



EASTERN ANALYTICS, INC.

BEST PRACTICES FOR ENRICHING YOUR DATA - TRANSCRIPT

COMBINING MICROSOFT POWER BI & AZURE AI FOR OPTIMUM RESULTS



Combining Power BI & Azure ML for Optimum Results

Welcome, everybody, to "**Best Practices for Enriching Your Data: Combining Microsoft Power BI and Azure AI for Optimum Results.**" Just to quickly let you know who we are, we're Eastern Analytics, a boutique consulting service, who helps customers maximize the value of their data through technology. We specialize in Microsoft Analytics, Power BI, and Azure.

As seasoned experts, we have the functional knowledge that allows us to understand our customers' sources and requirements, *and* the technical expertise to design and build systems that are robust, flexible, and secure. We assist our customers with their analytic needs, taking them through the 4Ds, design, development, deployment, and DevOps, and cover everything from technology advice and solution architecture, to support and staff augmentation.

Now, Scott Pietroski is Eastern Analytics Managing Partner and Senior Solution Architect. And for more than 25 years, he's been building out analytic platforms for corporations such as Bose, Adidas, Estee Lauder, and many more. He specializes in Microsoft Power BI and Azure and is excited to share with you today's presentation.

Hi, everybody. I'm Scott Pietroski. Welcome. Today's presentation, we're going to do an overview of Power BI's integration with Azure AI, which is also known as Cognitive Services and also Machine Learning.

We're going to talk about **Azure ML versus Power BI ML**. We're going to talk about the pros and cons of each environment. We're going to talk about the technical side of Azure ML and Power BI. We're going to talk about cost considerations and best practices. At the end, there will be a question and answer period. I have a lot of information packed into this, so we're going to hold off to the end for the questions just because we don't want to go down too much of a rabbit hole in a technical question, but we will answer them at the end, whatever questions you have.

So, Power BI and AI ML. Okay, as most people know, there's two flavors of Power BI. Most of you are probably using Power BI. It's very popular. There's the Power BI desktop, which is a standalone application. You basically do development work in the desktop. You can pull in your data, you can wrangle it, and you can build reports right in the desktop. A quick picture here of the desktop for people who are not familiar with it. Here, basically, when you go into the desktop, you would have a button up in the middle that would be transform data. When you click on transform data, you go into the power query editor. This is where you do your data wrangling. This is an example. We have a table over on the left, and we're going to be talking about using the AI Insights. Over in the top right-hand corner here, you can enhance your data using Azure Machine Learning, Vision, or text analytics.

Now, the Power BI service is basically the cloud service where you can publish your Power BI desktop data sets. You can publish them out there. You can create workspaces. You can restrict access to the workspaces, share the work with your teams, create reports, and a bunch of other things out there in the web. The Power BI service, not only can you publish Power BI artifacts out to the web service, but you can also train ML models out in the Power BI service as long as you have the premium level. This is a picture of the Power BI service. Most of you may be familiar with it. Inside of the Power BI service, you need to have the premium level. It has to be a premium user or premium account where you have different premium levels.

Combining Power BI & Azure ML for Optimum Results

You can tell it's premium because of this diamond icon on here. You can tell you're in the premium workspace. This is the workspace that I created here for this presentation. In it, we have a data set that was published from the desktop, but we also have data flows. It's in the data flows that you can actually create the ML models. You can then consume them out here on the cloud service. For AI ML integration, both the desktop and the web service allow you to consume AI ML models. Only the Power BI service allows you to create them. In the Power BI desktop, you cannot create them.

When we talk about **AI ML services in Power BI**, we want to think about **Azure's Cognitive services**. Cognitive Services. Cognitive Services are services that Microsoft basically has pre trained ML models and they provide it as a service. You can connect both the Power BI desktop and the service to these Cognitive Services. And there are two categories of services. So, you have text analytics. Inside of text analytics, you can basically pass the service a text string and it will do language detection, tell you what language it's in.

