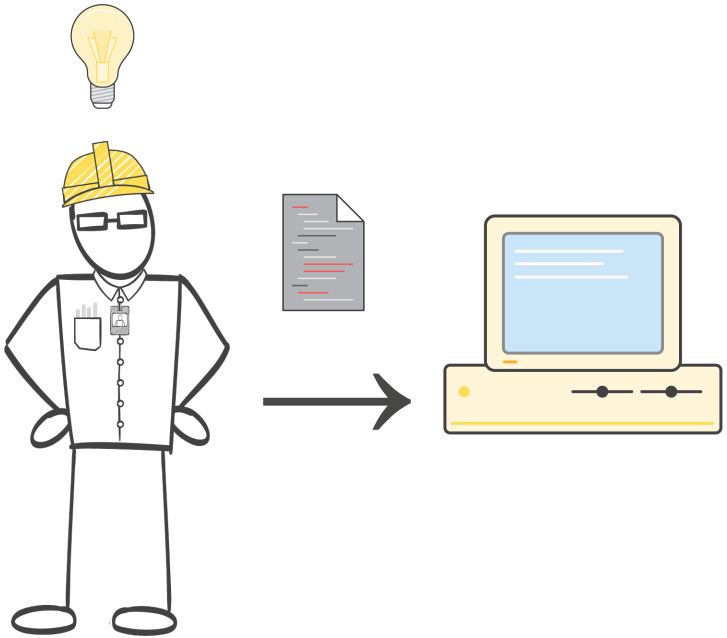
Machine Learning on AWS with Amazon SageMaker

Constantin Gonzalez
Principal Solutions Architect, Amazon Web Services
glez@amazon.de

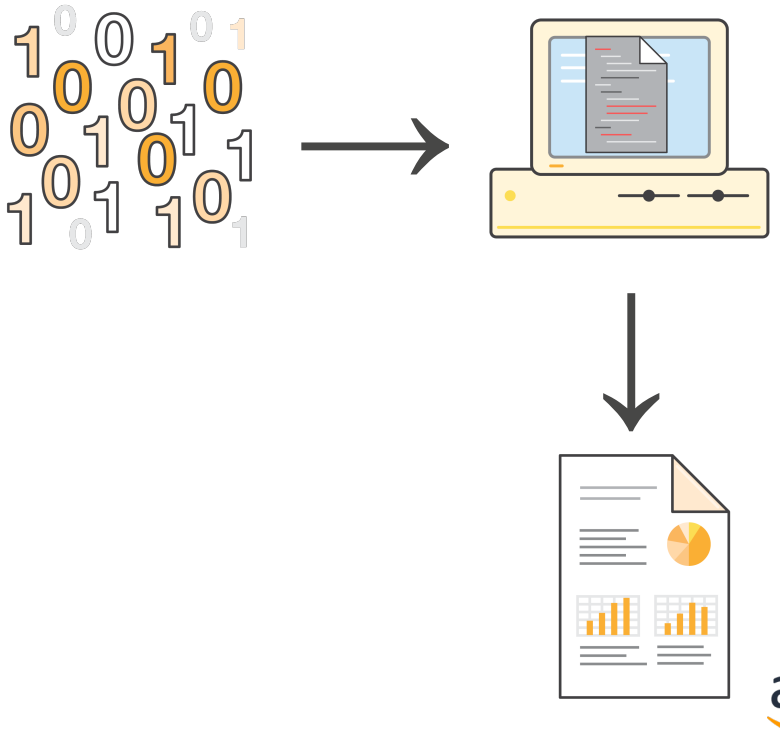
December 2017

Computer Programming (1936 – Today)

1.

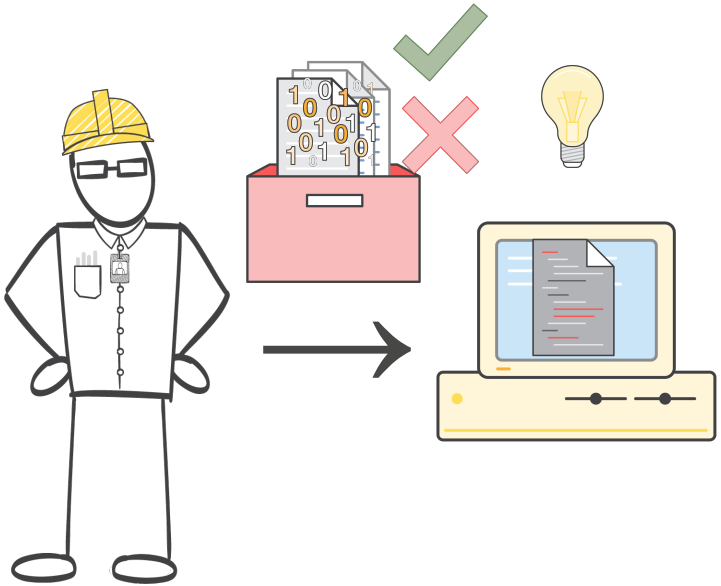


2.

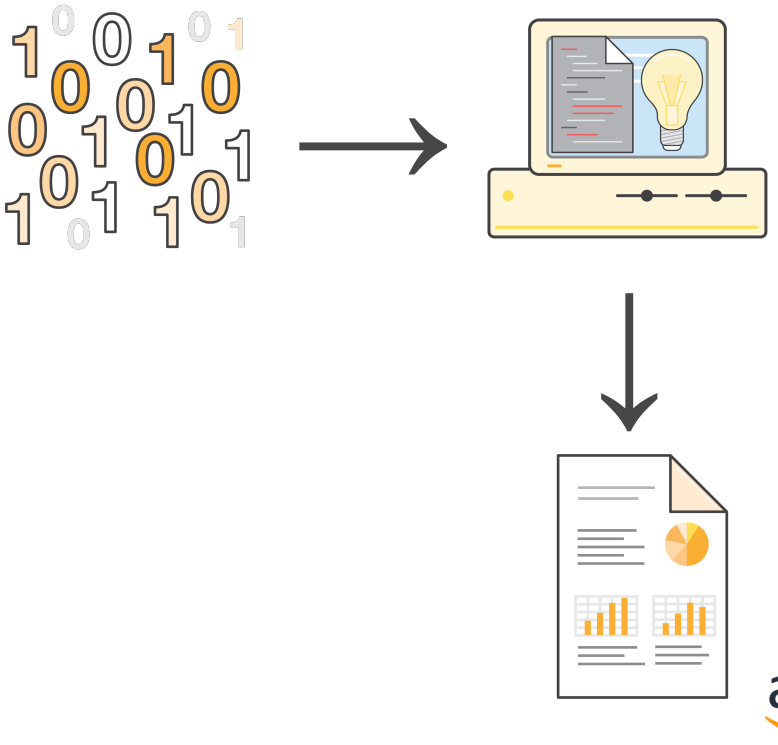


Machine Learning (1959 – Today)

1.



2.

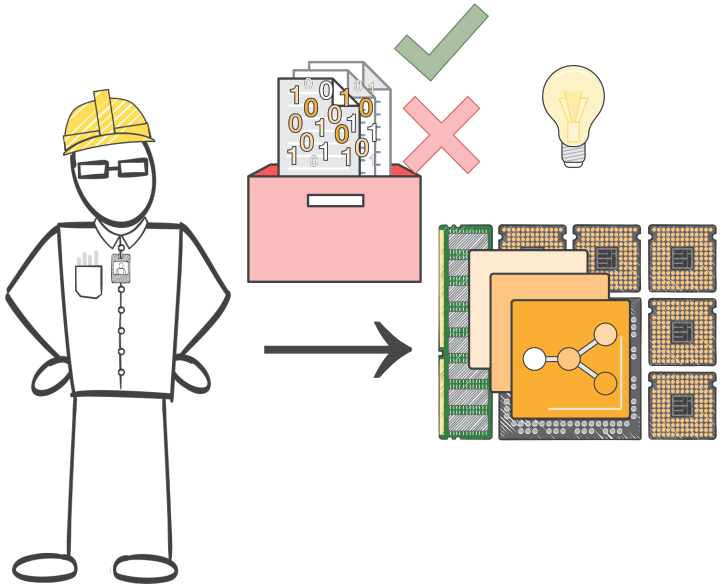


Machine Learning At Amazon (1995)

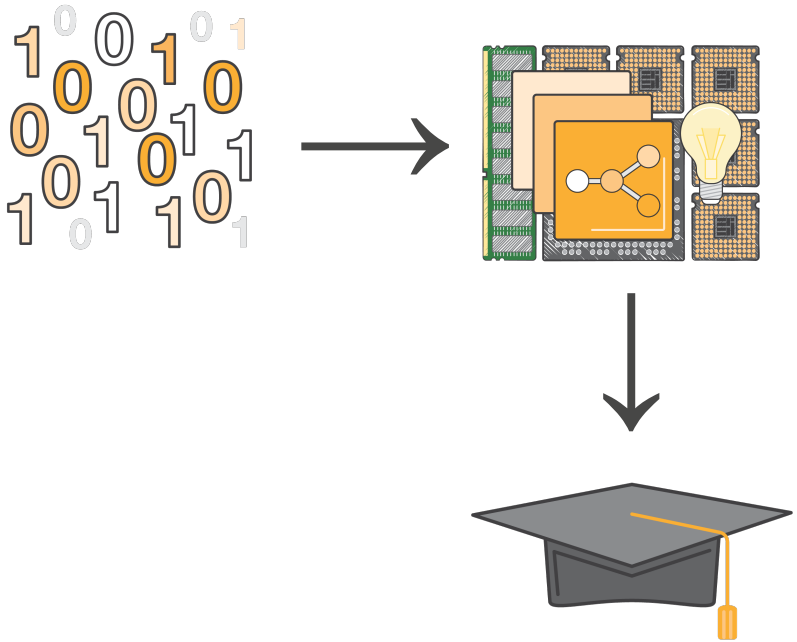
aws

Deep Learning (1986 – Today)

1.



2.

