Why the Future of Machine Learning is Tiny

Pete Warden, June 11, 2018

https://petewarden.com/2018/06/11/why-the-future-of-machine-learning-is-tiny/

Photo by Kevin Steinhardt

When Azeem asked me to give a talk at CogX, he asked me to focus on just a single point that I wanted the audience to take away. A few years ago my priority would have been convincing people that deep learning was a real revolution, not a fad, but there have been enough examples of shipping products that that question seems answered. I knew this was true before most people not because I'm any kind of prophet with deep insights, but because I'd had a chance to spend a lot of time running hands-on experiments with the technology myself. I could be confident of the value of deep learning because I had seen with my own eyes how effective it was across a whole range of applications, and knew that the only barrier to seeing it deployed more widely was how long it takes to get from research to deployment.

Instead I chose to speak about another trend that I am just as certain about, and will have just as much impact, but which isn't nearly as well known. I'm convinced that machine learning can run on tiny, low-power chips, and that this combination will solve a massive number of problems we have no solutions for right now. That's what I'll be talking about at CogX, and in this post I'll explain more about why I'm so sure.

Tiny Computers are Already Cheap and Everywhere

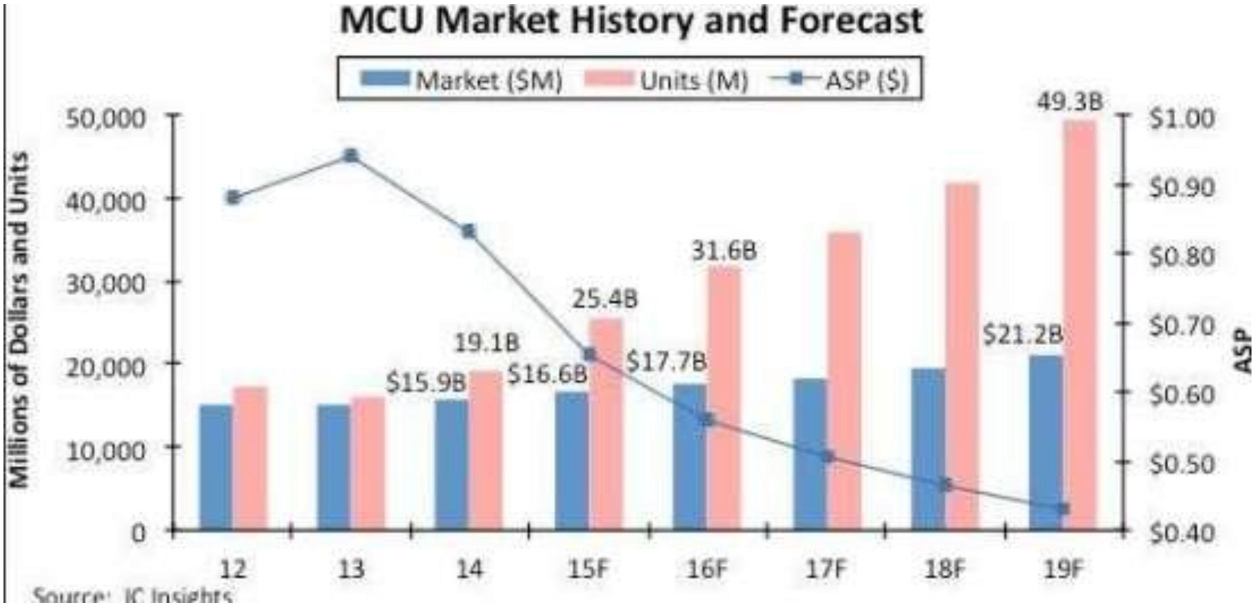

Chart from embedded.com

The market is so fragmented that it's hard to get precise numbers, but the best estimates are that over 40 billion microcontrollers will be sold this year, and given the persistence of the products they're in, there's likely to be hundreds of billions of them in service. Microcontrollers (or MCUs) are packages containing a small CPU with possibly just a few kilobytes of RAM, and are embedded in consumer, medical, automotive and industrial devices. They are designed to use very small amounts of energy, and to be cheap enough to include in almost any object that's sold, with average prices expected to dip below 50 cents this year.

