Azure Machine Learning Service

Technical Overview

Roberta Bruno – CSA AI

Azure AI

Azure Applied AI Services

Azure Cognitive Search

Azure Form Recognizer

Azure Bot Service

Azure Video Analyzer

Azure Metrics Advisor

Azure Immersive Reader

Azure Cognitive Services

Language

Vision

Speech

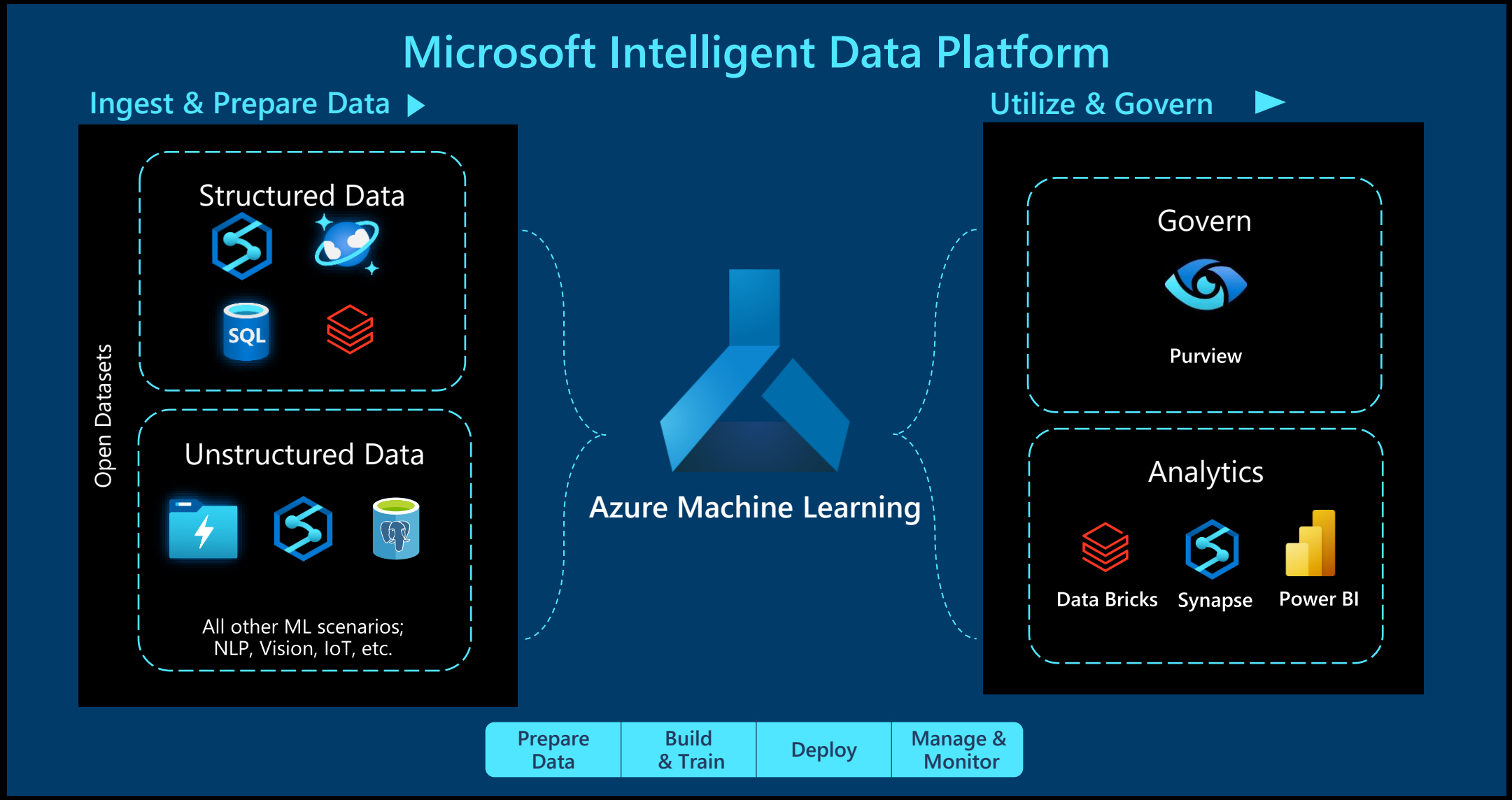
Decision

OpenAI

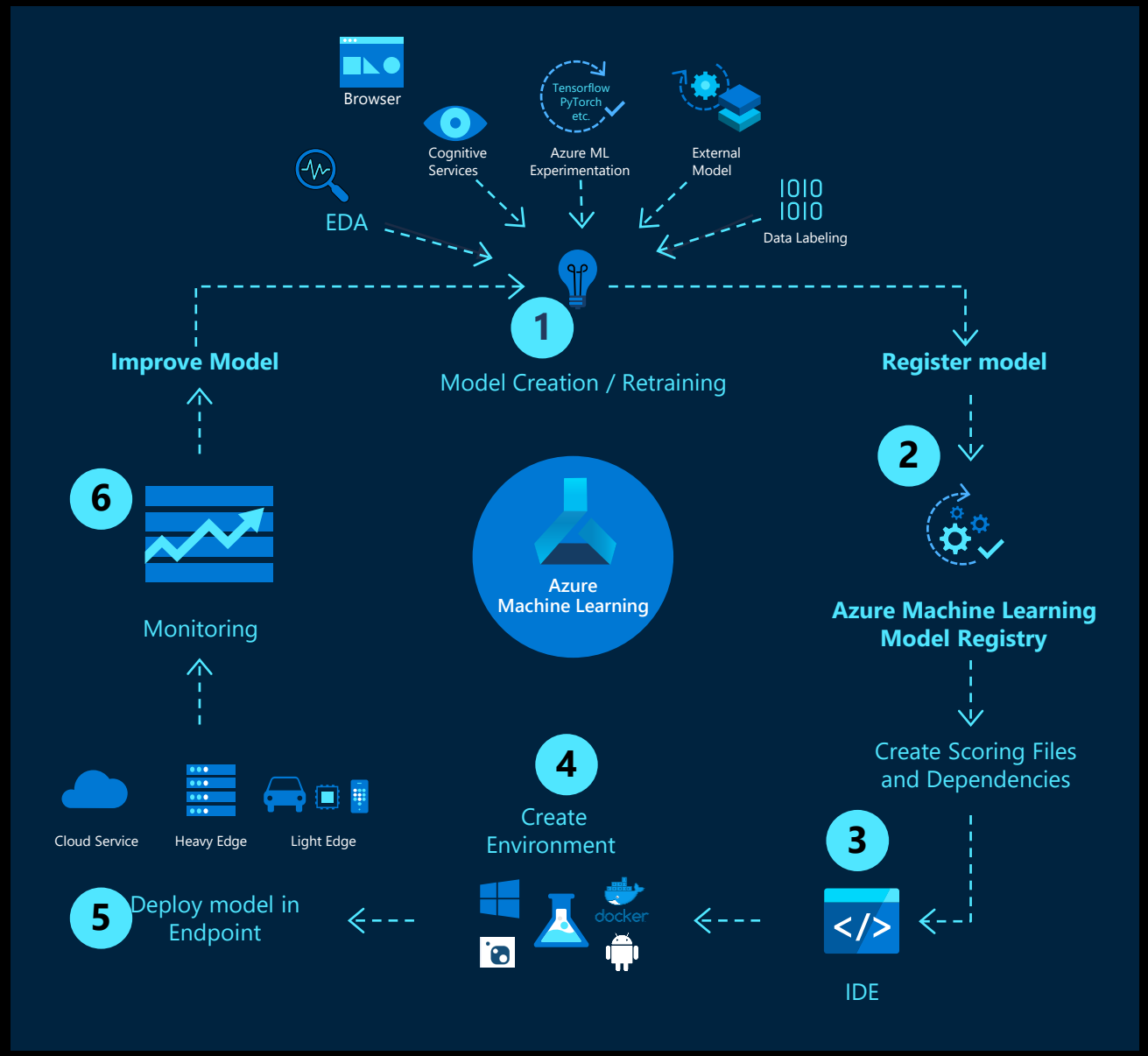
Azure Machine Learning

Azure Machine Learning

Turning Data into Business Insights



Azure ML Service can implement the end-to-end ML lifecycle



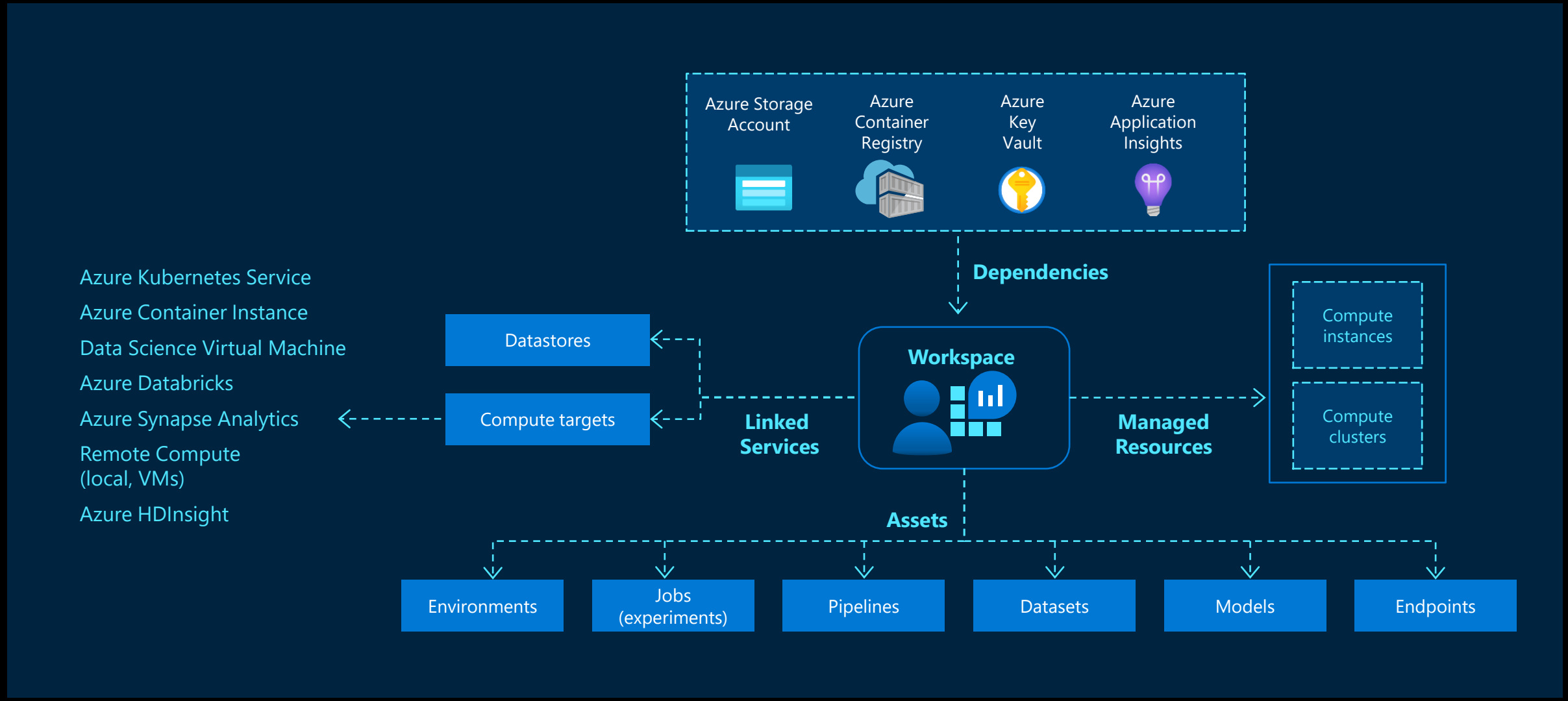
Workflow Steps

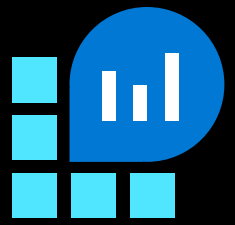
1. a) Develop machine learning training scripts in R, Java, Julia or C# using an editor of your choice - Designer, Notebooks, VS Code, Jupyter etc.
   b) Alternatively, use AutoML for no code
   c) Create and configure a compute target
   d) Submit the scripts to the compute target to run in that environment. The script runs are saved to an experiment.
   e) Review experiment for logged metrics from the current and past runs. If the metrics do not indicate a desired outcome, loop back to step a) and iterate on your scripts.
2. Once a satisfactory run is found, register the persisted model in the model registry.
3. Develop a scoring script.
4. Create an Azure ML environment that captures the dependencies to deploy the model.
5. Deploy the model using Endpoints
6. Monitor the model in production and identify when further improvements / adjustments are needed including model and data drift.

