Large-Scale Evolution of Image Classifiers

Esteban Real[1] Sherry Moore[1] Andrew Selle[1] Saurabh Saxena[1]
Yutaka Leon Suematsu[2] Jie Tan[1] Quoc V.Le[1] Alexey Kurakin[1]

[1]Google Brain

[2]Google Research

ICML, 2017
Presenter: Tianlu Wang

Outline

Outline

Motivation

- AlexNet, GoogleNet, VGG, ResNet...
- Designing neural network architectures can be challenging
- Discover network architectures automatically

Outline

Backgrounds

- Achievements: evolution algorithm outputs a **fully-trained** model with no human participation
- Drawbacks: significant computation
- Image classification, CIFAR-10, CIFAR-100

Outline

Neuro-evolution

- Weight evolution: back propagation
- Weight and architecture: *NEAT algorithm* (node and connection)

Outline

Non-evolutionary

- Bayesian optimization
- Reinforcement learning
- Q-learning

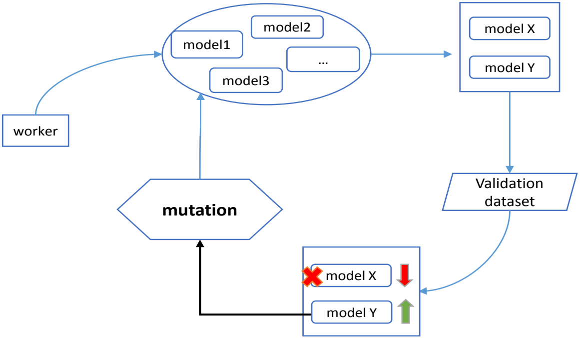
Outline

Algorithm Overview

- Input: a population of models, each model is a **trained single-layer nonconvolutional model** with learning_rate = 0.1
- Measurement: accuracy on validation dataset

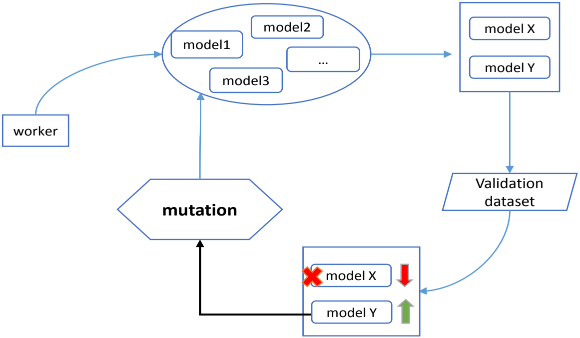
Algorithm Overview

- Input: a population of models, each model is a **trained single-layer nonconvolutional model** with learning_rate = 0.1
- Measurement: accuracy on validation dataset



Algorithm Overview

- Input: a population of models, each model is a **trained single-layer nonconvolutional model** with learning_rate $= 0.1$
- Measurement: accuracy on validation dataset



- When to stop?

Outline

Model Encoding

Individual model is encoded as a graph:

- Vertices
  - rank-3 tensor(image_width * image_height * channels)
  - activations(batch normalization with ReLU or plain linear layer)
- Edges
  - Identity connections
  - Convolutions

Model Encoding

Individual model is encoded as a graph:

- Vertices
  - rank-3 tensor(image_width * image_height * channels)
  - activations(batch normalization with ReLU or plain linear layer)
- Edges
  - Identity connections
  - Convolutions

Inconsistent input:

- pick and keep primary one
- reshape(interpolation/truncation/padding) non-primary ones

Mutations

The worker picks a mutation at random from a set:

- ALTER-LEARNING-RATE
- IDENTITY (effectively means keep training)
- RESET-WEIGHTS
- INSERT/REMOVE CONVOLUTION
- ALTER-STRIDE
- ALTER-NUMBER-OF-CHANNELS
- FILTER-SIZE
- INSERT-ONE-TO-ONE
- INSERT/REMOVE SKIP