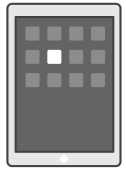


Artificial Intelligence At Amazon

Thousands Of Employees Across The Company Focused on AI

Discovery & Search

Fulfilment & Logistics



Enhance Existing Products

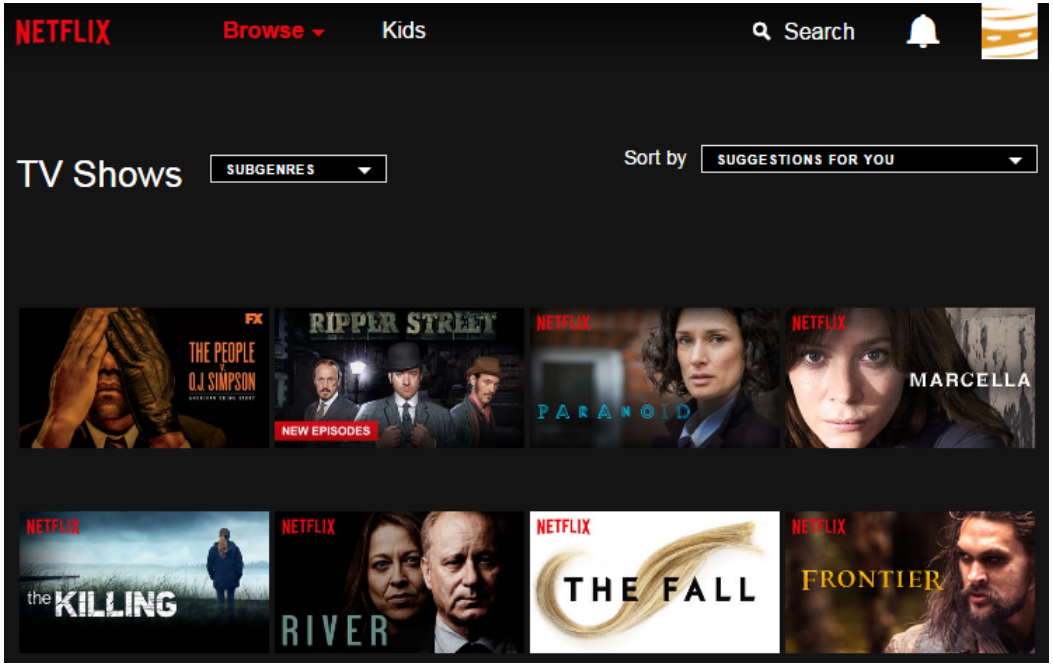
Define New Product Categories

Bring Machine Learning To All

aws

AWS Customers using AI

Netflix Recommendation Engine



Pinterest Lens

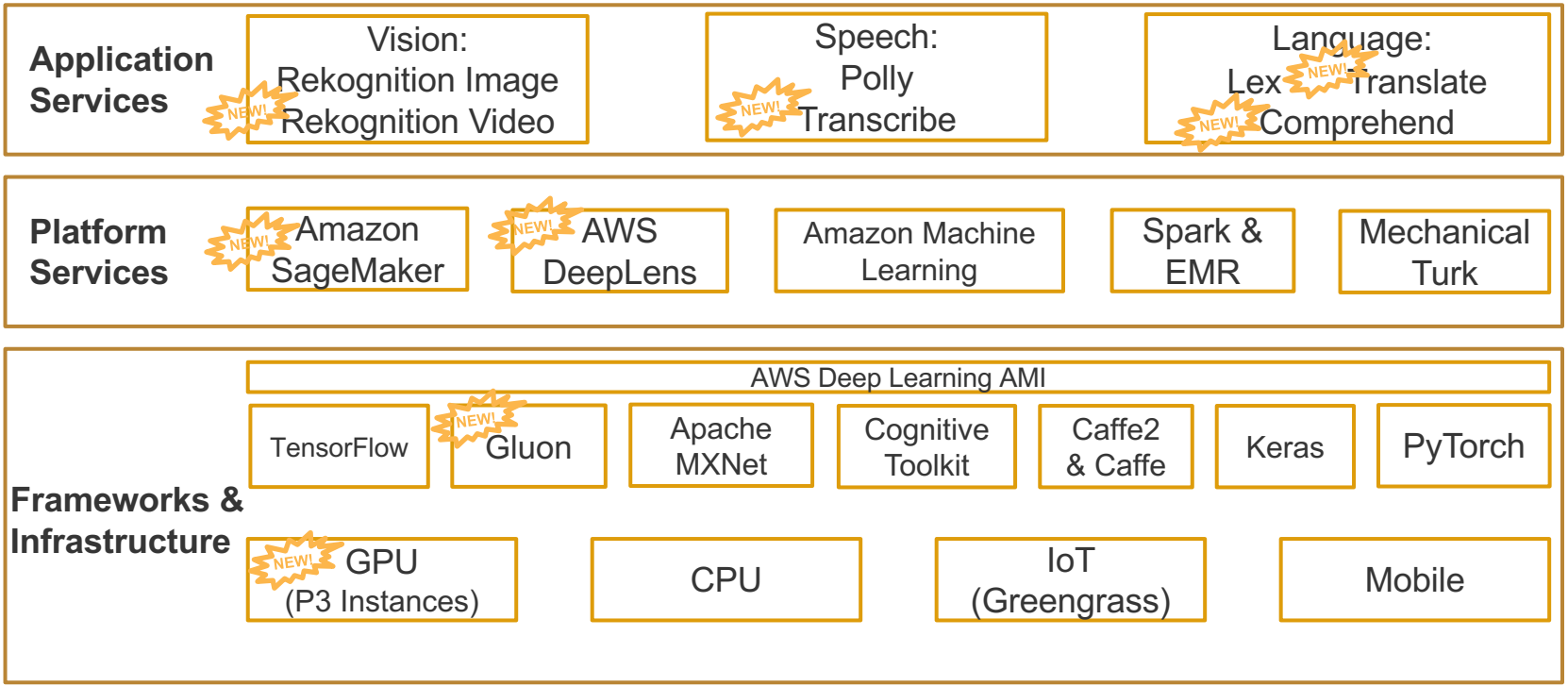
Grow your own Strawberries 36

Chocolate Strawberry Waffle Ball 24

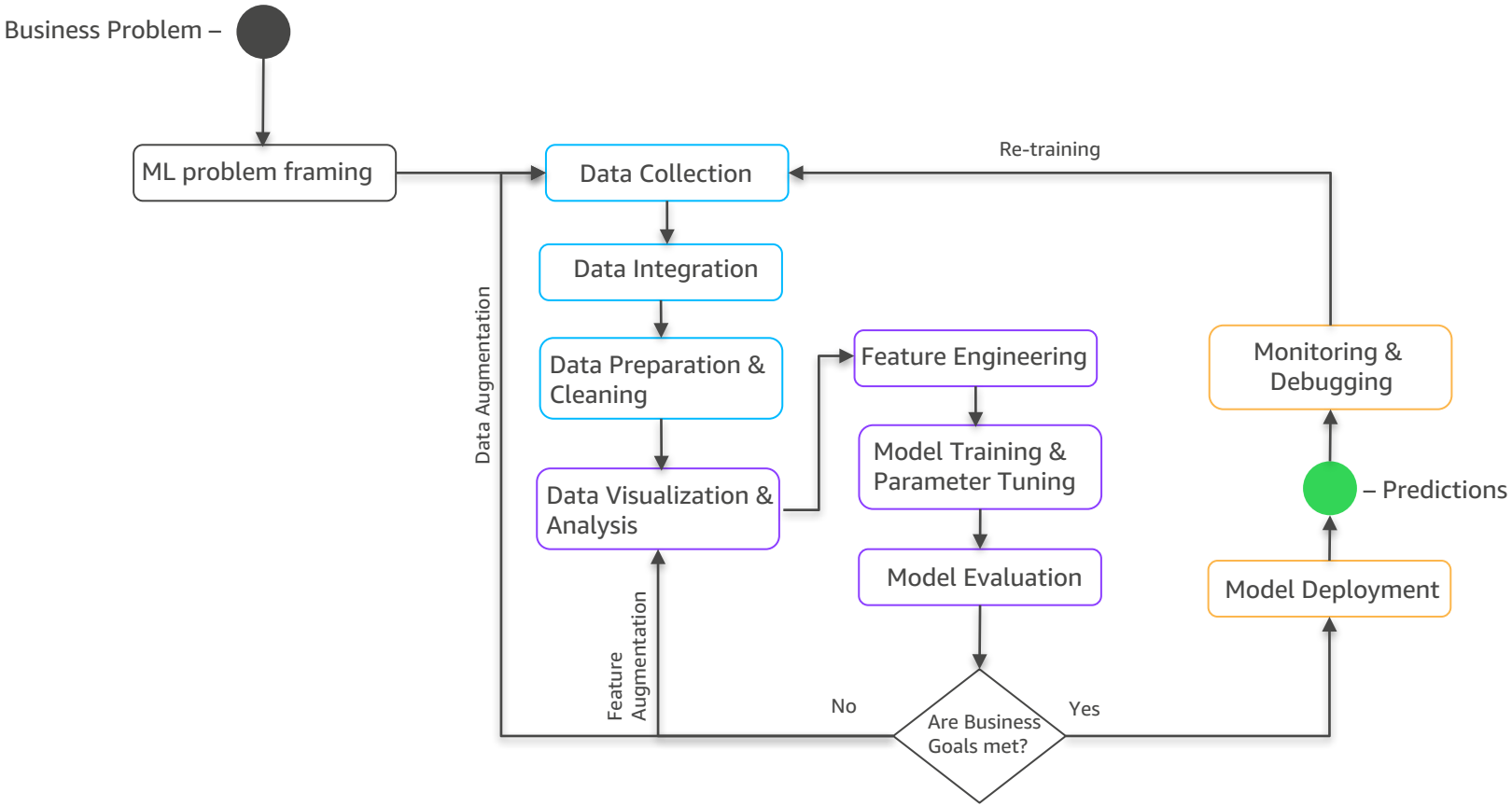
Adam Barton Delights

aws

AWS ML Stack

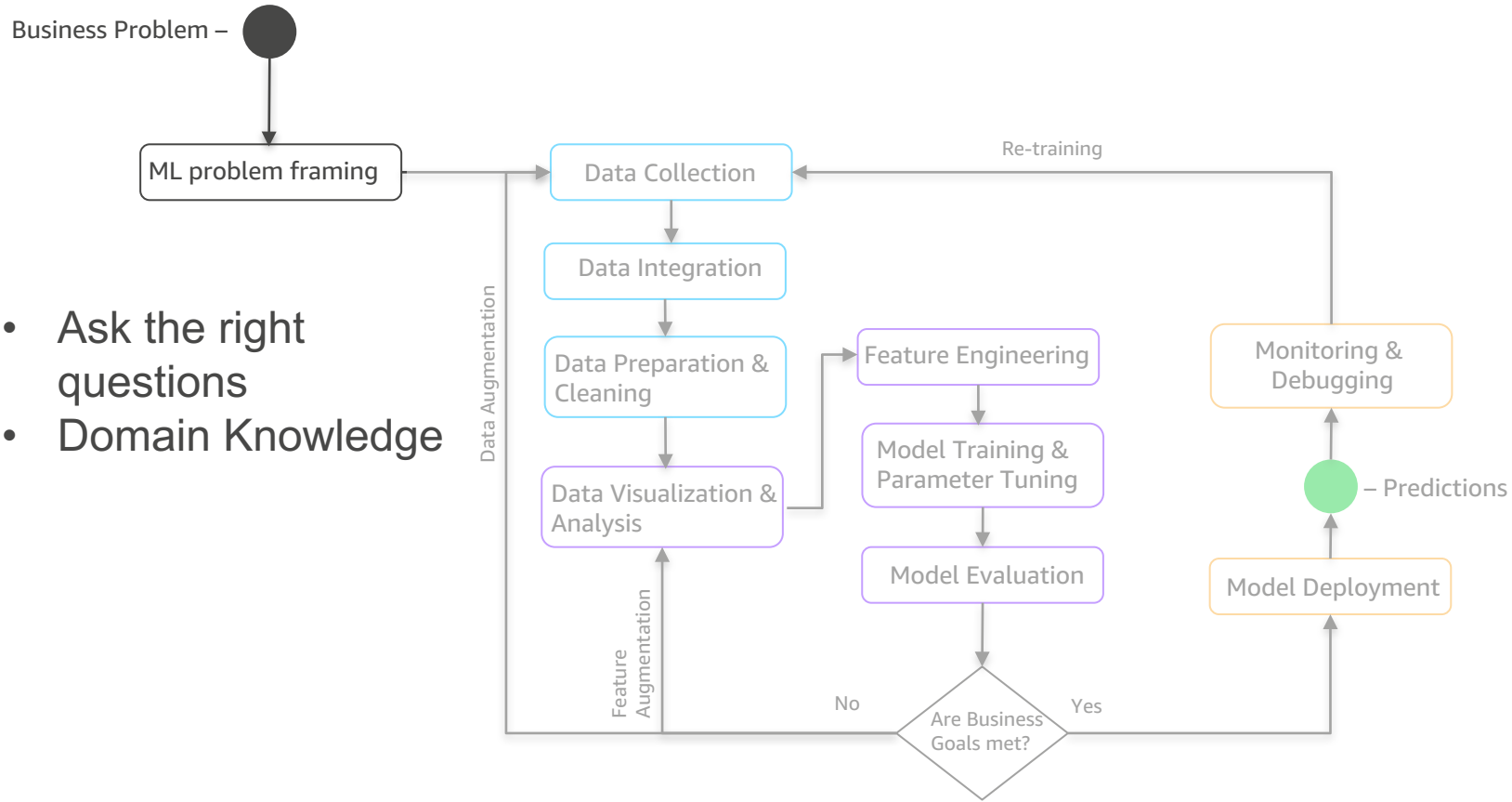

Application Services
Vision:
Rekognition Image
Rekognition Video
Speech:
Polly
Transcribe
Language:
Lex
Translate
Comprehend
NEW!
Platform Services