Azure Machine Learning Services

Technical Details

Azure Machine Learning





Workspaces

The workspace is the top-level resource for the Azure Machine Learning service. It provides a centralized place to work with all the artifacts you create when using Azure Machine Learning service.

The workspace keeps a list of compute targets that can be used to train your model. It also keeps a history of the training runs, including logs, metrics, output, and a snapshot of your scripts.

Models are registered with the workspace.

You can create multiple workspaces, and each workspace can be shared by multiple people.

When you create a new workspace, it automatically creates these Azure resources:

Azure Container Registry

Registers docker containers that are used during training and when deploying a model.

Azure Storage

Used as the default datastore for the workspace.

Azure Application Insights

Stores monitoring information about your models.

Azure Key Vault

Stores secrets used by compute targets and other sensitive information needed by the workspace.

Workspace Key Elements

Workspace

Models

Environments

Jobs (Experiments)

Deployment

Pipelines

Datastores

Compute Target

Data Labelling

Runs and Experiments

An experiment is a grouping of many runs from a specified script. It always belongs to a workspace. Information for every run is stored under the related experiment.

Experiment

Grouping of many runs from a given script.

Always belongs to a workspace.

Stores information about runs

Run

Produced when you submit a script to train a model. Contains:

Metadata about the run (timestamp, duration etc.)

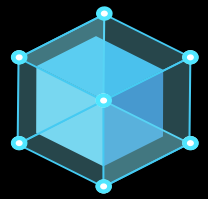
Metrics logged by your script.

Output files auto collected by the experiment, or explicitly uploaded by you.

A snapshot of the directory that contains your scripts, prior to the run.

Run configuration

A set of instructions that defines how a script should be run in a given compute target.



Models and Model Registry

A machine learning model is an artifact that is created by your training process. You use a model to get predictions on new data.

The model registry keeps track of all the models in your Azure Machine Learning service workspace.

Model

A model is produced by a run in Azure Machine Learning.

- Note: You can also use a model trained outside of Azure Machine Learning.

Azure Machine Learning service is framework agnostic — you can use any popular machine learning framework when creating a model.

A model can be registered under an Azure Machine Learning service workspace

Models created in Azure Machine Learning are versioned, with built in tools for comparing and selecting model versions

Once a model is created, you can generate a docker file or docker image and deploy the model anywhere

Model Registry

Keeps track of all the models in your Azure Machine Learning service workspace.

Models are identified by name and version.

You can provide additional metadata tags when you register the model, and then use these tags when searching for models.

You cannot delete models that are being used by an image.

ML Pipelines

An Azure Machine Learning pipeline is an independently executable workflow of a complete machine learning task. Subtasks are encapsulated as a series of steps within the pipeline.

An Azure Machine Learning pipeline can be as simple as one that calls a (Python) script, so may do just about anything.

Tasks

Pipelines should focus on machine learning tasks such as:

- Data preparation including importing, validating and cleaning, munging and transformation, normalization, and staging
- Training configuration including parameterizing arguments, filepaths, and logging / reporting configurations
- Training and validating efficiently and repeatedly. Efficiency might come from specifying specific data subsets, different hardware compute resources, distributed processing, and progress monitoring
- Deployment, including versioning, scaling, provisioning, and access control

Compute Targets

Compute Targets are the compute resources used to run training scripts or host your model when deployed as a web service

They can be created and managed using the Azure Machine Learning SDK or CLI.

You can attach to existing resources.

You can start with local runs on your machine, and then scale up and out to other environments.

Currently supported compute targets

Compute Target	Training	Inference	V1	V2
Local Computer	✓			
A Linux VM in Azure (such as the Data Science Virtual Machine)	✓			
Azure ML Compute	✓			
Azure Databricks	✓			
Azure Data Lake Analytics	✓			
Apache Spark for HDInsight	✓			
Azure Container Instance		✓	✓	
Azure Kubernetes Service		✓	✓	
Azure Synapse (preview)	✓			
Kubernetes (preview)	✓	✓		✓

Environments

Azure Machine Learning environments are an encapsulation of the environment where your machine learning task happens. They specify the software packages, environment variables, and software settings around your training and scoring scripts.

The environments are managed and versioned entities within your Machine Learning workspace. Environments enable reproducible, auditable, and portable machine learning workflows across a variety of computes.

You can use an Environment object on your local compute to:

- Develop your training script.
- Reuse the same environment on Azure Machine Learning Compute for model training at scale.
- Deploy your model with that same environment.
- Revisit the environment in which an existing model was trained.
- Azure Machine Learning provides a set of curated environments that are available for you to use by default in your workspace. The curated environments reduce the run preparation cost and allow for faster deployment.

Datasets and Datastores

Datastores are used for storing connection information to Azure storage services while Datasets are references to the data source locations

Azure Machine Learning Datasets

make it easier to access and work with your data. By creating a dataset, you create a reference to the data source location along with a copy of its metadata. Because the data remains in its existing location, you incur no extra storage cost, and don't risk the integrity of your data sources.

Datasets use datastores

to securely connect to your Azure storage services. Datastores store connection information without putting your authentication credentials and the integrity of your original data source at risk. They store connection information, like your subscription ID and token authorization in your Key Vault associated with the workspace, so you can securely access your storage without having to hard code them in your script.

Data Labeling

Azure Machine Learning data labeling is a central place to create, manage, and monitor labeling projects:

Coordinate data, labels, and team members to efficiently manage labeling tasks.