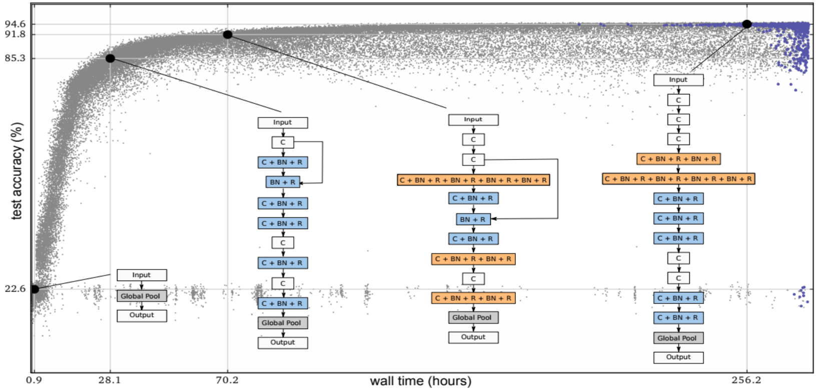
Outline

More Details

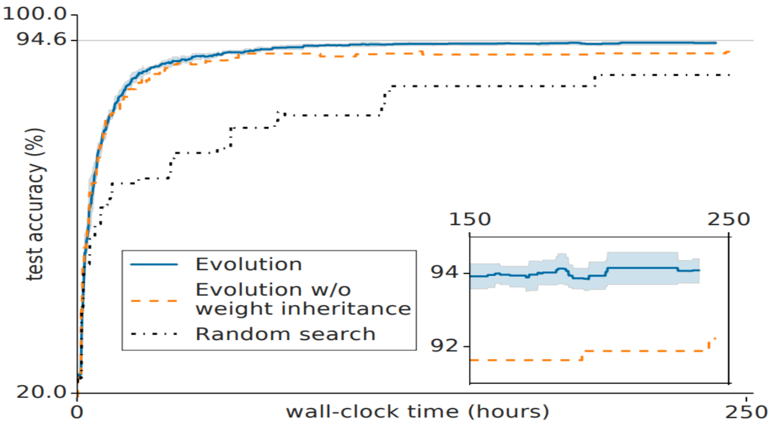
- Poor initial conditions(12th silde)
- 45,000 training; 5,000 validation; 10000 test
- SGD with momentum of 0.9, batch size 50, weight decay 0.0001
- Computation cost: floating-point operations
- Inherit parameters' weights whenever possible

Outline

Progress of an evolution experiment



Repeatability of results and controls



Outline

Compared to hand-designed networks

Study	Params.	C10+	C100+	Reachable?
Maxout (Goodfellow et al., 2013)	–	90.7%	61.4%	No
Network in Network (Lin et al., 2013)	–	91.2%	–	No
All-CNN (Springenberg et al., 2014)	1.3 M	92.8%	66.3%	Yes
Deeply Supervised (Lee et al., 2015)	–	92.0%	65.4%	No
Highway (Srivastava et al., 2015)	2.3 M	92.3%	67.6%	No
ResNet (He et al., 2016)	1.7 M	93.4%	72.8%†	Yes
Evolution (ours)	5.4 M	94.6%		N/A
	40.4 M		77.0%	
Wide ResNet 28-10 (Zagoruyko & Komodakis, 2016)	36.5 M	96.0%	80.0%	Yes
Wide ResNet 40-10+d/o (Zagoruyko & Komodakis, 2016)	50.7 M	96.2%	81.7%	No
DenseNet (Huang et al., 2016a)	25.6 M	96.7%	82.8%	No

Compared to auto-discovered networks

Study	Starting Point	Constraints	Post-Processing	Params.	C10+	C100+
Bayesian (Snoek et al., 2012)	3 layers	fixed architecture, no skips	none	–	90.5%	–
Q-Learning (Baker et al., 2016)	–	discrete params., max. num. layers, no skips	tune, retrain	11.2 M	93.1%	72.9%
RL (Zoph & Le, 2016)	20 layers, 50% skips	discrete params., exactly 20 layers	small grid search, retrain	2.5 M	94.0%	–
RL (Zoph & Le, 2016)	39 layers, 2 pool layers at 13 and 26, 50% skips	discrete params., exactly 39 layers, 2 pool layers at 13 and 26	add more filters, small grid search, retrain	37.0 M	96.4%	–
Evolution (ours)	single layer, zero convs.	power-of-2 strides	None	5.4 M 40.4 M ensemb.	94.6% 95.6%	77.0%

Outline

Improve the method

- Large population size
- More training steps
- Increase mutation rate
- Reset all weights

Summary

- Neuro-evolution starts from trivial initial conditions and yields fully trained models
- Construct large, accurate networks for two challenging and popular image classification benchmarks
- Large search space and high computation cost