

5 Cloud Intelligence Innovations

Recap of Microsoft Machine Learning & Data Science Summit and Ignite 2016

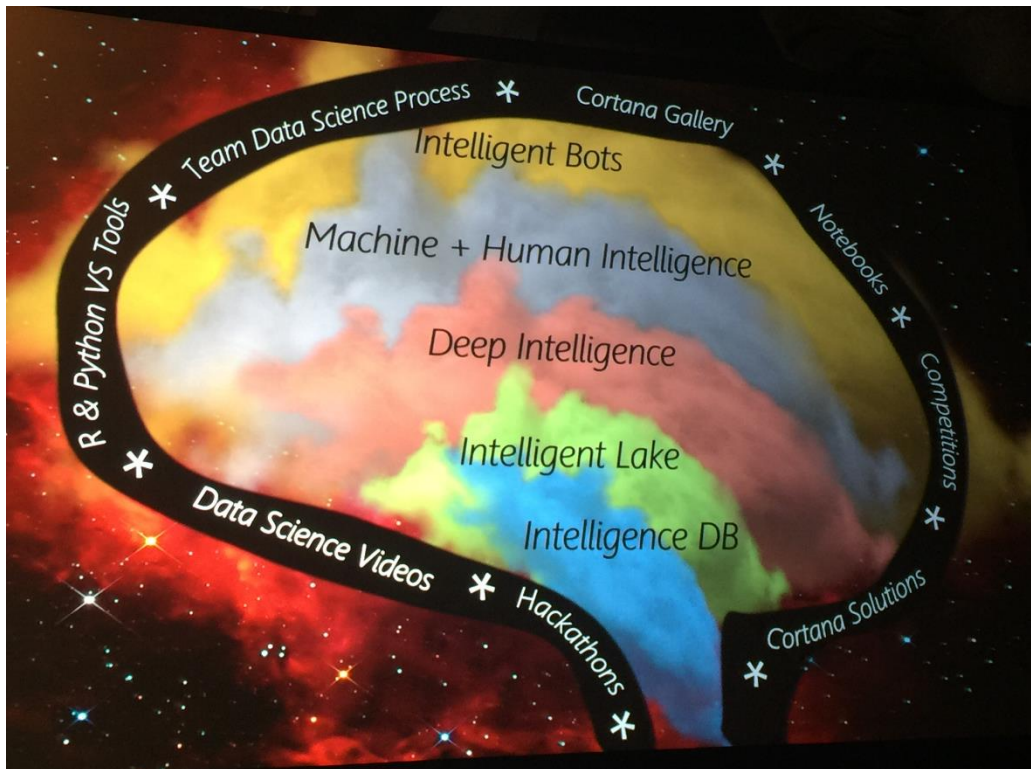
By Joseph Sirosh, Corporate Vice President, Microsoft Data Group

Last week, we held our *first-ever* [Microsoft Machine Learning & Data Science Summit](#) in Atlanta, a unique event tailored for machine learning developers, data scientists and big data engineers. I wanted to take some time this week to recap the key concepts, talks and customer scenarios that we presented there.



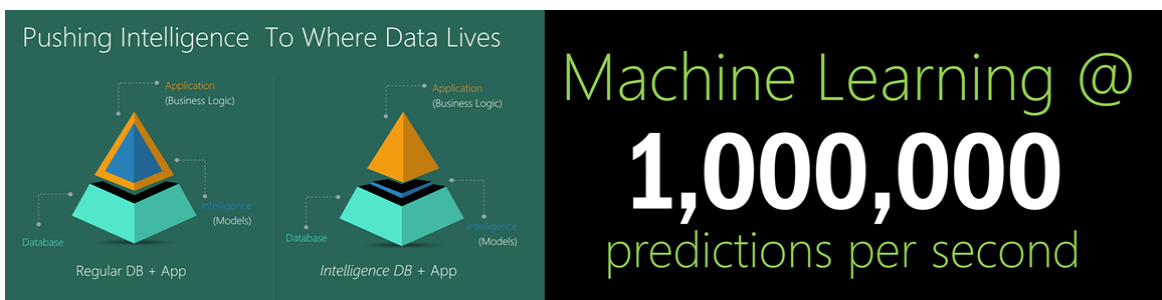
In my Summit keynote (you can [watch it online here](#)), I played the role of a tour guide, taking the audience on a journey through the core new patterns that are emerging as we bring advanced intelligence, the cloud, IoT and Big Data together. The cloud is truly becoming the “brain” for our connected planet. You’re not just running algorithms in the cloud, rather you’re connecting that with data from sensors from around the world. By bringing data into the cloud, you can integrate all of the information, apply ML and AI on top of that, and deliver apps that are continuously learning and evolving in the cloud. Consequently, the devices and applications connected to it are learning and becoming increasingly intelligent.

