HPC AI500: The Methodology, Tool, and Metrics for Benchmarking HPC AI Systems

Zihan Jiang
Institute of Computing Technology, Chinese Academy of Sciences
University of Chinese Academy of Sciences
AIBench Tutorial, ISCA’21
Outline

- Challenges
- Methodology
  - Workload Construction
  - Hierarchical Benchmarking Methodology
- Metrics
- Specification
- Ranking
Challenge 1: How to Achieve Both Representativeness And Simplicity

- Conflict between representativeness (SPEC series) and simplicity (HPL)
  - SPEC series, PARSEC, and TPC benchmarks emphasize the importance of the benchmarks being representative and diverse.
  - HPL (TOP 500) establishes the de facto supercomputer benchmark standard in terms of simplicity
    - Easy to port
    - Benchmarking cost is affordable
    - Metric is easily understandable

- In the HPC AI domain:
  - Massive AI tasks and different metrics, the FLOPS is not the only concern.
  - Not affordable to perform benchmarking at scale.
The micro benchmark HPL-AI evaluates the performance of the HPC AI system based on mixed precision LU decomposition, however, our evaluation results show a low percentage of LU decomposition in 19 workloads of AIBench.
Challenge 2: How to Achieve Repeatability

- AI nature is stochastic and AI workloads has randomness
  - Random seeds affect model initialization, data traversal order, etc.
  - Non-idempotence of floating-point operations
  - AI frameworks use different implementations of operators, such as convolution.
  - Multiple hyper-parameters, such as batch size, learning rate, and weight decay.
Example 2: Randomness

- The epochs to quality vary significantly under different random seeds.
Taking AI (deep neural networks) as the example: HPL-AI

- Misleading: Some optimizations improves FLOPS but deteriorate model quality.

No information of model quality

The huge input space of the kernel makes the benchmarking using micro benchmark very expensive.

A single kernel can not reflect overall performance.
While mixed-precision training significantly improve FLOPS, it can deteriorate the model quality.

The ResNet50 quality comparison

FLOPS comparison of ResNet50 model and operators
Outline

- Challenges
- Methodology
  - Workload Construction
  - Hierarchical Benchmarking Methodology
- Metrics
- Specification
- Case Study
- Ranking
The criteria for choosing the workloads

- Representativeness
- Repeatability
- Simplicity
- The requirements in the HPC Field

We choose AIBench training—by far the most comprehensive benchmarks—as the starting point for the design and implementation of HPC AI500 benchmarks.
How to Achieve Representativeness

- Microarchitecture-dependent perspectives
  - K-means Clustering; t-SNE for visualization

Computation and memory access pattern:
achieved_occupancy;
ipc_efficiency;
gld_efficiency;
gst_efficiency;
dram_utilization.
How to Achieve Representativeness

- Microarchitecture-independent perspective
  - K-means Clustering; t-SNE for visualization

Algorithm behaviors:
- Model complexity;
- Convergent rate;
- Computational cost.
How to Achieve Repeatability

Run-to-run Variation of Seventeen Benchmarks of AIBench Training. Note that Image-to-image and image generation variations are not reported due to a lack of a widely accepted metric to terminate an entire training session.