Amazon SageMaker
AWS DeepLens
Amazon Machine Learning
Spark & EMR
Mechanical Turk
Frameworks & Infrastructure
AWS Deep Learning AMI
TensorFlow
Gluon
Apache MXNet
Cognitive Toolkit
Caffe2 & Caffe
Keras
PyTorch
GPU
(P3 Instances)
CPU
IoT
(Greengrass)
Mobile

ML Process


Business Problem –
ML problem framing
Data Collection
Re-training
Data Integration
Data Augmentation
Data Preparation & Cleaning
Feature Engineering
Monitoring & Debugging
Data Visualization & Analysis
Model Training & Parameter Tuning
– Predictions
Model Evaluation
Model Deployment
Feature Augmentation
No
Are Business Goals met?
Yes

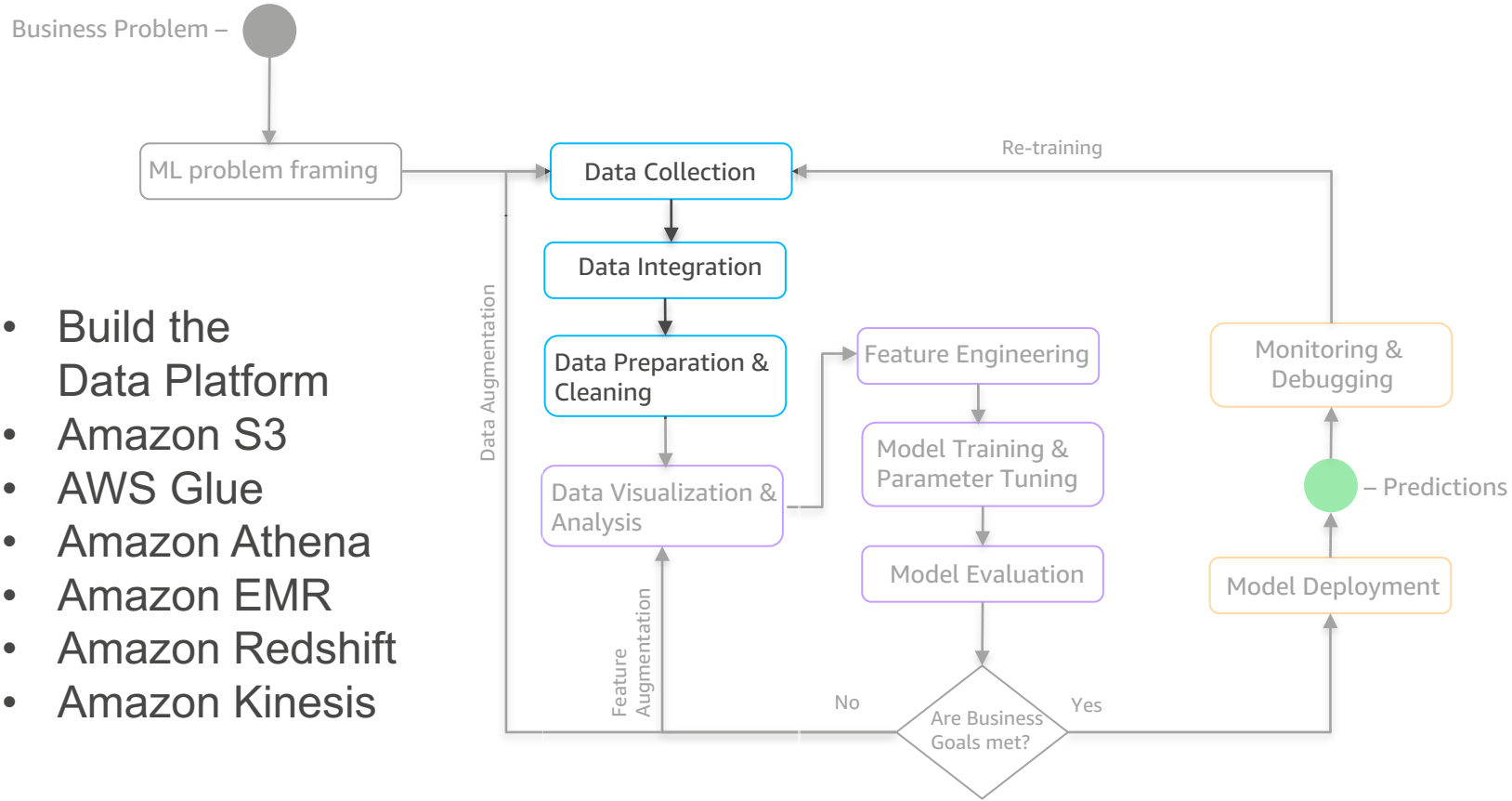
ML Process: Discovery

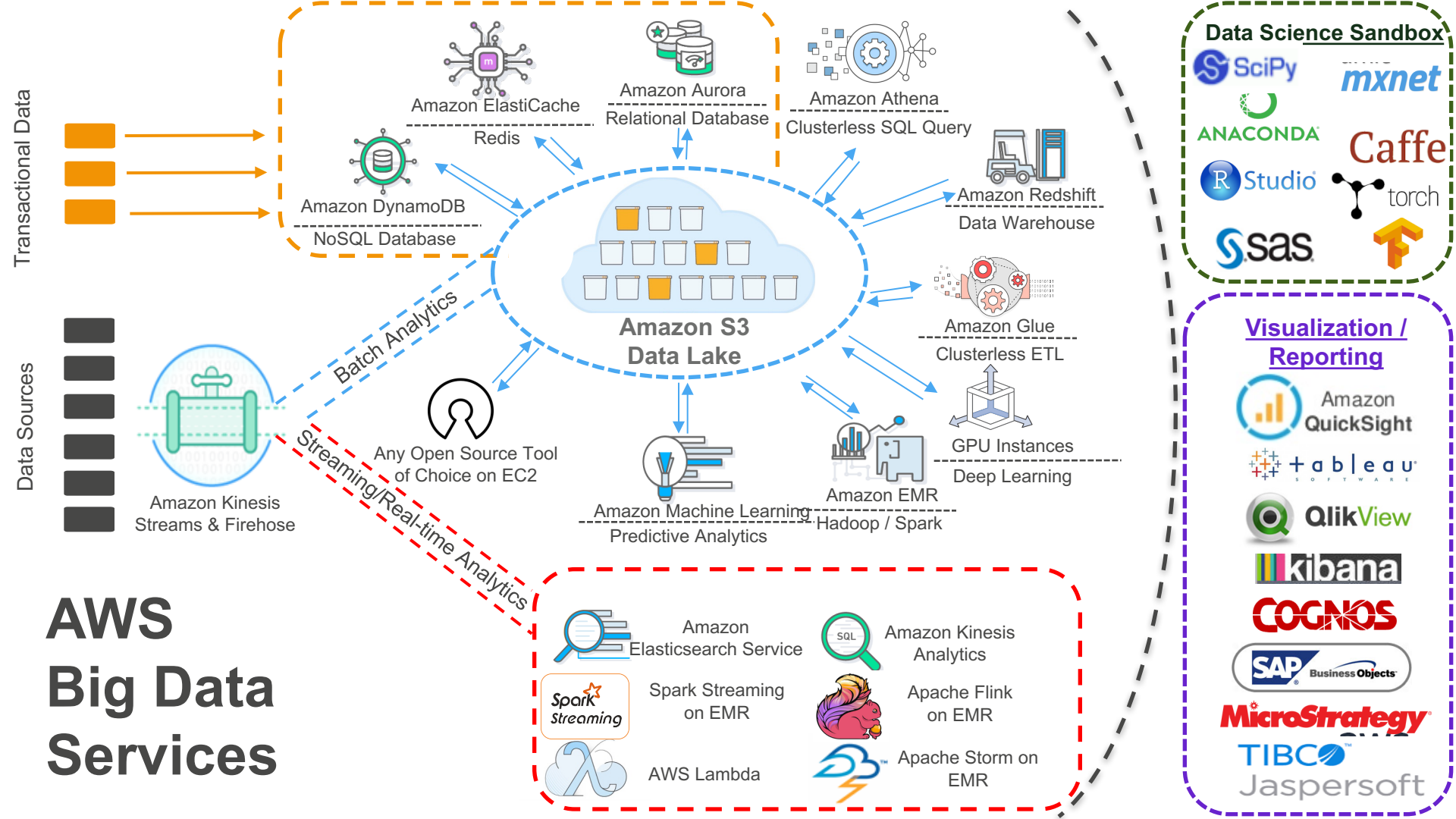
- Ask the right questions
- Domain Knowledge