I created a loose analogy of the five patterns that I presented in my keynote to the anatomy of the human brain. These five patterns are really about ways to bring data and intelligence together in software services in the cloud to build intelligent applications. I drill down into each layer or pattern in the sections that follow.



Intelligence DB

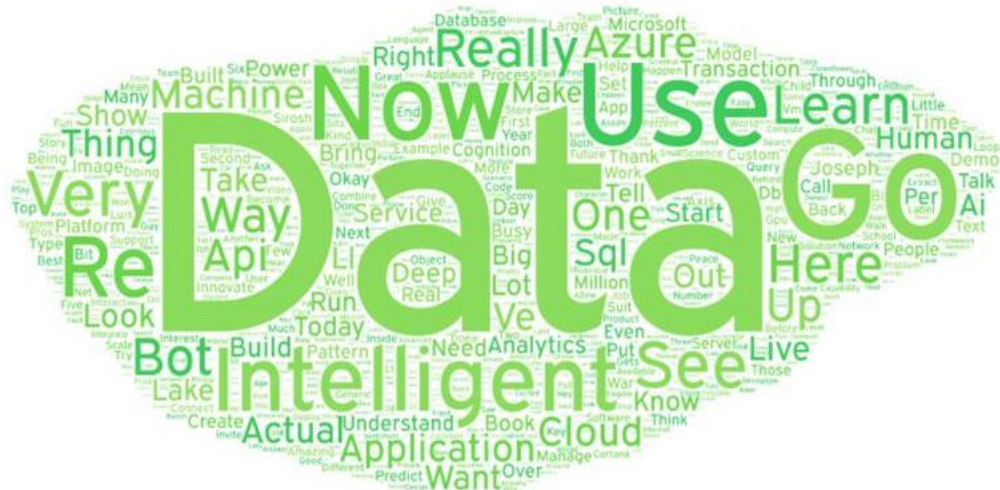
This is the pattern where intelligence lives with the data in the Database. Imagine a core transactional enterprise application built with a database like SQL Server. What if you could embed intelligence – i.e. advanced analytics algorithms and intelligent data transformations – *within the database itself* to make every transaction intelligent in real time? This is now possible for the first time with R and ML built into SQL Server 2016.



At the Summit, we showed off a fascinating demo titled **Machine Learning @ 1,000,000 predictions per second**, in which we showcased real-time predictive fraud detection and scoring in SQL Server. By combining the performance of SQL Server in-memory OLTP as well as in-memory columnstore with R and Machine Learning, apps can get remarkable performance in production, as well as the throughput, parallelism, security, reliability, compliance certifications and manageability of an industrial strength database engine. To put it simply, intelligence (i.e. models) become just like data, allowing models to be managed in the database, exploiting all the sophisticated capabilities of the database engine. The performance of models can be reported, their access can be controlled, and, moreover, because these models live in the database, they can be shared by multiple applications. No more reason for intelligence to be “locked up” in a particular app.

I invited two customers to showcase this pattern: **PROS**, which uses ML for revenue management in a SaaS app on Azure, and **Jack Henry & Associates**, who use it in loan charge-off prediction.

One of PROS' customers is an airline company that needs to respond to over 100 million unique, individualized price requests each day. They use Bayesian statistics, linear programming, dynamic programming and a slew of other technology to create price curves. To answer the responses, they need to be in under 200 milliseconds, so the responses must be incredibly fast. It's practically impossible for humans to do this – understanding the market economics using all available data and to do so in under 200 milliseconds. So you really need autonomous software handling this. The combination of SQL Server 2016 and Azure provided the unified platform and global footprint that made it a lot easier for PROS to accomplish this. In fact, 57% of the over 3.5 billion people who travel on planes annually are touched by PROS software powered by Azure – it's pretty amazing.