<table>
<thead>
<tr>
<th>No.</th>
<th>Benchmark</th>
<th>Variation</th>
<th>Runs</th>
<th>Time (Hour)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC-AI-C1</td>
<td>Image Classification</td>
<td>1.12%</td>
<td>8</td>
<td>76.25</td>
</tr>
<tr>
<td>DC-AI-C2</td>
<td>Image Generation</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>DC-AI-C3</td>
<td>Text-to-Text</td>
<td>9.38%</td>
<td>6</td>
<td>1.72</td>
</tr>
<tr>
<td>DC-AI-C4</td>
<td>Image-to-Text</td>
<td>23.53%</td>
<td>5</td>
<td>10.21</td>
</tr>
<tr>
<td>DC-AI-C5</td>
<td>Image-to-Image</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>DC-AI-C6</td>
<td>Speech Recognition</td>
<td>12.08%</td>
<td>4</td>
<td>42.78</td>
</tr>
<tr>
<td>DC-AI-C7</td>
<td>Face Embedding</td>
<td>5.73%</td>
<td>8</td>
<td>3.43</td>
</tr>
<tr>
<td>DC-AI-C8</td>
<td>3D Face Recognition</td>
<td>38.46%</td>
<td>10</td>
<td>12.02</td>
</tr>
<tr>
<td>DC-AI-C9</td>
<td>Object Detection</td>
<td>0</td>
<td>10</td>
<td>2.06</td>
</tr>
<tr>
<td>DC-AI-C10</td>
<td>Recommendation</td>
<td>9.95%</td>
<td>5</td>
<td>0.16</td>
</tr>
<tr>
<td>DC-AI-C11</td>
<td>Video Prediction</td>
<td>11.83%</td>
<td>4</td>
<td>2.11</td>
</tr>
<tr>
<td>DC-AI-C12</td>
<td>Image Compression</td>
<td>22.49%</td>
<td>4</td>
<td>5.67</td>
</tr>
<tr>
<td>DC-AI-C13</td>
<td>3D Object Reconstruction</td>
<td>16.07%</td>
<td>4</td>
<td>0.38</td>
</tr>
<tr>
<td>DC-AI-C14</td>
<td>Text Summarization</td>
<td>24.72%</td>
<td>5</td>
<td>6.41</td>
</tr>
<tr>
<td>DC-AI-C15</td>
<td>Spatial Transformer</td>
<td>7.29%</td>
<td>4</td>
<td>0.06</td>
</tr>
<tr>
<td>DC-AI-C16</td>
<td>Learning to Rank</td>
<td>1.90%</td>
<td>4</td>
<td>0.14</td>
</tr>
<tr>
<td>DC-AI-C17</td>
<td>Neural Architecture Search</td>
<td>6.15%</td>
<td>6</td>
<td>7.47</td>
</tr>
<tr>
<td>DC-AI-C18</td>
<td>Advertising</td>
<td>2.12%</td>
<td>4</td>
<td>2.28</td>
</tr>
<tr>
<td>DC-AI-C19</td>
<td>Nature Language Processing</td>
<td>19.51%</td>
<td>5</td>
<td>45.22</td>
</tr>
</tbody>
</table>
Simplicity is another important criteria for benchmarking. However, benchmarking an entire training session of all seventeen workloads in AIBench Training is extremely expensive, which reaches up to 10 days.

We emphasize that Image Classification, object Detection, and Learning-to-Rank achieve not only representativeness and repeatability, but also simplicity.
The requirements in the HPC Filed:

- **Dataset:**
  - HPC AI domains cover both commercial and high-performance scientific computing.
  - Currently, business applications are pervasive. AI for science applications lag but is promising. In general, the scientific data are often more complicated than that of the MINST or ImageNet data (e.g., resolution, channels).
The requirements in the HPC Field:

- Computation Complexity
  - The Computation comparison of three candidates (Single training batch).

<table>
<thead>
<tr>
<th>Workloads</th>
<th>Computation (FLOPs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image Classification</td>
<td>23 G</td>
</tr>
<tr>
<td>Object Detection</td>
<td>691 G</td>
</tr>
<tr>
<td>Learning to Rank</td>
<td>0.08 M</td>
</tr>
</tbody>
</table>

*Image Classification and Object Detection are chosen as two typical workloads for HPC AI benchmarking.*
Based on the existing analysis, we can conclude that Image Classification and Object Detection are the final candidates to construct the HPC AI500 benchmark.

We choose the most representative workloads and data sets from both HPC and commercial fields.

- **Extreme weather analysis (EWA)**
  - pioneering works that uses deep learning to tackle the classical HPC problem.
  - Gordon bell prize (SC’18).
  - Essentially object detection: detect different weather patterns in on climate image.

- **Image Classification**
  - De facto HPC AI benchmark used in many state-of-the-art works.
Outline

- Challenges
- Methodology
  - Workload Construction
  - Hierarchical Benchmarking Methodology
- Metrics
- Specification
- Case Study
- Ranking
HPC AI500 defines a comprehensive benchmarking methodology based on nine-layers system abstraction, divided into the following three levels: hardware level, system level, and free level.

