NEURAL INFORMATION PROCESSING SYSTEMS

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[1]

[1]Huawei Technologies Canada Co., Ltd.

[2]Dept. ECE, University of Alberta

[3]Huawei Kirin Solution, Shanghai, China

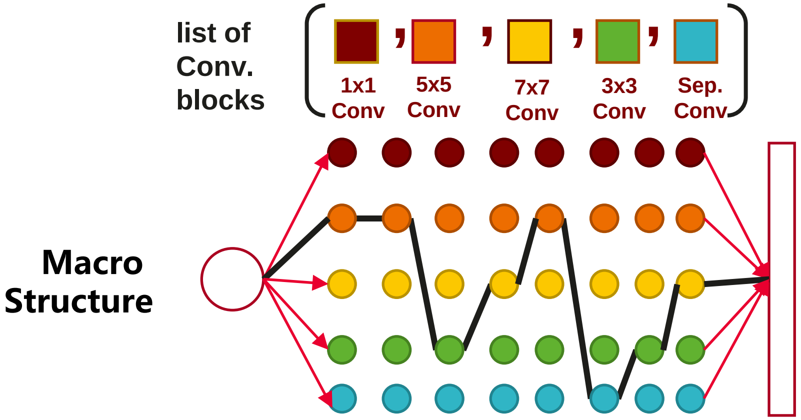
https://github.com/Ascend-Research/AutoGO

HUAWEI

UNIVERSITY OF ALBERTA

Background

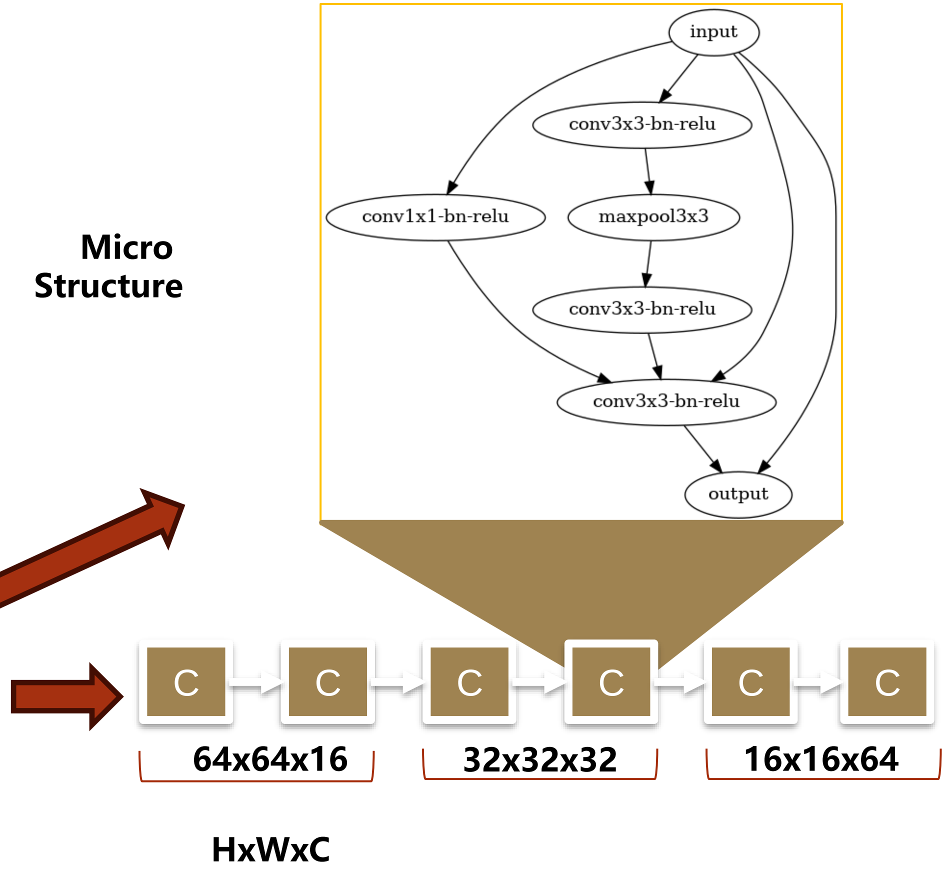
Neural Architecture Search (NAS) automates the DNN design.
Given a task, dataset and search space, we find architectures that obtain high accuracy and hardware-friendliness (e.g., FLOPs, latency, etc.)



