05202020 Build Kevin Scott Keynote

Build
Kevin Scott
Monday, May 19, 2020

KEVIN SCOTT (EVP, CTO, Microsoft): Welcome and thank you all for joining today. I'm so excited to be here with everyone virtually this year for Build.

I'm a maker and an engineer at heart, so this event has always been one of my favorite opportunities to connect with the folks out there doing the real innovation on top of the platforms and tools that we provide. That innovation is something developers have been driving forward for decades, since the dawn of the computing revolution.

And at Microsoft, our job has always been to empower you with the best possible tools for you to do your job, creating the world's digital future.

In a present defined by uncertainty, faced with daunting challenges and unprecedented need, what you all do is more important now than ever. And today I'm here to talk to you about what's coming next, the technology trends that will shape how software and development look in the future.

I'm lucky to have lived through several eras of the ongoing revolution in computing. I think you could argue that revolution is a bigger force shaping our world now than it ever has been.

Each era of the computing revolution has been defined by boundaries and constraints. We overcome the constraints and push past the boundaries, and we then forget the boundaries existed as we accelerate breakthroughs on the path to ubiquity.

The great author and futurist Arthur C. Clarke's *Profiles of the Future* contains a quote that I think embodies the spirit of exploration and discovery perfectly: "The only way of finding the limits of the possible is by going beyond them into the impossible."

There are a number of big picture developments that are going to change how we all approach technology in the future, the end of Moore's Law, the explosion of data, the evolution of the edge among them.

We've been able to take for granted that there's a PC on every desk and in every home for so long, that we forget how impossible a goal that must have seemed in the late '70s and early '80s.

Bill Gates wrote his first commercial program on an Intel 8008 microprocessor. The most powerful GPU used for training AI models right now is billions of times more powerful than this device. If you combined every 8008 ever made, that would still be several orders of magnitude less computing power than you carry around in your pocket in 2020.

And that wasn't all. The scarcity of compute at the beginning of the PC era was just one of many obstacles that had to be overcome before we could have personal computing so ubiquitous that we all have the luxury of taking it for granted.

And here's the interesting thing, these impossible sounding constraints that Bill Gates and a bunch of other personal computing pioneers confronted in the early days of that era seemed more like challenges and opportunities to them than obstacles that should deter them.

None of those pioneers could have accurately predicted what personal computing would become, but they had a pretty good idea of the trends that would enable what they were attempting, as well as the technical challenges that they had to overcome. They invested into those trends and focused their ingenuity and creativity on overcoming those challenges.

Today, I want to focus specifically on the trends and constraints that are influencing one of the most significant developments in the history of computing, the explosion of large scale machine learning models and rapid advancements in AI.

Microsoft has been able to do some amazing things to drive this trend forward, but none of the major shifts in our computing revolution have happened just because of the efforts of one company.

PCs became a phenomenon not because of the PC itself, but because of the breadth of things people did with them. PCs are a platform for others to build on top of. The internet changed the world not because of TCP/IP and HTTP, but because it was a platform for individuals and entrepreneurs and big businesses to create and innovate.

Even what you do every day on your mobile device isn't about the technology and the device, but more about the breadth of applications on those devices and the people, content and services those applications connect you to.

At every one of these inflection points, the leaps forward we made were because of the contributions of developers. Any platform is only as good as the developers who use it. For AI to achieve its full potential, it must be a platform, and this platform will need to be powered and delivered at truly unprecedented scale and democratized so everyone can innovate and build on top of it.

That's where all of you come in. The future of AI will be in the hands of the developers who can harness its powers and you won't need to be a data scientist or machine learning specialist to do so. For any developer, the wildest dreams of what you can achieve and create with technology will become accessible as we build AI at scale.

The reason that I wake up every morning excited about my job is not because we get to work on these things, but because we get to work on them in service of others. That's why I feel so fortunate to be speaking today with all of you, the explorers who are going to help us expand the boundaries of what is possible.

We'll step back for just a moment and start today off with what I know has been top of mind for everybody, the global crisis of COVID-19. The last few months have shown us why we need the kind of boundaryless thinking that Clarke spoke about to apply against a massive, intractable challenge that has impacted every single one of us. Developers all around the world have risen to the challenge presented by this pandemic to create and deploy new software and tools with unprecedented speed.