Here a link to [the blog about SQL Server-as-a-Scoring Engine @1,000,000 transactions per second](#). I also encourage you to play with [Machine Learning Templates with SQL Server 2016 R Services](#), which include templates for Online Fraud Detection, Customer Churn Prediction and Predictive Maintenance using [SQL Server R Services](#). Also, here are some related interesting sessions from the Summit for you to watch:

- [Data Science for Absolutely Everybody.](#)
- [Insanely Practical Patterns to Jump Start Your Analytics Solutions.](#)
- [The Future of Data Analysis.](#)
- [Putting Science into the Business of Data Science.](#)

Intelligent Lake

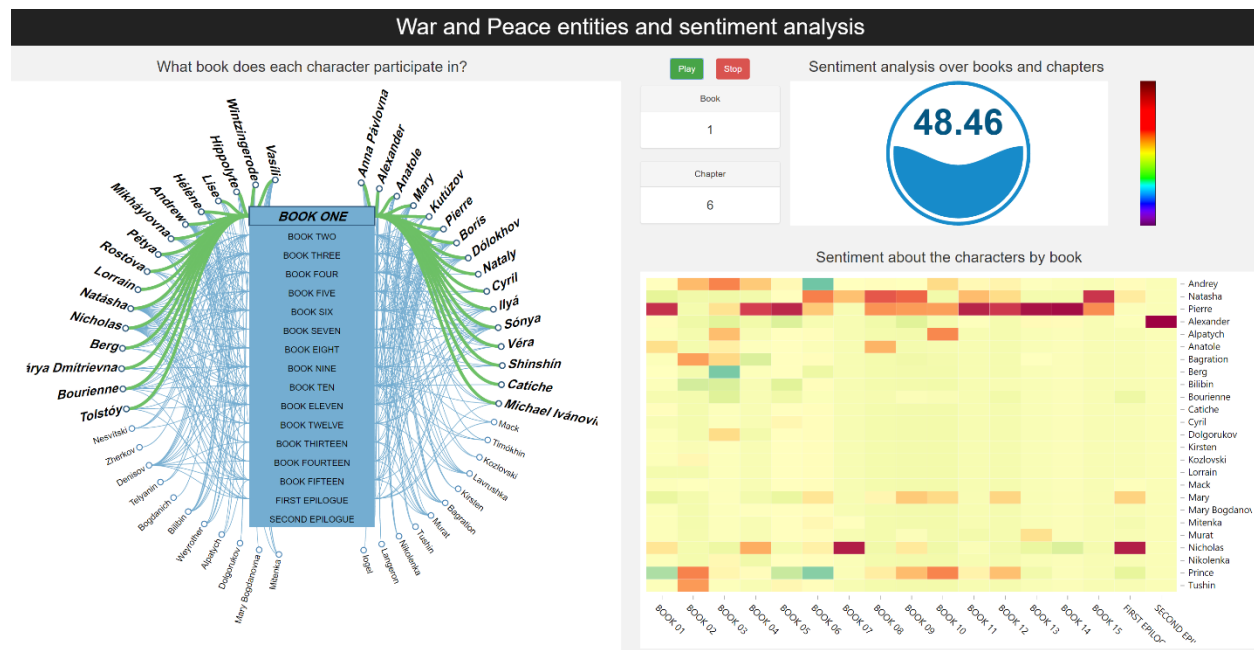
The era of big cognition is here. Think about this... software thus far has been “sterile”, unable as it is to understand or use human emotions, or combine it with anything else. Could we join emotion signals with other auxiliary data and use that in some intelligent and responsible way? While this might have seemed like science fiction not too long ago, this is all becoming possible now. You can extract emotional sentiments from images, videos, speech and text, and you can do it in bulk. This is the sort of capability that can be brought to you by the *Intelligent Lake* pattern. The fact that you can *query by content* unstructured data, e.g. images or videos, and extract emotions, age, and all sorts of other features in a data platform such as Azure Data Lake is just incredible. You can now join emotions from image content with any other type of data you have and do incredibly powerful analytics and intelligence over it. This is what I mean by **Big Cognition**. It's not just extracting one piece of cognitive information at a time, not just about understanding an emotion or whether there's an object in an image, but rather it's about joining all the extracted cognitive data with other types of data, so you can do some real magic with it.

