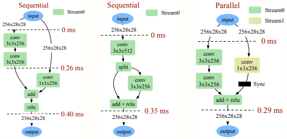
AutoGraph: Optimizing DNN Computation Graph for Parallel GPU Kernel Execution

Yuxuan Zhao, Qi Sun, Zhuolun He, Yang Bai, Bei Yu. *The Chinese University of Hong Kong*.

Introduction



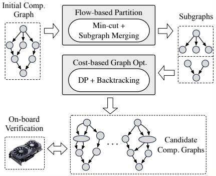
(a) PyTorch (b) TASO (c) Ours

- ✦ Sequential kernel execution is of low efficiency.
- ✦ Existing graph optimization methods break the inter-operator parallelism.

Our Contributions:

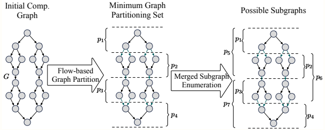
- ✦ We propose a novel dynamic programming + backtracking search algorithm to find optimization solutions efficiently.
- ✦ Leveraging customized cost and multi-stream, our method achieves the SOTA performance.

Overview



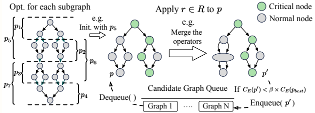
Details of AutoGraph

Flow-based Graph Partition


Initial Comp. Graph
G
Flow-based Graph Partition
Minimum Graph Partitioning Set
p_1
p_2
p_3
p_4
Merged Subgraph Enumeration
Possible Subgraphs
p_5
p_6
p_7

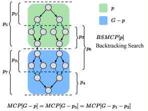
- ✦ Reduce the search space while maximizing optimization opportunities.

Backtracking Search via Customized Cost



- ✦ We propose the mixed critical path cost as the selection criteria.

DP-based Optimization Solution Search



- ✦ Optimize the current subgraph with backtracking search and reduce the problem to a sub-problem.

On-board Verification

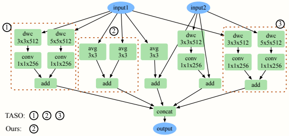
- ✦ We leverage GPU multi-stream to exploit the inter-operator parallelism of the computation graph.

Results

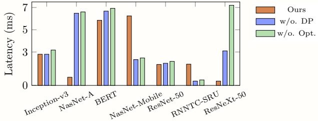
End-to-end Model Inference Latency

Model	JIT	TASO+JIT	IOS	Nimble	TASO+Nimble	Ours
Inception-v3	8.839	7.819	3.788	3.174	2.928	**2.799**
ResNet-50	4.566	4.554	3.284	2.144	1.988	**1.905**
ResNeXt-50	7.540	7.369	3.056	7.708	5.933	**2.892**
NasNet-A	13.891	10.843	9.583	6.483	13.086	**5.850**
NasNet-Mobile	10.155	8.085	3.821	2.320	6.540	**1.883**
RNNTC-SRU	1.496	1.307	-	0.486	**0.387**	**0.387**
BERT	11.011	9.026	-	6.923	6.473	**6.240**

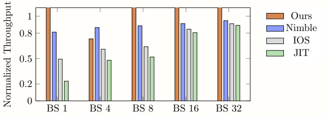
Case Study on NasNet Cell



Ablation Studies on Different Settings



Ablation Studies on Different Batch Sizes


Normalized Throughput
Ours
Nimble
IOS
JIT
BS 1
BS 4
BS 8
BS 16
BS 32