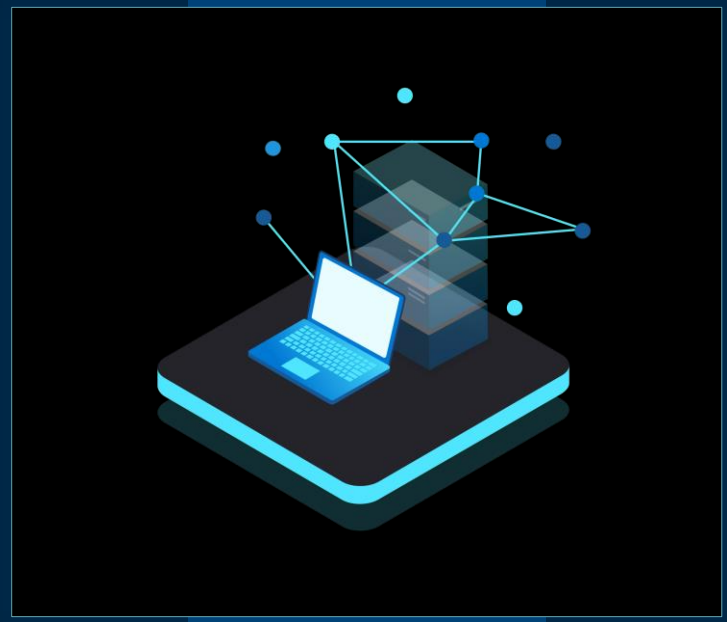
Track progress and maintains the queue of incomplete labeling tasks.

Start and stop the project and control the labeling progress.

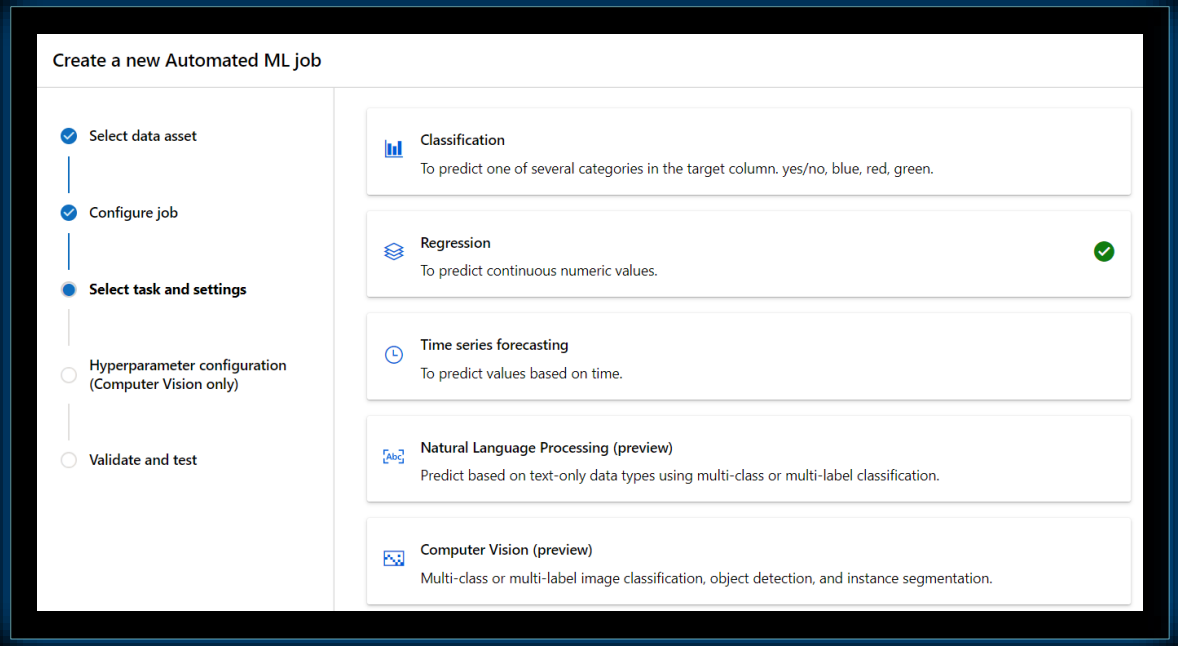
Review the labeled data and export labeled in COCO format or as an Azure Machine Learning dataset.

