IOTech

My 2025 Predictions for the Edge
by Brad Corrion, CTO, IOTech Systems

In January of this year, I joined IOTech Systems as CTO, stepping into the shoes of the esteemed Jim White, who rumor has it is enjoying a relaxing retirement. Jim set the tradition of making annual predictions for the coming year in Edge Computing, and as my first year comes to a close it is my turn to look into the crystal ball and record some predictions to paper. Here are Jim's 2024 predictions, which I did not re-read until after my draft was complete, so it was pleasing to see the continuity of thought expressed in both sets of predictions.

As a first time prognosticator, I suppose the purpose of these predictions is that it opens up a view of the world from our vantage point: assisting customers deploying edge computing with our open-source based middleware platform. This year, rather than covering a few different trends, I will cover the impacts, both obvious and not so obvious, from one major trend. So for a company that doesn't peddle generative AI technologies or Large Language Models (LLMs), their place in the zeitgeist makes it worth better understanding how they impact the Edge, Edge Computing and IoT. I also act as a Venture Partner for Monsoon Venture Fund, an Arizona-focused early stage VC firm, which means we see a healthy pipeline of startups incorporating LLMs in all sorts of use cases. While it feels like these technologies just arrived, it should be noted that the largest models are already edging towards commodity, which means the race is now on for quality, price and domain differentiation (I highly recommend reviewing this presentation from Benedict Evans). In short, it would be a dereliction of duty to not cover this trend.

What makes this a fun exercise is that considering the role of an LLM creates an opportunity to re-evaluate what is the work we want to do, and to re-consider the assumptions that have shaped the purpose of organizational charts and fundamental investment criteria for decades.

Prediction 1: LLMs and edge compute will be aided by human-centric operations - aka "Personal Interactions as Code"

It's in the name! The use of natural language for input and output makes these tools most suitable for human interactions and aligned with that theme is that some of the most expensive work to complete are human-facing: addressing patients in health care and addressing customers (and employees) in retail/service/public settings. And note, these areas are traditionally expensive because of the hiring and maintaining skilled staffing to serve broad populations at scale. Startups and disruptive ideas are held back by the organizational momentum and sheer logistics of assembling teams - but LLMs teamed with natural language processing and a communications platform are delivering "personal interactions as code" — significantly reducing the cost to deploy, at scale, effective engagement services.

One area to observe: Tech-centric retailers are already deploying LLMs at the edge. Home Depot has built an LLM to help employees answer customer questions, trained on domain text, corporate knowledge bases and previous interactions. Other retailers are right there too. If you aren't at least playing with these tools you are already behind.

For those of us considering industrial controls and related embedded computing scenarios, not only are we not yet prepared for natural language use cases, but have a ready made list of excuses to avoid LLMs: our latencies and frequencies don't accommodate LLMs as we see them today, the embedded operational domains utilize carefully curated structured data, and that analytical systems need to provide consistently correct results. These factors play right into the two weaknesses of LLMs (for now), but don't let your domain blinders deceive you. There are still excellent utilizations of LLMs for our domain.

Prediction 1: LLMs and edge compute will be aided by human-centric operations - aka “Personal Interactions as Code”

Edge Computing is an umbrella term that conveys the use of cloud-native technologies to deploy modern software applications in geographically distributed locations — close to the source of data that is being acted upon. The entire edge computing domain is based on transitioning established embedded systems teams to support new technologies such as AI, micro services, event-driven architectures, innovation, and so on while maintaining uptime and reliability standards for software deployed in hard to access locations.

For the entire job scope, the ideal candidate for the job doesn’t exist, and so we will be satisfied to take an embedded engineer and teach them about cloud technologies, or take a cloud-native graduate and teach them about the embedded domain. Either way we will be stretching small teams in challenging ways to cover all of the jobs to be done.