They don't get much attention because they're often used to replace functionality that older electro-mechanical systems could do, in cars, washing machines, or remote controls. The logic for controlling the devices is almost unchanged from the analog circuits and relays that used to be used, except possibly with a few tweaks like programmable remote control buttons or windshield wipers that vary their speed with rain intensity. The biggest benefit for the manufacturer is that standard controllers can be programmed with software rather than requiring custom electronics for each task, so they make the manufacturing process cheaper and easier.

Energy is the Limiting Factor

Any device that requires mains electricity faces a lot of barriers. It's restricted to places with wiring, and even where it's available, it may be tough for practical reasons to plug something new in, for example on a factory floor or in an operating theatre. Putting something high up in the corner of a room means running a cord or figuring out alternatives like power-over-ethernet. The electronics required to convert mains voltage to a range circuitry can use is expensive and wastes energy. Even portable devices like phones or laptops require frequent docking.

The holy grail for almost any smart product is for it to be deployable anywhere, and require no maintenance like docking or battery replacement. The biggest barrier to achieving this is how much energy most electronic systems use. Here are some rough numbers for common components based on figures from Smartphone Energy Consumption (see my old post here for more detail):

- A display might use 400 milliwatts.
- Active cell radio might use 800 milliwatts.
- Bluetooth might use 100 milliwatts.
- Accelerometer is 21 milliwatts.
- Gyroscope is 130 milliwatts.
- GPS is 176 milliwatts.

A microcontroller itself might only use a milliwatt or even less, but you can see that peripherals can easily require much more. A coin battery might have 2,500 Joules of energy to offer, so even something drawing at one milliwatt will only last about a month. Of course most current products use duty cycling and sleeping to avoid being constantly on, but you can see what a tight budget there is even then.

CPUs and Sensors Use Almost No Power, Radios and Displays Use Lots

The overall thing to take away from these figures is that processors and sensors can scale their power usage down to microwatt ranges (for example Qualcomm's Glance vision chip, even energy-harvesting CCDs, or microphones that consume just hundreds of microwatts) but displays and especially radios are constrained to much higher consumption, with even low-power wifi and bluetooth using tens of milliwatts when active. The physics of moving data around just seems to require a lot of energy. There seems to be a rule that the energy an operation takes is proportional to how far you have to send the bits. CPUs and sensors send bits a few millimeters, and is cheap, radio sends them meters or more and is expensive. I don't see this relationship fundamentally changing, even as technology improves overall. In fact, I expect the relative gap between the cost of compute and radio to get even wider, because I see more opportunities to reduce computing power usage.

We Capture Much More Sensor Data Than We Use

A few years ago I talked to some engineers working on micro-satellites capturing imagery. Their problem was that they were essentially using phone cameras, which are capable of capturing HD video, but they only had a small amount of memory on the satellite to store the results, and only a limited amount of bandwidth every few hours to download to the base stations on Earth. I realized that we face the same problem almost everywhere we deploy sensors. Even in-home cameras are limited by the bandwidth of wifi and broadband connections. My favorite example of this was a friend whose December ISP usage was dramatically higher than the rest of the year, and when he drilled down it was because his blinking Christmas lights caused the video stream compression ratio to drop dramatically, since so many more frames had differences!

There are many more examples of this, all the accelerometers on our wearables and phones are only used to detect events that might wake up the device or for basic step counting, with all the possibilities of more sophisticated activity detection untouched.

What This All Means For Machine Learning

If you accept all of the points above, then it's obvious there's a massive untapped market waiting to be unlocked with the right technology. We need something that works on cheap microcontrollers, that uses very little energy, that relies on compute not radio, and that can turn all our wasted sensor data into something useful. This is the gap that machine learning, and specifically deep learning, fills.