At both the Summit and Ignite, we showed a Big Cognition demo in which we used U-SQL inside [Azure Data Lake Analytics](#) to process a million images and understand what's inside those images. These images are everyday pictures of real people, people walking their dogs, people attending weddings, people in

suits going to work, etc. We took the DLL that sits behind the Cognitive Services Vision API and put it inside U-SQL (thanks to the incredible extensibility of ADL) and processed those million different faces, extracting emotions from them, and combining it with other data to do some incredible analytics.

The collage features the 'Extensible Intelligent Lake' logo on the left, the 'Microsoft Cognitive Services' website in the top right, and a central 'Big Cognition' text overlay. Below the logo is a grid of faces with emotion labels. To the right of the central text are examples of image recognition, including a person on a beach and a dog, with captions like 'I think it's a man standing on a beach with a dog' and 'captionbot.ai'. At the bottom right, there's a section titled 'Microsoft Just Released the Best iPhone App Ever' showing a person holding a phone.

And we didn't stop there. We went beyond image analytics and had fun with analytics on top of a large volume of text, in particular, the book War and Peace, written by Leo Tolstoy in the 1800s. It's a great book and you should read it if you haven't: about 1,500 pages long, spread over 15 books, with 50 major characters. We took the intelligence APIs and our big data engines and created something really interesting on top of that. We extracted key phrases, and then used Text Analytics API (also available as part of Cognitive Services) to get deep insights into the story. We created an "emotional heat map", showing when the characters appeared in the book and what the sentiment was at that time. Using this, even someone starting off with very little understanding of War and Peace can benefit from an incredibly rich, interactive guide on top of the text.



While War and Peace is a really massive book, it isn't really big data, per se. We ran it in an engine that will let you do this against an arbitrarily large amount of text and you can generate the same type of experience for millions of books in the Library of Congress or even the entire Internet. This is the magic of mashing up the capabilities of the Cognitive Services APIs with an extensible big data engine such as Azure Data Lake.

Here are relevant sessions from the Summit for you to watch:

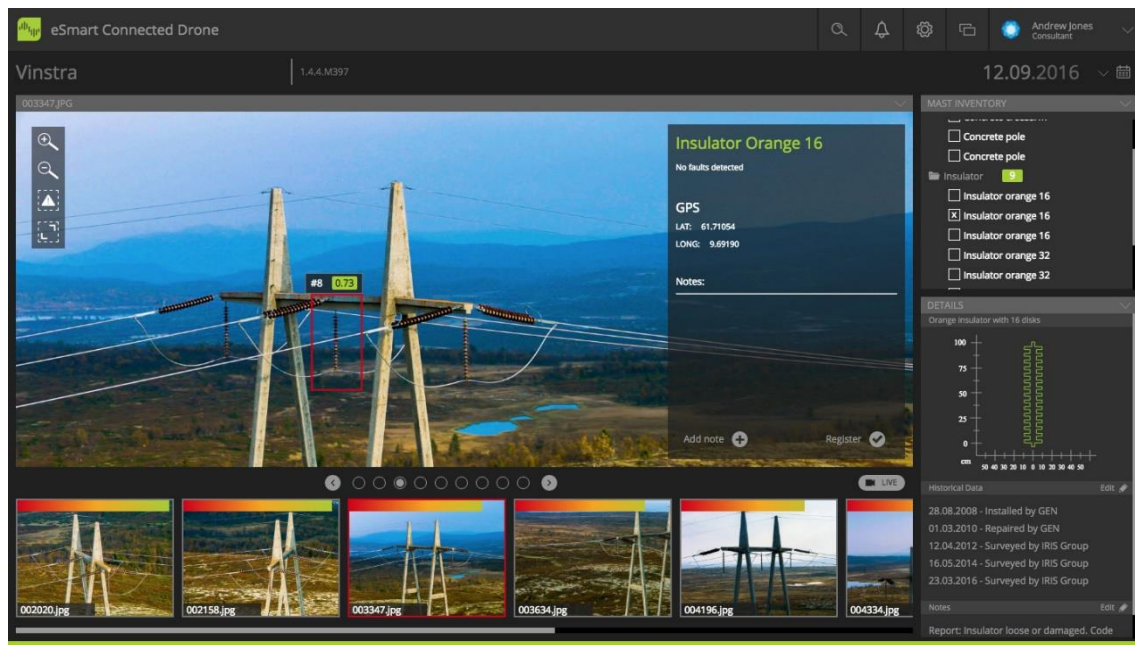
- [Cognitive Services: Making AI Easy.](#)
- [Killer Scenarios with Data Lake in Azure with U-SQL.](#)
- [Building a Scalable Data Science Platform with R on HDInsight.](#)
- [R and Spark as Yin and Yang of Scalable Machine Learning in Azure HDInsight.](#)
- ['Spilling' Your Data into Azure Data Lake: Patterns & How-Tos.](#)
- [Go Big \(with Data Lake Architecture\) or Go Home!](#)
- [Big, Fast, and Data-Furious...with Spark.](#)
- [Patterns and Practices for Data Integration at Scale.](#)
- [Optimizing Apache Hive Performance in HDInsight.](#)