<table>
<thead>
<tr>
<th>Problem Domain (Datasets, Target quality, Epochs)</th>
<th>Metrics: Valid FLOPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learning rate policies and batchsize setting</td>
<td>Other hyper-parameters settings</td>
</tr>
<tr>
<td>Workload (The Subset of AI Bench)</td>
<td></td>
</tr>
<tr>
<td>Image Classification</td>
<td></td>
</tr>
<tr>
<td>Extreme Weather Analytics (Object Detection)</td>
<td></td>
</tr>
<tr>
<td>Programming Model</td>
<td></td>
</tr>
<tr>
<td>AI Framework (e.g. TensorFlow)</td>
<td></td>
</tr>
<tr>
<td>AI Accelerators, interconnection, and libs (e.g. GPU, CUDA, NVLink)</td>
<td></td>
</tr>
<tr>
<td>Communication libs (e.g. Horovod)</td>
<td></td>
</tr>
<tr>
<td>OS</td>
<td></td>
</tr>
<tr>
<td>Hardware (e.g. CPU, Network)</td>
<td></td>
</tr>
</tbody>
</table>

Layer 9
Layer 8
Layer 7
Layer 6
Layer 5
Layer 4
Layer 3
Layer 2
Layer 1

Outline

- Challenges
- HPC AI500 Methodology
  - Workload Construction
  - Hierarchical Benchmarking Methodology
- Metrics
- HPC AI500 Specification
- Case Study
- Ranking
Using VFLOPS to unify the computation and model quality.

\[
VFLOPS = FLOPS \times \text{penalty\_coefficient} \quad (1)
\]

The penalty\_coefficient is used to penalize or award the FLOPS if the achieved quality is lower or greater than the target quality. Its definition is described as follows:

\[
\text{penalty\_coefficient} = \left(\frac{\text{achieved\_quality}}{\text{target\_quality}}\right)^n \quad (2)
\]

achieved\_quality refers to the achieved quality in the evaluation; target\_quality refers to the target quality defined in HPC AI500 problem domain.

The value of n is a positive integer, which is used to define the sensitivity to the model quality. The higher the number of n, the more loss of quality drop. As EWA (Object Detection) has much more stringent quality requirement than that of Image Classification. We set n as 10 for EWA and 5 for Image Classification by default.
Outline

- Challenges
- Methodology
  - Workload Construction
  - Hierarchical Benchmarking Methodology
- Metrics
- Specification
- Ranking
Problem Domain
- Extreme Weather Analysis: detect the patterns of extreme weather, essentially Object Detection. The application that wins Gordon Bell Prize.
- Image Classification: ResNet50/ImageNet is a de facto benchmark for optimizing HPC AI systems.

Dataset
- The extreme weather dataset: 16 channels, 768*1052, 2 TB
- ImageNet 2012: 3 channels, 256*256, 136 GB

Model
- Faster-RCNN
- ResNet-50 V1.5
## HPC AI500 Benchmark Suites

<table>
<thead>
<tr>
<th>Problem Domain</th>
<th>Model</th>
<th>Dataset</th>
<th>Target Quality</th>
<th>Training Epochs</th>
<th>Comm Lib</th>
<th>AI accelerators Lib</th>
<th>AI Framework</th>
</tr>
</thead>
<tbody>
<tr>
<td>EWA</td>
<td>Faster-RCNN</td>
<td>The extreme weather dataset</td>
<td>mAP@[IoU=0.5]= 0.35</td>
<td>50</td>
<td>MPI, NCCL2</td>
<td>CUDA, cuDNN, NCCL</td>
<td>TensorFlow</td>
</tr>
<tr>
<td>Image Classification</td>
<td>ResNet50 V1.5</td>
<td>ImageNet 2012</td>
<td>TOP1 Accuracy= 0.763</td>
<td>90</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Image Classification**
  - Model: ResNet50 V1.5
  - Dataset: ImageNet 2012
  - Target Quality: TOP1 Accuracy= 0.763
  - Training Epochs: 90
  - Comm Lib: MPI, NCCL2
  - AI accelerators Lib: CUDA, cuDNN, NCCL
  - AI Framework: TensorFlow
Outline

- Challenges
- Methodology
  - Workload Construction
  - Hierarchical Benchmarking Methodology
- Metrics
- Ranking
# The VFLOPS Ranking

