Tony (00:04):

Welcome to Code Together, a podcast for developers, by developers, where we discuss technology in trends and industry. I'm your host, Tony Mongkolsmai.

(00:16):

Today, we're going to talk about enabling machine learning for exascale computing. We're lucky enough to be joined by Taylor Childers, who is a member of the Argonne Leadership Computing Facility data science group where he works on multiple projects that require deep learning and simulation to be run on large supercomputers, including preparing for Aurora, the upcoming exascale computer.

(00:35):

He has a PhD in particle physics and worked at the CERN laboratory in Geneva, Switzerland, for six years before coming to Argonne in 2013. Welcome to the podcast, Taylor.

Taylor (00:44):

Thanks.

Tony (00:46):

Let's just jump right into it. So we're talking about a large-scale HPC platform in Aurora. And we're also talking about how you're going to leverage AI to do really cool things, leveraging all that hardware. Why is now the right time to combine this AI and deep learning and leverage it for science?

Taylor (01:07):

The AI revolution started 10 to 12 years ago. And it was really ignited by industry. Industry had large datasets; places like Google and Facebook. And they had large computing resources.

(01:22):

Facebook and Google have large warehouses of computing resources located all over the world. And people were providing them with relatively well-labeled datasets that they could use to run machine learning algorithms and train those algorithms to do things for them.

(01:42):

Now, in the basic sciences, I'm personally familiar with high energy physics. We collect very large datasets from our particle accelerators. And we're interested in understanding the basic laws of the universe.

(01:58):

We can use techniques like AI, similar to what the industry was using for, say, facial recognition or object ID. We can use those to identify objects on our detectors with our large datasets.

(02:14):

And with computer resources like Aurora, which are available to basic science researchers like those in high energy physics, we can leverage those the same way that Google and Facebook leveraged their large clusters that they have in their warehouses.

(02:31):

So we have the big datasets. We've had those big datasets for a long time. Now we're starting to see the ability to have large computing resources that we can use to do the AI trainings that are required and require those large resources.

Tony (02:48):

That's pretty cool. Actually, it's interesting because you're actually taking what industry has done and making sure that it works for something in the scientific field.

(02:59):

In the past, it used to be the HPC science field-led industry. Now it sounds like, in this case, with AI, we're flipping it around the other way.

Taylor (03:07):

That's right. The experiments at the LHC, for instance, led to the worldwide grid, worldwide cloud for scientists. And that predates AWS or Google's Cloud.

(03:22):

When it came to AI, industry definitely outpaced basic sciences and moved that ball forward and really opened the door to possibilities. So we're playing a little bit of catch-up. But with machines like Aurora, we should be back in the lead, especially in the basic sciences that's where the DOE and public sector shine, whereas in the private sector, they are much more worried about different kinds of problems that their users are interested in.

Tony (03:53):

Cool. And so with that, what kind of models are you leveraging? You mentioned computer vision. How are you taking those and what problems are you actually looking to solve in science today at Argonne National Laboratory?

Taylor (04:06):

So, in preparation for Aurora, we actually put out a call for early science projects. These are projects that will get first access to Aurora before anyone else.

(04:18):

And the idea is they work with us through the development process with Intel. And we try to make sure that those applications are ready to run on day one when Aurora is turned on.

(04:29):

Some of those projects include, for instance, a fusion reactor research group. They're a group who are involved with some of the fusion device development around the globe. And one of the things you have to worry about when you're running a fusion reactor... Imagine you have a container and you're trying to hold very hot plasma, temperatures of the sun, inside, using magnetic fields. And you don't want that plasma to touch your container because it'll melt.

(05:03):

So one of the challenges is as we advance that technology... In the beginning, we were using small plasmas. If we lost containment, it wasn't a big deal. Nothing was damaged because everything was really small.

(05:19):

Now as we grow the technology and we are able to hold the plasma stable for a longer period of time, we need to scale up that technology to put them into production. Well, you want to know ahead of time if your plasma is unstable, and it needs to be quick.

(05:38):

In our current project, our goal is to know within 30 milliseconds that we are on a path to losing the plasma and handwritten human algorithms tend to be slow compared to artificial intelligence algorithms that are developed with modern networks.

(06:00):

So this fusion group is using what's called a recurrent neural network, which is really good for processing sequential time data. And they're trying to use that to process all the sensors that exist on a fusion device in order to predict when an instability will occur so that they can avoid losing the plasma in an unsafe way.