Installing, configuring, troubleshooting and maintaining edge compute systems carries a non-trivial workload, and the technical debts are often incurred after the innovation team has moved on to the next project. Fleet management for edge computing presents the great intersection of challenges and opportunities: 1) mostly repetitive tasks, 2) spread across a large number of distributed compute nodes, 3) with tiny but impactful differences (such as site-specific IoT configurations). Lately I’ve been asking teammates to embrace nonlinear management capabilities: simply put, no-one has the time to manually touch 10,000 fleet systems, even if it is just to add a node into a queue for a software update. This is getting pretty close to the sweet spot for an LLM. But rather than have a software system manage 10,000 interactions with a human, flip the tables around: let’s enable one human to manage 10,000 interactions with edge computers. Well designed platforms will expose easily consumable data that can be mapped back to a generated series of scripts or queued actions. “Update the container tag version to 3.2 across all deployed instances of Edge Central”, “Create a report identifying how many nodes are using unsafe versions of OpenSSH”, or “parse these log files and explain why the device discovery is failing at site 234.” The road has already been paved to use LLMs to support Kubernetes deployments - and the space will just keep getting more exciting.

Prediction 2: “Little AI” will continue to be a driving force of data consumption for Edge Computing

While LLMs (“Big AI”) continue to dominate media and technical spheres due to the sheer volume of investments and infrastructure required to create a foundation model, let us not forget “simple” AI, Deep Learning, aka Artificial Narrow Intelligence aka “Little AI”, which is very relevant to the Edge Computing domain.

Little AI dominated our Edge conversations for years, but adoption momentum has been tempered by the recognition of the investments required to build repeatable flows that collect data, clean it, train a model from it, validate it, and finally evaluated by said models deployed to thousands of sites (not to mention the teams and specialties that had to be acquired to make this all possible). This is a multi-disciplinary process that must be honed and practiced by teams working with their partners vendors (such as us), and across this multi-year span we are now seeing practical applications entering deployment.

Prediction 2: "Little AI" will continue to be a driving force of data consumption for Edge Computing

This is where we traditionally come in: customers want to move data to points where the data is refined into higher value forms, and our software builds the roads and trucks that enable the data movement. Along the way we assist with the complex problems like connecting to data sources, collecting that data in volume and normalizing it into standard formats for collection, training and analytical purposes. For years we have been helping organizations come to terms with the data, pathways and refining what they need and the good news is that our customers succeeding in these areas. Some are still amassing hordes of training data, and others are proving the successes of Edge AI such as the growing field of carbon conservation in the production of chilled water for offices and factories. These are very new use cases, that use very old forms of data, but excitingly are sampled at frequencies unheard of in building automation circles, fused with data from other sensors, IoT devices and sources that never mattered in previous eras. The resulting system should be repeatable, maintainable and functional for years, and the tools are modular and well documented.

The interesting contrast to me is that while Little AI requires a practiced approach to data gathering and hygiene, LLMs garner so much attention because it appears so simple: just hoover up as much of the web as you can, figure out meaningful relationships, and ask it questions that make very compelling demonstrations. They could basically ignore the need for structured data albeit with extra investments on the results side to ensure safety in spite of the latent biases of the training data (pick your poison: or you clean the data going in or you clean the data coming out). At the end of the day your teams still need repeatability, sustainability, and economy that are not yet proven for Big AI as they are understood for Little AI.

Another distinction is that while it is much easier to stand on the shoulders of giants with Big AI to get some immediate encouraging results, standing on the shoulders of the Little AI pioneers just shows you how much work is still in front of you. This distinction is important because it takes more effort to resource and fund the long term needs of a Little AI project vs the easy sales pitch to management with such early, promising demonstrations, regardless of the unknown unknowns that lie ahead.

Prediction 3: LLMs are changing the expectations for software solutions

This is more succinct prediction, but don't let the brevity dissuade of its importance. It is a rephrasing of something I learned working with local founders: users now expect software to deliver results, not just enable access to the results. In other words, don't give me a spreadsheet, give me the template, or better yet, give me the result calculated from a populated template, or better still: just give me the result. Technology can now hide complexity, so please hide it from me and my teams.

This shift in expectation is being reinforced every time a generative AI or Big AI demo is given to an audience. Expectations go up, patience for complexity goes down.

The resulting pressure on technology suppliers will be an expectation for solutions that do more with fewer obligations.

CTOs and CIOs have long used a People/Process/Things matrix to manage their relative investments in technology required to run their organization. The perceptions of Big AI are upsetting the balance of that matrix. Normally deciding the weighting of each is burdened by relative need (weaker people need stronger processes, weaker technology requires stronger people), but if the cost of technology is so cheap, and the technology can satisfactorily map in the processes, then it can effectively overweight the balance to "Things" — and fundamentally reducing the need to invest in People.