ML Process: Integration – Data Architecture

- Build the Data Platform
- Amazon S3
- AWS Glue
- Amazon Athena
- Amazon EMR
- Amazon Redshift
- Amazon Kinesis

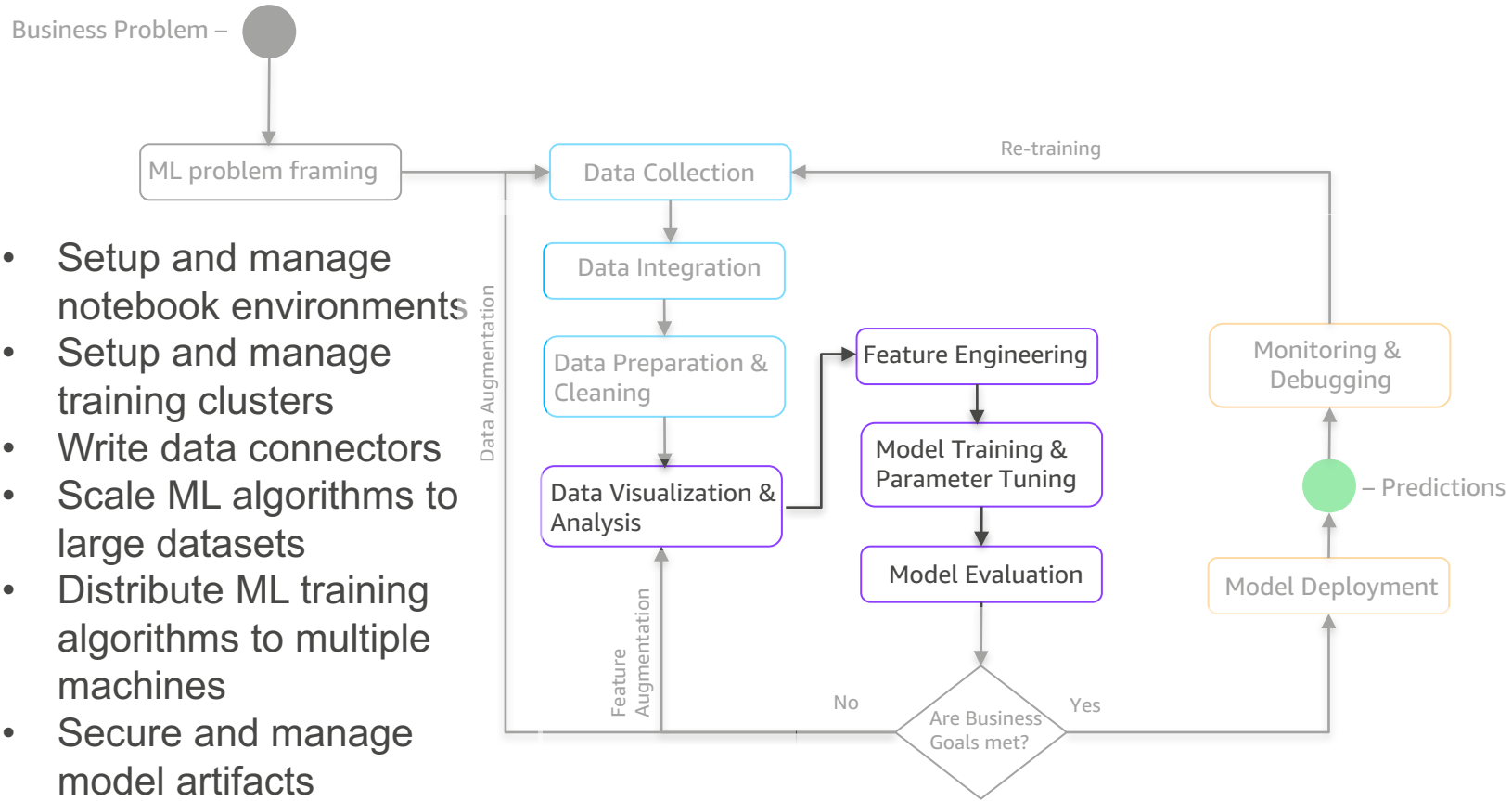




AWS Big Data Services

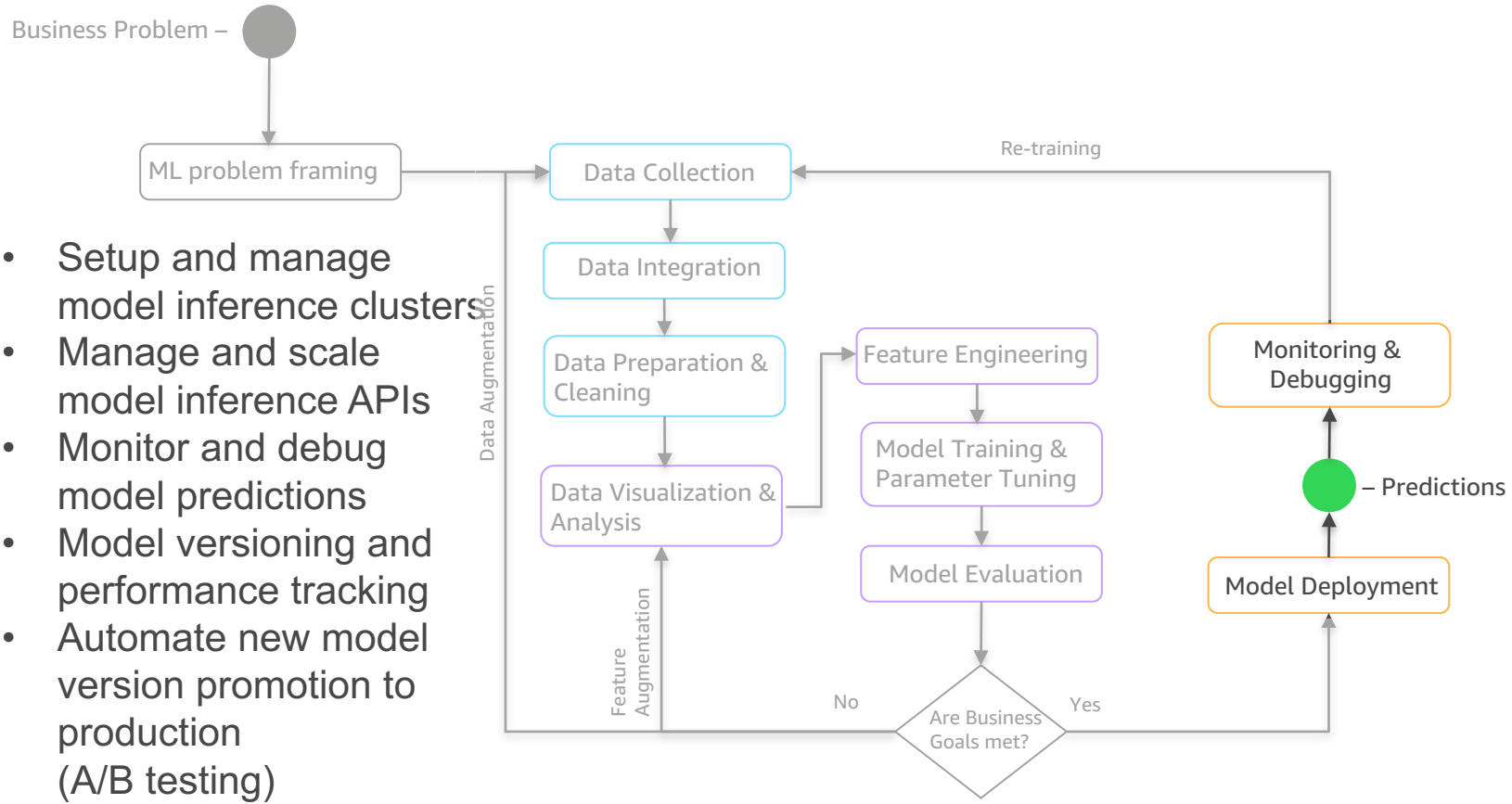
ML Process: Model Training

- Setup and manage notebook environments
- Setup and manage training clusters
- Write data connectors
- Scale ML algorithms to large datasets
- Distribute ML training algorithms to multiple machines
- Secure and manage model artifacts



ML Process: Model Deployment

- Setup and manage model inference clusters
- Manage and scale model inference APIs
- Monitor and debug model predictions
- Model versioning and performance tracking
- Automate new model version promotion to production (A/B testing)