Tony (06:26):

So with AI, it sounds like we're able to do something that, before, we weren't able to. Is that right? Were the hand-coded algorithms not really allowing us to do what we needed to do to make strides forward around the nuclear fusion science?

Taylor (06:41):

That's right. Humans are really good at pattern recognition in the world around us. When we talk about data analysis, trying to do pattern recognition, often what we do is we go through a series of steps that reduces the complexity of our data until we can put it in a 2D plot of data that we can then visually inspect.

(07:04):

And maybe go back and we do it again, and we do it again. We iterate that way. Well, a computer isn't limited to two-dimensional pattern recognition. And AI has really shown that it can handle multidimensional data and find patterns and be able to accurately predict, in those high-dimensional spaces, what's going to come.

(07:29):

And so, when you take the hundreds of temperature, pressure or magnetic sensors around the fusion device and you feed those into a neural network, it can make sense of all of those at the same time. It can analyze them in useful ways to find the patterns that we're interested in.

(07:52):

We have to be able to guide it through our training process. For instance, we simulate, or also, we have some actual recorded data of failures. We then use those to train the AI and say, this is what a failure looks like. This is all the data that we collected leading up to that failure. And the AI can then learn from our previous mistakes and try to help us identify them in future so that we don't make them again, essentially.

Tony (08:22):

What's really interesting there is AI is really good at identifying things it's been able to be trained on a lot. Just out of curiosity, if you know off the top of your head, how many failures do we actually have that we feed into this model so that we can get some good output from it?

Taylor (08:35):

I think the current dataset is on the order of hundreds of examples.

Tony (08:41):

Okay. That's interesting. I think we usually think... At least myself, as a layperson, when I think of nuclear fusion, I feel like it's something we do once in a really long time, like the space shuttle launch that NASA is trying to do.

(08:53):

You feel like it happens once in a long time. But really, it happens a lot more often, it sounds like. And you guys are really trying to grow this as quickly as possible.

Taylor (09:00):

Yeah. We have these facilities, the smaller ones, that have been in existence. And they record their data all the time and so we have this data available and now we're trying to use the AI methods.

(09:14):

Even though we only have hundreds of examples, obviously, those data are recorded at a very high rate. So in the end, the amount of data to process is quite large because you are considering measurements at the microsecond scale.

(09:30):

So you've got microseconds. And you're interested in milliseconds of a warning. So there's quite a bit of data there to mine.

Tony (09:39):

That makes sense. You've had systems that are big before at Argonne National Laboratory. But Aurora is a whole another scale of compute power and processing power. How are you guys planning on leveraging that versus what you guys had before?

(9:55):

So we're talking about a factor 10 increase in computing power. We're also talking about an architecture change. Going from a full CPU machine, which is what we've had for decades, to a hybrid CPU/GPU machine where the GPUs are really capable of handling linear algebra.

(10:28):

And machine learning is just a lot of linear algebra. Even though it always sounds very complicated, it's really just a bunch of polynomials that need to be computed across matrices and vectors.

(10:39):

The GPUs are really excellent at doing those, doing them fast, and getting you the results that you need. And that's another reason why we think, in the fusion research case, where you can hit that 30-millisecond requirement because you're able to do so many computations per second.

Tony (10:58):

And you also mentioned that you have a lot of data. And how, as the models have grown and the datasets have grown, that's really allowed us to do more novel things in the AI and science space.

(11:09):

And moving all of that data can be very expensive and very painful. So I know that Aurora is targeting a way to leverage Intel's Distributed Asynchronous Object Storage (DAOS) as part of their workflow. Can you talk about how that helps enable you to get better science and AI work done?

Taylor (11:26):

Yeah. When you're using a supercomputer, you've got Aurora, which is going to have more than 60,000 GPUs. And we're doing leadership science. This means that when we run projects on our systems, our goal is to use the full machine as one machine.

(11:47):

So our biggest projects are going to be running all 60,000 GPUs to do single, coordinated calculations. Not only do they need to be able to communicate with each of those GPUs, with our high-bandwidth network, but we also need to be able to get data in and out.

(12:06):

And a supercomputer is very different than your desktop machine. Your desktop has all that it needs. It has a hard drive. It has the memory, the compute cores. In a supercomputer, you put all of those things in separate spots. So you've got all your computer power in one place. You've got all your hard drives in another place.