Deep Learning is Compute-Bound and Runs Well on Existing MCUs

One of my favorite parts of working on deep learning implementations is that they're almost always compute-bound. This is important because almost all of the other applications I've worked on have been limited by how fast large amounts of memory can be accessed, usually in unpredictable patterns. By contrast, most of the time for neural networks is spent multiplying large matrices together, where the same numbers are used repeatedly in different combinations. This means that the CPU spends most of its time doing the arithmetic to multiply two cached numbers together, and much less time fetching new values from memory.

This is important because fetching values from DRAM can easily use a thousand times more energy than doing an arithmetic operation. This seems to be another example of the distance/energy relationship, since DRAM is physically further away than registers. The comparatively low memory requirements (just tens or hundreds of kilobytes) also mean that lower-power SRAM or flash can be used for storage. This makes deep learning applications well-suited for microcontrollers, especially when eight-bit calculations are used instead of float, since MCUs often already have DSP-like instructions that are a good fit. This idea isn't particularly new, both Apple and Google run always-on networks for voice recognition on these kind of chips, but not many people in either the ML or embedded world seem to realize how well deep learning and MCUs match.

Deep Learning Can Be Very Energy-Efficient

I spend a lot of time thinking about picojoules per op. This is a metric for how much energy a single arithmetic operation on a CPU consumes, and it's useful because if I know how many operations a given neural network takes to run once, I can get a rough estimate for how much power it will consume. For example, the MobileNetV2 image classification network takes 22 million ops (each multiply-add is two ops) in its smallest configuration. If I know that a particular system takes 5 picojoules to execute a single op, then it will take (5 picojoules * 22,000,000) = 110 microjoules of energy to execute. If we're analyzing one frame per second, then that's only 110 microwatts, which a coin battery could sustain continuously for nearly a year. These numbers are well within what's possible with DSPs available now, and I'm hopeful we'll see the efficiency continue to increase. That means that the energy cost of running existing neural networks on current hardware is already well within the budget of an always-on battery-powered device, and it's likely to improve even more as both neural network model architectures and hardware improve.

Deep Learning Makes Sense of Sensor Data

In the last few years its suddenly become possible to take noisy signals like images, audio, or accelerometers and extract meaning from them, by using neural networks. Because we can run these networks on microcontrollers, and sensors themselves use little power, it becomes possible to interpret much more of the sensor data we're currently ignoring. For example, I want to see almost every device have a simple voice interface. By understanding a small vocabulary, and maybe using an image sensor to do gaze detection, we should be able to control almost anything in our environment without needing to reach it to press a button or use a phone app. I want to see a voice interface component that's less than fifty cents that runs on a coin battery for a year, and I believe it's very possible with the technology we have right now.

As another example, I'd love to have a tiny battery-powered image sensor that I could program to look out for things like particular crop pests or weeds, and send an alert when one was spotted. These could be scattered around fields and guide interventions like weeding or pesticides in a much more environmentally friendly way.

One of the industrial examples that stuck with me was a factory operator's description of "Hans". He's a long-time engineer that every morning walks along the row of machines, places a hand on each of them, listens, and then tells the foreman which will have to be taken offline for servicing, all based on experience and intuition. Every factory has one, but many are starting to face retirement. If you could stick a battery-powered accelerometer and microphone to every machine (a "Cyber-Hans") that would learn usual operation and signal if there was an anomaly, you might be able to catch issues before they became real problems.

I probably have a hundred other products I could dream up, but if I'm honest what I'm most excited about is that I don't know how these new devices will be used, just that the technological imperative behind them is so compelling that they'll be built and whole new applications I can't imagine will emerge. For me it feels a lot like being a kid in the Eighties when the first home computers emerged. I had no idea what they would become, and most people at the time used them for games or storing address books, but there were so many possibilities I knew whole new worlds would emerge.

The Takeaway

The only reason to have an in-person meeting instead sending around a document is to communicate the emotion behind the information. What I want to share with the CogX audience is my excitement and conviction about the future of ML on tiny devices, and while a blog post is a poor substitute for real presence, I hope I’ve got across some of that here. I don’t know the details of what the future will bring, but I know ML on tiny, cheap battery powered chips is coming and will open the door for some amazing new applications!