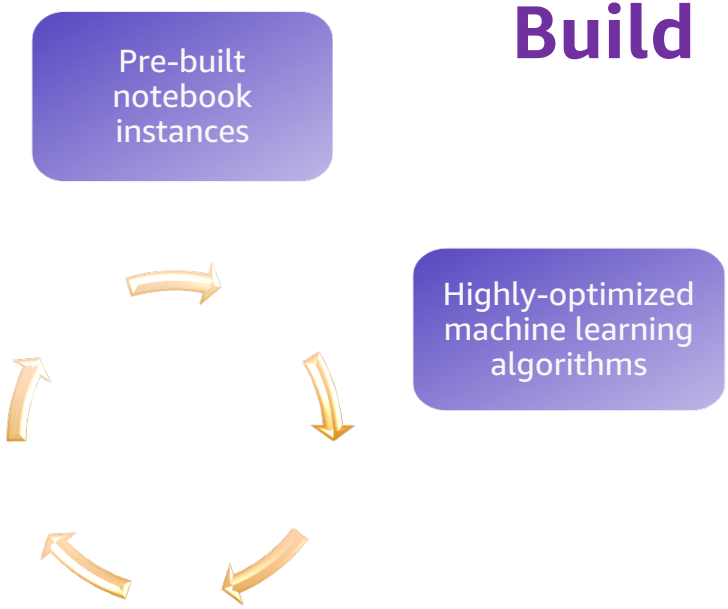


Amazon SageMaker

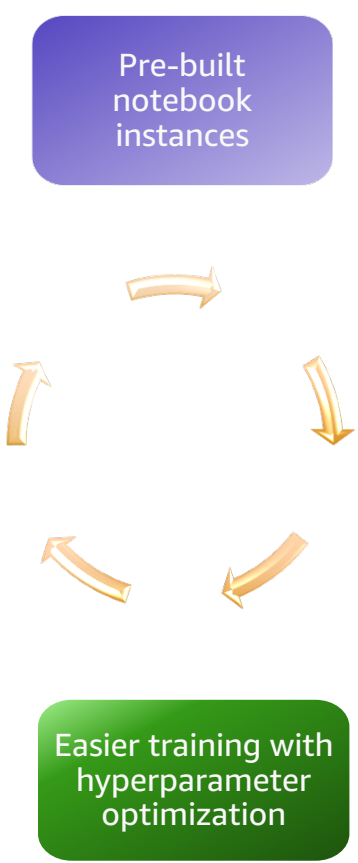
A **fully managed service** that enables **data scientists** and **developers** to quickly and easily **build** machine-learning based models **into production** smart applications.

aws

Amazon SageMaker



Amazon SageMaker



Build

Highly-optimized machine learning algorithms

scikit learn

mxnet

GLUON

Train

One-click training for ML, DL, and custom algorithms

TensorFlow

PYTORCH

aws

Amazon SageMaker

Build

Pre-built notebook instances

Highly-optimized machine learning algorithms

scikit learn

mxnet

GLUON

Train

One-click training for ML, DL, and custom algorithms

TensorFlow

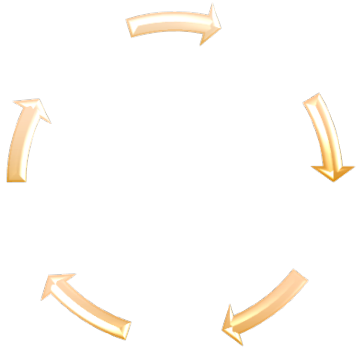
PYTORCH

Easier training with hyperparameter optimization

Deploy

Deployment without engineering effort

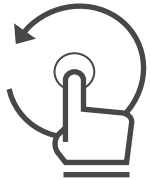
Fully-managed hosting at scale



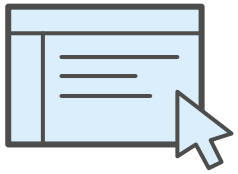
aws

Amazon SageMaker

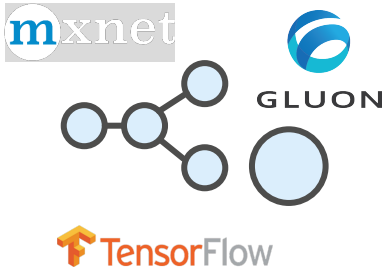
Build, train, and deploy machine learning models at scale



