A Benchmark Proposal for Massive Scale Inference Systems

(Work-In-Progress Paper)

Meikel Poess[†]
Server Technologies
Oracle Corporation
Redwood Shores, CA, USA
meikel.poess@oracle.com

Raghu Nambiar
Advanced Micro Devices
Santa Clara, CA, USA
raghu.nambiar@amd.com

Karthik Kulkarni
Cisco Systems, Inc
San Jose, CA, USA
kakulkar@cisco.com

ABSTRACT

Many benchmarks have been proposed to measure the training/learning aspects of Artificial Intelligence systems. This is without doubt very important, because its methods are very computationally expensive, and, therefore, offering a wide variety of techniques to optimize the computational performance.The inference aspect of Artificial Intelligence systems is becoming increasingly important as the these system are starting to massive scale. However, there are no industry standards yet that measures the performance capabilities of massive scale AI deployments that must perform very large number of complex inferences in parallel.

In this work-in-progress paper we describe TPC-I, the industry's first benchmark to measure the performance characteristics of massive scale industry inference deployments. It models a representative use case, which enables hard- and software optimizations to directly benefit real customer scenarios.

KEYWORDS

Benchmarking; Artificial Intelligence; Information System

ACM Reference format:
Meikel Poess, Raghu Nambiar, and Karthik Kulkarni. 2019. A Benchmark Proposal for Massive Scale Inference Systems. In *ACM/SPEC International Conference on Performance Engineering Companion (ICPE'19 Companion), April 7–11, 2019, Mumbai, India.* ACM, New York, NY, USA, 4 pages. DOI: https://doi.org/10.1145/3297663.331098

1 Introduction

Until recently Artificial Intelligence (AI) benchmarks have been measuring the performance of a single applications that fit in a server or a set of servers. Today, many commercial AI systems are deployed on very large, complex systems in many cases distributed across multiple datacenters operating on very large datasets. Measuring the performance of such systems calls for a benchmark framework that can encapsulate the complexity of such systems. With respect to hardware it must include multiple servers running in multiple data centers, user interfaces, network communication and disk I/O as well as high availability. Beyond the inference component it must include mechanisms that enable high concurrent access with potential load balancing capabilities to assure optimally utilized systems and to guarantee latency and throughput requirements.

Many of the existing artificial intelligence (AI) benchmarks concentrate on the performance and accuracy levels of training AI models. Training is usually categorized as an occasionally conducted, single-user task that is highly compute and memory intensive. Job completion times, resource consumption, and costs associated are important metrics to differentiate systems. Inference, to the contrary, is usually done frequently by many users concurrently, and, depending on its use case, it may demand certain response times. From system level perspective, metrics that measure throughput and latency of inference are as important as metrics measuring resource consumptions.

TPC benchmarks have certain unique characteristics. TPC-I is designed keeping those characteristic in mind. It is a complete system level benchmark for characterization of massive scale inferencing based on a well-trained model. The motivation behind TPC-I is the lack of benchmarks while there are increasing number of industries across many verticals that are actively using AI systems for decision making. To design TPC-I, we studied two use cases: Highway Toll System and Airport Security.

TPC-I is modeled after an airport security system. Airport security has been a major concern across the world. The US Customs and Border Protection Agency has deployed an Automated Passport Control (APC) system [6] in 63 international airports. This facial recognition system is situated in the primary inspection area of an airport to expedite the entry process for international travelers by providing an automated process. Photos of each passenger's passport and face are obtained through a self-service reader. If this program was to be expanded to all US airline travelers, it would need to serve 2.6 *million passengers at 4,898 public airports according to FAA*

[*]Article Title Footnote needs to be captured as Title Note
[†]Author Footnote to be captured as Author Note

ICPE '19, April 7–11, 2019, Mumbai, India.

ACM ISBN 978-1-4503-6239-9/19/04...$15.00.
DOI: http://dx.doi.org/10.1145/10.1145/3302541.3313098

travel statistics [15]. Assuming a uniform distribution of passenger arrival at APC stations in a typical 18h airport day, this is one passenger every 0.0249 seconds.

While the exact details of APC are unknown to the authors, it represents a typical use case of a large scale, high throughput with low latency inference system. Since the workload can be scaled to large system, it is suitable for capacity planning. The model can be extended to other use cases such as automatic id verification at airports, which is done manually today.

The remainder of the paper is organized as follows: Section 2 presents related work. Section 3 describes the current state of TPC-I, including a description of its use case, System Under Test and workload driver definition, data set and scaling and performance metric and execution rules. We conclude our paper with Section 4.

2 Related Work

There are multiple benchmarks that focus on machine learning. The most cited once are DeepBench [4], DAWNBench **Error! Reference source not found.** and MLPerf [9]. DeepBench focuses on measuring the performance of basic operations that are commonly used for deep neural networks on specified hardware configurations. It does not go through an actual training or inference process, but instead focuses on a set of operations most commonly used in deep learning.