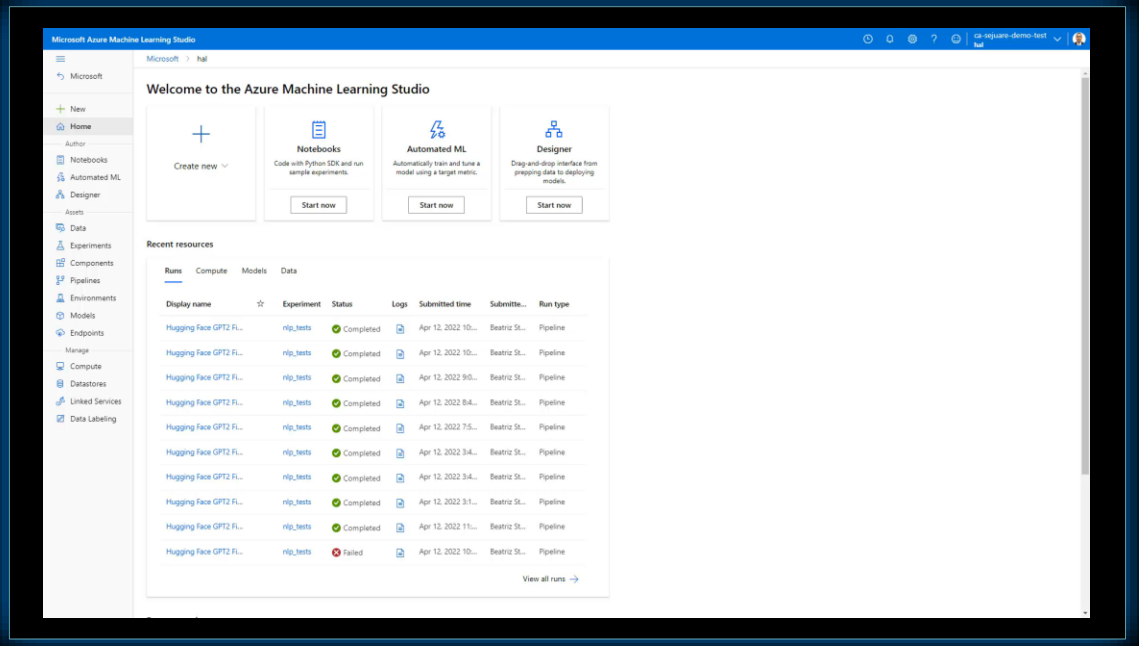
Azure Machine Learning

Authoring



Automated Machine Learning





AutoML for Tabular Data

Classification

Regression

Time Series Forecasting

AutoML for Images

Image Classification

Image Classification Multi-Label

Object Detection

Image Segmentation

AutoML for NLP

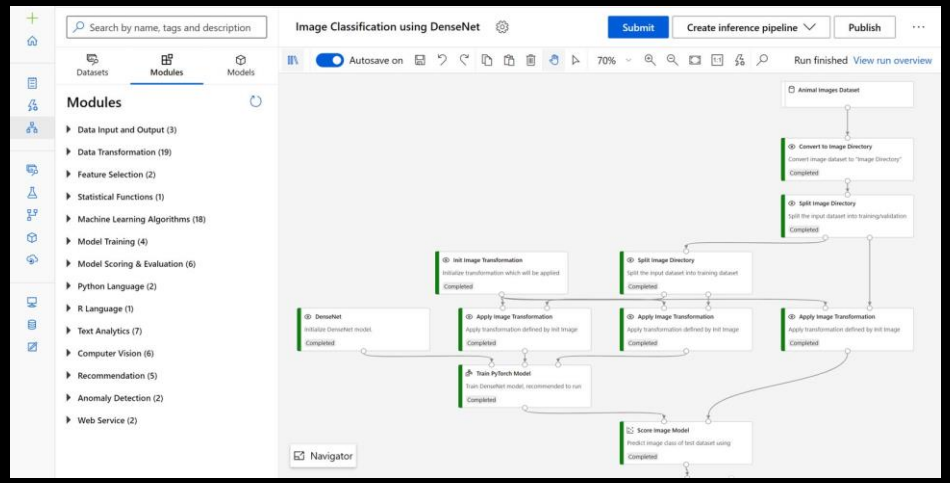
Text Classification

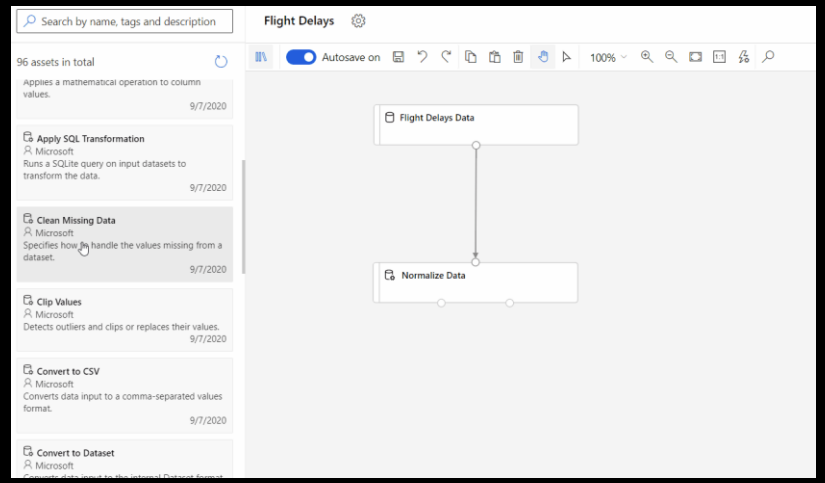
Text Classification Multi-Label

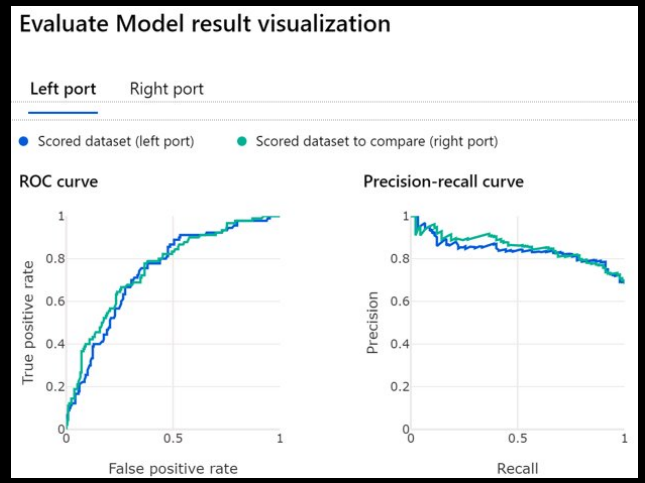
Text NER (Named-Entity Recognition)

AutoML Code Generation

Azure Machine Learning Designer







What is the Azure Machine Learning Designer?

Designer is an UI interface which enables you to build machine learning pipelines with drag-n-drop experience and simplify the publishment and deployment of your pipelines.

Key advantages

Connect to your own data with ease.

Hundred of pre-build components help you to build and train models without writing code.

Automate model validation, evaluation and interpretation in your pipeline.

