THE LOTTERY TICKET HYPOTHESIS

FINDING SPARSE, TRAINABLE NEURAL NETWORKS

PRESENTATION ACCOMPLISHED BY BOREVSKY ANDREY

What are **Subnetworks?**

Definition

Neural subnetwork is a sequence of simple operations that are part of a greater architecture, but able to function separately.

What are **Subnetworks?**

Definition

Neural subnetwork is a sequence of simple operations that is part of a greater architecture, but able to function separately.

Example

Convolutional Neural Network.

One block of 1) convolution 2) activation function & 3) batch normalization can be thought of as subnetwork.

What is a **Pruning technique**?

Problem

Neural networks are over-parameterized.
This results in a waste of both computation and memory.

What is a **Pruning technique**?

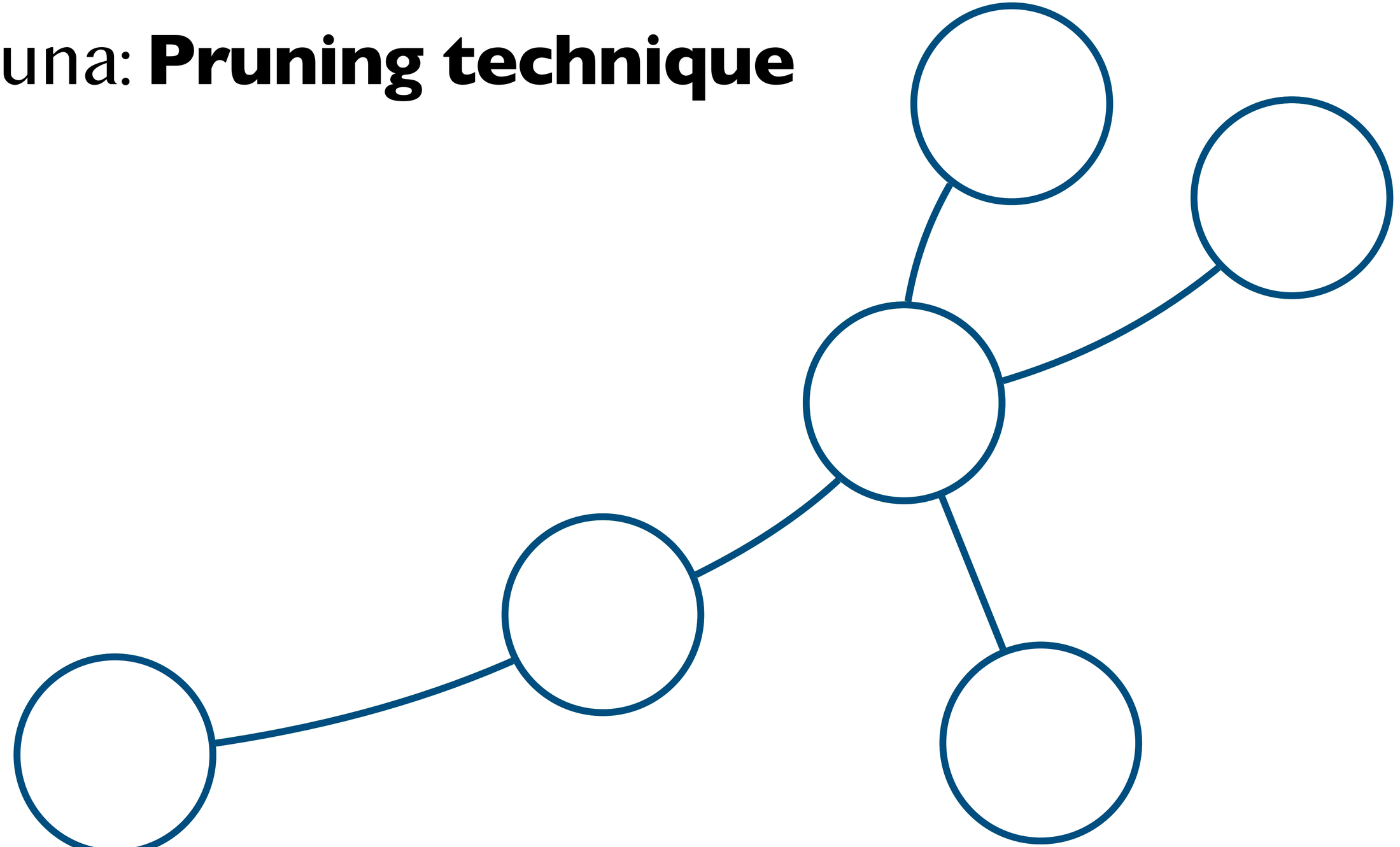
Problem

Neural networks are over-parameterized. This results in a waste of both computation and memory.

Definition

Algorithms, eliminating unnecessary weights from neural networks and preserving model's accuracy.