(12:28):

And so DAOS is the infrastructure that's being developed by Intel to connect our storage to the compute so that you can bring your data in, you can take your data out. So that's a key part of our system and is really important to ensure that the researcher can do the work that they need to do.

(12:54):

AI is very data-intensive. Our trainings tend to require lots of data coming through each GPU, and you tend to do that in a cyclical way. So it'd be really interesting to see the speeds that people pull off of DAOS as we deploy at the end of this year.

Tony (13:14):

Yeah. For sure. The bottleneck, obviously, is how do you get all of the data to the compute to make sure that you can really leverage those 60,000 GPUs. For most people listening probably, that sounds like a very normal number if you're an HPC person. But for the average person, 60,000 GPUs must sound like a crazy number for them.

Taylor (13:33):

It is impressive even if you're used to it.

Tony (13:38):

And when you talk about how we get all of that data there, when I was working on a data center for Intel, trying to build out a scalable system for AI, we obviously knew that networking and storage were going to be really important to us as well.

(13:50):

Can you talk a little bit how some oneAPI components that Intel is providing, like the One Collective Communications Library (oneCCL), is being leveraged? And then how Intel's one Data Analytics Library (oneDAL) are helping make sure that you guys are able to do the compute-intensive things you need in a way that's really efficient?

Taylor (14:10):

Yeah. People who do AI have a common set of tools. Largely, these are Python tools supported by either the open community or the industry folks like Google and Facebook, who are the big supporters of TensorFlow and PyTorch. Those are the leading AI frameworks currently. And then you have Python libraries like scikit-learn.

(14:36):

And we've been working with Intel to prepare for Aurora and make sure that these applications are ready for Aurora. For instance, with oneDAL, oneDAL encompasses a lot of the traditional statistical learning or machine learning and algorithms like K-Means and these sorts of things for clustering or regression. And those live in scikit-learn.

(15:02):

So whenever one of our researchers from, say, high energy physics or biology comes along with their dataset and their application that they've been running on their laptop or on their university cluster, they're coming with those libraries.

(15:18):

And Intel has been working with us to backend those libraries with the toolkits that Intel have been providing like oneDAL. So oneDAL has those optimized algorithms inside that make the best use of Intel GPUs and CPUs as they can.

(15:38):

So when you install your Python libraries and you want scikit-learn, there's going to be the option to install oneDAL on the backend so that when you're running on Aurora... Or, say, hey, in the future when you've got your laptop and it has an Intel GPU in it, you're going to be able to run that the same way you would anywhere else. And you're going to get the benefits of the speed-ups from the GPU.

(16:306):

In the case of TensorFlow and PyTorch, it's the same thing between things like oneDAL and oneCCL. oneCCL is very important for the networking, for communicating between the GPUs.

(16:17):

When you're training on a large-scale neural network, typically, you are training in what we call a data-parallel way. So that means that you've got your model on every GPU. And you have to synchronize your model parameters periodically. And to do that, you have to communicate.

(16:39):

While every GPU is processing or training the model on different data, the models need to stay synchronized. So we do that by communicating over the network. oneCCL is able to do that for us between GPUs on the same computer or GPUs across the network.

(17:00):

And a tool like Horovod, for instance, which is a tool that we often use for data-parallel training, having that oneCCL plugin for the backend is very important. So we've been working to have that ready for Aurora so we can do that data-parallel training across the 60,000 GPUs.

Tony (17:21):

That's great. Yeah. The networking part is, obviously, such a big deal. We were talking about the datasets. How big of datasets are you actually talking about? When you guys are doing this analysis, are you guys talking hundreds of gigabytes? Tens of gigabytes? What size are you guys actually using for your datasets?

Taylor (17:39):

It varies a lot based on the domain of science. In high energy physics, we probably used to be leading the data size race 20 years ago, before everyone had a camera on their phone and Facebook.

(17:53):

But that quickly outpaced us once the Facebooks were able to collect photos and data from everyone. But we're talking about petabytes per year collected from machines. And then, of course, all of the additional simulation and analysis software that goes on top of that.

(18:14):

Researchers come in with different sizes. So they might have a small portion of that data that they want to analyze. They may bring in large portions of it that they want to analyze. Usually not bringing in petabytes.

(18:26):

But some of our most intensive applications produce tens of petabytes per year on our machines, and that was at previous scales. And it tells you something when you're going to increase the compute capacity by a factor of 10.