Deploy models and publish endpoints with a few clicks

Notebooks Feature Set

Compatible with Jupyter

Fully compatible with Jupyter notebooks

Improved Notebook editor

Monaco editor, IntelliSense and Gather

Git Integration and Terminal

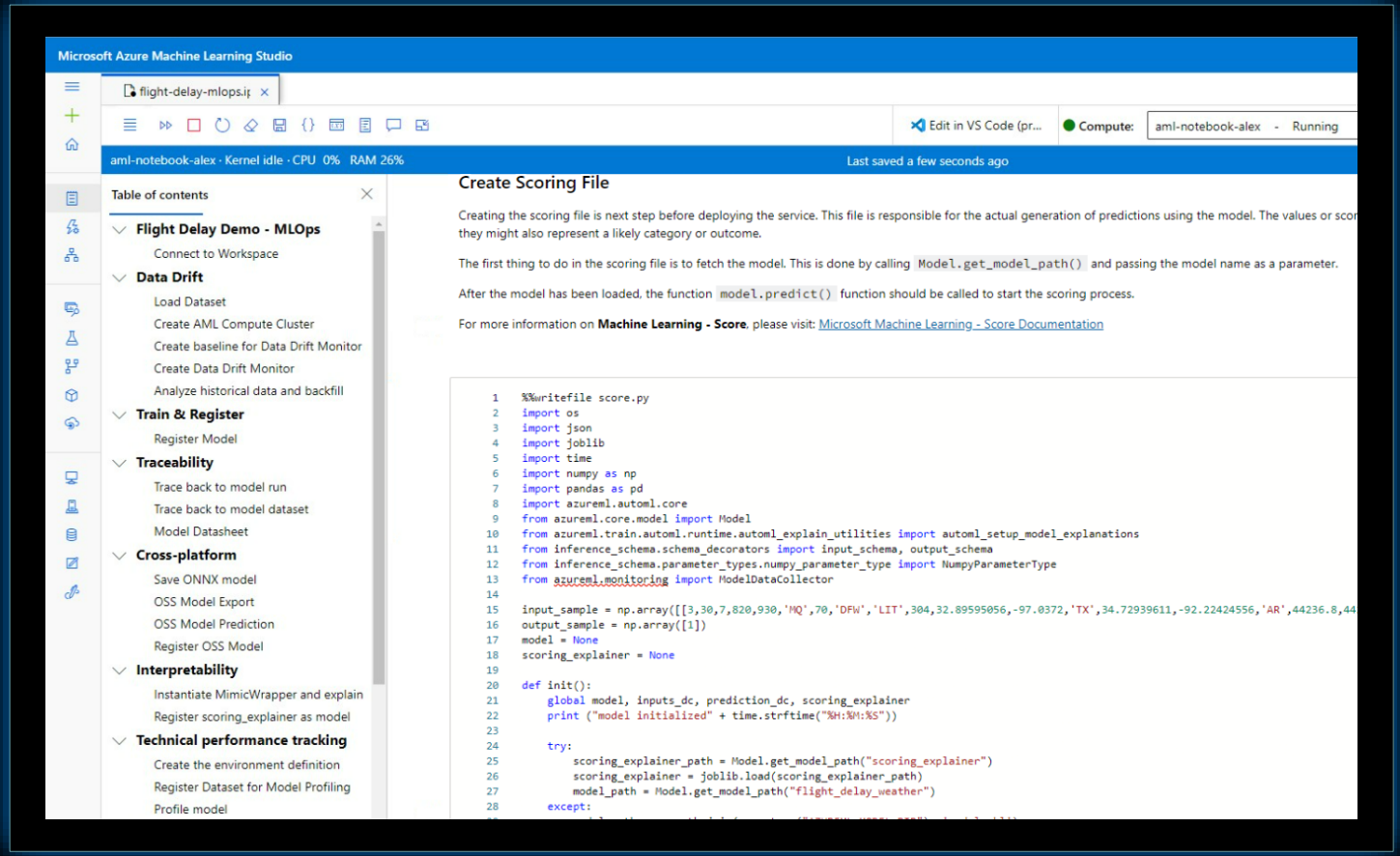
Collaborate with Git and use the terminal

