Request for Comments: AIXPRT 1.0

Last revision: April 11, 2018

Introduction

This request for comment (RFC) explains the need for a new artificial intelligence (AI)/machine learning benchmark, shows how the BenchmarkXPRT Development Community[1] plans to address that need, and provides the preliminary design specifications for a new benchmark tentatively named AIXPRT. Use these instructions to comment: Request for comments.

We are looking for feedback and suggestions on these specifications from anyone interested in shaping the future of machine learning benchmarking. Please share this document with anyone who may be interested, and encourage them to join the BenchmarkXPRT Development Community.[2]

The need for a new AI benchmark

Based on input from BenchmarkXPRT Development Community members and the AI community at large, there is a need for an independent, third-party benchmark that functions as an industry standard for measuring the speed and accuracy of platforms running workloads in the AI space.

For several reasons, current AI performance evaluation tools are not well-suited for widespread adoption. For example, they tend to rely on theoretical peak metrics that do not accurately represent real-world application performance.

Workloads that do try to be more representative of real-world performance often lack ways to ensure consistent implementation across test platforms. When testing open-source networks on different hardware and software stacks, inconsistencies in test configuration and methodology, such as varying batch sizes, different data sets, and non-standardized reporting practices, can lead to inaccurate and misleading performance comparisons.

Finally, existing AI benchmarks typically require complex setups and are difficult to replicate for reproducible benchmarking. The fine tuning required to set up such tests on each system makes it exceedingly difficult to produce apples-to-apples comparisons.

AIXPRT development goals

Scope

The potential scope of an AI benchmark is as vast as AI: it could measure performance at the edge, in the cloud, or end-to-end. It could focus on training performance or inference performance for any number of relevant workloads, such as object detection, natural language processing, image categorization, and computer vision. It could use datasets with different sizes and sources.

[1] What is the BenchmarkXPRT Development Community?

[2] How to join

For AIXPRT 1.0, we will necessarily limit that scope so that we can create a viable first iteration. Initially, the benchmark will focus on measuring inference performance for workloads related to computer vision, and it will target single-node client or edge devices.

For this initial version, we propose selecting existing workload and test datasets, repackaging them, improving their ease of use, and analyzing their results to find better ways to present the data.

Workload scenarios

Within the scope of computer vision, we propose considering the following workload categories and corresponding neural network algorithms:

- Image classification: Image classification involves tasks such as identifying the type of animal in an image. The algorithms we propose to consider include Inception (GoogLeNet), ResNet, and MobileNet.
- Object detection and localization: Object detection and localization involve identifying the contents and their spatial location in an image and drawing bounding boxes around them. The algorithms we propose to consider include Single Shot Detection (SSD), You Only Look Once (YOLO).
- Segmentation/pixel-level classification: Image segmentation tasks attempt to classify each pixel in an image according to the object it is a part of. The algorithms we propose to consider include Fully Convolutional Networks (FCN), though this area is evolving rapidly and we may add additional algorithms.

Test data

A major development focus will be determining the test data to use for the AIXPRT 1.0 workloads. We propose using large general-purpose datasets, such as ImageNet and Stanford NMT, as well as vertical-focused datasets, such as Cityscapes. Using data from well-known sources will add credibility to AIXPRT.

We propose using different levels of data precision for different purposes. We propose supporting 32-bit floating point for a baseline comparison across different platforms. In addition, to provide a “best to best” comparison, we would use the lowest (best) level of precision supported on each platform, which might be 8-bit or 16-bit floating point.

In general, we propose using relatively small batch sizes to represent edge-inference use cases. This is more representative of many edge cases, such as vision sensors in autonomous cars or security camera streams supported by network video recorders.

Metrics and scores

In the long term, it is important to develop a consistent scoring methodology. To speed implementation, it is possible that early versions of AIXPRT may present only raw results for each workload.