- Search Space
  - Macro Structure: ResNets, MobileNets, etc.
  - Micro Structure: Cell-based NAS-Bench-101 or 201.

- Problems? The search space is predefined.
  - By expert knowledge/heuristics
  - Bounds on performance limits.
  - May not be hardware friendly.

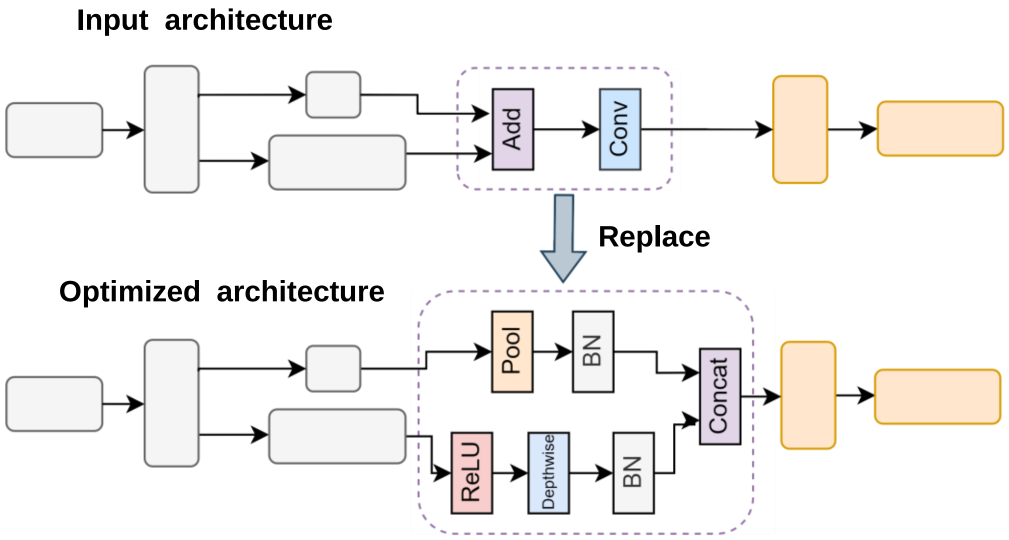
We cannot assume this cell
Is optimal at all these resolutions



Our Contribution: AutoGO

Framework for optimizing networks for performance and hardware-friendliness.

- Adjust low-level Computational Graphs.
- Data-driven mining of computational segments from benchmarks.
- Tests on popular CV tasks like classification, segmentation, etc.
- Applicable in deployment scenarios – we use it to optimize power and latency on proprietary networks for Huawei NPUs.



Building a Segment Database

Computational Graphs:

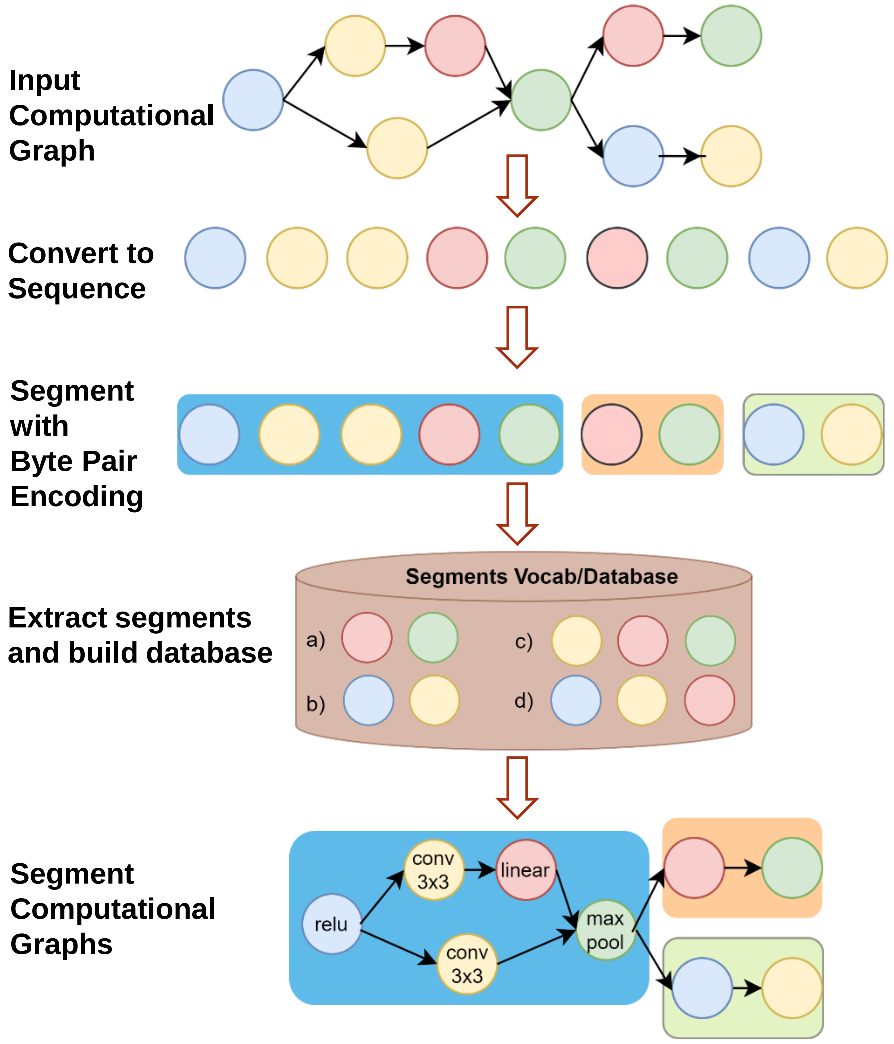
- DAGs with primitive operation nodes (e.g., Conv, Add, ReLU).
- Encode spatially-sensitive features like I/O HWC.

Data Driven Extraction:

- Use topological sort to convert graphs into sequences.
- Apply Byte-Pair Encoding (BPE), tokenization from NLP.
- This is a form of Frequent Subgraph Mining, used to build database.

Segments:

- CG subgraphs extracted from existing NAS Benchmarks.
- Can vary in #nodes, #edges, topology, inputs, outputs, etc.
- Unit of mutation in AutoGO.