DAWNBench [10] is an end-to-end benchmark suite to measure the performance of training and inference while also reporting the cost for those operations. It has three distinct workloads, two of which are image classification workloads using the ImageNet 2012 data set [11] and image data set [8]. The third workload is an NLP based question answering workload using Stanford Question Answering Dataset (SQuAD) dataset [13]. These workloads are more end-to-end in the sense that users can report four different values i.e., training time, training costs, inference time and inference costs for each of the three individual datasets. Training on image classification on ImageNet dataset requires the trained model to have 93% or greater accuracy. Training on image classification on CIFAR dataset requires the trained model to have 94% or greater accuracy. The training on the question answering model on the SQuAD dataset requires an F1 score of 0.75 or greater. DAWNBench benchmark also considers inference latencies as the time taken to infer or classify 1 image either with ImageNet trained model with 93% accuracy or CIFAR data trained model with accuracy of 94% or time taken to answer one SQuAD question using a model with a F1 score of at least 0.75. All of these latency times are calculated as the average time taken when done this over 10,000 questions or images.

The third benchmark is MLPerf [9]. It is a machine learning (ML) benchmark suite for measuring the performance of ML software frameworks, ML hardware accelerators, and ML cloud platforms. MLPerf's workloads cover a broad spectrum of use cases. There are seven different AI/ML categories, which includes image classification, object detection, speech to text, translation, recommendation, sentiment analytics and reinforcement learning. This benchmark can model two scenarios: (i) closed model where changes to the model are minimal, primarily intended to test the underlying hardware and software systems, (ii) open model where the model is continuously enhanced using ML.

The three benchmarks calculate the training time and latencies for inference, but do not benchmark throughput of a system from inference point of view after an AI model is deployed. The proposed benchmark is to measure the inference throughput of a system and the price per inference of such a system, which has deployed the model for classification or identification.

3 Benchmark Description

TPC benchmarks fall into one of two classes, enterprise or express benchmarks [1]. Enterprise benchmarks are technology agnostic, i.e., instead of defining specific steps to be executed by a specific product or groups of products, they define domain specific workloads on an abstract level. Express benchmarks, on the other hand, define very specific steps developed for a group of products. The steps can be scripted and provided in form of a benchmark kits.

The decision whether to use the enterprise or express model for TPC-I was carefully weighted. It was decided to use the enterprise model, mainly because AI technology is still evolving at a very rapid paste. Enterprise benchmark, being technology agnostic, entices existing system providers to improve performance by adding new technologies to their systems. It also welcomes entirely new system providers that have technologies that are radically different from existing once and that could not have been anticipated at the time of benchmark development.

3.1 Use Case

TPC-I is modeled after a massive scale facial recognition system as described in section 2. The mixture and variety of operations measured by TPC-I are not designed to exercise all possible operations used in large scale AI systems. And they are certainly not limited to those of a facial recognition process. They rather capture the variety and complexity of typical tasks executed in a realistic large-scale AI system characterized by:

- Receiving and processing of high resolution photos
- Running inference on complex models
- Fulfilling latency and throughput requirements

3.2 SUT Driver and Communication Definition

A TPC-I test configuration consists of one or multiple System Under Tests (SUT), Driver System (DS) and Communication Interfaces (CI). A SUT consists of:

- One or more inference processing units,
- All front-end systems to support the backends, e.g. data communication processors; cluster controllers and workload balancers,
- All hardware and software components of all networks required to connect and support the SUT components,
- Data storage media to satisfy high availability requirements.

The DS provide Remote Camera Emulator (RCE) functionality that emulate the target camera population during the benchmark

run. TPC-I being an enterprise benchmarks allows for various system architectures to be benchmarked. Figure 1 shows a sample 3-Tier test system configuration, with a RCE layer, client and server systems. We group those parts of the system that will be part of the System Under Test (SUT). For instance, the Remote Camera Emulators and its Network are not part of the SUT. The SUT contains all components that will be priced.

