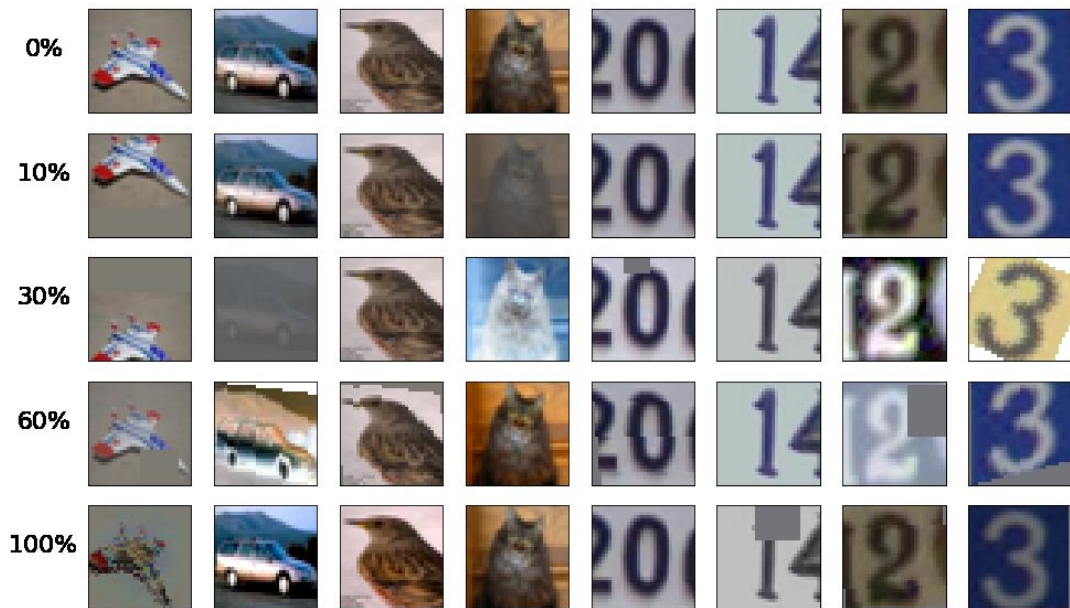


Population Based Augmentation

Efficient Learning of Augmentation Policy Schedules

Daniel Ho, Eric Liang, Ion Stoica, Pieter Abbeel, Xi Chen

Efficiently learn data augmentation policies to improve neural network performance.



Data Augmentation

Most models only use basic data augmentation strategies.



Original



Horizontal Flip



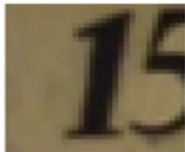




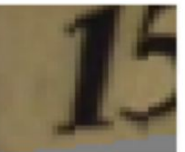
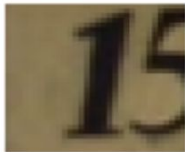

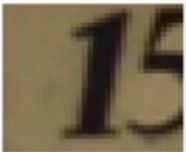



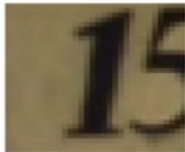



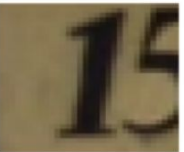
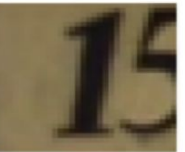
Pad & Crop



Rotate

Augmentation with AutoAugment

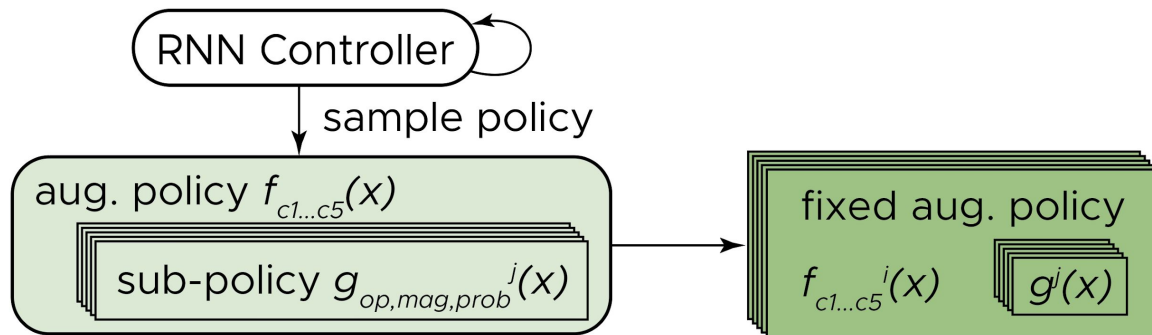
Learns operations to apply with certain probability and magnitude.