It'll do key phrase extraction as another service, which is where it pulls up the block like you would see in a Google page results. That's what it's doing is key phrase extraction from that page. It'll also do sentiment analysis to tell you whether or not maybe a blog post is positive, negative, or neutral.

Another category of AI ML is vision. It's image recognition. What the service does is it actually adds tags to your image, or should I say, returns tags about your image. The tagging, an example that I have here, Man on a Horse. If you had a picture of a man on a horse and you uploaded it to this Vision Cognitive Services, it might return tags, one tag that says a man, one tag that says a horse, maybe a tag that says a man on a horse. And if there was a picture of a mall bar sign in there, like in the old mall bar advertisements for the cigarettes, it would also return a tag with the term mall bar in it. So it would actually OCR the image, recognize that there's text on it and return that as well.

Now for Azure ML, in Power BI, you can consume any model that you create inside of Azure ML on the Azure ML platform. You do need to be authorized to access it, so you need access to the workspace under your particular user ID, but you can consume any model. Here's a picture here of the desktop. In the desktop, I happen to go up here and select Text Analytics in the Power Query Editor, and I get a pop up screen. Sure enough, these are the different functions that are available, sentiment analysis, key phrase extraction, etc. You can just basically add the field here that you want to perform the inference on, and it will add additional fields, which are the results.

The same thing happens in the Power BI service, but it's a little bit different. The Power BI service, basically we have the cognitive services, and these are the ones I just mentioned between image recognition and the text analytics. But also what it's showing here is a couple of other things. It's showing two machine models that we trained out on the Azure ML platform, so we could actually consume them just as easily as we consume the Cognitive Services. It's also showing us here Power BI machine learning models. So out in a data flow in the Power BI premium service, I've trained a model, and that model now is available for use here as well inside of a data flow.

Combining Power BI & Azure ML for Optimum Results

So this is power query that I'm on out on the service and you can consume ML models, whether it's Azure ML, standard Microsoft services or Power BI models. If we think about Azure ML versus Power BI ML, most people are familiar nowadays with machine learning at a high level.

What is Azure ML? Azure ML is a standalone environment for machine learning. It's a machine learning platform out in Azure. It has seamless integration with Power BI, so you can consume the models as I showed you in the screen before. It includes Auto ML functionality similar to what happens in Power BI, which I'll explain in a moment. It includes Auto ML functionality, but it's presented in a different way. Azure ML is built for a data scientist, a technical person. It's an ML platform. It's designed for enterprise level ML. You can use the GUI to do it, or it can be completely code driven. You can do it all in Python under the hood and consume all the different objects on the platform in Python. But it provides a common framework for data scientists. You can create Jupyter Notebooks in there and consume everything.

A quick look at the **Azure ML platform**. This is the Azure ML platform. Basically, I happen to be in a pipeline here. Think of this as a data flow, but it's flowing through the specific data science and ML steps needed. This particular one ingests data and then reduces the columns to what it needs and then enhances certain elements to the records. Then in this case, it splits the data and trains two different algorithms to create two different models to see which one is more accurate. On it, you have a graphical GUI interface where you can do all of the data science steps that you would need. Behind each one of these activities is actual Python code called and calling Python methods and functions. You can also do all of this inside of a notebook. It's got Auto ML. What is Auto ML? Auto ML is something where you upload a data set, you pass it all of your features, which are the things you're going to use to predict a result, and then you also pass it the result or the thing you want to infer. It then goes through, it takes that data set and says,

Okay, I know what I want to do my predictions on the features, and I also know the column that I want to predict. It will go through and take that data, clean it, do preparation steps, and then after that it will loop through a bunch of predetermined algorithms and basically create a bunch of different ML training jobs, one per algorithm. It will loop through that, do one per algorithm and save off all of the artifacts for each job and find you what it thinks is the most accurate algorithm to use and train the model. It's very powerful. It's a great place to start. Then after that, you can always enhance it.