Let's join Peter Lee, who's been leading some extraordinary efforts within Microsoft to channel the creativity and passion of our own employees to help.

PETER LEE (CVP, Technology & Research): Thanks, Kevin, and hi, everyone. As all of you know, this pandemic has had terrible consequences all around the world. From the loss of life to social isolation to an economic decline that rivals the Great Depression, we've all been confronted with profound challenges.

But, you know, this crisis has also brought out the best in humanity. People and organizations around the world have been working together, often virtually, to build the tools, the technologies and the solutions that people need, people like healthcare workers on the frontlines or researchers and scientists racing to find the new drugs and vaccines that we all need.

We here at Microsoft have been especially inspired when we've had a chance to work together and code together with these types of people and organizations. And as one example, I'd like to share a snippet of a Teams conversation I recently had with Rod Hochman, who is the CEO of the Providence Health System, on their deployment of the Microsoft Health Bot.

Hi, Rod Hochman. It's really a privilege to talk to you. It's been such a great collaboration between Providence Health and Microsoft, and I'm really just pleased that we have this chance to kind of catch up on things during this pandemic crisis.

ROD HOCHMAN (CEO, Providence St. Joseph Health): Boy, talk about a time for collaboration and the great ability to bring technology and science and clinical care together. As hard as this has been, it's also been amazing to watch all of that happen.

PETER LEE: Yeah, and we still have a lot more to do. And then, of course, after that first patient, you started to see a huge influx of inquiries from concerned citizens, people who were seeking advice and help if they were feeling symptoms. Can you say a little bit about how the COVID-19 self-assessment bot called Grace helped?

ROD HOCHMAN: Sure. Immediately when you have a pandemic epidemic, the next thing is going to be how do you triage people so that they don't go to the wrong place. And immediately, what we worry about with someone with COVID-19 is that we're going to quickly overwhelm all of our emergency rooms. We'd infect everyone there and we'd be out of business. The name of the game is get people assessed way before they come to the emergency room.

Well, how do you do that? And how do you make people – you know, what kind of tools do they have so they can figure out am I sick enough? Should I go? If I do, how do I get in touch with

someone? Quickly, with Grace and creating a bot which could sort out for an individual, if I had X, Y and Z, then what do I do, and then once they did that, where do you direct them? Well, do you go to the emergency room? Do you go to express care? Do you call your doctor up? Do you get a telehealth visit?

Quickly, what we were able to do with a bot was get people information, and then importantly, be able then subsequently direct them to places other than emergency room or an ambulance to go, and that probably is one of the things that saved our hospitals. And we were able to triage folks without getting overwhelmed.

PETER LEE: It was just so great working with your tech teams collaboratively on that bot. And in fact, you then connected us to the CDC to really help them establish the same kinds of capabilities. It was just a fantastic collaboration.

ROD HOCHMAN: Because I think we had well over 150,000 that used the bot, 170,000 other time users, and the exchange of information was 3.2 million messages. And that was just in the space from about early March till up until a little bit before now.

PETER LEE: Incredible.

ROD HOCHMAN: And that led to about 13,000 televisits a day, 13,000 a day. We're completely bypassing having to bring people into primary care offices, but getting them the right care at the right place. And then you look at the ability for a tool like this to be spread to everywhere so that we could be triaging patients, whether you're in New York, whether you're somewhere else, and be able to do it consistently.

PETER LEE: It's amazing what even a small application of AI can do to help a health system respond to a crisis. In fact, the COVID-19 Health Bot has now been deployed at over 1,500 healthcare sites in 23 countries around the world, and it's helped over 32 million people self-assess their own symptoms.

But it doesn't stop there. We're also donating massive amounts of GPU computing power to organizations like ImmunityBio, Folding@home, and great research groups around the world. They all really amount to working together. No one organization, no one person is going to be able to beat this thing. It's working together, developing new technologies together that is going to be necessary to overcome this crisis.

And so, for all of you, we look forward to serving you and to working together with you. Please stay safe and healthy, everyone. Thank you for all that you're doing.

KEVIN SCOTT: Thank you, Peter. As a company and as an industry, we're on a journey to solve the world's most intractable problems through the massive power and creative potential of AI. One of the biggest developments in the field over the past couple of years has been the training of very large, deep neural networks to discover generalized language representations that can be reused in many different applications.