Optuna: **Pruning technique**

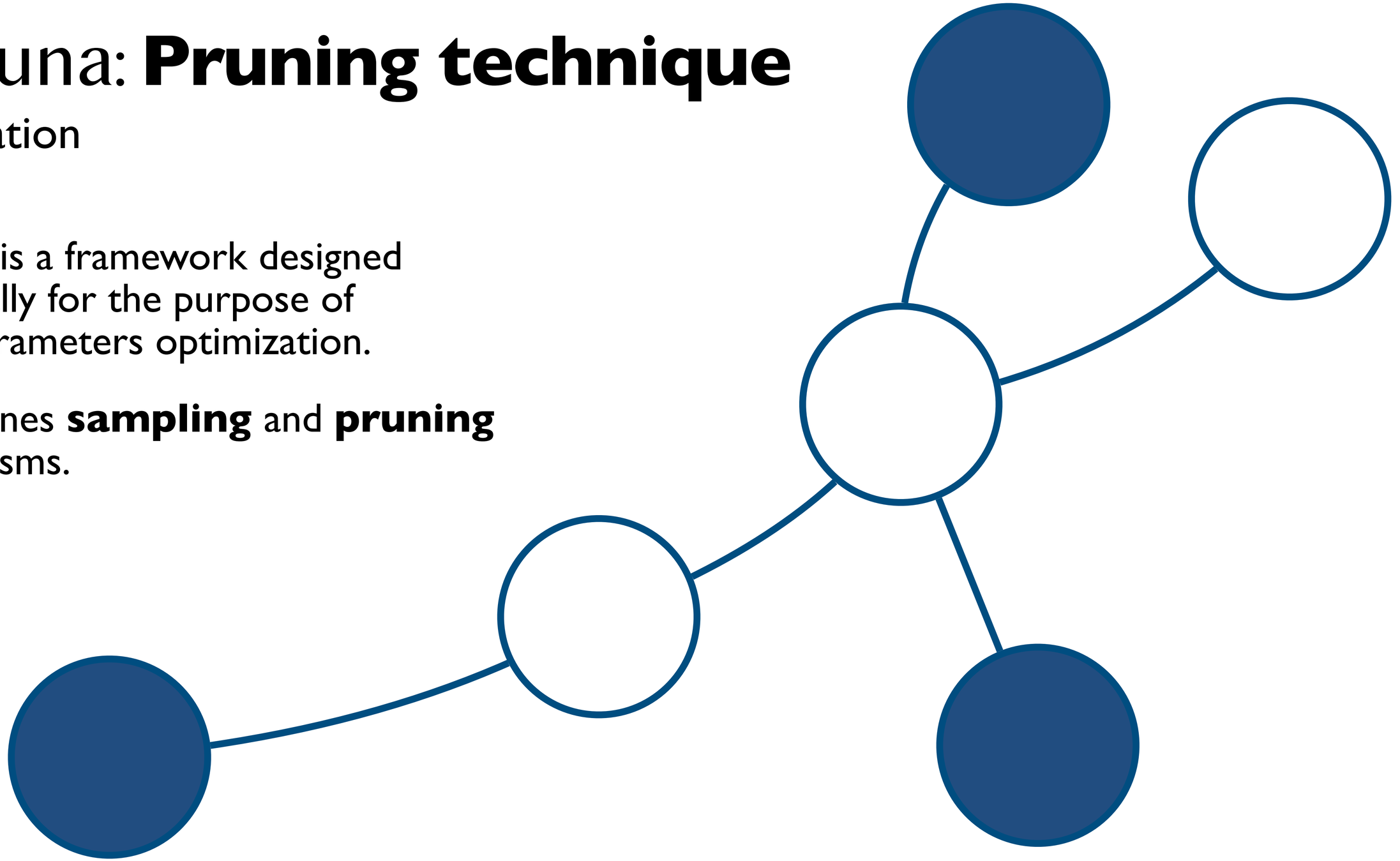


Optuna: **Pruning technique**

Explanation

Optuna is a framework designed specifically for the purpose of hyperparameters optimization.

It combines **sampling** and **pruning** mechanisms.