Azure Machine Learning integration

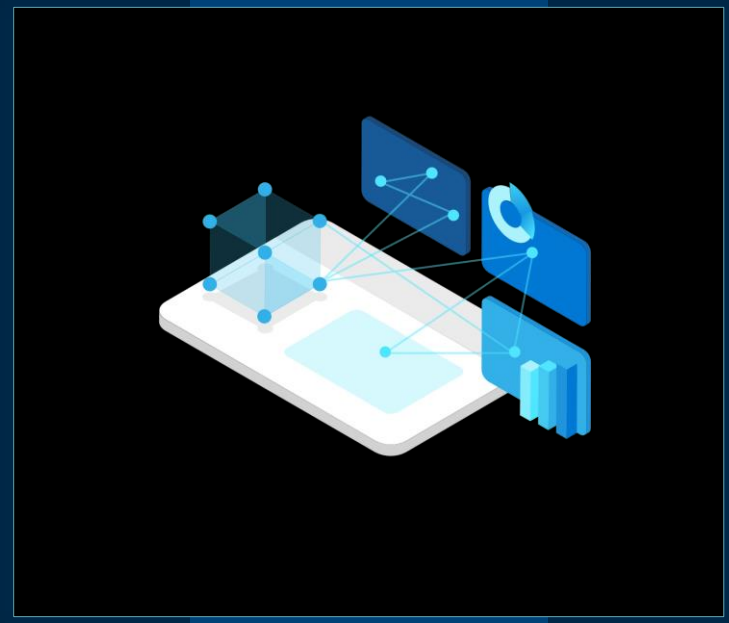
One-click access to all AzureML assets

VSCode Integration

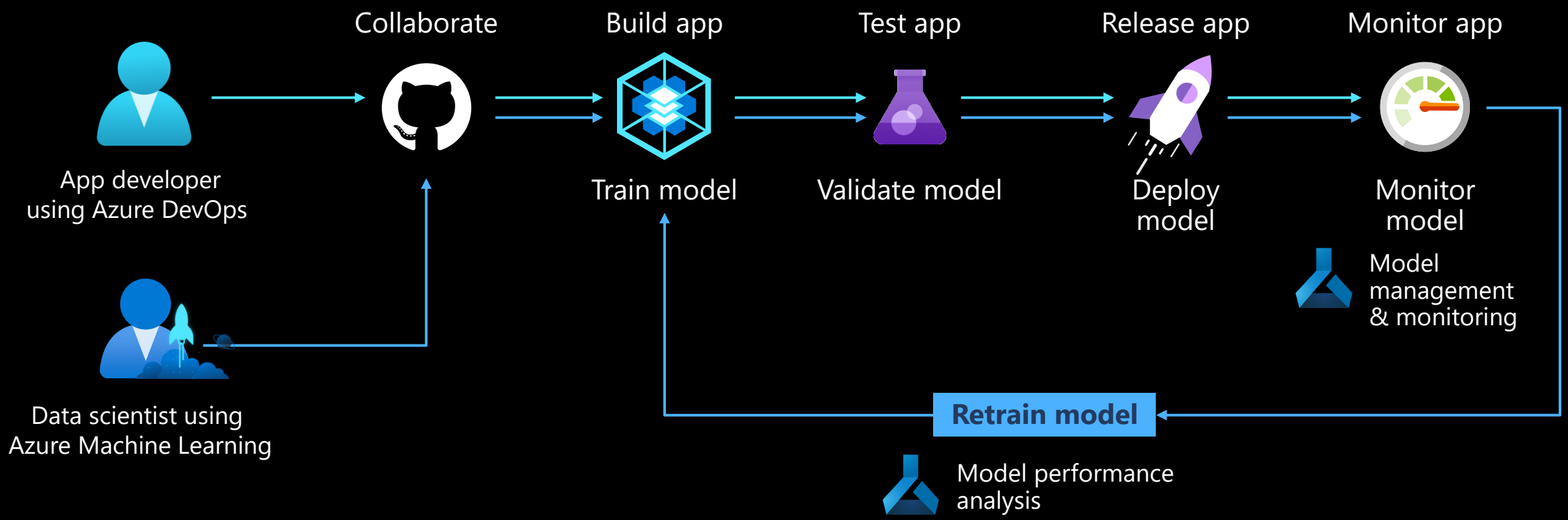
Transition to full fledged editor



MLOps



MLOps Workflow



Model reproducibility

Model validation

Model deployment

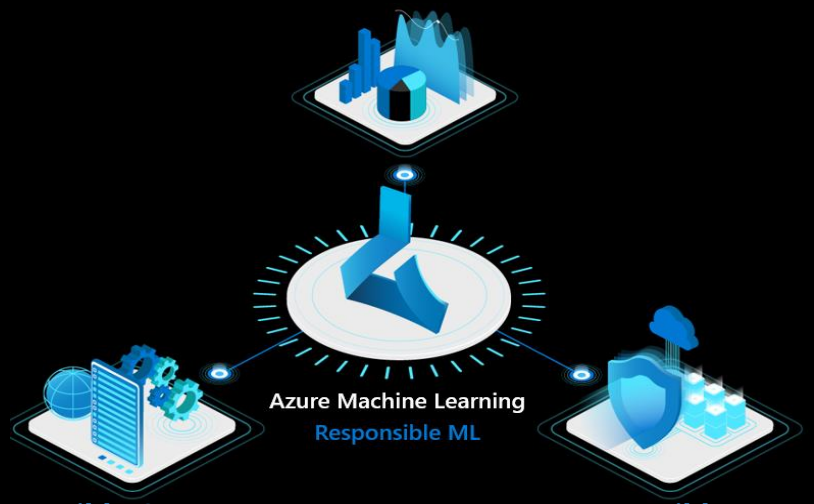
Model retraining

Responsible ML

GA in 2022

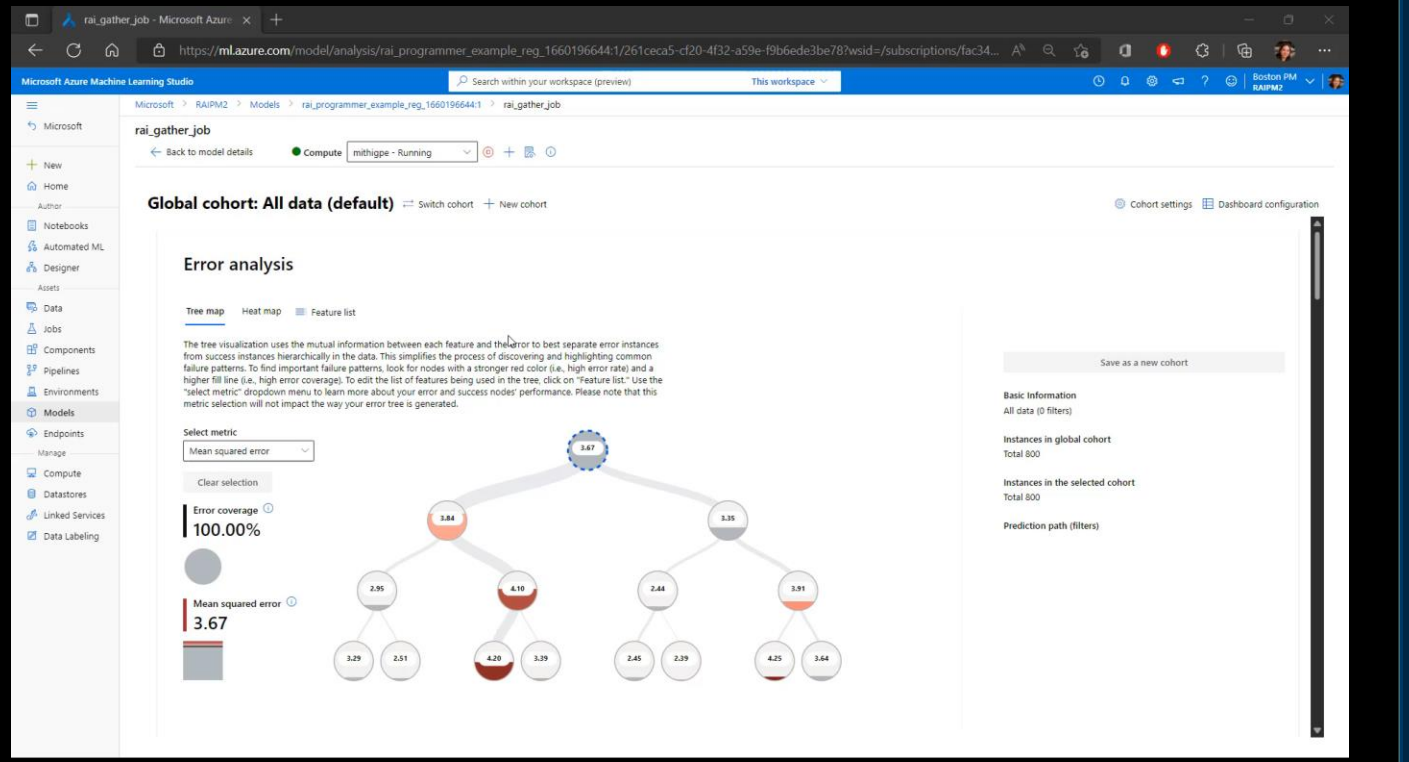
Responsible AI Dashboard in Azure Machine Learning