These new models train on very large volumes of unlabeled data using new techniques known as self-supervised learning. Models trained using these new techniques have achieved levels of performance in natural language processing that have upended the field and are now being applied to other domains.

Even though the models have already become very big, their performance continues to improve as they get bigger. Given that these models are very large and do most or all of their training on unlabeled data, one of the primary things limiting their performance is having an abundance of compute power to throw at the training task.

The implications of this and what it means for AI are massive. To tell you more about this trend, let me introduce you to my technical advisor, Luis Vargas, who's been leading and aligning our cross-company initiative to bring together these large scale models and new computing architectures that together can power AI at unprecedented scale.

LUIS VARGAS (Partner Technical Advisor): Thanks, Kevin.

My name is Luis Vargas and I work in Kevin's team, helping drive our AI strategy. In the next nine minutes we'll talk about AI at scale, what it is and why it matters. I have three announcements for AI developers and a demo of AI, settling a longstanding dispute from Star Wars.

As you understand more than most, the explosion of internet data and large amounts of compute from the cloud have pushed the fast progress of AI. Microsoft has contributed to this progress by advancing the state of the art in areas like speech recognition, computer vision and natural language understanding. And we have made many of these capabilities available to everyone as Azure Cognitive Services.

Much of this has built on supervised learning where we train AI models from scratch to perform a task by using large amounts of labeled data. Because of this dependency, learning is constrained by human effort.

Over the past year, we have seen the emergence of a new trending approach called self-supervised learning where AI models can learn from large amounts of unlabeled data. For example, models can learn about language by reading large volumes of text and predicting words and sentences as they do so. No labels are needed, because the labels are in the data itself. The words and the sentence is being predicted.

As the model does this billions of times, it gets really, really good at understanding how words relate to each other on different contexts. This results in the model acquiring a very good understanding of the language.

The more data that is used for learning, the richer the understanding. Thankfully, in today's world, there is an immense amount of web data that we can crawl and curate though Bing.

The language understanding and codec in the model can then be fine-tuned to support a large amount of language tasks, such as sentiment analysis, search, question and answering, or summarization, all this through a process called transfer learning. This removes the need to train task-specific models from scratch and increases the accuracy of tasks, while dramatically reducing the need for labelled data.

Now, besides data, the size of a model determines its ability to learn and encode the complexity of language. The model size is described by its number of parameters, roughly the number of connections in this neural network. The more parameters that a model has, the better it can capture the difficult nuances of language.

A year ago, the largest AI models in the world had around 1 billion parameters. Our Turing NLG, Natural Language Generation model, the largest today, has 17 billion. A rough analogy is that we have gone from third grade reading and comprehension level to a high school level.

Let's look at it in action.

Let's start by asking the model a couple of random questions. The model has learned facts, concepts and grammar from large amounts of data, including all of Wikipedia, tens of millions of web pages and thousands of books.

When did we first land on the moon? Notice that the model is not doing a query against a database or an index, but instead understanding the semantic meaning of the question, relating it to the world of information that it has learned, and synthesizing a grammatically correct sentence.

What are the Enterprise's main weapons in Star Trek? And those are phasers and photo torpedoes. Nice.

Let's now copy and paste a document that the model has never seen before, like this memo from Satya about the Microsoft response to COVID-19. And let's ask the model some questions about it.

How many messages does the healthcare bot respond to? You can find the answer here, 1 million messages per day. Great.

What was the first for the University of Bologna? With some semantic understanding of the content, you can find the answer here. They moved 90% of its courses for 80,000 students online to Teams in three days, which is awesome.

How are Power Apps and Power BI being used? They're being used to manage bed count and inventory of critical supplies and sharing that information with others across the region. Amazing.

How are scientists using GitHub? To power a distributed computing project to assist researchers developing potential therapeutics using volunteer computers, which is fantastic.

Before we move on, as I promised in the beginning, let's finally settle the longstanding Star Wars debate of who shot first in the Mos Eisley cantina. By copying the Wikipedia article about the relevant scene, I'm asking the model, who shot first. There you go, Solo shot first.

If you watch the video again, notice that response is not found explicitly in the article. Once again, the model has understood the content and synthesized the appropriate answer.

Now, let's ask the model to do something different, to summarize content. Let's start with something that most are familiar with, the story of Romeo and Juliet. By copying it from Wikipedia, I'm asking the model to generate a summary. As the model reads the text, it abstracts the most important ideas from it and stitches them together into a cohesive summary, tweaking it as it makes progress.