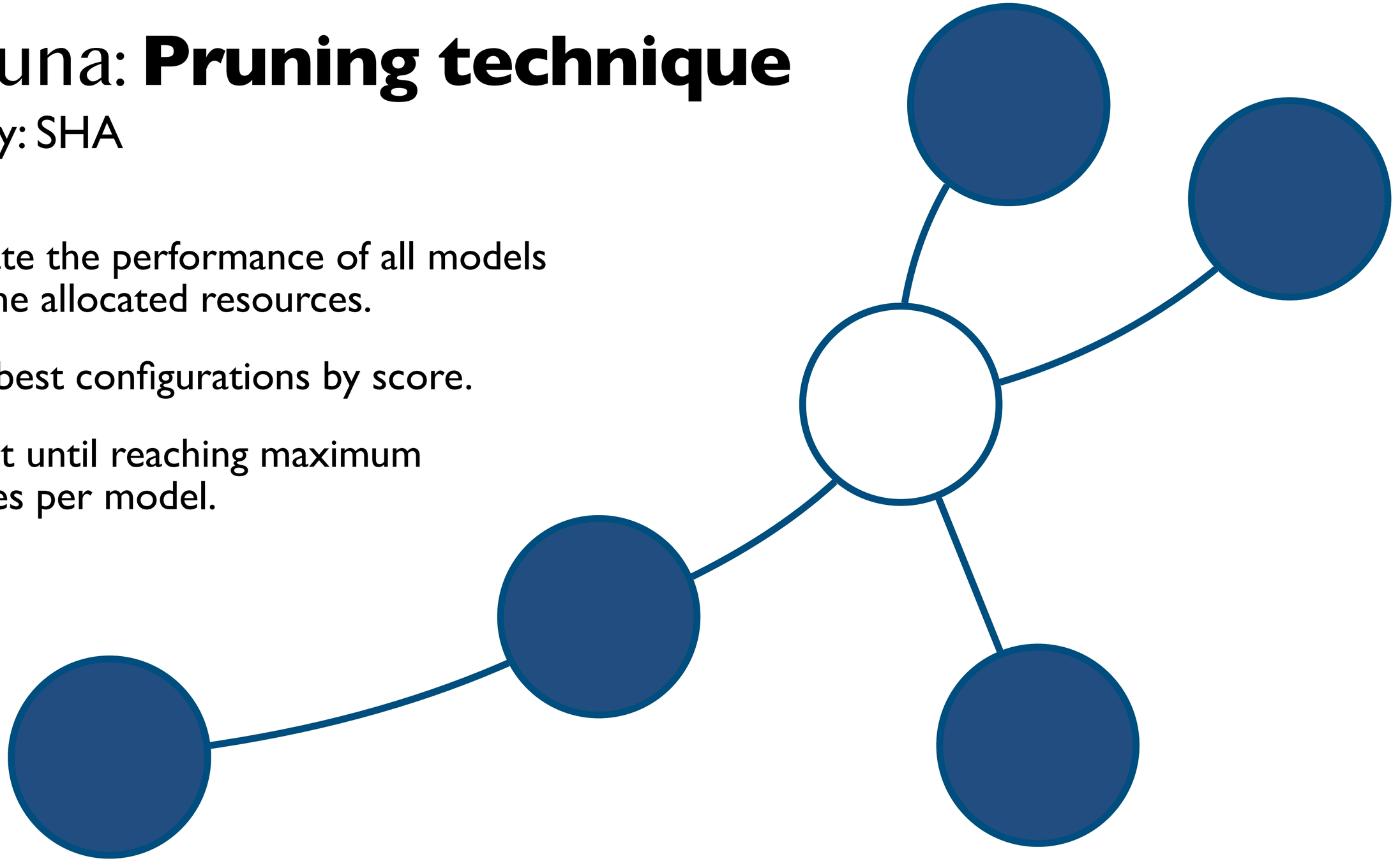
Optuna: **Pruning technique**

Strategy: SHA

1. Evaluate the performance of all models within the allocated resources.

2. Keep best configurations by score.

3. Repeat until reaching maximum resources per model.



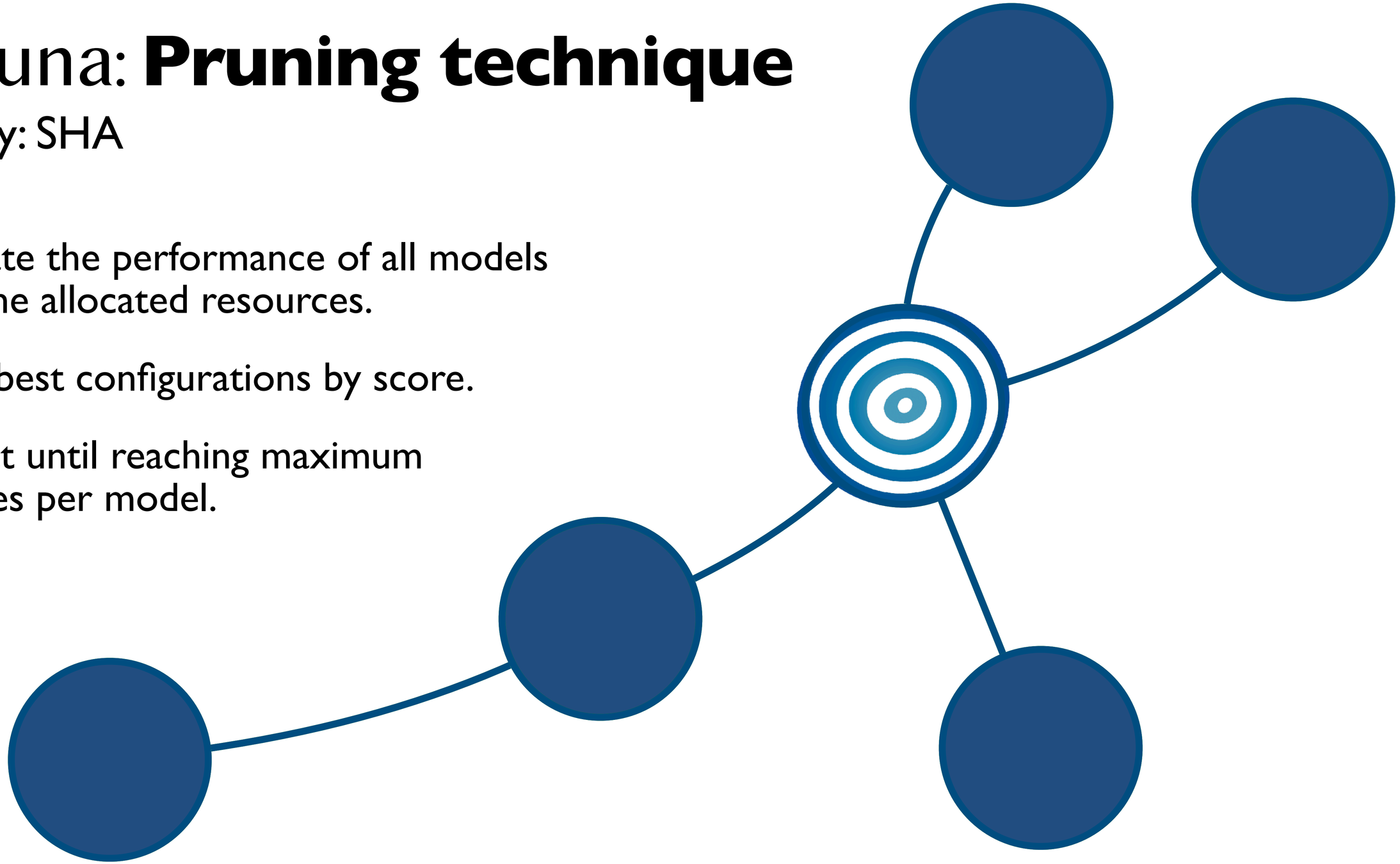
Optuna: **Pruning technique**

Strategy: SHA

1. Evaluate the performance of all models within the allocated resources.

2. Keep best configurations by score.

3. Repeat until reaching maximum resources per model.



Hm, if a network can be reduced in size…

Hm, if a network can be reduced in size…

why don’t we train the smaller architecture instead?

Well, there is a problem…

What is the **problem?**

Statement

- Architectures uncovered by pruning are harder to train from the start, reaching lower accuracy than the original networks
- Training a pruned model from scratch performs worse than retraining a pruned model.
- The sparser the network after pruning, the slower the learning and the lower the eventual test accuracy.

What is the **solution?**

The Lottery Ticket Hypothesis

A randomly-initialized, dense neural network contains a subnetwork that is initialized 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.

What is the **solution?**

The Lottery Ticket Hypothesis: Explanation

1. Train 1st NN

