AutoGO: Automated Computation Graph Optimization for Neural Network Evolution

Mohammad Salameh[1], Keith G. Mills[1,2], Negar Hassanpour[1], Fred X. Han[1], Shuting Zhang[3], Wei Lu[1], Shangling Jui[3], Chunhua Zhou[3], Fengyu Sun[3] and Di Niu[2]
[1]Huawei Technologies Canada, [2]Dept. ECE, University of Alberta, [3]Huawei Kirin Solution, Shanghai, China

Abstract

Real-world DNN optimization is a challenge for ML engineers.

Heavy use of expert-driven, predefined design spaces.

We present a framework for evolving neural networks:
AutoGO: Automatic Graph Optimization.

Our contributions:
- Database of computational *segments* for mutation.
- Novel Predecessor-Segment-suCcessor (PSC) predictor accurately estimates mutation performance change.
- Mixed Integer Linear Programming (MILP) for functionality.
- Further refine best architectures in existing benchmarks.
- Optimize architectures for CV tasks, like ResNets/VGG for Segmentation/Pose Estimation, EDSR/FSRCNN for SR.
- Demonstrate real-world deployment applicability by improving already-lightweight architectures for mobile phone deployment using cycle-accurate counter.

Computational Graphs as Segments

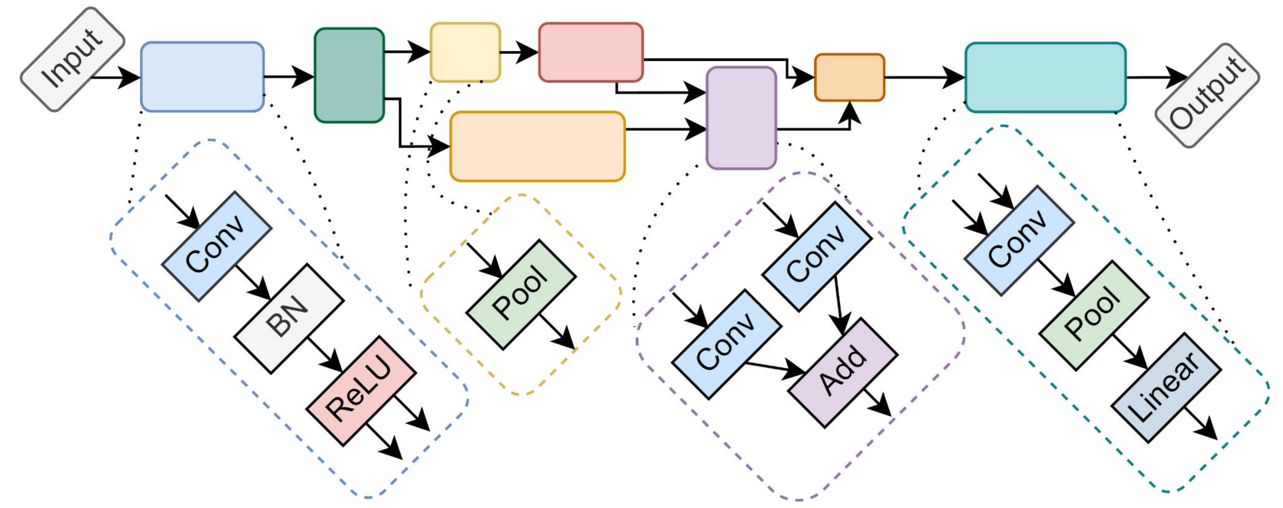


Fig. 1 in paper: DNN partitioned into disjoint subgraphs, called segments. Each segment contains a variable number of nodes, edges, inputs, etc., and are the AutoGO unit of mutation.

We mine segments in a data-driven manner using topological sort and Byte-Pair Encoding (BPE); tokenization from NLP.