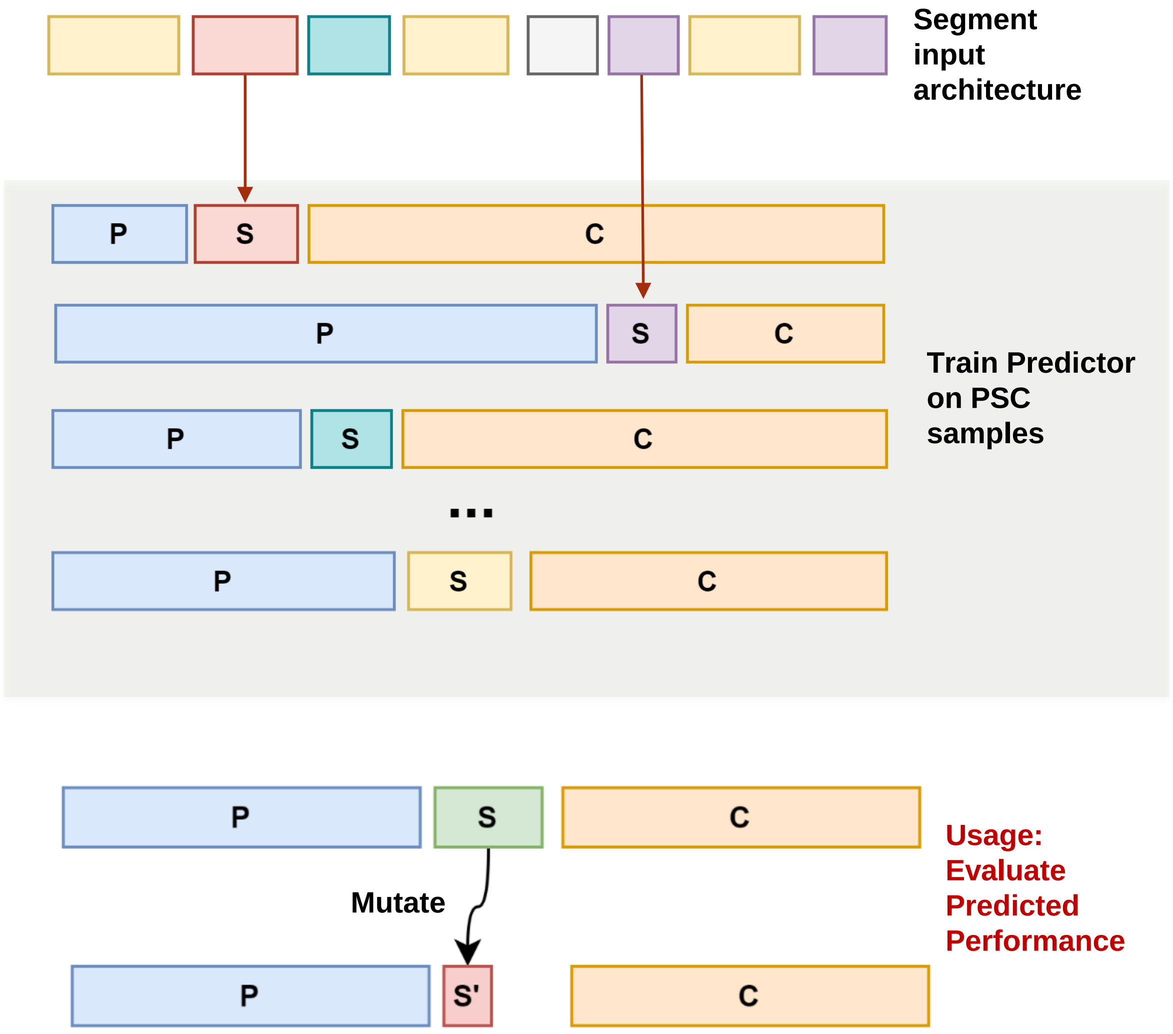
PSC and Mutation-driven search

PSC:

- 3 components of an architecture we mutate.
  - Segment S, to replace with S' from the database
  - Predecessor P
  - suCcessor C
- Any CG consists of many P, S, C permutations.

PSC Predictor:

- Designed for Segment mutation-based NAS.
- Aware and sensitive to the mutation context.
- GNN encodes P, S and C subgraphs separately, so changes in performance for mutant architectures are attributed to mutating S -> S'.
- Use an MILP to ensure network functionality.



Results

Image Classification

Family	Method	ImageNet Top-1	Delta Acc	FLOPs (Giga)	Delta FLOPs
ResNet-50	Baseline	74.02%	--	6.29	--
	AutoGO Arch 1	75.34%	+1.32%	6.71	+6.68%
	AutoGO Arch 2	75.66%	+1.64%	5.88	-6.52%
ResNet-101	Baseline	75.09%	--	13.76	--
	AutoGO Arch 1	76.56%	+1.47%	13.66	-0.73%
	AutoGO Arch 2	75.69%	+0.60%	13.35	-2.98%

Image Classification, Semantic Segmentation And Pose Estimation

Family	Method	ImageNet Top-1	Delta Acc	Cityscapes mIoU	Delta mIoU	PCK	Delta FLOPs
VGG16	Baseline	74.18%	--	65.36%	--	85.92%	--
	AutoGO	74.91%	+0.73%	66.91%	+1.55%	85.99%	-21.00%

Super Resolution

Family	Method	Set5 PSNR	Set14 PSNR	BSD100 PSNR	Urban100 PSNR	Manga109 PSNR	Delta FLOPs
EDSR	Baseline	36.89	32.57	31.39	29.14	36.08	--
	AutoGO Arch 1	38.01	33.62	32.18	31.56	38.49	-16.31%
	AutoGO Arch 2	37.97	33.55	32.16	31.53	38.47	-21.99%
	AutoGO Arch 3	38.01	33.58	32.16	31.46	38.44	-25.53%