$$f(x, w_0) \xrightarrow{SGD} \begin{cases} L(f(x_v, w_j)) = \min_w L(f(x_v, w)) = \ell \\ \text{test accuracy} = a \end{cases}$$

What is the **solution?**

The Lottery Ticket Hypothesis: Explanation

1. Train 1st NN

$$f(x, w_0) \xrightarrow{SGD} \begin{cases} L(f(x_v, w_j)) = \min_w L(f(x_v, w)) = \ell \\ \text{test accuracy} = a \end{cases}$$

2. Train 2nd NN with mask $m \in \{0,1\}^{|w|}$

$$f(x, w_0 \odot m) \xrightarrow{SGD} \begin{cases} L(f(x_v, w_{j'})) = \min_w L(f(x_v, w)) = \ell' \\ \text{test accuracy} = a' \end{cases}$$

What is the **solution?**

The Lottery Ticket Hypothesis: Explanation

1. Train 1st NN

$$f(x, w_0) \xrightarrow{SGD} \begin{cases} L(f(x_v, w_j)) = \min_w L(f(x_v, w)) = \ell \\ \text{test accuracy} = a \end{cases}$$

2. Train 2nd NN with mask $m \in \{0,1\}^{|w|}$

$$f(x, w_0 \odot m) \xrightarrow{SGD} \begin{cases} L(f(x_v, w_{j'})) = \min_w L(f(x_v, w)) = \ell' \\ \text{test accuracy} = a' \end{cases}$$

3. Prediction of hypothesis

$$\exists m : \begin{cases} \text{commensurate training time: } j' \leq j \\ \text{commensurate accuracy: } a \leq a' \\ \text{fewer parameters: } \|m\|_0 \ll \|w\| \end{cases}$$

What is the **solution?**

The Lottery Ticket Hypothesis: Explanation

1. Train 1st NN

$f(x, w_0) \xrightarrow{SGD}$

2. Train 2nd NN with mask

3. Prediction of hypothesis

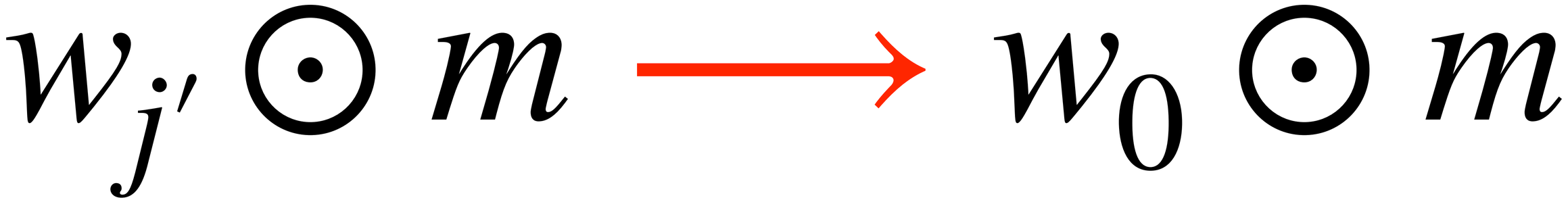
$\exists m$

$\|w\|$

$f(x, w_0 \odot m)$ is a winning ticket!