This will play out in surprising ways I am certain. A peer asked: when will we recognize the first VC-backed startup with a unicorn valuation and only one employee?

Prediction 4: Platforms appear to be at risk from the LLM onslaught, but instead they will be mutually beneficial for adoption at the edge

The traditional role of a platform is to provide a consistent discovery and utilization covering a set of related technologies. Consistency of style leads to efficiencies in adoption and understanding by fellow teammates and community members. Example, the WIN32 API for Windows application development, Posix for UNIX, tensor flow and PyTorch for consistent AI model environments, and EdgeX Foundry for IoT data collection, normalization and consumption. A good platform has multiple sides that reinforce utilization: encouraging different types of users to collaborate, producing a win-win for all participants (the more vendors build to a standard, the more users adopt it, a virtuous cycle).

Organizations recognize the value of platforms when they realize that they are creating large amounts of technical debt by supporting multiple deployments of competing yet largely similar technologies — a platform approach carries value if for no other reason than to reduce the number of technologies in deployment or to make it easier to stitch together value added investments being made by different teams across a company or even an ecosystem. When teams can forget the boring platform stuff, they can move on the fun, higher value work.

Prediction 4: Platforms appear to be at risk from the LLM onslaught, but instead they will be mutually beneficial for adoption at the edge

That's when the open source nature of a platform becomes important: if simply for an organization to insure their consolidated strategy against vendor lock-in, but more importantly to create a community of like minded users to cultivate the forward growth of the platform.

But how does the LLM change the face of the platform? Couldn't the embedded knowledge of an LLM allow developers to create vertical offerings, skipping right over the messy bits in the middle? Perhaps, but after a few iterations, they may have the same problem with technical debt. Or worse, the LLMs no longer generate results the way they did prior. My opinion is that they will grow stronger from each other:

A. The documentation and GitHub repositories of open source platforms means those platforms have already been ingested by LLMs: in some of the earliest versions of ChatGPT it could reasonably create new plugins for EdgeX Foundry device services, and we have seen success parsing vendor specifications to aid device provisioning.
B. The combination of defined, standard and well documented APIs means it is easy for LLM-based systems to access information and create integrated solutions.
C. It is preferable to accept the output of an LLM that generates some glue code around an established platform vs a vertical solution generated out of memory. IP rights, license models, repeatability, and other factors will favor inclusion of the platform.
D. Lastly, the more widely a platform is deployed (or is easily deployable), the more capabilities you offer for LLM-generated workflows, or even locally hosted LLM models, making for a great foundation to adopt LLM-based workflows across an organization.

Summary

My favorite definition of generative AI is summed up like this: "ChatGPT Is a Blurry JPEG of the Web." What is fun about this definition is that blurry JPEGs are still recognizable, and that most of us deal with daily work that is reasonably covered by even a poor assimilation of the necessary result (to wit: not all employees are superstars yet the corporate mission carries forward).

The question I will leave the reader with is whether these trends are converging to address the growing challenge of retiring experts?

As an example, in the building automation space, one of our customers said every building has a "building whisperer" — that person who knows how everything works, where the hidden switches are, what to do when the unlabeled red light is blinking on the control panel. The challenge is that just as the embedded technologies are now aging out, so are the experts that have nursed them along for years.

In the IoT world, a lot of these deployments are "set and forget" - installed, configured, and largely forgotten. Once their caregivers retire we are left with a serious skills and experience gap in the labor market. Can a joint utilization of modern edge computing frameworks combined with the embedded technical chops of an LLM, conspire to make up for the loss of tribal knowledge? We are investing that the answer is yes.

In truth each of these predictions is worth clicking deeper into in order to evaluate the underlying assumptions, and to better consider possible impacts. To that end, in 2025 we'll be launching a podcast or series of webinars to dig into these and other topics. Keep on the lookout for more announcements and hope to see you in the conversation.

For additional information on our products, contact us at
info@iotechsys.com

Visit our website **www.iotechsys.com**

IOTech Systems © 2024. All products and company names listed are trademarks or tradenames of their respective companies.
All specifications are subject to change without further notice.