Along with that, we have data assets. We can connect and pull in data to this platform from all different systems. Everything happens under the hood as jobs. It tracks everything. You can create pipelines, you can create custom components, things that you use that aren't automatically defined, etc. It's a data science platform designed for machine learning. If we go to the Power BI ML, now, Power BI ML, basically, you create your models as part of a data flow. It's part of the Power BI service. It's not a standalone platform that you have to stand up. It's a separate set of functionality built into Power BI that allows you to train and predict.

Combining Power BI & Azure ML for Optimum Results

It's 100 % Auto ML, so it's not modifiable. You can't go through an Azure ML and customize it. It's not modifiable as wizard-nondatabased, and it really includes a whole bunch of functionality to support non data scientists. Think of data analysts. When you think of Power BI ML, think of a strong data analyst. It walks the Auto ML Wizards and actual functionality built into its Auto ML will go through and do things like clean the data. It'll help you with feature selection rather than assuming you know what you're doing. It'll look at cardinality, it will look at all sorts of things to figure out what are correlations, figure out what are the best features to use, automatically restrict on those, and then train your model for you.

One of the nice things is that because it's a service...In the Azure ML world, you have access to every single component and everything that's generated every step of the way. The Power BI ML, it's a service. What they do is they additionally output some of the artifacts that are created and some of the analysis that is created as part of its logs of its run so that you can get to it and still try to figure out under the hood exactly what it's doing.

This is just a quick picture of the wizard inside of the Power BI service and its ML. It's a very basic wizard. There's not a lot of settings you need to do other than tell it or agree with that it's going to tell you, 'Oh, it looks like you're trying to predict a number. This is a regression, correct?' Things like that. It's very basic. It'll walk you through just clicking next four or five screens and then start its training runs.

Azure ML versus Power BI ML, what are the pros and cons? Why would I use one over the other? The pros of Azure ML is the biggest pro to me is that the results can be used outside of reporting. A good example of that was we had a project where we did for a customer, we built what's called a data enrichment platform. We took all of the global sales for this customer and all of their competitors' products around the world, and we move that data to a Master Data Management Service. So all of the products, global products and these products in this market segment were inside of a Master Data Management System.

And we then used ML to basically categorize all of the competitors' products so that they were in line with the same categorizations that the customer had. By doing that, they were able to go through and compare all of their competitors' products to their own products on an apples to apples basis, products categories, how it fell onto their product hierarchy, subcategories, all those different things, and they could then do really accurate market share analysis. That's an example of why you would want to use something outside of just your reporting environment.

Another pro is that Azure ML is completely customizable. You've got your Auto ML that we talked about. You've got your GUI driven, which is how you saw a pipeline created or an example of one. You've also got notebook based, which means you can really make it do anything you want. It comes with the data science libraries, all the standard stuff so I can learn the rest of it. It comes with that installed automatically on the computers and you can reference it. It's created for a data scientist.

Combining Power BI & Azure ML for Optimum Results

What are some of the cons of this? Well, some of the cons is you have to implement it. It's the standalone environment. It requires a working knowledge of the platform, spinning up computes, those kinds of things, authorizations, all the rest of it. Then it also requires at least a basic knowledge of data science to get started.

If we look at Power BI, Power BI is ML. The pros is it's wizard driven, it's easy to use, it's built for a data analyst. It also includes some additional things in its auto ML such as feature suggestions, warnings. Basically warnings could be anything from cardinality to unbalanced classes, things like that, the basics of machine learning. Then it's also included in your Power BI service, so you don't have to do anything, you just have to use it.

The cons is there's not that much flexibility when the accuracy is low. One of the things that we did over in that data enrichment platform with Azure ML was not only did we export the inferences, but we also exported the probabilities. So then human beings then went in and looked at the low probability predictions and corrected them if they were needed. And once they were corrected and the data was signed off on, we then were able to use it downstream in analytics.

But just as importantly, we were also able to use that approved accurate data set as a training data set again so that our future models would become more accurate. The variety of the data they had seen is more accurate than the previous time they were trained. That's a limitation on Power BI's ML. Power BI's ML, there's not much flexibility when it's low, there's no way to correct bad inferences, and the inferences are not available outside of Power BI, not in an automated fashion. There's no API to get them. You can right click on an Excel and download it.