It won the initialization lottery with a combination of weights and connections capable of learning.

If reinitialised randomly, it no longer matches original network's performance



Unique to the work, each unpruned connection is reset to its initialization value.

Mastering the **pruning technique**

Identifying winning tickets

1. Randomly initialize a neural network $f(x, w_0)$
2. Train the model till reaching minimum validation loss
3. Prune $p\,\%$ of least-magnitude weights, hereby creating a mask m
4. Reset the remaining weights to initial values w_0
5. Run the winning ticket $f(x, m \odot w_0)$

Mastering the **pruning technique**

Identifying winning tickets

1. Randomly initialize a neural network $f(x, w_0)$
2. Train the model till reaching minimum validation loss
3. Prune $p\,\%$ of weights, hereby creating a mask m
4. Reset the remaining weights to initial values w_0
5. Run the winning ticket $f(x, m \odot w_0)$

Iterative pruning

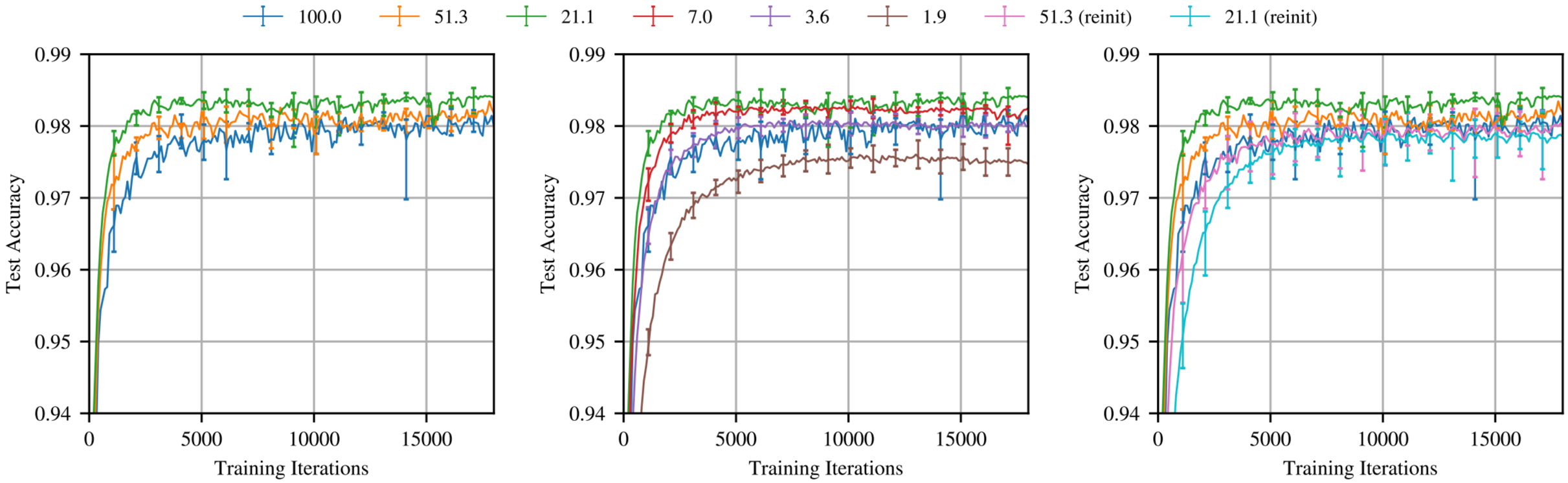
One-shot pruning

Iterations VS One-shot

- Although iterative pruning extracts smaller winning tickets, repeated training means they are costly to find.
- According to results, one-shot pruning does indeed find winning tickets.