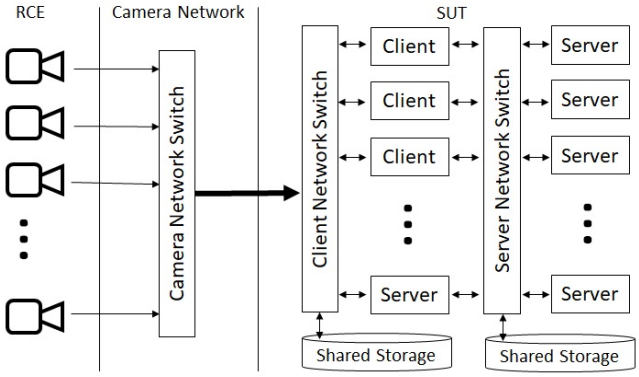

Figure 1: 3-Tier Sample Test System Configuration

3.3 Data Set and Scaling

The throughput of the TPC-I benchmark is driven by the number and activity of each APC station, which is emulated by one RCE. We are still in the evaluation process to determine which data set fits TPC-I. Of course, getting real person image data is imposible because of privacy concerns. Hence, we will need to settle on a data set that results in similarly complex inference operations. One data set being considered is Google's Street View House Numbers (SVHN) [16]. One of the largest advantages of SVHN is its size. It contains over 600,000 digit images. The complexity of its images, i.e. digits and numbers in natural scene images, makes it a very attractive data set compared to MNIST. Hence, SVHN resembles a significantly harder real world problem of recognising digits and numbers in natural scene images compared to MNIST.

TPC-I has similar scaling requirements than TPC-C. TPC-I requires that in order to obtain a higher throughput, more RCEs must be configured resulting in more photos being sent for image recognition, i.e. inference in the backend. TPC-I is designed to scale just as more APC stations are bing added to the system. However, certain latency requirements must be maintained as TPCI is scaled up. Each added APC must not exeed a passanger response time of 186s, which is the maxium of the negative exponential distribution of the user interaction in an APC and the tolerable time passengers are likely to accept interacting with an APC. Like the APC interaction profiles themselves, the frequency of the individual interaction profiles are modeled after realistic scenarios.

The intent of the scaling requirements is to maintain the ratio between the APC load presented to the system under test, the required space for storage, and the number of APC stations generating the workload.

3.4 Performance Metric and Execution Rules

The execution rules and metric are two fundamental components of any benchmark definition and, they are probably the most controversial when trying to reach an agreement between different companies in a benchmark consortium. The execution rules define the way a benchmark is executed, while the metric emphasizes the portions of a benchmark that are measured. We describe them in one section since they are intrinsically connected to each other and they are equally powerful in how they control performance measurements.

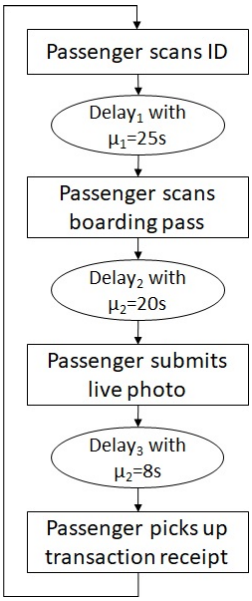

Figure 2: Passenger APC Interaction Profile

Execution rules and metric are modeled after TPC-C. The interaction profile between the airline passenger and the APC system is depicted in Figure 2. Once an APC system is available to a passenger he first scans his ID, then after *Delay₁* the passenger scans his boarding pass, after *Delay₂* the passenger takes his picture and submits it . Finally, after *Delay₃* the passenger picks up his transaction receipt and leaves the station for the next passenger.

Each delay is taken independently from a negative exponential distribution as follows: $Delay_i = -\log(r) * \mu_i$ with $i \in \{1,2,3\}$ and $r \in (0,1)$ a random number uniformly distributed. The completion time, i.e. response time (RT), is measured in the RCE with a resolution of 0.1 seconds. It is defined as the delta between the start time (T_1) and the end time (T_2) of the passenger interaction, i.e. $RT = T_2 - T_1$.

The TPC-I transaction mix represents the clearance process of a passenger by the APC. The metric used to report Maximum Qualified Throughput (MQTh) is the number of clearances per minute. MQTh is reported as "Clearances per minute TPC-I" CpmI. The duration of the test includes the ramp up time and the steady state time. When taking the MQTh the system must be in a steady state for an uninterrupted minimum of 60 minutes. The steady state must represent the true sustainable throughput of the SUT. Although the requirement of the measurement interval is at most 60 minutes, the system under test must be configured such that it could sustain the reported tpmC for a continuous period of at least 18 hours without manual intervention.

3.5 Pricing

The fact that a price/performance metric and an availability date metric is mandatory in TPC benchmarks since the first TPC benchmark was introduced in 1989 has been proven vital to the customres of the TPC. The TPC has taken a general approach to assuring that pricing of benchmarked systems is reported cor-

rectly and has real world relevance. As a consequence reporting of a three-year verifiable cost of ownership for all the components used in the system is mandatory in all TPC benchmarks. To regulate pricing, the TPC publishes the TPC Pricing specification.

TPC pricing specification require that the total price must be within 2% of the price a customer would pay for the configuration. Test sponsors are required to publish an updated price if future changes vary the price of the configuration outside of this range. As this requirement stands for the active life of the benchmark result, specific price quotations used in the benchmark do not require any "good for a number of days" statements. Line item pricing is required, although bundles of components that are identifiable with a single product identifier are allowed. Line items should be priced at the value that the supplier would sell for quantity=1.