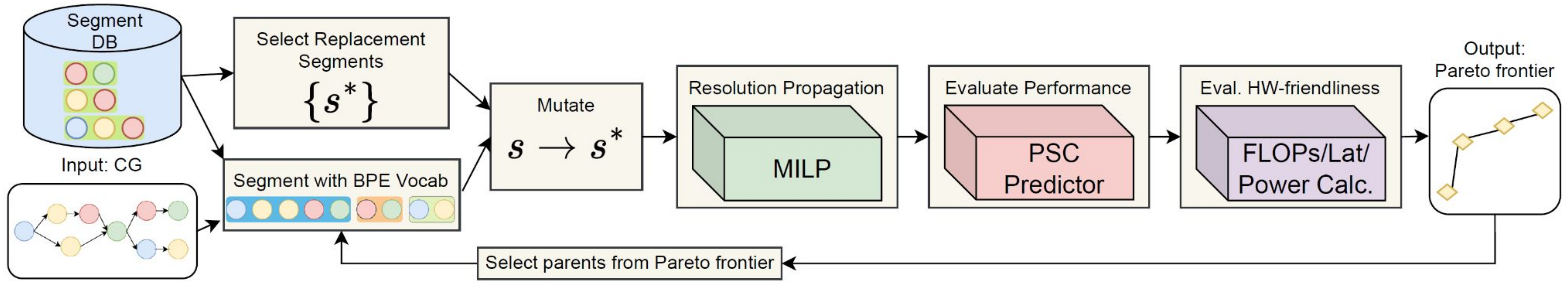


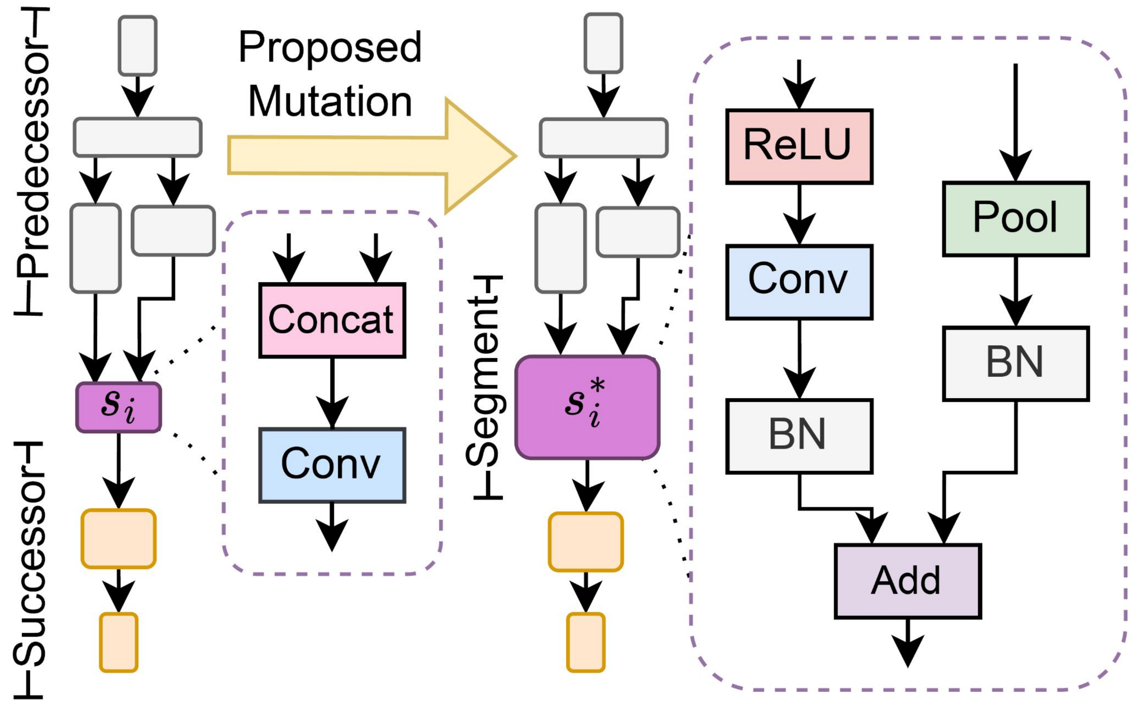
Fig. 2 in paper: AutoGO takes the CG of an architecture and Segment DB as inputs, producing a Pareto frontier as output.

Predecessor, Segment and suCcessor

Assign CG segments into 3 groups, ***P***, ***S***, ***C***, for mutation:

1. Predecessor ***P*** - Architecture input to ***S***.
2. Segment ***S*** - Specific architecture piece we mutate.
3. suCcessor ***C*** - Everything after ***S*** to the output.

Mutation: Replacing ***S*** with ***S****, drawn from Segment DB:



PSC Predictor: Novel neural predictor which is sensitive to performance change as a result of segment mutation.

Tab. 1 in paper: Rank correlation (SRCC) on 5 families.

Arch. Family	GNN	PSC 1:1 Ratio	PSC
NB-101	0.627 ± 0.031	0.666 ± 0.025	**0.849** ± 0.054
NB-201	0.809 ± 0.016	0.865 ± 0.015	**0.983** ± 0.003
HiAML	0.010 ± 0.013	0.170 ± 0.042	**0.734** ± 0.031
Inception	0.209 ± 0.037	0.066 ± 0.071	**0.496** ± 0.022
Two-Path	0.023 ± 0.018	0.236 ± 0.043	**0.724** ± 0.022