**Image Classification, Free Level, July 2, 2020**

<table>
<thead>
<tr>
<th>Rank</th>
<th>Source</th>
<th>VPFLOPS</th>
<th>AI Accelerators</th>
<th>Framework</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Fujitsu[1]</td>
<td>31.41</td>
<td>Tesla V100*2048</td>
<td>MXNet</td>
</tr>
<tr>
<td>3</td>
<td>Sony[3]</td>
<td>10.02</td>
<td>Tesla V100*2176</td>
<td>NNL</td>
</tr>
<tr>
<td>5</td>
<td>Preferred Network[5]</td>
<td>2.41</td>
<td>Tesla P100*1024</td>
<td>Chainer</td>
</tr>
<tr>
<td>6</td>
<td>Berkeley[6]</td>
<td>1.95</td>
<td>KNL*2048</td>
<td>Intel Caffe</td>
</tr>
<tr>
<td>7</td>
<td>Intel[7]</td>
<td>1.27</td>
<td>KNL*1536</td>
<td>Intel Caffe</td>
</tr>
<tr>
<td>8</td>
<td>IBM[8]</td>
<td>0.75</td>
<td>Tesla P100*256</td>
<td>Caffe</td>
</tr>
<tr>
<td>9</td>
<td>Facebook[9]</td>
<td>0.7</td>
<td>Tesla P100*1024</td>
<td>Caffe2</td>
</tr>
</tbody>
</table>
### Auxiliary Metrics: Time and Quality

<table>
<thead>
<tr>
<th>Rank</th>
<th>Source</th>
<th>Time(min)</th>
<th>AI Accelerators</th>
<th>Framework</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Fujitsu[1]</td>
<td>1.2</td>
<td>Tesla V100*2048</td>
<td>MXNet</td>
</tr>
<tr>
<td>2</td>
<td>Google[2]</td>
<td>2.2</td>
<td>TPU V3*1/204</td>
<td>TensorFlow</td>
</tr>
<tr>
<td>3</td>
<td>Sony[3]</td>
<td>3.7</td>
<td>Tesla V100*2176</td>
<td>NNL</td>
</tr>
<tr>
<td>4</td>
<td>Tencent[4]</td>
<td>6.6</td>
<td>Tesla P40*2048</td>
<td>TensorFlow</td>
</tr>
<tr>
<td>5</td>
<td>Preferred Network[5]</td>
<td>15</td>
<td>Tesla P100*1024</td>
<td>Chainer</td>
</tr>
<tr>
<td>6</td>
<td>Berkeley[6]</td>
<td>20</td>
<td>KNL*2048</td>
<td>Intel Caffe</td>
</tr>
<tr>
<td>7</td>
<td>Intel[7]</td>
<td>28</td>
<td>KNL*1536</td>
<td>Intel Caffe</td>
</tr>
<tr>
<td>8</td>
<td>IBM[8]</td>
<td>50</td>
<td>Tesla P100*256</td>
<td>Caffe</td>
</tr>
<tr>
<td>9</td>
<td>Facebook[9]</td>
<td>60</td>
<td>Tesla P100*1024</td>
<td>Caffe2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rank</th>
<th>Source</th>
<th>Quality</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Fujitsu[1]</td>
<td>75.1</td>
</tr>
<tr>
<td>2</td>
<td>Google[2]</td>
<td>76.3</td>
</tr>
<tr>
<td>3</td>
<td>Sony[3]</td>
<td>75</td>
</tr>
<tr>
<td>4</td>
<td>Tencent[4]</td>
<td>76</td>
</tr>
<tr>
<td>5</td>
<td>Preferred Network[5]</td>
<td>74.9</td>
</tr>
<tr>
<td>6</td>
<td>Berkeley[6]</td>
<td>75.4</td>
</tr>
<tr>
<td>7</td>
<td>Intel[7]</td>
<td>74.6</td>
</tr>
<tr>
<td>8</td>
<td>IBM[8]</td>
<td>75</td>
</tr>
<tr>
<td>9</td>
<td>Facebook[9]</td>
<td>76.3</td>
</tr>
</tbody>
</table>

---

**Notes:**
- ISCA'21
- Institute of Computing Technology, Chinese Academy of Sciences
- Bench Council

---
The first HPC AI benchmark based on scientific application and dataset:

- HPC AI500 has been listed as a key reference by Jack Dongarra (The founder of TOP500)
- https://www.benchcouncil.org/HPCAI500/index.html
Thanks!