Deep Intelligence

The third pattern I talked about is *Deep Intelligence*. Azure now enables users to perform very sophisticated deep learning with ease, using our new GPU VMs. These VMs combine powerful hardware (NVIDIA Tesla K80 or M60 GPUs) with cutting-edge, high-efficiency integration technologies such as Discrete Device Assignment, bringing a new level of deep learning capability to public clouds. Joining me on stage last week was one of our partners, **eSmart Systems**, who presented a demo of their *Connected Drones* which are capable of inspecting faults in electric power lines using image recognition.



eSmart Systems is a small, dynamic and young company out of Norway that made an early decision to build all of their products completely on the Microsoft Azure platform. Their mission is to bring big data analytics to utilities and smart cities, and Connected Drone is their next step in delivering value to customers through Azure. The way they put it, "*Connected Drone is a way of making Azure intelligence mobile*". The objective of Connected Drone is to support inspections of power lines which, today, is performed either by ground crews walking miles and miles of power lines, or through dangerous helicopter missions to aerially monitor these lines (if there is one place you don't want humans to be in helicopters, it's over high power lines). With Connected Drones, eSmart uses deep learning to automate as much of the inspection process as possible.



Connected Drones cover all stages of the inspection process from the initiation of the inspection plan to the planning of the drone mission, and then the execution of the mission. As they fly over power lines, the drones stream live data through Azure for analytics. eSmart Systems uses different types of neural networks including deep neural networks (DNNs) to do this, and these are all deployed on Azure GPUs. They analyze both still images and videos from the drones and are able to recognize objects in real time. The system can process video from between 10 to 50 frames a second at the moment. Among the many challenges they've faced, one that they called out on the stage is *class imbalance*, where they have many different types of objects that need to be recognized and some of those are more common than others. In order to prevent a biased classifier, the solution they came up with is to mix real and synthetic images to train their models. This is truly, *Machine Teaching*. I call it *Mission NN-possible*!



Another example of the power of deep learning was demonstrated in Microsoft CEO Satya Nadella's keynote at Ignite, where he showcased the magical capability of bringing DNNs running in SQL Server to an application that's being run by home improvement retailer **Lowe's**. You can watch the demo [here](#) to appreciate how they've created a magical experience around kitchen remodeling, by combining HoloLens with the ability for DNNs to recognize images and match a personal style in an augmented reality environment.

Today, to implement the Deep Learning pattern, you can use the [Data Science VM](#) which is already pre-loaded with deep learning toolkits or use [Batch Shipyard toolkit](#), which enables easy deployment of

batch-style Dockerized workloads to Azure Batch compute pools. Both are ideal for Deep Learning, so give them a try.

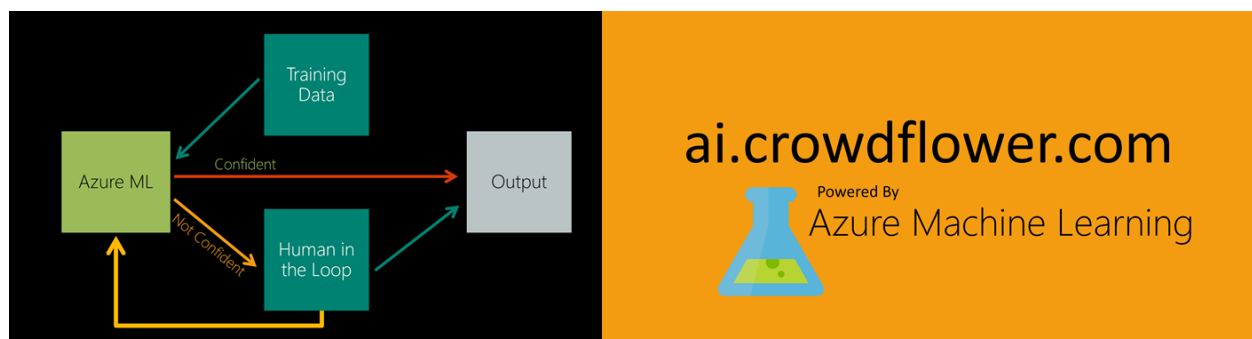
Here are some interesting related sessions from the Summit:

- [CNTK: The Microsoft Cognition Toolkit: Democratizing the AI tool chain: Open-Source Deep-Learning like Microsoft Product Groups.](#)
- [Unlock Real-Time Predictive Insights from the Internet of Things.](#)
- [Deep Learning in Microsoft R Server Using MXNet on High-Performance GPUs in the Public Cloud.](#)
- [Intelligent Virtual Reality Kitchen Design: Behind the Scenes.](#)

Human + Machine Intelligence

In this fourth intelligence pattern, we use humans to enhance machine learning. Humans can be used in two places. The first is to generate labels for the data that you want to learn from, i.e. humans become teachers to machines. The second is when you have predictions of lower confidence and you add human intelligence to get a better, higher-confidence result. The challenge here is how to automate this. How does one put a man behind a machine? How do you interact with a human through an API? And how do you combine both machine and human intelligence in software? Microsoft partnered with [CrowdFlower](#) to solve this problem, combining machine intelligence with human intelligence using crowdsourcing services to offer dramatically better accuracy than what either could do on their own. You can now easily create ML models using human-labelled training data and deploy these models using humans-in-the-loop in situations where a model's predictions fall below a customer defined confidence threshold.

For instance, in machine-based text translation, if ML is not confident about a translation, it can reach out to a native-language speaker and he or she can assist in correcting the translation.