	Original	Sub-policy 1	Sub-policy 2	Sub-policy 3	Sub-policy 4	Sub-policy 5
Batch 1						
Batch 2						
Batch 3						

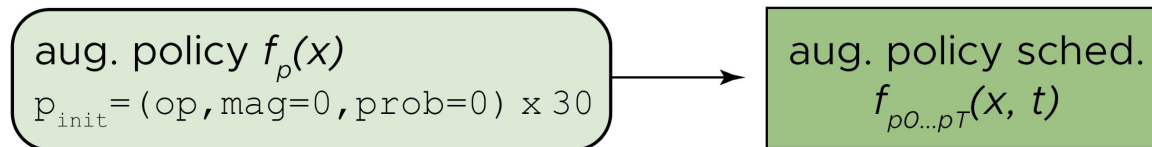
What's the catch?

AutoAugment is too computationally expensive to learn. Our algorithm, PBA, uses 1000x less compute.

AutoAugment: Trained with PPO on 15,000 child models

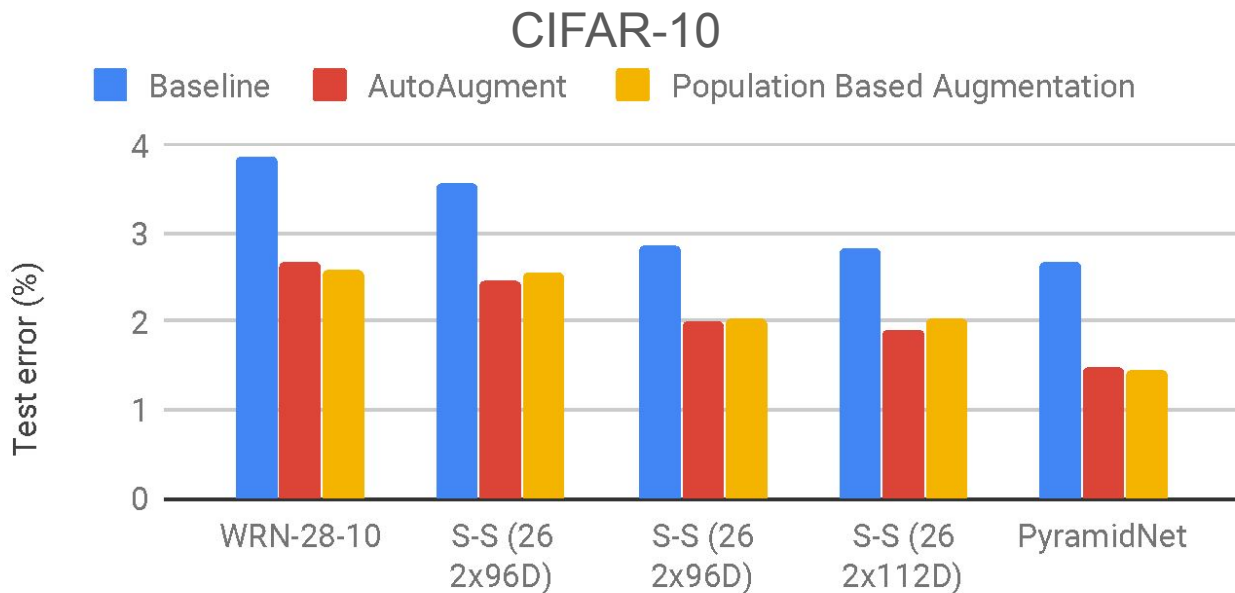


PBA: Trained with 1 run of PBT on 15 child models



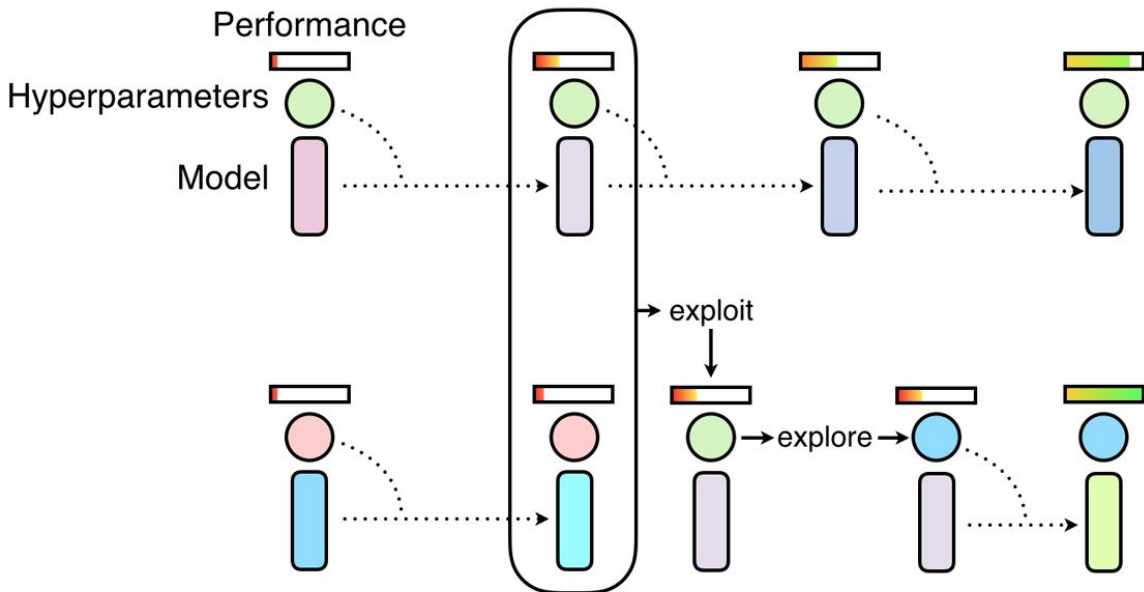
Population Based Augmentation (PBA)

PBA learns CIFAR augmentation policy in 5 GPU hours.
AutoAugment learns in 5,000 GPU hours.