See how the completed summary contains the main points. Romeo and Juliet fall in love. Juliet's parents force her to marry Paris. Romeo poisons himself, and Juliet stabs herself. And it discards other points less relevant, like Tybalt and Mercutio.

Let's now summarize a complete long article about Microsoft's commitment to be carbon negative by 2030. Once again, as the model makes progress on the article, it abstracts the most important ideas into the summary, reassessing the relevant priority and tweaking the summary accordingly, while maintaining cohesiveness.

Given the length of the article and the importance of multiple aspects, this is going to be a much richer summary. We refer to the current trend of increasingly larger AI models and their ability to power a large number of tasks as AI at scale. We believe that this trend will continue into the hundreds of billions and eventually trillions of parameters.

Training models which trends to hundreds of billions of parameters requires big clusters of hundreds of machines with specialized AI accelerators interconnected by high-banded networks inside and across the machines. Azure provides us with such clusters, for example with the latest with NDv2 series with V100 GPUs, connected by NVLink and Infiniband.

Training models at scale requires the ability to distribute and optimize the training of the models with large amounts of data across all of the AI accelerators in the cluster. When models are so large, better training requires on one hand splitting the neural network layers across multiple machines, and on the other, partitioning the huge training dataset into batches to simultaneously train multiple instances of the model across the cluster, while combining the learning.

Over the past year, we have developed state of the art optimization techniques to make these processes significantly more efficient by reducing redundancy in the state, stored and replicated across the GPUs.

We recently open sourced these techniques into a process library called DeepSpeed to enable everyone to train AI models 10 times bigger and five times faster on the same infrastructure.

We have continued innovating at a fast rate, and I'm happy to announce that today we're open sourcing the second version of DeepSpeed, which now enables you to train models up to 20 times bigger and 10 times faster.

Besides the speed, Microsoft Research has created various other complementary state of the art training optimizations targeting different levels of the software stack between a framework like PyTorch or TensorFlow and the hardware. We wanted to make it easy for you to use all these training optimizations together with a framework and the hardware of your choice.

I'm happy to announce that today, we are bringing together the optimizations from DeepSpeed, as well as other libraries from Microsoft, to the popular ONNX runtime. This adds support for highly efficient training, besides inferencing, to this framework agnostic and hardware agnostic AI runtime.

AI at scale and the reuse of large AI models is transforming the way that we operate at Microsoft, and you can see this already showing up in our products. For example, Bing fine tunes our Turing language model to support search and question and answering of the web, and then Word does this over documents, Outlook uses it to suggest replies to emails, and the most relevant documents for a meeting. And Dynamics uses it to suggest actions to a seller, based on its interactions with a customer.

However, language learned from web data is sometimes not enough to support the specific domain needs of our products, for example, the productivity domain in Office, the business domain in Dynamics, or the professional domain in LinkedIn. Every organization uses its own distinct vocabulary that a language model must learn.

To achieve this, we're training one personalized language model per organization, by augmenting our best Turing model with the data generated within the organization itself. As with the best model, we use self-supervised learning to do this without the need for labeled data. This allows us to automate the learning process within each organization, while ensuring privacy.

AI at scale is improving not only our models' understanding of language but also their understanding of relationships between language and other types of data. We're training large self-supervised models jointly across text and images to support tasks combining these, like searching visually for images with a text description in Bing.

We're automating the capturing of images for accessibility in Office. Similarly, we're training models across text and video, for example to automate the creation of outlines with the sections identified in a video.

Finally, we're experimenting with adding four data dimensions into our training, like the layout in Word documents and in PowerPoint slides to automatically generate one from the other, or customer activity sequences in Dynamics to identify potential issues and opportunities.

We think of our large scale AI models and the systems that enable their use as a new software platform, one that enables faster innovation and better collaboration.

We believe that this platform needs to benefit everyone. And so, I'm happy to announce that in the coming weeks, we'll open source our Turing language models, together with recipes to use them in Azure Machine Learning. We're really excited to see what developers build with this.

Back to you, Kevin.

KEVIN SCOTT: Thank you, Luis. The growth and development of these massive ML models is going to represent such a huge step forward in how we approach and use technology. As we just discussed, our primary constraint to growing these models further is the availability of affordable compute. With Moore's Law slowing, we have to find other mechanisms to efficiently scale compute to meet the needs of modern AI.