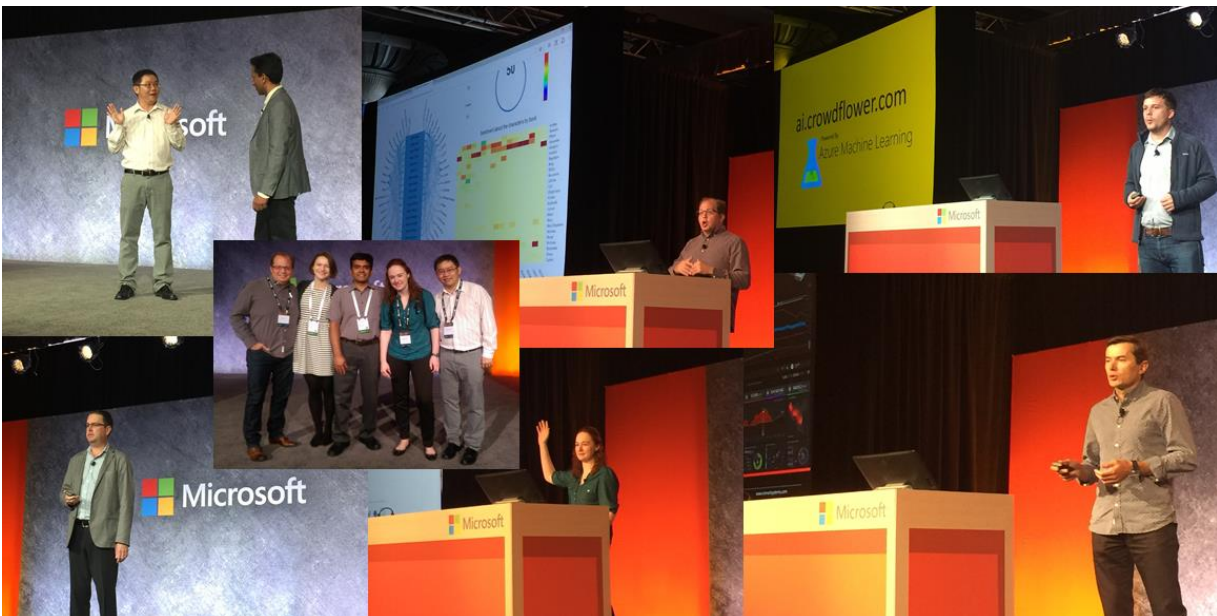


Another big application of this pattern is routing support tickets. Although this sounds simple, it's actually rather complicated. For those of you who run consumer apps or services, you must be accustomed to getting tons of customer feedback. If you process this feedback right and in a timely manner, you can deliver an amazing customer experience. But one big issue is that, if your app or service were to experience exponential growth, your support costs also go up exponentially, because as a business you do want to try your best to look at every support issue/ticket. To solve this problem, you can use [CrowdFlower AI](#). First, you label all your past tickets with the category that you want for them and then send it to [Azure Machine Learning](#). This actually works really well with over 95% accuracy for many of our customers. But businesses want perfect support (that's how they differentiate themselves), and often that's 99% or better accuracy for processes in their value chain. So, where a prediction is not confident, it can be routed through a human who can help get it labeled. The amazing thing is that you can feed that data back into your ML model and make it better over time. Data scientists call this *active learning*, and, to business customers, this translates into big cost savings. The resulting outcome is that, as your company scales exponentially, your support costs scale linearly. If this sounds relevant to your organization, take a look at Crowdflower AI and watch the Summit session titled [Building Real-World Analytics Solutions Combining Machine Learning with Human Intelligence](#).

Intelligent Bots using Cognitive Services on Azure

The fifth and last pattern is called *Intelligent Bots*, or *Intelligent Conversations-as-a-Platform*.

Conversational platforms are the next frontier in human interaction models – rather than having hundreds of apps on your phone with which you interact for different things, you just go to one single conversational platform of your choice and talk to agents or bots that live in that platform. You can design bots for customer service or shopping or arbitrary other services that you need to deliver. Today, we support a number of powerful intelligence capabilities to support this scenario. Inside the [Cortana Intelligence Suite](#), there is a [Bot Framework](#) that allows you to build your own bots. You also have APIs in Cognitive Services such as LUIS, the [Language Understanding Intelligence Service](#), which allow you to do key phrase extraction from the natural language that people are typing, so you can understand customers' queries. You can also reason over it and run sentiment analytics. As a result, you can use all that information to construct the right response back or generate the right queries into the back-end systems, so you can pull just the right information for the customer during the chat session (see the [Ignite Bot](#) that we demonstrated last week). That's how you integrate intelligence with the conversational framework to deliver powerful chat applications to your customers.



If you go to the Bot Framework gallery page, you will see a large number of bots, infused with intelligence, already pre-created for you.

One other interesting bot I'd like to highlight lives on the Microsoft support page, providing answers to customers who need support. This is making an enormous difference to our customers as they can get their answers directly in a chat session with the support agent, as opposed to being placed on hold on a phone line and having to wait for a while before their query can be answered. We are just starting to scratch the surface of this and there will be further enhancements to this bot as we continue to learn from these interactions.

The combination of the Bot Framework with LUIS, with Cognitive APIs, with Intelligence DB, Intelligent Lake and Deep Intelligence allows the creation of a very rich set of implementations. If you use Intelligent Lake with the Bot Framework, you can have extremely sophisticated deep analytics on a very large amount of data standing behind the bot to inform it with the most intelligent response. Watch [Bots as the Next UX: Expanding Your Apps with Conversation](#) to learn more about this. I firmly believe that bots infused with intelligence will really change our world.



