AI Matrix – Synthetic Benchmarks for DNN

Wei Wei, Lingjie Xu, Lingling Jin, Wei Zhang
Alibaba Inc
{w.wei, lingjie.xu, l.jin, wz.ww}@alibaba-inc.com

Tianjun Zhang
University of California, Berkeley
tianjunz@berkeley.edu

Introduction

Nowadays, almost all deep learning benchmarks fall into the category of collective benchmarks. They have some **drawbacks**:

- selectively collected from real DNN applications and get outdated as models evolve
- consist tens to even hundreds of applications which spans from different areas. It takes long time to run all of the tests.
- typically open-sourced or copyright authorized where they are not representable to application of interests especially the applications are proprietary

We propose a framework to generate synthetic DNN benchmarks to address the above drawbacks. The synthetic approach with automatically generated DNNs has **advantages** below:

- represents the workload characteristics of any applications of interests
- combines the applications from different areas into one or few synthetic benchmarks
- synthetic benchmark can be updated automatically anytime to reflect new changes of models
- serves as proxies for any application of interests but is proprietary.

The generated benchmarks called AI Matrix, serving as a performance benchmarks matching the statistical workload characteristics of a combination of applications of interests.

Methodology

Application Monitoring System
- Collect the workload characteristics data of running applications
- Input size, input channel, kernel size, kernel stride, etc.

↓

Workload Analysis
- Deploy nearest neighbor algorithm to aggregate the operations
- Cluster the convolution operations by input size and filter size

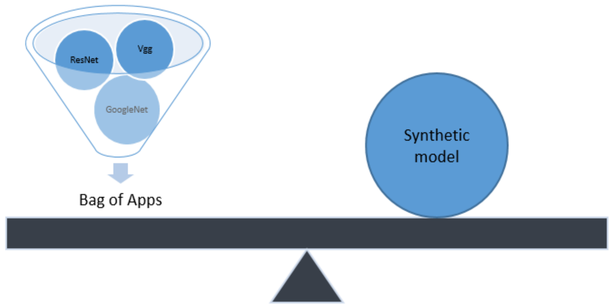
↓

Workload Synthesizer
- Generate synthetic CNNs with matched fitness for each group
- Number of filters is adjustable
- Apply genetic algorithm to search optimal solution

↓

Synthetic Benchmarks

$$Fitness = 0.5 * \frac{|MAC - MACreal|}{MACreal} + 0.5 * \frac{|WP - WPreal|}{WPreal}$$



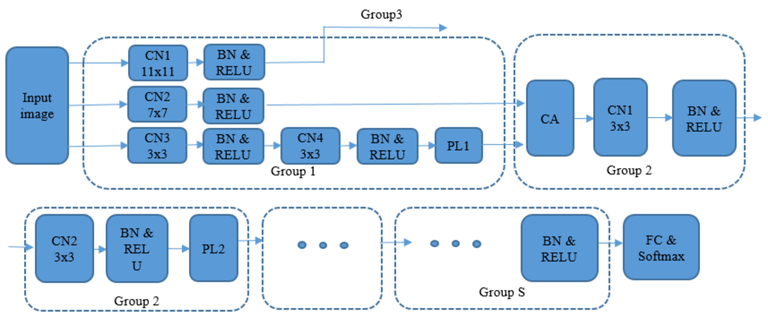
Experiment and Result

Experiment on data from mixing 3 classic models

We select 3 different classic models Alexnet, Vgg16, GoogleNet, running once for each model on our monitoring system and log the inference execution data. After workload analysis, the statistical information of layers are collected and clustered shown below.

Group	Group center (HxW)	Counts	filter size, filter stride
1	224x224	1	11,4
		1	7,2
		2	3,1
2	112x112	2	3,1
3	56x56	4	3,1
		1	1,1
4	28x28	4	5,1
		5	3,1
		8	1,1
5	14x14	3	5,1
		13	3,1
		22	1,1
6	7x7	2	3,1
		10	1,1

For each group, genetic algorithm is applied separately to find the optimal number for each kernel filter (L=10 for each binary string). The solution is based 50 iterations evolution with 5000 individuals.

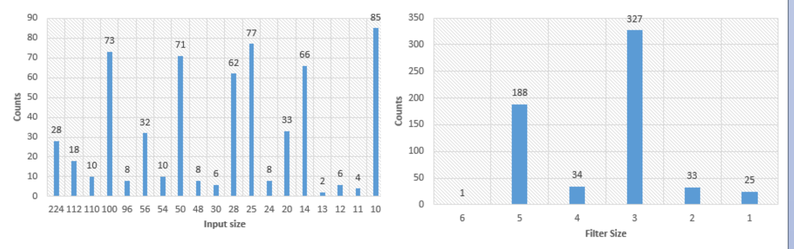


Group	3 models (MACs, warps)	Synthetic model (MACs, warps)	Fitness Error (%)
1	2156022912,2802996	2156050176,2803928	0.02
2	2774532096,2792888	2774419200,2859720	1.19
3	4983881728,3184980	4986729216,3184876	0.03
4	5280485376,2221154	5280615536,2221030	0.00
5	2199505920,2789028	2199066688,2788961	0.01
6	109734912,885229	109735598,885337	0.00

Experiment on data from Alibaba

Similar to previous experiment. We collect some DNN application data through our Alibaba AI platform. It logs layers' statistical data of different models running on it in certain period of time. It also gives the flexibility of synthetic benchmark that could be generated based on data collected in any period of time. It represents all the models running in that period of time.

The operations with statistical data collected on our platform are shown below. The fitness values and corresponding errors are also summarized below.



Group	Real models (MACs, warps)	Synthetic model (MACs, warps)	Fitness Error (%)
1	3728928152,7139725	3729030144,7277052	0.96
2	6129559271,11169027	6129236736,11165590	0.01
3	14495976374,9817393	14500048640,9813450	0.03
4	11699424510,4706169	11699536710,4708200	0.02
5	3330293660,3374419	3330837398,3374523	0.00
6	258530570,575312	258472240,573424	0.17

Conclusion

The objective of this work is to develop an innovative framework that can be used to generate synthetic DNN benchmarks which matches the statistical workload characteristics of real applications of interests. We conduct an experiment on data from mixing 3 classic models of Alexnet, Vgg16 and Googlenet as simple illustration of our framework. We then validate our framework on some real data collected from Alibaba AI platform. With the help of genetic algorithm optimization, our generated synthetic model could represent not only the statistical distribution of layer parameters but also workload characteristics of all the collected real models.

Our future work is to consider integrating the block architecture, e.g, inception module and residue module, into the synthetic models. RNN based models will also be considered for synthetic models.

Alibaba Group 阿里巴巴集团