We're looking at the complete system for machine learning and AI, from data center power and cooling to networks to competing architectures, all the way up to operating systems, programming languages and frameworks. As a computer scientist and a systems researcher, this is honestly the most exciting period that I've experienced in the evolution of computing infrastructure. And all of this unprecedented innovation is converging in the service of building amazing new AI supercomputing technology.

We've been building increasingly powerful systems for AI for a while. Late last year, we completed work on our first AI supercomputer and handed it over to scientists and engineers to start using in their work. The site, Top500.org, keeps the list of the world's most powerful supercomputers.

And I'm excited to announce today that our cloud-hosted supercomputer would make that list. By our measures, it's one of the top five largest supercomputers in the world. It uses 285,000 CPU cores and 10,000 GPUs connected within extremely fast network. This machine is purpose built for the kinds of massive distributed models that Luis talked about earlier. That gives this supercomputer all the benefits of a dedicated appliance paired with the benefits of a robust modern cloud infrastructure.

One of the most exciting things about this machine is that it's cloud hosted. It's Azure, so because of the power of the cloud, we were able to develop and deploy this in just six months. And because it's Azure, it benefits from our carbon-neutral commitments as a company.

Let me step back for a second because I'm sure there's at least a few of you out there right now who are saying, well, it's pretty cool, but I'm not exactly in the market for one of the world's largest supercomputers just this minute.

The work that we've done to get here has a clear benefit to any Azure customer. Our advances in large scale clusters, industry leading network design and a software stack to control it all are applicable to the Azure product roadmap.

A good analogy here might be the way automotive technology is pioneered in high-end racing before making its way into the cars that you and I drive every day. Hybrid power trains, all-

wheel drive, APS and more; except in this situation, developers get the benefit of the output as well.

Through the large ML models hosted on this supercomputing infrastructure, this new kind of computing power is going to drive amazing benefits for the developer community, empowering a previously unbelievable AI software platform that will accelerate your projects, large and small.

Just like the ubiquity of sensors and smartphones, multi-touch, location, high-quality cameras, and accelerometers enabled an entirely new set of experiences, the output of this work is going to give developers a platform to build new products and services. Having access to ubiquitous, high-quality language translation, understanding, summarization and reasoning is going to help supercharge developers. It will increase not only your own productivity, but also the capabilities of the products and services you develop.

The machine we're talking about today was built for our friends and partners at OpenAI who are going to use it to power their industry-leading AI research. A few days ago, I had the pleasure of talking about how these amazing advances in AI supercomputing will translate to real-world impact for developers and organizations with a very special guest.

My guest today is Sam Altman. Sam's one of the most successful entrepreneurs and investors in the industry, and we've had the good fortune of working with Sam through OpenAI, where he's CEO. I'm super happy to you with us here today, Sam.

SAM ALTMAN: Thanks for having me.

KEVIN SCOTT: We're talking today about AI supercomputers and these gigantic models that we've been working on, and in a very real sense, you all are at the very, very frontier. You are doing the most ambitious work in AI today. And so, I just wanted to get your perspective about what it is that you think we're going to be able to do with these really large models, and in particular how they're going to affect developers and how they do their work, and what they're able to build.

SAM ALTMAN: It's a great question and it's exciting for us because we view ourselves as an AI development and deployment company. We actually want to build these large-scale systems and see how far we can push it. And as we do more and more advanced research and scale it up into bigger and bigger systems, we begin to make this whole new wave of tools and systems that can do things that were in the realm of science fiction only a few years ago.

People have been thinking for a long time about computers that can understand the world and sort of do something like thinking. But now that we have those systems beginning to come to fruition, I think what we're going to see from developers, the new products and services that can be imagined and created are going to be incredible. I think it's like a fundamental new piece of computing infrastructure.

KEVIN SCOTT: Yeah, and that's one of the things that you and I have chatted about a lot. We have finally gotten to the point with these very big models where transfer learning, the ability to

train a model for once and then use it and a bunch of different applications or scenarios, that's finally working with these big models.

And so, it's really exciting. You can start thinking about the models themselves as a platform. And so, I'd love for you to chat a little bit about the incredible breadth of things that you all have been doing. It's not just natural language, right?