So Azure ML versus Power BI ML. What's the **cost breakdown** on it? In Azure ML, the workspace and the storage account is very basic. We're looking at \$50 to \$100 a month for that, which is fairly inexpensive. The actual cost that comes in is the required computes for it. This is a cost estimate just for a really simple basic deployment of pushing out one or two models into production. But that's going to require \$2,400 a month estimate for a compute instance, which is on demand. You need that running when you're developing your notebooks and you turn it off.

Also for the compute cluster, that's on demand, turns itself off when you're done training. The inference cluster is a dedicated instance. What that is is in order for you to consume models, let's say in Power BI, you need to publish that model and have it live as an endpoint so that Power BI can consume it. Well, in order to be live and as an endpoint, it basically has to be up and running. That is the most expensive part is your inference cluster because it has to be running all the time. Once again, we're looking at maybe about \$500, \$600, something like that. But that's just for one workspace and a couple of models published. If we go over to Power BI ML, you need the premium service or user ID to basically be able to use that ML functionality. We've got the Power BI premium, you can do it on a per user basis, which is \$20 a month. Think of Power BI Pro is 10, it's twice as much, \$20 a month to be able to use the ML features. Also the Power BI premium capacity, if you wanted to do it for your whole organization, starts at about 4,000 a month and then just goes upwards with the amount of compute you want to have for your own dedicated premium capacity. Most importantly, only premium users inside of Power BI ML can consume, or inside the Power BI service can consume the ML features. Azure ML and Power BI ML.

Combining Power BI & Azure ML for Optimum Results

Let's talk a little bit about **data preparation**. I think I might have skipped a slide. It might have skipped on me. Hold on. Okay, I actually did, and this will interest you, this slide.

We want to talk about the technical side of Azure ML versus Power BI ML. First of all, Azure ML is platform as a service. It's part of their AI offering. The data size and volume is unlimited in Azure ML, meaning that you can size your computes accordingly. You consume your data out of different sources and then use it. All of the data inside of Azure ML is stored within your own Azure ML storage account. The data is never out on the Azure ML service sitting out there. You take your data, you consume it off of a BLOB storage, the compute does this thing and then you export your results, whether it's models or the rest of it, that also can get saved to your own storage account. Access control, it's role based authorizations, so you can control every single thing that you need to.

Then model retraining is orchestrated through the data factory. Now, why is that important? Well, it doesn't cost a lot to run typical data factory jobs if you're not doing a lot, but you still will end up doing your training runs, if you're doing any background inference runs, those kinds of things you do through the Azure Data factory. If we take a look at an Azure ML set up, inside of here, we've basically got your workspace, we've got an inference cluster. That's the expensive part because it has to be up and running all the time. We've also got a key vault for security, we've got application insights, and last but not least, a storage account. There's not a lot to it, but you still need to stand it up. This would be for one instance. It would be one tier in your landscape if you had a multi tier landscape. If you look at Power BI ML, it's built into the Power BI service. The model size is limited to 100 gigabyte if it's a premium user license, 400 gigabyte if it's a premium account. Data stored in the Power BI service, it looks to me like they're going to add capabilities to store in SQL Server, but it's not there yet.

Only premium users can access the premium flows. So if you're going to have an area by finance or something, then you'll need to have whatever finance users are going to consume it. They also have to be upgraded to premium. And then model retraining is manual. So it's performed at the data flow level. There's no way to trigger it. If we look at it here, basically, this is just a screenshot of where we have a model. You do have to select retrain. It doesn't happen automatically.

