NIPS-like confidence mark: 4/5

На ICLR-2019 статья получила Best paper award.