(18:46):

Now, when it comes to AI, often, your data science is fixed, unlike traditional simulation. In simulation sciences, you're simulating something, and so you produce a lot of data, whereas, in AI, you're producing a model. And the model is typically comparatively smaller in memory than the data that you required it to train on.

(19:13):

So it's the same amount of data. And then you do your training in a cyclical way. And, obviously, some of the projects in the future are interested in things like continuous learning where you constantly have a new stream of data coming in and you're updating an old neural network to include that new information.

(19:34):

So in that case, you would have a live stream of data either coming from your detector or an outside source. So that's one of the things we'll be working on with Aurora is live data streams updating ML models, machine learning models, as they collect more data.

(19:50):

You asked about data sizes. I would probably say that some of our biggest users are probably going to have data sizes in the range of a petabyte they would actually try to do training on.

Tony (19:58):

That's gigantic, though. It's such an order of magnitude over what we've been using, what we think about as large datasets. So I know that you also have some kind of next-gen, I'll say, Intel-type hardware. You guys have an AI testbed.

(20:04):

How are you guys planning on leveraging that testbed so you can look beyond where you're at now, knowing what Aurora is going to look like? What are you guys looking towards in the future, using that testbed?

Taylor (20:16):

Yeah. As a research lab, it's important for us to gauge what's happening now. Obviously, there's a bit of a hardware renaissance in computing, where you've got Intel coming back to the GPU world in a big way. And you've got other countries investing in custom architecture development.

(20:38):

The European supercomputers, Chinese supercomputers are all using very different hardware. So it's important for us to be engaged in following that activity.

(20:49):

The AI testbed is an outgrowth of that goal. Our testbed is really geared toward partnering with industry, putting their hardware on the floor where our researchers can play with their systems, give them feedback, and have that be a two-way street where they get feedback that they need, and we get early experience with really custom chips that are intended specifically for artificial intelligence.

(21:23):

And there's a lot of different methods that are being employed as far as how to design the chips to make them the most effective. And so we currently have... Just counting quickly. We currently have five systems on the floor that researchers can gain access to and play with.

(21:41):

And one of the ways we want to also keep track, for the next few years, where we need to be investigating how to allow our users to effectively use future architectures.

Tony (21:56):

Yeah. So that gives you the hardware look of what technologies you're going to be leveraging. As we build out this AI for Science program at Argonne and leveraging Aurora, where do you hope that this kind of technology takes us in the next couple of years?

Taylor (22:12):

That's a good question. So a lot of what you see right now that's very popular in industry are these foundational models, these models that can encompass a large corpus of information.

(22:29):

And one of the hard things in basic sciences is being able to fully understand everything that came before. And with AI doing very well at language modeling, language understanding, it would be really interesting to be able to build up some basic foundational models that have parsed historical publications, results that curate domain data.

(23:07):

You can imagine having a foundational chemistry model that has processed all the historical publications about chemistry. And, of course, you can make it as specialized as you want. It might be specific to some corner of the field that you're interested in.

(23:25):

And I'm sure it will start that way because making monolithic things is hard. But having a generic resource that you can quickly check, has this been done? What was the result? How is the study, point me at the paper? Even that would be a huge tool for modern scientists where our domains are so split into the nitty-gritty details of every little corner of the science.

(23:59):

In high energy physics, there are just so many things that you could focus in on and study this one thing for your whole life. And every field is that way. Biology is a huge, diverse ecosystem of people studying the most complicated systems.

(24:21):

It would be really beneficial to have these corpuses of knowledge in an easily accessible way versus now I have to go Google search for all the PDFs that relate to my research. Right? That's just a real challenge.

(24:39):

I think that is something we could easily achieve in the next five years because we have the AI technology. Now we just need to do the work. Obviously, I don't expect private industry to do that for us. And if they did, they would probably keep most of it for themselves because it would be really useful, especially for materials R&D and so forth.

Tony (25:00):

Cool. It's almost like democratizing science in some sense. Right? Making things discoverable.

Taylor (25:04):

Yeah. Yeah. Discoverable science. Exactly.

Tony (25:09):

Well, that'll be great. Okay. I think we're about out of time here. I'd like to thank Taylor for joining us. Thanks, man. It was great talking to you.

Taylor (25:17):

Yeah. No problem. Thanks for having me.

Tony (25:19):

And thank you to our listeners for joining us again on the Code Together Podcast. We'll be back with some more interesting topics for you to listen to.