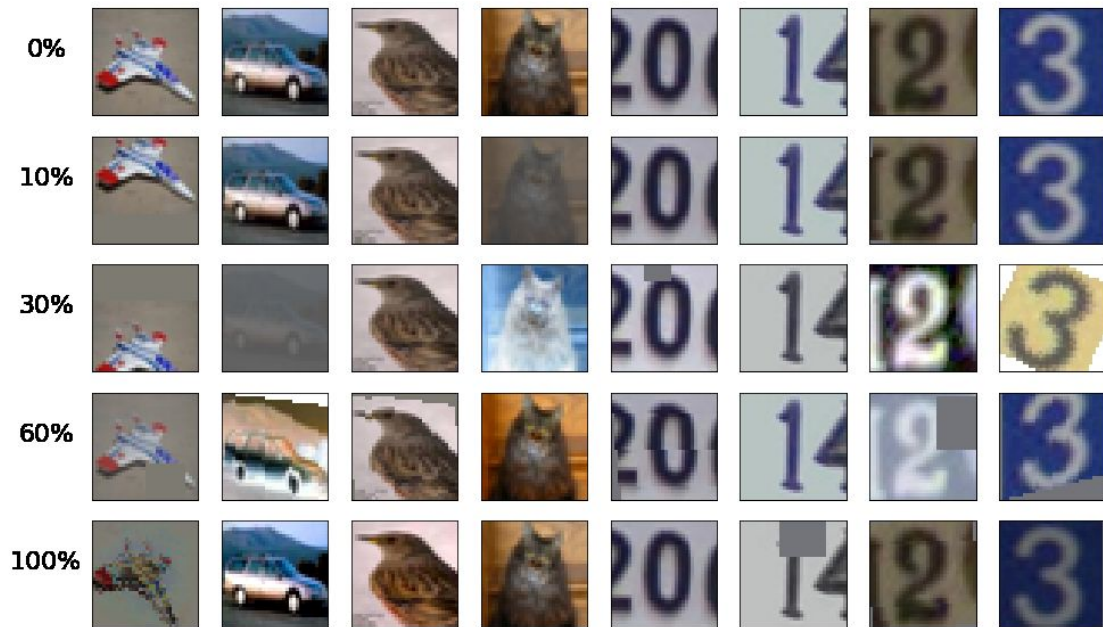
How is the augmentation schedule learned?

Hyperparameter search using a mix of evolutionary algorithms and random search to discover adaptive augmentation policy schedule quickly.



Learned Augmentation Policy Schedules

Effect of Population Based Augmentation applied to images showing stronger augmentations as training progresses.



Thank you!

Population Based Augmentation

Daniel Ho, Eric Liang, Ion Stoica, Pieter Abbeel, Xi Chen

Poster: Pacific Ballroom **#134**

Code: <https://github.com/arcelien/pba>

Contact: Daniel Ho (daniel.ho at berkeley.edu)