Results on Computer Vision Tasks

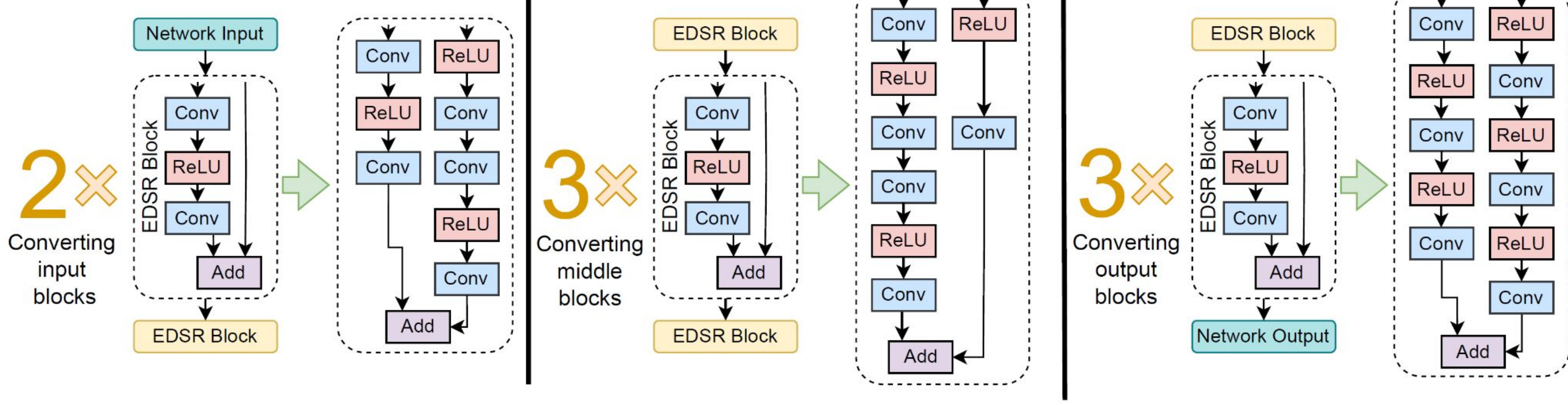
We apply AutoGO to optimize classical networks that are not part of NAS-Benchmarks for Computer Vision Tasks like Classification (ImageNet), Semantic Segmentation (Cityscapes), and Human Pose Estimation (MPII).

Tab. 3: AutoGO improves ResNet-50/101 & VGG-16.

Architecture	ImageNet Top-1/5	Cityscapes mIoU	MPII PCK	FLOPs [1e9]	Lat. [ms]
ResNet-50 Original	74.02%/91.22%	63.42%	82.36%	6.29	7.18
ResNet-50 AutoGO Arch 1	75.34%/92.16%	65.88%	**84.07**%	6.71	7.50
ResNet-50 AutoGO Arch 2	**75.66**%/**92.45**%	**66.65**%	82.70%	5.88	6.92
ResNet-101 Original	75.09%/91.94%	65.92%	82.77%	13.76	15.86
ResNet-101 AutoGO Arch 1	**76.56**%/**93.09**%	**67.12**%	83.59%	13.66	15.56
ResNet-101 AutoGO Arch 2	75.69%/92.15%	66.38%	**84.64**%	13.35	15.36
VGG-16 Original	74.18%/91.83%	65.36%	85.92%	30.81	4.65
VGG-16 AutoGO	**74.91**%/**93.23**%	**66.91**%	**85.99**%	24.34	4.20

Tab. 4: AutoGO improves EDSR Super Resolution PSNR.

SR Architecture	DIV2K	Set5	Set14	BSD100	Urban100	Manga109	FLOPs [1e9]	Lat. [ms]
EDSR Original	36.19	36.86	32.57	31.39	29.14	36.09	141	18.04
EDSR AutoGO Arch 1	**37.28**	**38.01**	**33.62**	**32.18**	**31.56**	**38.49**	118	15.38
EDSR AutoGO Arch 2	37.27	37.97	33.55	32.16	31.53	38.47	110	14.52
EDSR AutoGO Arch 3	37.25	**38.01**	33.58	32.16	31.46	38.44	105	13.81



Real-world Mobile Deployment

AutoGO further optimizes an already-lightweight, proprietary U-Net-like architecture for denoising on a mobile phone using a cycle-accurate counter.

Tab. 6: Reducing power/latency of a denoising network.

Denoising	PSNR	ΔLatency	Power [mW]	ΔPower	FLOPs [1e9]
Base Model	139.4	–	724.59	–	17.05
AutoGO	**139.9**	-24.94%	657.82	-9.21%	16.26

Example Mutations

Right - Fig. 4 in paper: AutoGO replacing a ResNet-50 residual block with a complex segment from HiAML. MILP adjusts channels/resolution to ensure architecture functionality.

Below - Paper Fig. 9: Trio of key mutations AutoGO performs on EDSR for Super Resolution (SR).