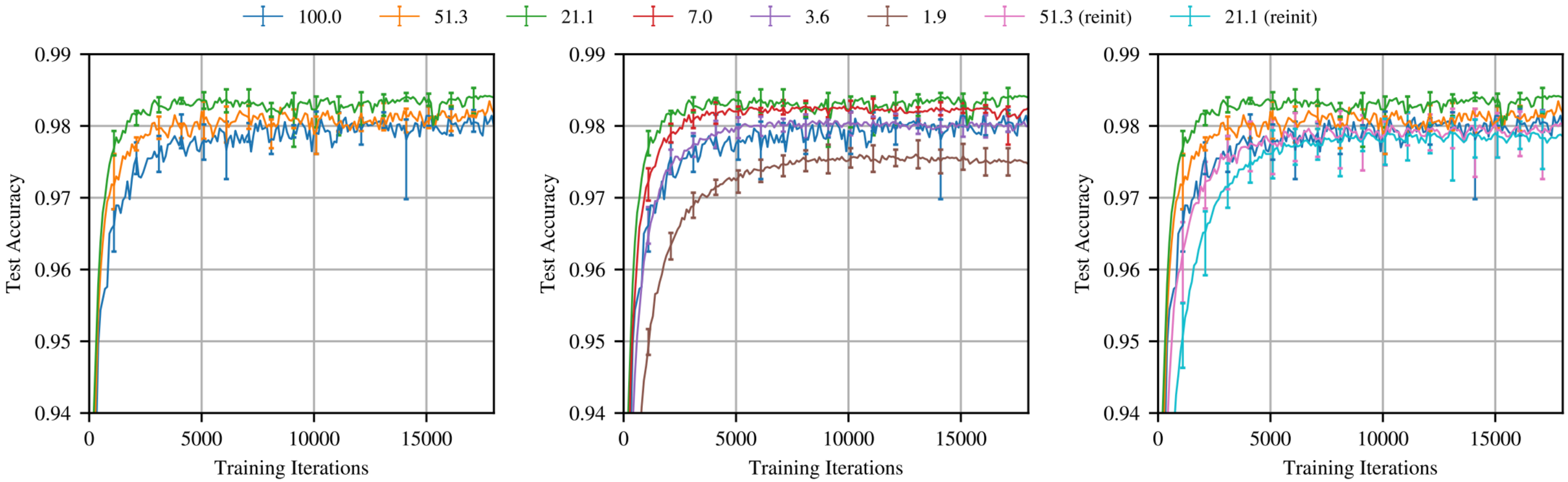
Fully-Connected Neural Network

Networks learn faster and reach higher test accuracy the more they are pruned (till 21%).



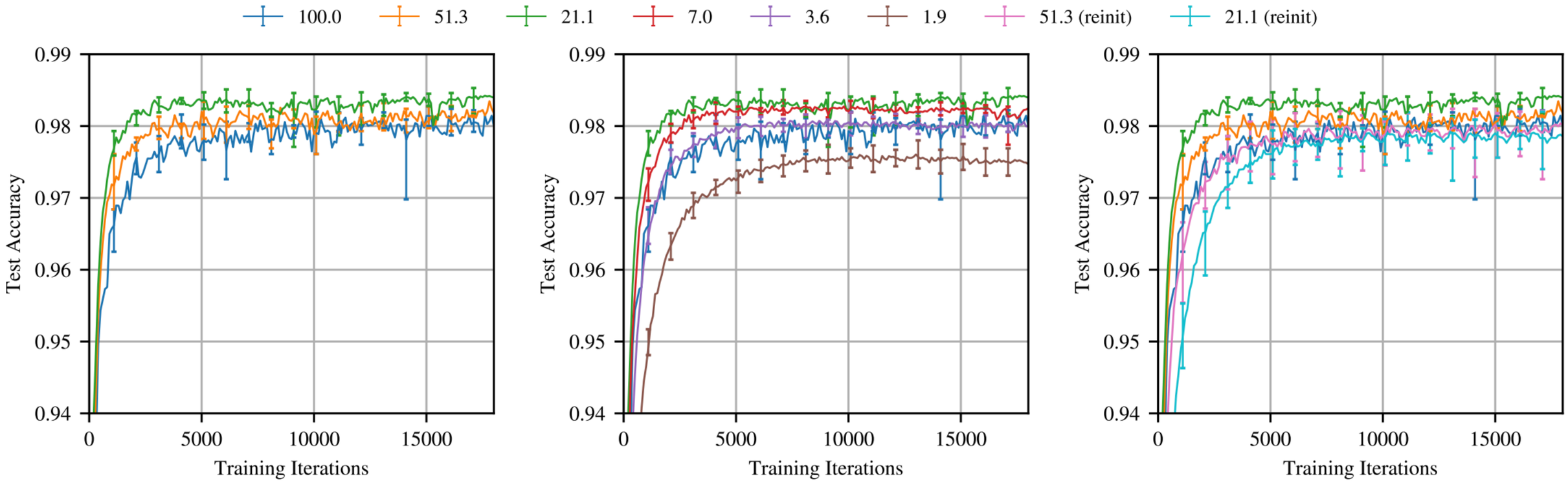
Fully-Connected Neural Network

Further pruning causes learning to slow, returning to the early-stopping performance of the original network.