Discount information must be complete enough that the final discount could be determined from other information included in the quotation. Each line item could have an individual discount, or a group of line items could be discounted at a specific rate, or the entire configuration could be discounted at a specific rate. Discounts could be based on quantity, total dollar volume, or the relative value of the combination of components included in a discount. Discounts may not depend on any purchase other than the components included in the benchmark's priced configuration.

Price quotations may be for the exact number of items bid or for a smaller number with an indication that it is good for quantities greater than the number. Since price quotations may not be dependent on past or future purchases, quotations may be used in more than one benchmark result, if they apply.

3.6 Auditing

Another key aspect of TPC benchmarks are the audit process. For enterprise TPC benchmarks the TPC mandates an independent audit by a certified auditor. While the TPC-I specification defines the benchmark specific rules for auditing, the actual audit is conducted by an independed, third party auditor. These audtiors undergo a rigorous examination process before they become certified TPC auditors, which is a requirement to audit TPC benchmarks. We are very grateful to our auditors as they serve a dual purpose. They make sure that the benchmark result they are auditing is compliant with the TPC-I specification. On the other hand, auditors continuously supply feedback to the TPC about issues they encounter in the field. This feedback is directly taken to the subcommittees to implement improvements to their benchmarks.

4 Conclusion

This work-in-progress paper described the current state of the TPC's efforts to create a benchmark in the Artificial Intelligence work space. We showed TPC-AI is an initiative to characterize massive scale inference representative many real-life scenarios. While there are still open issues and tough decisions ahead the authors plan to ratify TPC-I in the TPC this year to supply the industry with a much needed large scale benchmark in the AI space.

ACKNOWLEDGMENTS

We would like to thank the TPC subcommittee members for their continued support in creating a sound and relevant benchmark to measure the performance of AI systems. It is with no doubt the TPC's rule book that allows for a fair and competitive environment.

REFERENCES

[1] K. Huppler and D. Johnson. TPC express - A new path for TPC benchmarks. In Performance Characterization and Benchmarking - 5th TPC Technology Conference, TPCTC 2013, Trento, Italy, August 26, 2013, Revised Selected Papers, pages 48–60, 2013.

[2] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, Andrew Y. Ng Reading Digits in Natural Images with Unsupervised Feature Learning NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011.

[3] TPC Policies Version 6.12: http://www.tpc.org/tpc_documents_current_versions/pdf/policies_v6.12.0.pdf

[4] The Garfors Globe "100,000 Flights a Day" https://garfors.com/100000-flights-day-html/

[5] Federal Aviation Administration "Air Traffic By The Numbers "https://www.faa.gov/air_traffic/by_the_numbers/

[6] US Depertment of Transportation "U.S. International Air Passenger and Freight Statistics Report " https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=2ahUKEwi93_v53OzfAhUG2FQKHXBzAAIQFjAAegQIC-BAB&url=https%3A%2F%2Fwww.transportation.gov%2Fpolicy%2Faviation-policy%2Fus-international-air-passenger-and-freight-statistics-report&usg=AOvVaw1Q0JJa8RuGZydES3E4LLgQ

[7] US International Air Passenger and Freig Statistics https://www.transportation.gov/sites/dot.gov/files/docs/mission/office-policy/aviation-policy/321101/us-international-air-passenger-and-freight-statistics-march-2018.pdf

[8] DeepBench Gitup location: https://github.com/baidu-research/DeepBench

[9] MLPerf Home Page: https://mlperf.org/

[10] DAWNBench: An End-to-End Deep Learning Benchmark and Competition Cody Coleman, Deepak Narayanan, Daniel Kang, Tian Zhao, Jian Zhang, Luigi Nardi, Peter Bailis, Kunle Olukotun, Chris Ré, Matei Zaharia

[11] ImageNet Dataset: ImageNet: A Large-Scale Hierarchical Image Database

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li and Li Fei-Fei

Dept. of Computer Science, Princeton University, USA http://www.science.smith.edu/classwiki/images/8/8d/Imagenet_hierarchical_image_databaseP1.pdf

[12] CIFAR dataset: CIFAR-10 (Canadian Institute for Advanced Research)

Alex Krizhevsky and Vinod Nair and Geoffrey Hinton https://www.cs.toronto.edu/~kriz/cifar.html

[13] SQuAD data set: SQuAD: 100,000+ Questions for Machine Comprehension of Text Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, Percy Liang https://rajpurkar.github.io/SQuAD-explorer/

[14] EAutobahn: http://www.eautobahn.de/html/zahlen_und_daten.html

[15] Air Traffic By Numbers, Federal Air Traffic Administration https://www.faa.gov/air_traffic/by_the_numbers/

[16] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, Andrew Y. Ng Reading Digits in Natural Images with Unsupervised Feature Learning NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011.