We did the cost breakdown. A little bit about data prep. Azure ML, basically, and data prep. ML is ML. You're going to want to ensure that your training data reflects real life. Make sure that you're not pruning out different kinds of document types or whatever and then having those flow through in your inference data because you'll get bad inferences. Make sure it reflects real life. Data engineering can happen anywhere. It could happen outside of Azure ML. It can happen outside of Power BI ML. It can happen within both environments. Prepare your data as you would for any other ML projects. Simple stuff, remove nulls, the outliers, standardize it, normalize it, deal with class imbalances, all the typical things. Select features based on relevance, engineer features if you need them. That's a very strong, powerful thing. The accuracy of most ML typically comes down to feature selection and feature engineering, at least to increase the accuracy. Correct your class imbalance, as I mentioned, and then Auto ML on both environments perform some of the data prep. They get it to the basic level that it needs to be, but it may not be done completely.

Combining Power BI & Azure ML for Optimum Results

Okay, **best practices**. In Azure ML, create a workspace by functional area. So if you're going to have finance models and things like that, create a separate workspace for finance. The area can become very cluttered quickly, even if you have good naming conventions and things. So separate workspace by functional area, create the pipelines, and then start cloning them for fast prototyping. You can even create separate pipelines that name specifically for clones that have all the activities you need to just swap out the parameters on each one to quickly get to trying to see whether or not ML is useful for your use case. Use Auto ML as a starting point. It's saved off in a Jupyter Notebook. You can then use that Jupyter Notebook and do whatever you want.

Use predefined roles to restrict users and minimize cost. Minimizing cost, how does that roles affect that? Well, you don't want people spinning up computes and leaving computes running, et cetera, if they shouldn't be. The roles will keep people in their lane, exactly what they're doing, and they're very fine grain, so you can actually go live on one box if you want. Last but not least, deal with data drift through auto retraining. So whether you want to use the data set monitors or whether you want to come up with your own data drift detection mechanism, you'll want to deal with data drift. Power BI ML. All data prep concepts still apply for even in Power BI. Use power query for data engineering if you need to.

Most importantly, the data consistency is a must, meaning that if you're going to take data and then send it into Auto ML and Power BI, that data is going to go through certain prep steps in Auto ML. Then when you pass additional data into that model for inference, it'll go through the same prep steps, which is fine. But make sure that you don't do steps to your training data multi-tier before you hand it into Power BI to train and not do those exact same steps to your inference data before you try to pass that into the model.

There needs to be consistency. Now, create ML models in their own data flows. This allows you to promote the ML work separate from the rest of your daily delivery stuff. Only apply cognitive services once per row. What I mean by that is cognitive services can be very expensive. Let's say you were doing a sentiment analysis. Well, every time you hit that data set and hit refresh and that table with the ML model is going through sentiment analysis again, you're going to pay for that again and again and again and again. It can be very expensive. I have one customer that has nine hundred and ninety million tweets and Facebook posts. Using cognitive services, even though on their sliding scale, it was like \$40,000. Use it sparingly, use it once, and then monitor and retrain your models as frequently as you can.

Last but not least, **into production**. We've got multi-tiered Azure ML. Azure ML supports a single or multi tiered landscape. The roles are there and the system is defined enough that you can actually just have one Azure ML environment that only certain people are authorized to push out models and publish them and use them in production.

Combining Power BI & Azure ML for Optimum Results

When I say in production, multi-tier meaning you consume them somewhere as live and the roles will restrict that. It supports single multi-tier landscape. Roles restrict what people can do. It's Git enabled for its workbooks or its notebooks. It also works with Microsoft DevOps, which is Git as well. Endpoints can be consumed by the data factory, so you can easily do what you need to from the data factory. Think of your data factory as well if you're going to use Azure ML. Then just follow your typical promotion process. If you have a three-tier landscape for your other environments, then there's nothing to matter with having an Azure ML three tier landscape and just don't have the computers running and things, the expensive parts running, but you can still have your Dev QA prod box, etc.

Last but not least, power BI ML into production, use deployment pipelines. Most of you are familiar with the service. We have the deployment pipelines and you can move things up. After you promote them up to the next workspace, you're going to want to retrain your models after promotion. Then you're also going to want to refresh any downstream objects or models that consume them. Once you've retrained it in the workspace, update your other data sets or data flows so that basically they'll be up to date.