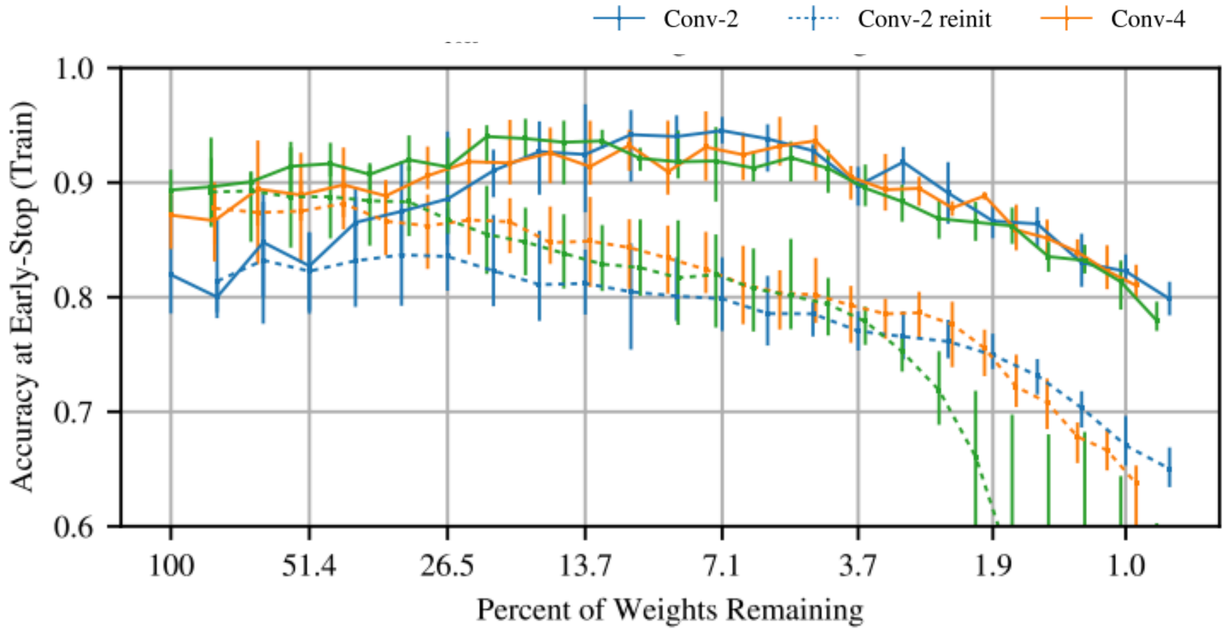
Fully-Connected Neural Network

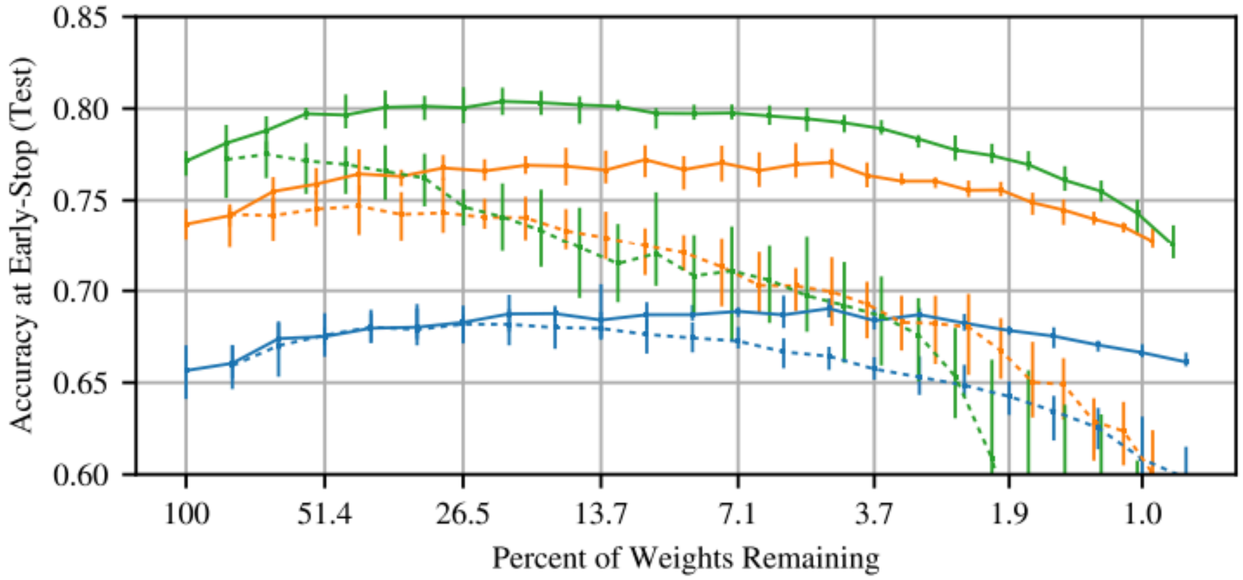
Where the winning tickets learn faster as they are pruned, they learn progressively slower when randomly reinitialized.