End-to-End Machine Learning Platform



Zero setup

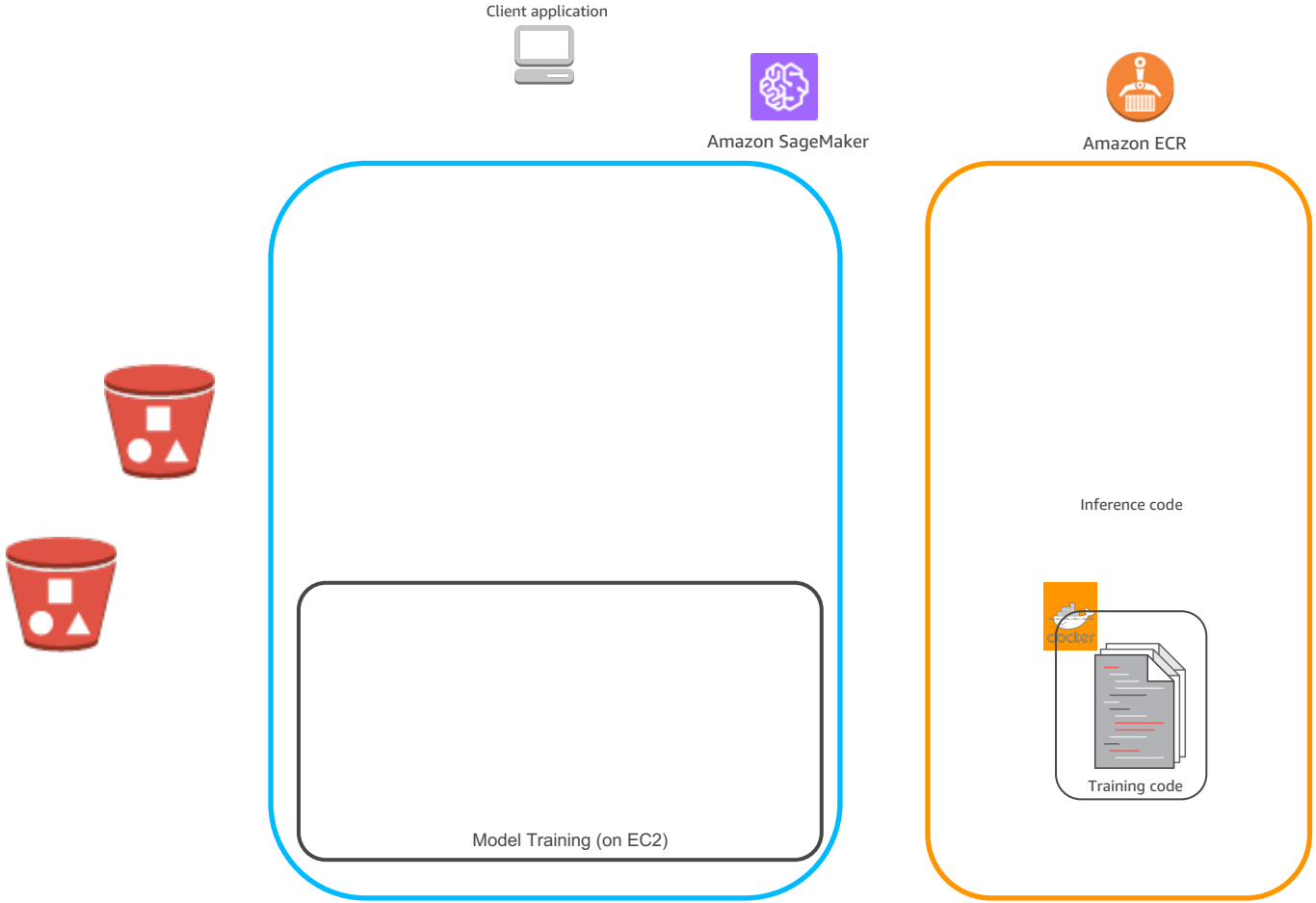


Flexible Model Training

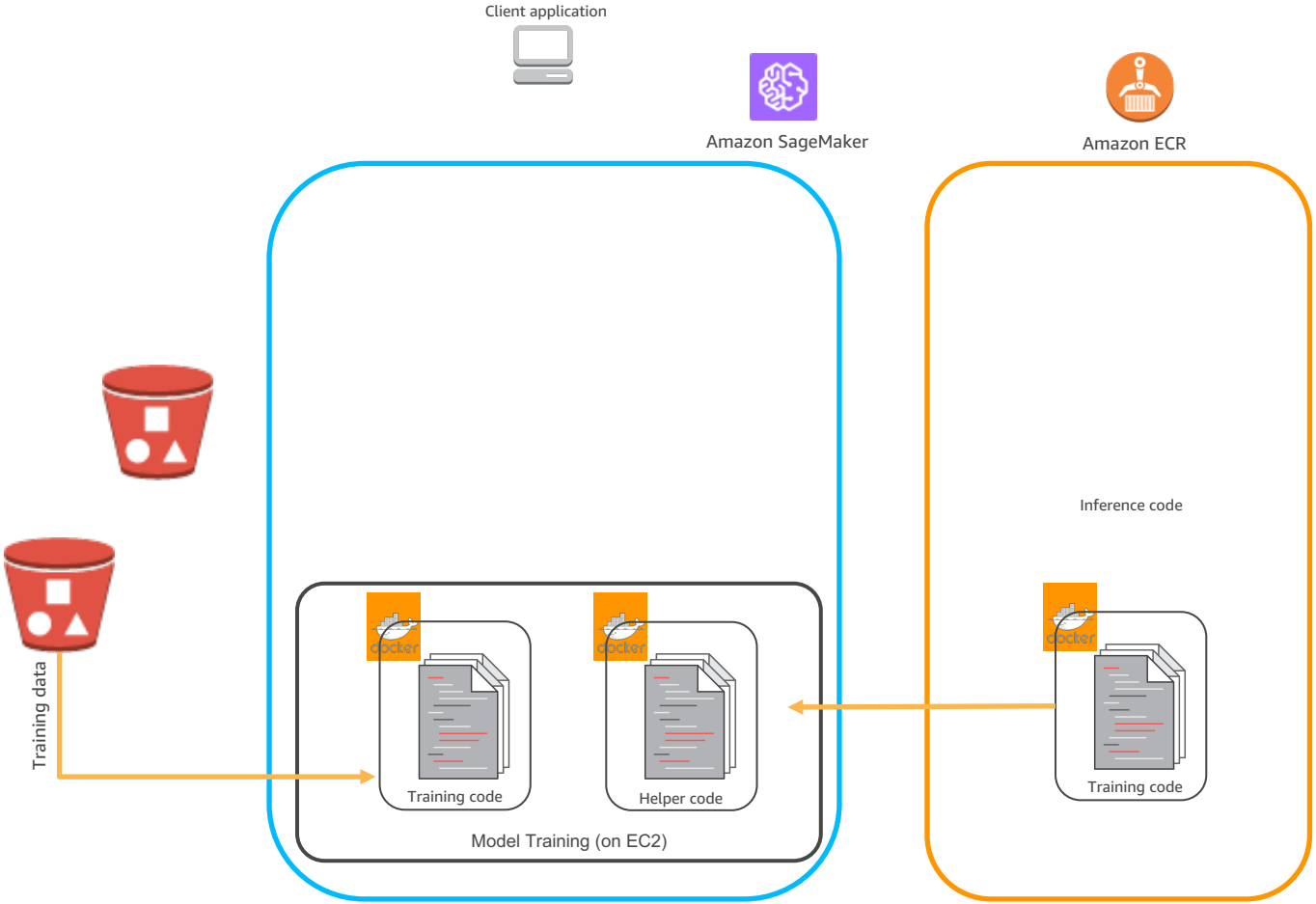
Pay by the second

aws

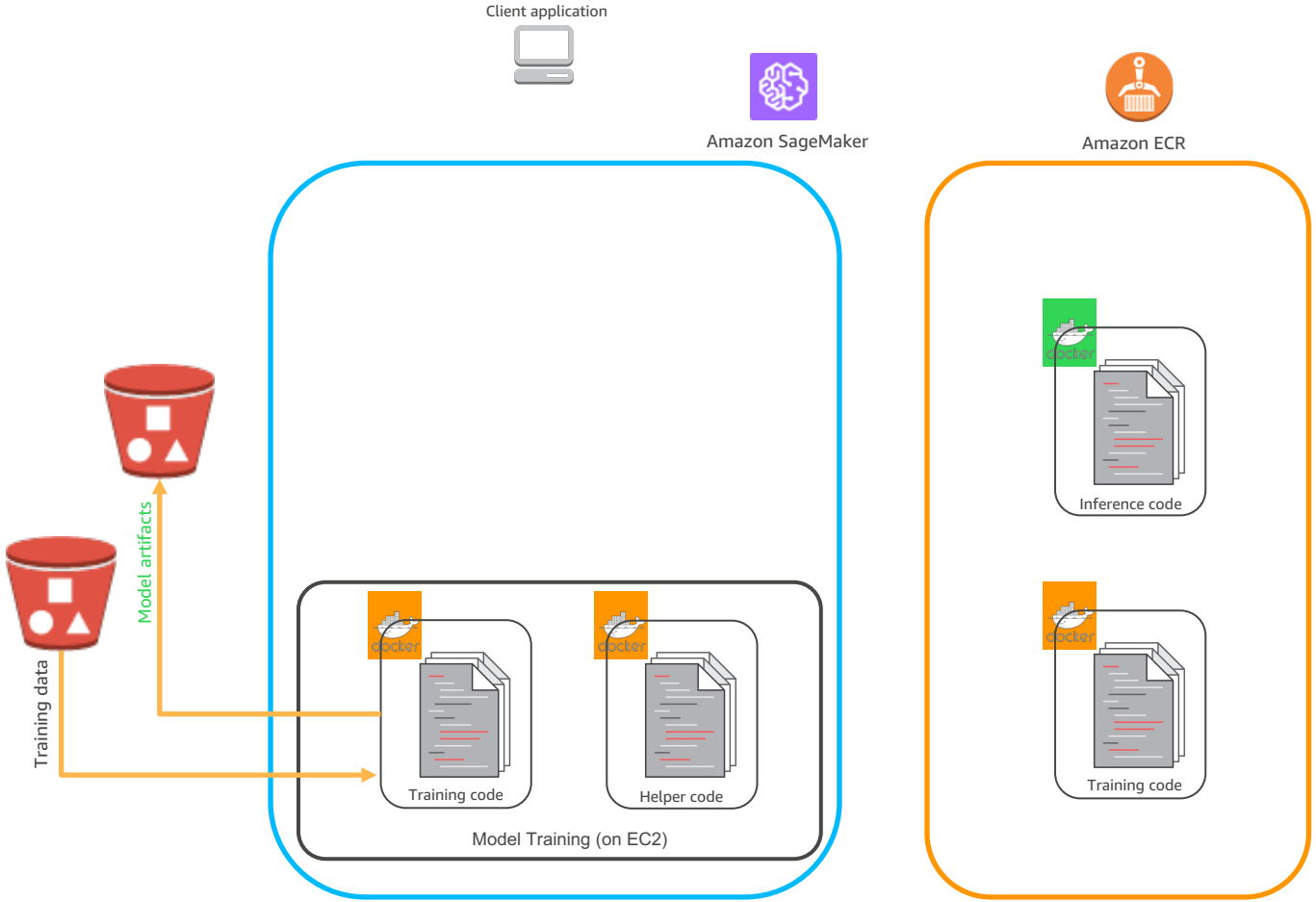
Behind the scenes



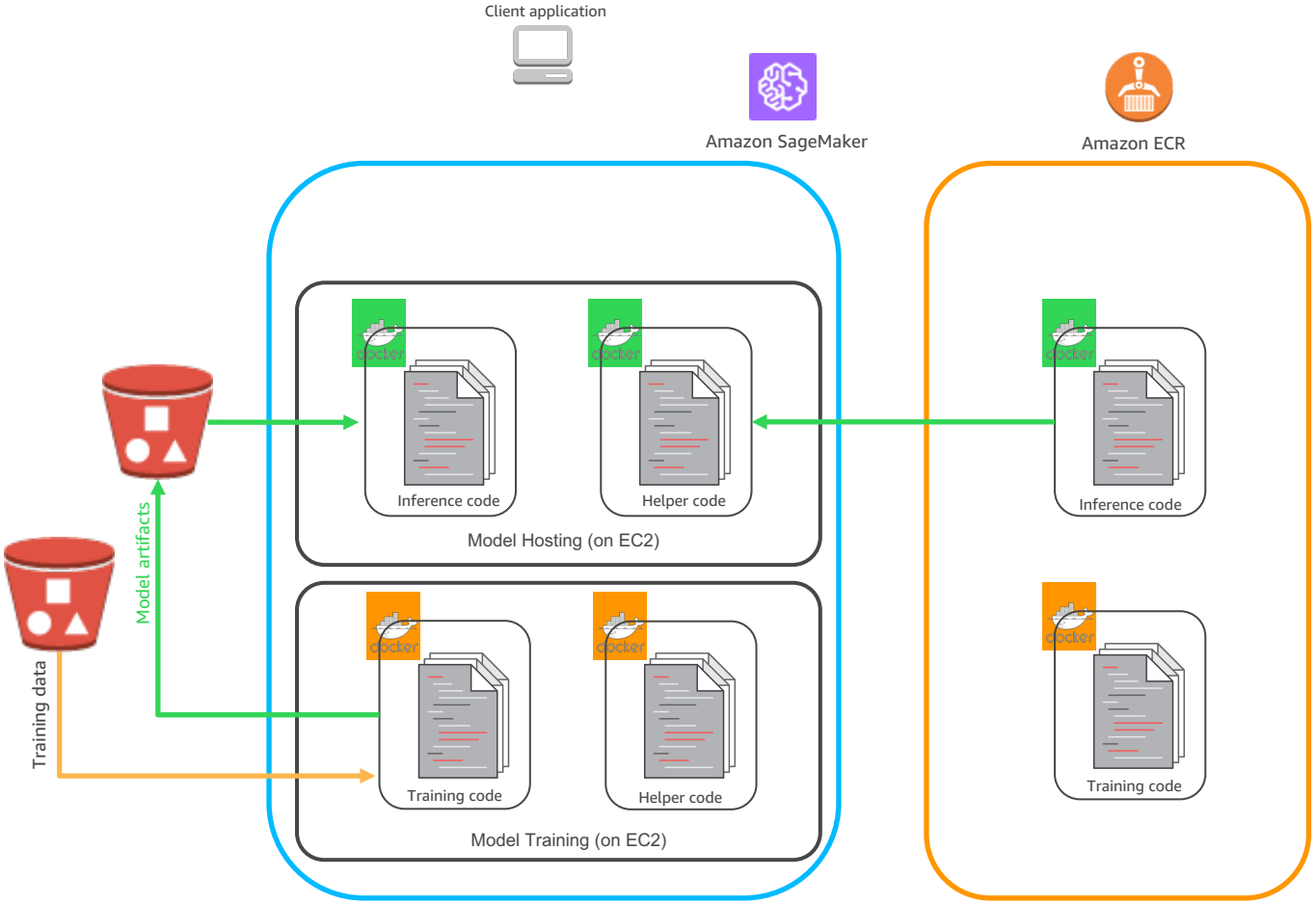
Behind the scenes



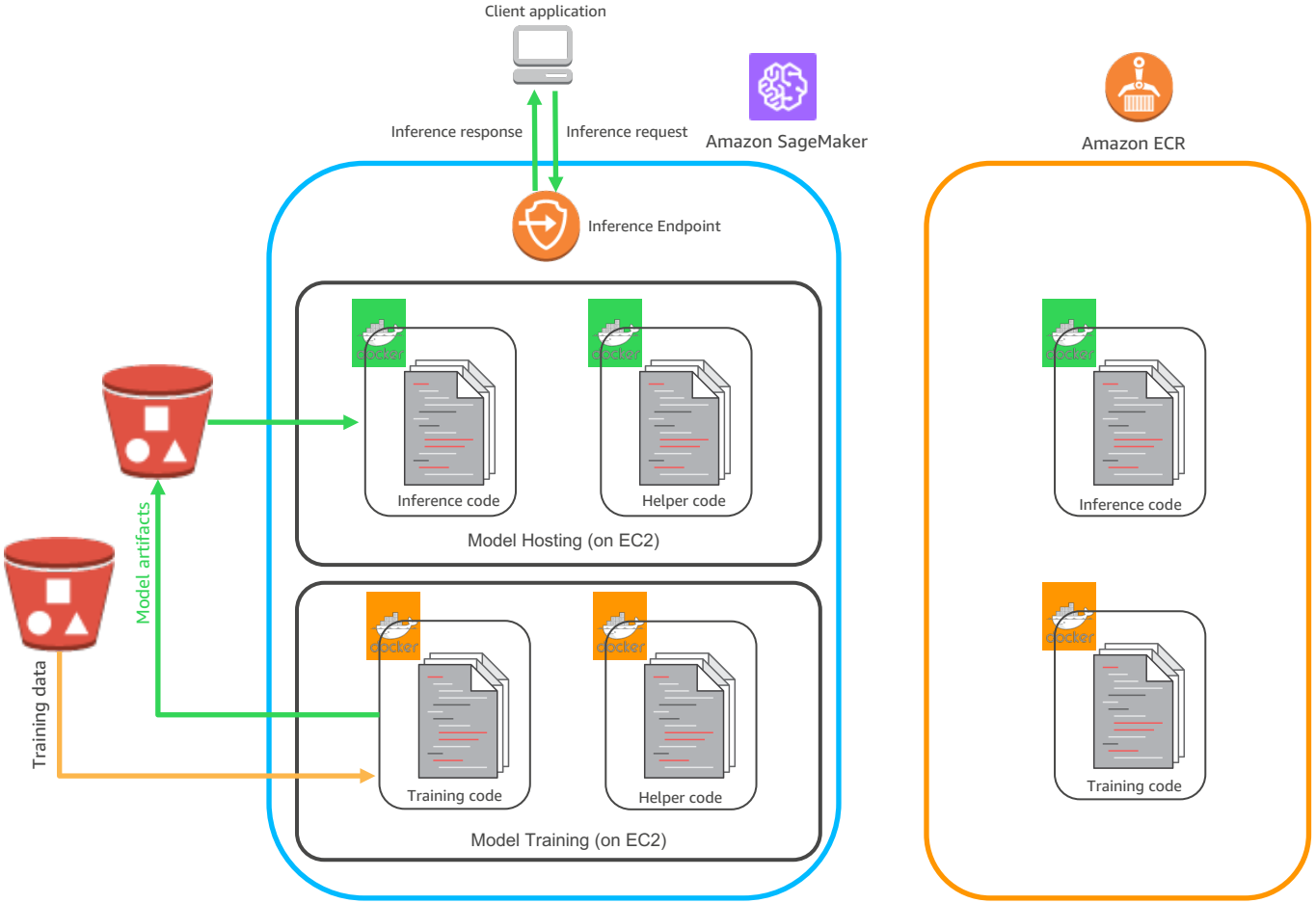
Behind the scenes



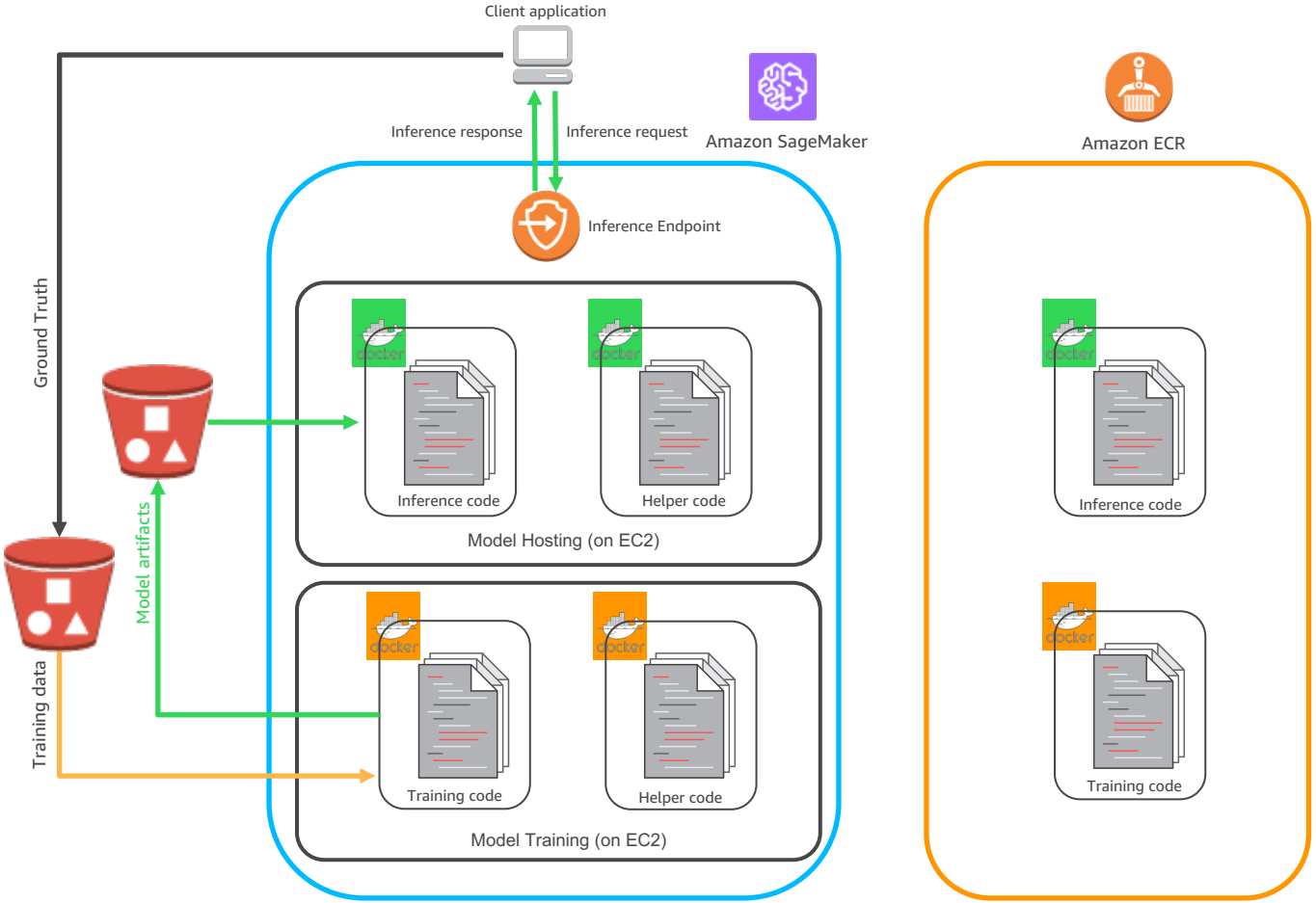
Behind the scenes



Behind the scenes

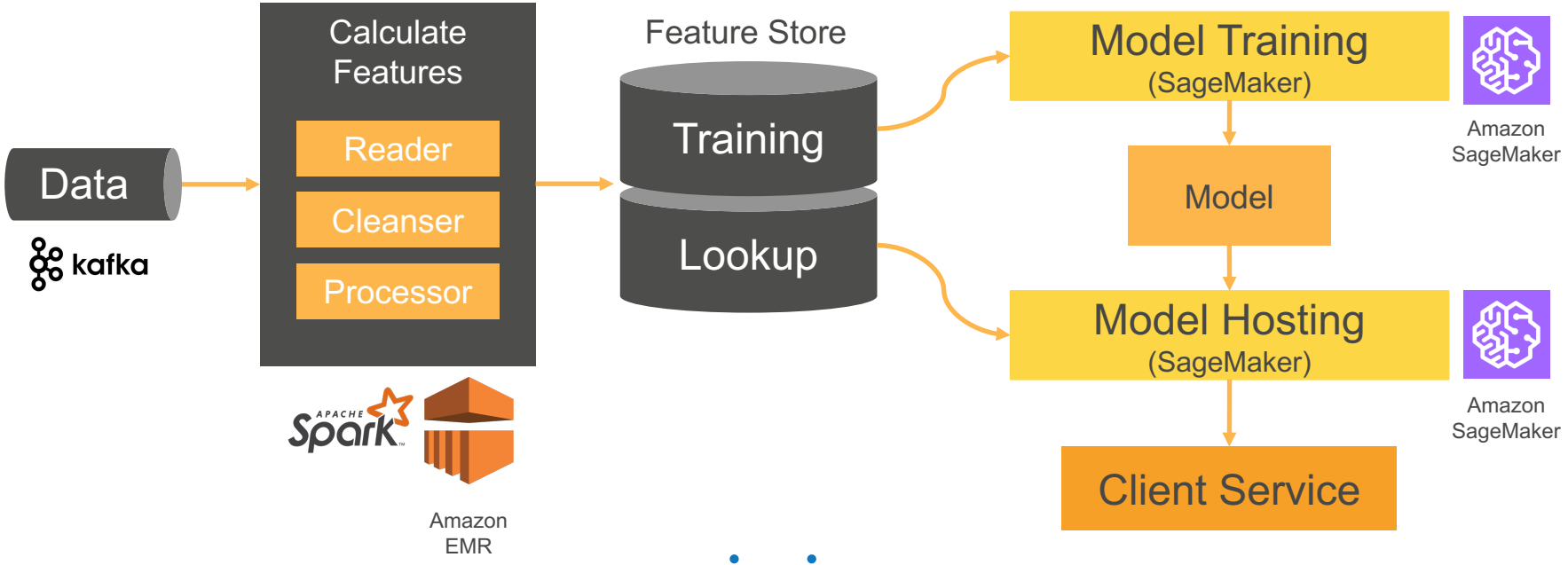


Behind the scenes



Customer Example: Intuit

Near real-time fraud detection in AWS using Amazon SageMaker

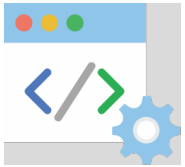


intuit

aws

Amazon SageMaker

1

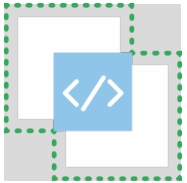


Notebook Instances

2

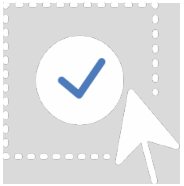
Algorithms

3



ML Training Service

4



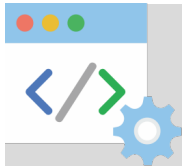