I know I put a whole bunch of stuff into this and give you a whole bunch of information. I think it's time for the **Q&A**. I'm just looking through the questions here.

The first question we have is, **with Azure ML, do we have to have a separate development and production box?** No, you don't. It'll just support anywhere from... Even if you go to Microsoft's support and look at it, they even will say that you can have a single dual or three tier landscape. It's up to you as to exactly what you're trying to do with it. If it's a pilot, all the rest of it.

Another question is, **are there any recommendations on how to keep the Azure ML cost low?** Most importantly is your compute. One of the expensive things that people don't like is the fact that the compute that you use for working with your notebooks and actually doing your notebook development, those computes need to be running while you're doing it, and they don't shut off automatically. So, you want to be very conscious about that. The other compute clusters shut off automatically, and you can size your inference cluster based upon your needs. But that's probably the big sticking point for it when it comes to cost, is make sure that the developers shut off their computes.

It looks like our last question is, **once you go live, how do you make sure the ML is up to date?** You make sure it's up to date by basically retraining your models. You can go through and you can look for data drift. Data drift is basically where your data set is changing and the variety of attributes inside of your data set is changing enough that you're concerned the model may not recognize the patterns. You could do that the old fashioned way, which is basically to convert some of your attributes into numbers, almost like you would vectorizing inside of ML. You convert them to numbers and then you deal with maybe the average or the sum total, and then you look for a variance. You store the variance, you store the number that you calculated out maybe a month ago, and then you take the data set a month later, you look at the value and then you compare the two and if the variance is off any certain percentage, then you decide to retrain.

Combining Power BI & Azure ML for Optimum Results

Here's another one that came up. Another question just popped in. **Performance issues during data retrieval, how to generally address do's and don'ts?** The data retrieval, I assume that you're talking about on the power BI side for data retrieval. You will run into cases that, at least that I've run into cases where if I have too many models that are used inside of my data set, let's say, and I usually ran into this, creating demos for this, actually I ran into it. That usually has to do with the size of your inference cluster and whether or not you have enough memory allocated and you have a big enough inference cluster to be hit because Power BI will hit them in parallel. I think there's also a setting in Power BI that you can tell it to retrieve data sequentially instead of in parallel. I would probably either reduce your model count that you're using within your data flow or inside of your data set, and then also deal with your inference cluster so that it's sized right compared to the performance load on it. Because the performance load is really either a peak or dead. You're either hitting the thing or for your inference cluster or nothing's happening on it. It's just sitting there. You're just going to have to weigh it and try to work your way through the peaks and valleys. That's what I would recommend.

That looks like the last question that we had. There's another one. **Have we run into many times where ML doesn't work?** Yes, there's all sorts of times that you go in and you try to work your way through an experiment and you just don't have the data. You don't have the variety of the data that you need for ML to be as accurate as it needs to be. Yes, ML doesn't always work. It doesn't always work. That's just the nature of the beast. That's it.

Thank you everybody for attending. Sorry if I didn't get to anybody's questions. We'll be following up with everybody after this just to check to see if you have any questions, to get feedback from this, to see whether this was helpful. You can always, hopefully, follow us. We're out on LinkedIn. We do do webinars periodically. Every maybe two to four weeks, we've got different webinars and different topics. You can basically, if you come to our page, Eastern Analytics out on LinkedIn, would love to have you follow us.

We've also got a new series coming up, Building Blocks, where we'll talk about individual components of your analytics stack. It's every other week. Every other week we talk about something different, whether it's data bricks and unity catalog, or maybe just data bricks and ingestion, or maybe just joins in Power BI queries, and merge data. We cover a whole bunch of different topics, and that series is just starting up. Thank you, everybody, for joining us. Hopefully, I was able to provide you some information.

**For more information about Eastern Analytics, or to schedule a free consultation, visit us at
Analytics-People.us
Or call us at:
781-783-7610**