Convolutional Neural Network

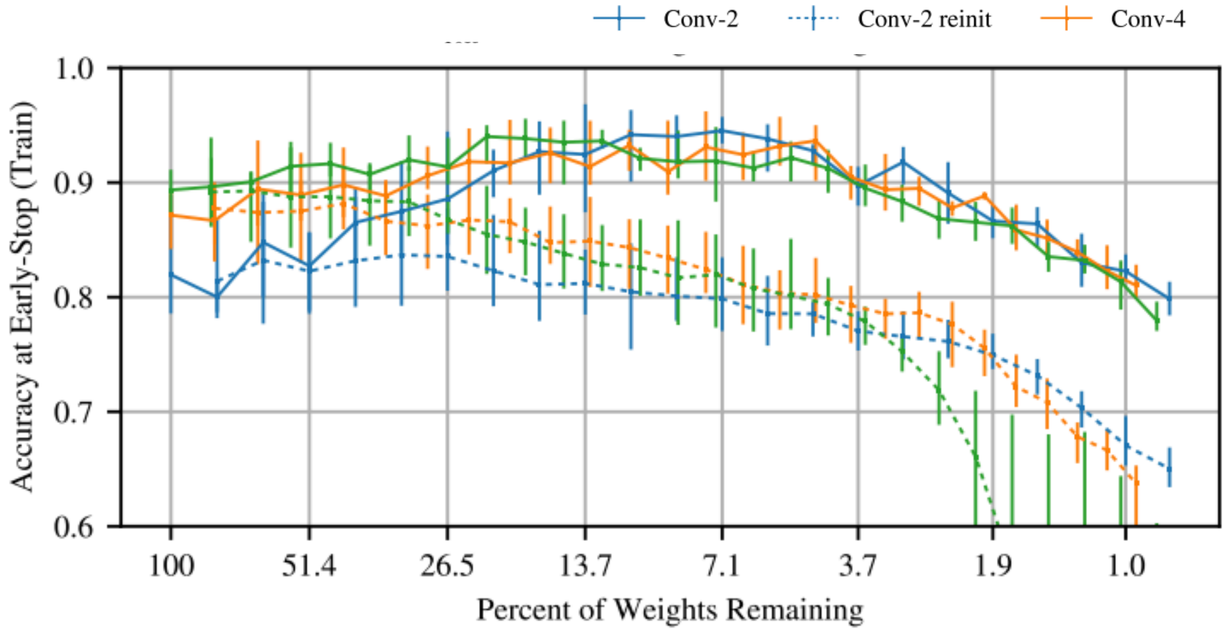
Previous theses are bolstered by the results from CNN.

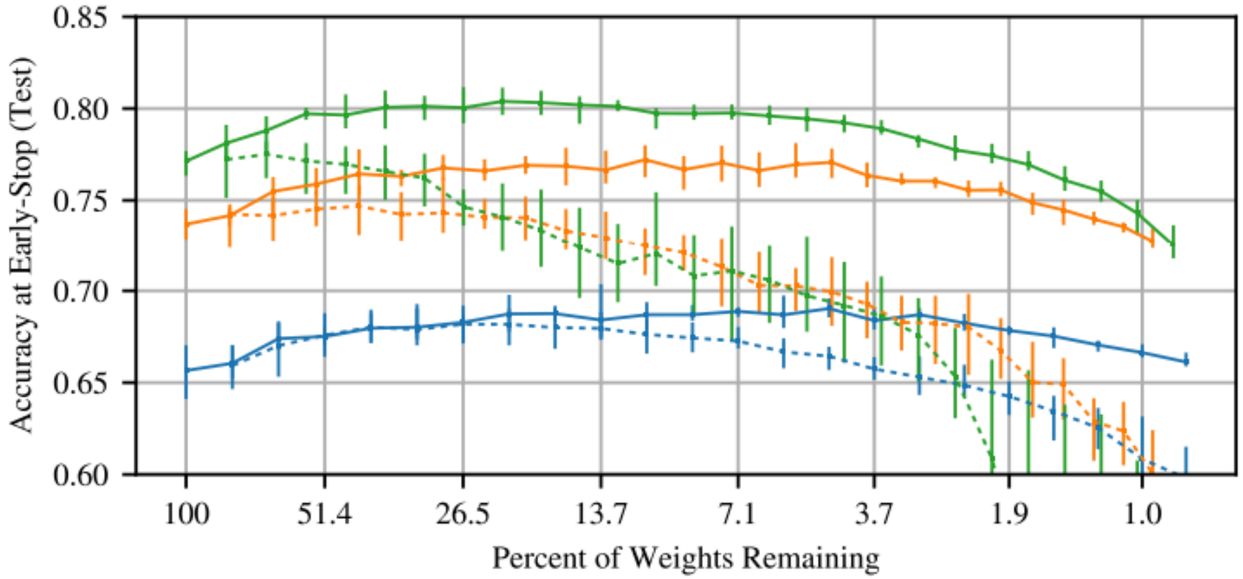




Convolutional Neural Network

The gap between test and training accuracy is smaller for winning tickets, indicating they generalize better.





What is **Dropout?**

Definition

This method randomly disables a certain fraction of units on each training iteration, in results ameliorating accuracy.

It can be said that dropout also creates a subnetwork.

Dropout + Winning ticket = ?

We continue to find winning tickets when using dropout. Accuracy becomes higher.

These improvements suggest that iterative pruning strategy interacts with dropout in a complementary way.

Real-life models: VGG & ResNet

Experiments on the models confirmed all the presented statements.

Global pruning was used:

- The lowest-magnitude weights were removed collectively across all convolutional layers.

- When all layers are pruned at the same rate, smaller layers become bottlenecks, preventing from identifying the smallest possible winning tickets.