ML Hosting Service

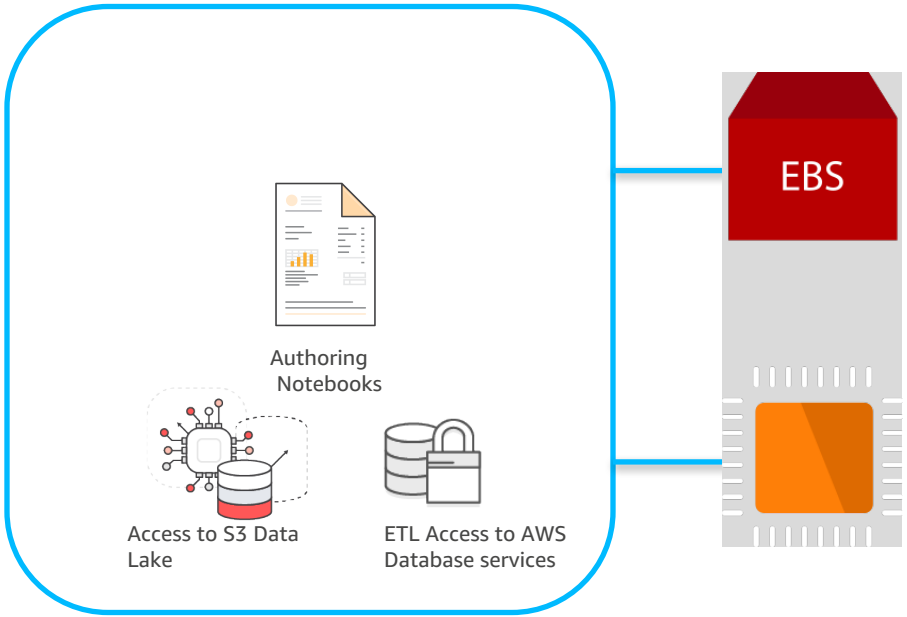
aws

Notebook Instances

1

Zero Setup For Exploratory Data Analysis





"Just add data"

- Recommendations/Personalization
- Fraud Detection
- Forecasting
- Image Classification
- Churn Prediction
- Marketing Email/Campaign Targeting
- Log processing and anomaly detection
- Speech to Text
- More…

aws

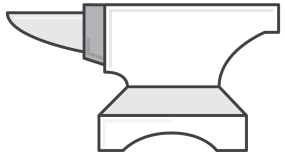
Algorithms

2

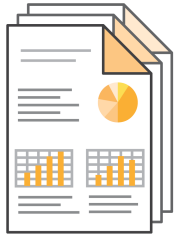
Amazon SageMaker: 10x better algorithms

Streaming datasets, for cheaper training

Train faster, in a single pass



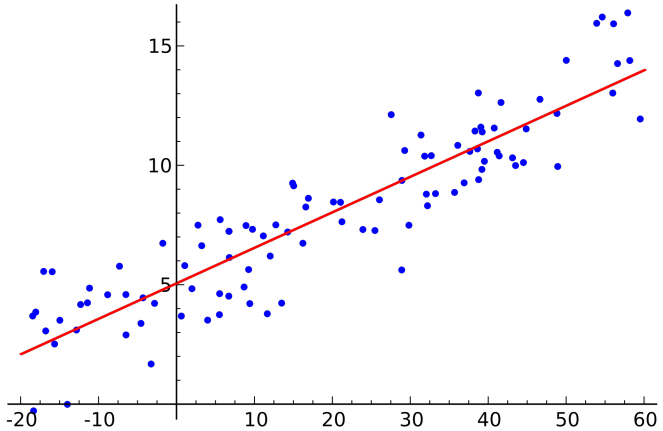
Greater reliability on extremely large datasets



Choice of several ML algorithms

Linear Learner

- Find linear equation that best approximates the data
- Supervised
- Supports:
  - Binary classification