UI and UX design

At this early point in the design process, we welcome input about the UI. We do believe it’s important to design the UI with these guidelines:

- The target platform for version 1.0 is Ubuntu Linux, so we will use Linux package management to make the installation of AIXPRT and workloads as simple as possible.
- The UI for AIXPRT should make it as easy as possible to configure, test, and display results.
- The UI needs to be extensible, to accommodate future workloads.
- The UI should allow the tester to run workloads with different parameters where appropriate.
- The UI should display the results in a consistent and easily readable format.
- The UI should allow users to save results.
- The UI should automatically gather as much test environment configuration information as possible. Testers still may need to enter some information manually.
- The UI will not automatically upload results to the XPRT database, so companies can run it on systems under development. Even so, the UI must allow testers to save results and make it easy to upload the results and configuration information of the current run, or any saved run, to the XPRT database.

Test system requirements

The table below shows what we hope will be the target platforms for AIXPRT 1.0. AIXPRT workloads will unavoidably require support in each relevant Inference SDK/Runtime. Therefore, we cannot guarantee all workloads will work on all platforms.

	Intel	**NVIDIA**
Inference SDK/Runtime	**Intel CV-SDK**	**NVIDIA TensorRT**
Operating system	**Ubuntu Linux 16.04 or higher**	**Ubuntu Linux 16.04 or higher**
Hardware	**Intel Xeon, Core, Atom, Altera FPGA, Movidius VPU**	**NVIDIA Titan, GTX, Jetson, and Tesla**

In future versions of AIXPRT, we hope to use the Qualcomm Snapdragon NPE SDK and Apple Core ML frameworks to develop workloads for Android and iOS. We will not do so in the first version.

Reporting requirements

We want to strike a balance between making the results as useful as possible and making it difficult to misuse the results. Our current thinking follows, but we welcome input.

Testers may publish without prior approval, provided they upload results and configuration information to the XPRT database. We would develop a process whereby testers can challenge results. If challenged, BenchmarkXPRT Development Community administrators will investigate the results and determine if the run is legitimate.

An open-source benchmark

Currently, we plan to make AIXPRT open source. In addition to the usual obvious advantages of making the benchmark open source, doing so may help increase marketplace acceptance of the benchmark. An open-source model offers unique challenges for a benchmark, because while open source encourages a constantly evolving product, benchmarking requires a product that remains static to enable platform comparisons over time.

We would especially like input on this area before we develop a proposal for handling an open-source benchmark.

Request for comments

Please send us your comments on AIXPRT in general, the proposed AIXPRT 1.0 requirements, or our open-source strategy.

Please provide feedback via email to benchmarkxprtsupport@principledtechnologies.com before May 13, 2018. BenchmarkXPRT Development Community administrators will collect and evaluate feedback to this RFC and publish the final design specification after this date.

About the BenchmarkXPRT family

The BenchmarkXPRT tools are a set of free apps that help you test how well devices do the kinds of things you do every day. The BenchmarkXPRT suite currently includes the following tools:

WebXPRT tests the performance and Web browsing capabilities of any device with Internet access.

TouchXPRT is a Universal Windows Platform app that tests the performance of Windows 10 devices.

MobileXPRT tests the responsiveness of Android devices.

CrXPRT tests the responsiveness and battery life of Chromebooks.

HDXPRT uses commercial applications to test the capabilities and responsiveness of PCs.

BatteryXPRT measures the battery life of Android-based phones and tablets.

We designed these apps to test a wide range of devices on a level playing field. When you look at results from XPRTs, you get unbiased, fair product comparison information. For a guide to which XPRTS to run on your devices, see our Infographic: Consult the right XPRTs.

What is the BenchmarkXPRT Development Community?

We built BenchmarkXPRT around a unique community model. Community membership is open to anyone, and there are many ways to participate. Members of the BenchmarkXPRT Development Community can be involved in every step of the process for AIXPRT and the other XPRT benchmarks that are of interest to them. They can give input on the design of upcoming versions, contribute source code, and help test the benchmarks. Community members have access to previews of upcoming benchmarks and to the source code of released benchmarks.

The community helps us pursue diverse input during the design process, which in turn makes the tests more representative of real world activity. Granting access to the source code serves to increase confidence in the code and makes it easier to improve future versions of the benchmarks.

For more information, please watch the video introduction to the community and consult the BenchmarkXPRT FAQ page. To see just how far the reach of the BenchmarkXPRT community extends, check out our infographic.

How to join

If you are not currently a community member, we encourage you to join. Registration is simple, and does not obligate you to participate.