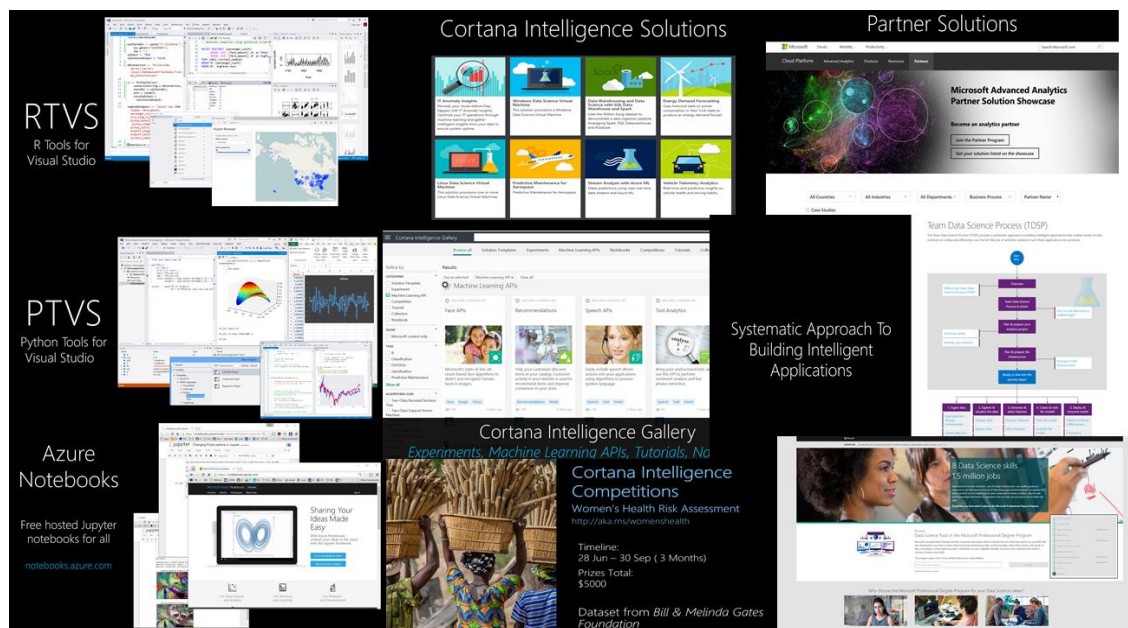
To quickly recap: I talked about five patterns for implementing intelligent applications – the *Intelligence DB*, the *Intelligent Lake*, *Deep Intelligence*, *Combining Machine and Human Intelligence*, and *Intelligent Bots*.

To support these patterns today, Microsoft provides an incredible wealth of tools and solutions:

- We have very powerful tools in [R](#) and [Python tools for Visual Studio](#) to enable R and Python development.
- The [Cortana Intelligence Gallery](#) offers solution templates from not just Microsoft but data scientists all over the world.
- We also launched [Notebooks.Azure.com](#), where we've taken Jupyter notebooks (previously IPython) and now host them on Azure. There are now universities using these notebooks for teaching classes, and companies using them in collaborative ML and Data Science projects. These notebooks live entirely in the cloud and can be accessed by a large number of people in an organization to contribute both text and code.
- We also host several [Cortana Intelligence Competitions](#). Our last competition was on [Women's Health Risk Assessment](#) and just finished a few days ago. We started off a new [Decoding Brain Signals](#) competition, where we've taken data from recordings of human brains and asked people to predict what a human being was seeing, based on those signals. The results from the latter are just amazing: People from all over the world participated and were able to beat the results that the original neuro-scientists had achieved, i.e. in terms of accuracy of predicting what people were seeing/thinking.
- We also offer [Cortana Solutions](#) for quickly starting your intelligent application development. It enables you to quickly build Cortana Intelligence Solutions from pre-configured solutions,

reference architectures and design patterns. You can make them your own with the included instructions.

- We have also developed the [Team Data Science Process \(TDSP\)](#) which provides a systematic approach to building intelligent applications. It enables teams of data scientists to collaborate effectively over the full lifecycle of activities needed to turn intelligent applications into products.
- We've also kicked off a number of hackathons recently, including one at [Georgia Tech](#) and another at [Columbia University](#), using these above tools and solutions.
- Furthermore, in an effort to provide the latest resources to current and aspiring data scientists, including the skills and practical experience they need to be successful in this high growth field, Microsoft has [opened registration for the data science track of the Microsoft Professional Program \(MPP\)](#). The MPP program features a comprehensive curriculum and hands-on labs, designed with input and participation from industry leaders, universities and learning partners, and helps to close the skills gap and create new opportunities in this field. Since we announced this program in July, the response has been truly inspiring - over 50,000 people have expressed an interest to enroll, and many pilot participants are providing excellent feedback on the program. If you've been thinking about a career in this field, we hope you'll consider enrolling at academy.microsoft.com.



A few final links to additional Summit sessions you will find really interesting:

- [See, Hear, Move: Changing Lives with Data Science.](#)
- [What is your Data Science Super Power? Tips and Tricks to a Successful Data Science Project.](#)
- [Data Science Doesn't Just Happen, It Takes a Process. Learn about Ours...](#)

All of this is super exciting. We are at the cusp of the intelligence revolution today. When the steam engine was first invented more than 200 years ago, nobody knew that it would herald the industrial revolution. Today, the cloud and the intelligence platforms you're seeing on the cloud, be it from Microsoft or others, represent the "steam engine" of our intelligent future.

The **Intelligent Revolution** is truly beginning.

Joseph

[@josephsirosh](#)