Responsible Development
Fairness
Explainability

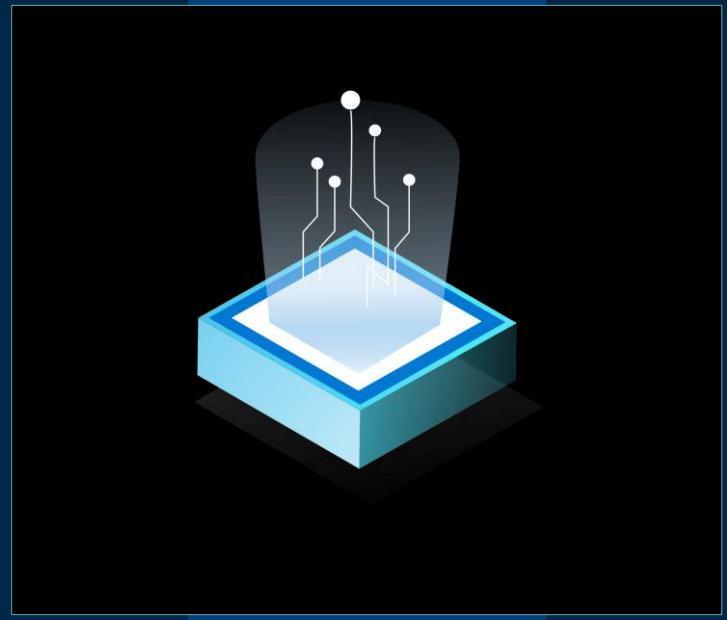


Responsible Governance
Transparency
Accountability

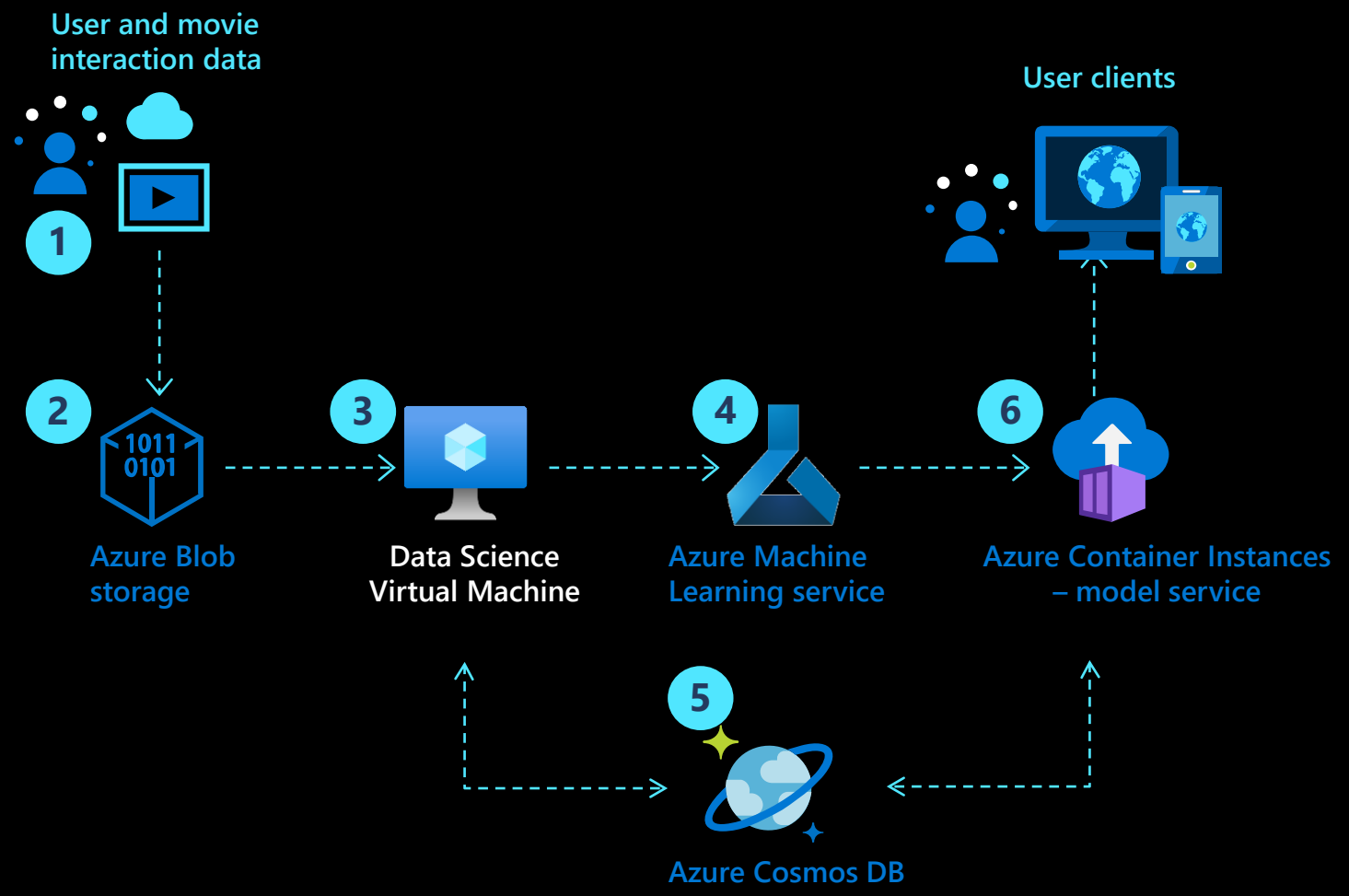
Responsible Usage
Tools



Architecture Examples



Movie recommendations on Azure



Microsoft

Q+A Session

Microsoft

Thank you

Microsoft AI