SAM ALTMAN: No. We're interested in trying to understand all the data in the world, so language, images, audio and more. But really, it's all data, right? It's all just, you know, you get input about the world and you try to understand the fundamental patterns and concepts of what's going on there, and then use that to build systems that can do useful things for people.

And the fact that the same technology and these same systems, as you mentioned, can solve this very broad array of problems and understand different things in different ways, that's really exciting to us. That's been the promise of these more generalized systems that can do a broad variety of tasks for a long time. And as we work with the supercomputer to scale up these models, we keep finding new tasks that the models are capable of.

KEVIN SCOTT: Yeah. And so, I know you've got a demo to show us today, so I'm going to be tremendously excited. I think this might be the first time that I'm seeing this. This is an application of what these really, really big models can do that might be super interesting for developers.

SAM ALTMAN: Yeah. We've talked about how these massive language models trained on supercomputers can do all these different kinds of natural language tasks, but the models actually go much further. When we fine tune them on specific data, we find that they can do things that we didn't expect. And one thing we were curious about was what we could do with code generation. Could we help developers write code?

And so, we used the Microsoft supercomputer, our generative text models, and we fine-tuned it on thousands of opensource GitHub repositories. And we'll roll the video and show you what happened.

I'm going to show what an opening ended language model can do when applied to code generation. The model I'm using today was trained on code from thousands of opensource GitHub repositories using the same unsupervised techniques as our GPT models. Let's try it out.

I'll start with a pretty basic Python task. I want to write a function to check whether or not a string is a palindrome. I'll start by writing the function signature and a quick comment string.

And now, I'll ask the model to generate the code that it thinks should come next. Great. The model got it right.

But that was a pretty simple example. If I search for it "is palindrome on stack overflow," I'm sure I'd find something similar, so let's step it up a bit.

This time, I'll write a function that definitely wasn't in the training set. That's a trend list indices for elements that are palindromes, and at least seven characters. And let's see what code the model generates.

Great. That looks right, and it even used the "is palindrome" function from above. This time, we can also rest assured that it didn't just copy some snippet from the training set. The model came up with a unique solution to a unique requirement. Let's step it up again.

This time, I'll write an order class composed of items. I've pasted in some code defining the item class and the properties of my order class.

Now let's write some methods for the order class. Let's start by bringing a method to compute the total price. But let's add a twist. Let's apply a discount to items that are palindromes. Let's see what the model suggests.

It helpfully writes a comment string for us. Ah, but that's not quite right. I only wanted the discount applied to palindromes, not to the whole order. Let's edit the comment to clarify what I wanted. Compute the total price and return it. Apply discount to items whose names are palindromes.

Let's try again. So far, so good. Great. That's much better.

Actually, I wanted the default discount to be 20% off, not 80% off. But that's an easy enough fix. There we go. Let's round it out by printing a receipt for the order. Print the total price and the price of each item.

Here we go. Perfect. That's just what we wanted. And it even used the compute total price method that the model and I wrote together.

Beyond helping implement functions, we've also seen the model excel at other development tasks like writing unit tests. All in all, this model can generate useful and context-aware code suggestions that would make any developer more productive. Tooling built on these kinds of models will allow you to spend less time on repetitive, time consuming coding tasks and focus more on the creative aspects of writing software.

KEVIN SCOTT: Yeah, it's truly incredible what these models are capable of doing, and it's clear that they're going to assist us in being more productive across all of our work, from reasoning over documents to writing emails to building software.

Thanks so much for your time, Sam. It's been a real pleasure.

SAM ALTMAN: Thank you, Kevin.

KEVIN SCOTT: We're actually already using AI powered tooling at Microsoft today. Visual Studio and Intelicode's AI assistance saves developers time by learning from the input of the broader coding community, and research teams are working to take these huge models and distill

them down to a size where they're practically usable in Visual Studio. You can check out the Intelicode talk later this afternoon to learn more.

Hopefully, you can imagine the possibilities once we have this kind of massive power supporting AI platforms for you to build and innovate on. Another super important place where AI will be used as a platform in the future will be at the intelligent edge, building a bridge between the physical and digital worlds.

Let's join Lila Tretikov to share some of the ways this has been developing in the recent past and what it will look like in the future.

LILA TRETIKOV (CVP of Technology): Thanks, Kevin. It is impressive to see our extra-large AI supercomputers and what they enable our developers to do. But we have something equally cool to show you in the next nine minutes. This time, our AI will be extra small and extra distributed. It will run across numerous smart, interconnected devices, enabling you to program them as one intelligent system. To enable this, you'll see our upcoming programming and deployment models, Dapr and ORM (ph), networked by 5G APIs from AT&T, supported by ONNX-based AI models that seem to separate between the edge and the cloud.

And finally, say hello to my friend Spot here, a Boston Dynamics robot. Spot will literally run AI on the edge. Thank you, Spot. Good boy.

These past few months have rapidly accelerated how important using AI on intelligent edge is across a host of scenarios. Frictionless access, hands-free computing and personal protective equipment became required nearly overnight.

At Cardinal Teen Hospital in Taiwan, workers helped protect patients and staff from the spread of COVID-19 by deploying computer vision at hospital access points. In France, doctors are using HoloLens to collaborate and share information with their patients.

In education, for the first time ever, instead of working together on campus, all first-year medical students from Case Western Reserve University School of Medicine practiced anatomy at home.

These examples build on intelligence systems work we have previously done with Unilever, Shell, Merck, and many others. We are eye-witnessing another evolution. All things around us are becoming smart things. You will program and manage them as one AI-powered, cloud-to-edge intelligent system. With such a system, we can reimagine and reprogram our physical world to help manage our work, automate redundant tasks, see, hear and collaborate across thousands of miles, as well as keep us safer when we can't be there in person.

We want to show you how you will be able to build such systems yourselves. Let's take a look at a prototype application showing some of our forward-looking, AI-on-the-edge features.

This app uses an infrared camera and an AI model trained to detect license plates to help with curbside food pick up from my favorite hypothetical takeout pizza restaurant. You see the

application detecting the license plate of a person picking up their to-go order. Notice that if we lose network connectivity, the license plate detection fails.

We can't have that. We need zero business downtime, so we're going to fix this by allowing the application to run on the edge when disconnected and on the cloud when the network becomes available.

This is our architecture. Our camera is already connecting to an edge device, which right now use an ONNX model in the cloud. We're going to change these edge microcircuits to also leverage a local ONNX model when disconnected.

We could also configure this system to run exclusively on the edge, never sending anything to the cloud at all for privacy across regions. We're using the Dapr, Distributed Application Runtime, which offers this simple microservices-based programming model.

This is our solution in Visual Studio Code. This wrapper of a 5G network API from AT&T allows us to query 5G network conditions, and our code will leverage the 5G API to allow us to switch between calling cloud and edge based on connectivity. We'll create a route to handle the network disconnecting, and we ride the logic to use the local API model.

For deployment, our Dapr microservice is already configured with Azure Arc to run local and in the cloud. Azure Arc uses the deployment specs to automatically deploy to our edge Kubernetes cluster when we check in our changes. Now that we've made our changes and Azure Arc has deployed them, that's your test.

As you can see, the application works with zero downtime, with only slight decrease in confidence levels when running just on the edge. And as the network is restored, the AI adjusts back to cloud based mode.

This isn't all, though. We've detected our customer coming to the restaurant. Now we need to deliver their food. To deliver the takeout, we'd like to show how we're integrating our work with autonomous robotics, in this case, with the Boston Dynamics team and Spot, who you've met earlier. Gina is going to help us show this work.

GINA TRIOLO (Software Engineer, AI & Robotics): Thanks, Lila. I'm going to show how our intelligent edge solution can leverage AI to have a Boston Dynamics Spot robot look around, use our edge AI to find the customer's car and carry their takeout to them. We've mounted an edge device on this Spot robot to allow us to control the robot autonomously and deploy our AI model using ONNX runtime.

Here, I'm showing the code where we use the ONNX AI model on the edge device that will help us recognize the license plate using images captured from the robot's cameras. We will use this code to leverage the robot's camera perspective and control the robot's position to navigate to the car with our license plate.

Lastly, we've built this using custom commands, leveraging speech and language understanding in Azure Cognitive Services to activate Spot to deliver the food. Putting everything together, we'll ask Spot to deliver the order.

Hey, Spot. Deliver ordered to Lila.

Note that in the future, Spot could potentially pick up and load the order into the car. Lila will tell us more about the complexity and training this kind of AI.

Good job, spot. Over to you, Lila.

LILA TRETIKOV: Yum, pizza. Thank you, Gina. We just used Visual Studio Code, the AT&T 5G network APIs, ONNX for AI on edge and cloud, Azure Arc, Dapr and ORM, and the Boston Dynamics robot to build an app that intelligently crosses the cloud and edge.

As Gina mentioned, to deliver the groceries would require precise control. We need a high degree of confidence for navigation path, traversing stairs and controlling speed on approach. To build such a system, you need reinforcement learning paired with expert guided teaching in the same way you would learn how the master a sport. This training requires hundreds of thousands, if not millions, of repeated attempts to balance controls and forces that act upon our system.

I would like to announce a public preview of Project Bonzai, the machine teaching component of our economist systems platform, which enables training such models. Bonzai is not for robotics only, though. Our customers have used it to optimize and bring up autonomy to a wide range of scenarios, from complex process controls in the chemical industry to machine tuning, calibration and motion control in manufacturing, and even making apparel goods and food processing.

The next wave of computing will enable us to manage the world handsfree, remote, even in the most dangerous and difficult conditions. Smart things will be our remote hands and eyes, and we are building intelligence systems to empower you to program them to solve your most critical needs. These critical systems must protect our security and privacy.

At Microsoft, we believe it is our responsibility in partnership with developers like yourselves to innovate transparently, respectfully and responsibly. We are committed to enabling developers around the globe to have a lasting impact on society through our ethical AI principles and AI for Good initiatives. With Microsoft providing AI platforms, safeguards and guidelines, we really look forward to seeing the intelligence systems you build for your customers. Thank you.

KEVIN SCOTT: Thank you, Lila. I really hope that what you've seen today excites and inspires you. As I said before, that's really why we're here, to provide great tools and platforms for the people who are tirelessly contributing so much passion, imagination and ingenuity at the frontlines of the technology revolution every day.

We saw such amazing examples of how that has taken place across so many different facets of the digital and the physical world, and how the two can be connected to help us tackle some of our most daunting global challenges, with the explosion of machine learning models and AI

platforms driven by superpowered infrastructure, how anybody will soon be able to harness computing power that was unthinkable even a few years ago to create and innovate at a scale that we've never seen.

I really encourage all of you to put your acute thought and attention towards the trends in computing that we discussed today. I truly believe that what happens in the coming years when we see the maturity of things like AI at scale, unsupervised machine learning and reinforcement learning will be a fundamental alteration to the fabric of what technology means to us and what it can achieve. And it's going to impact every aspect of your work as developers sooner rather than later.

Everything that we talked about here is real technology that we have today. This is stuff that we've been thinking deeply about for a long time. We have the resources and the charter to do so. But in just a few years, every developer at every level will have to think the same way about AI, about the edge, about how to apply machine learning.

In some ways, you could even think about the intersection we're seeing of these developments between AI and the sciences as the new calculus. In the 18th century, the introduction of calculus gave us the field of physics and helped us to better model and understand the world around us. Increasingly, neural networks are playing that role today in the sciences.

This has particularly taken place in biology. Think of some of the efforts we saw earlier to use email to rapidly understand and cope with COVID-19. That won't be the last existential crisis we face as a species, and these new tools are going to help us bear the weight of many elements of an uncertain future with more solid footing.

There are a couple of things that need to happen for any of this to have a net positive impact on our society. Firstly, all of it will have to be available to everyone. We must democratize access to these tools so that anyone can use them. Much of the onus to do that is on us, the platform builders.

But all of you share in this responsibility as well. Whether you're building a mobile app, creating new tools for precision agriculture, personalized medicine or commerce, the benefits that flow from the power of scaled AI will need to reach as many people as possible. The barriers to entry must be lowered.

Secondly, we must take the care to develop this new platform thoughtfully, accountably and ethically. Microsoft is already committed to a set of responsible AI principles that are core to all our AI development efforts. But all of us in the technology ecosystem need to be vigilant that the AI products and applications we create are inclusive, equitable and human centered.

My ask of you today is not just to help us build a future on top of these new AI platforms. It's to do your part to help everybody on the planet share in the empowerment that can come from reaching a new frontier in our shared technology journey. That endpoint is our mission at Microsoft, but we can't achieve it without all of you.

Thank you all so much for tuning in. Be safe, be well, and please never stop building and dreaming beyond the boundaries of what's possible.

END