  - Multiclass classification
  - Linear regression
  - Floating-point or test
  - CSV or recordIO-protobuf
- Parallel training of multiple models with automatic hyperparameter optimization

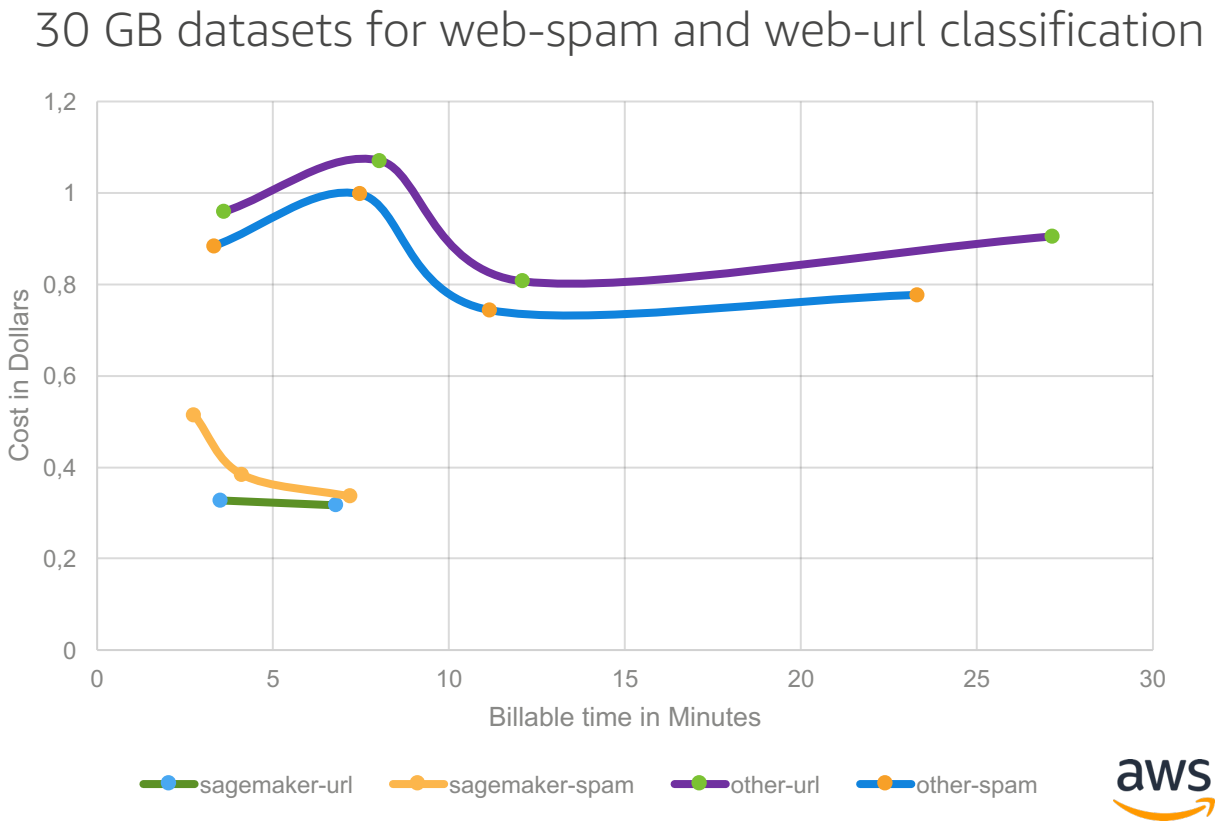


aws

Linear Learner

Regression (mean squared error)	
SageMaker	Other
1.02	1.06
1.09	**1.02**
0.332	**0.183**
0.086	0.129
83.3	84.5

Classification (F1 Score)	
SageMaker	Other
0.980	**0.981**
0.870	**0.930**
0.997	**0.997**
0.978	0.964
0.914	0.859
0.470	**0.472**
0.903	**0.908**
0.508	0.508



Factorization Machines

- Extends linear model to pair-wise interactions between features
- Good for high dimensional sparse datasets, e.g.:
  - Click prediction
  - Item recommendation systems
- Supports binary classification or linear regression
- recordIO-protobuf data format

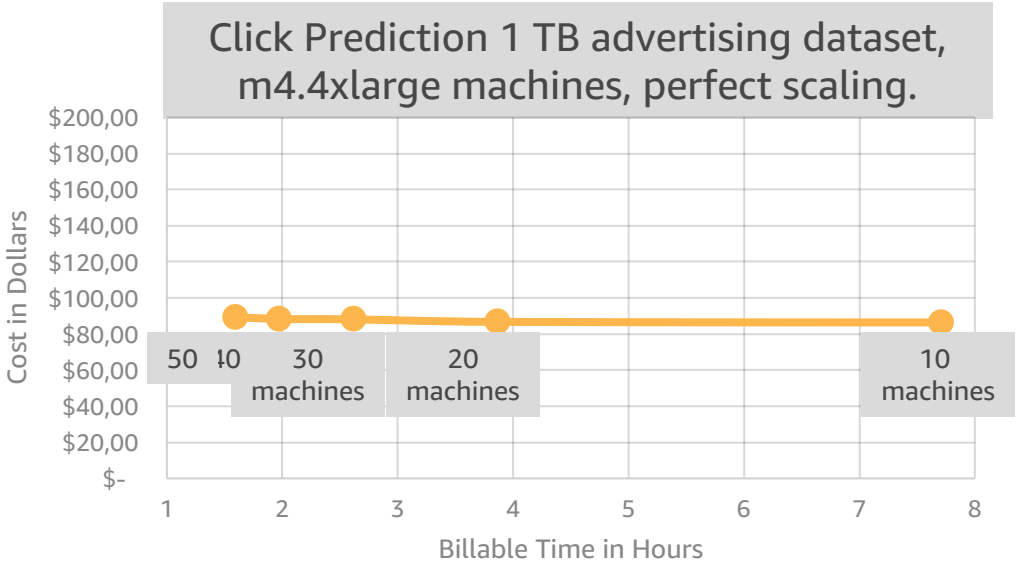
$$\tilde{y} = w_0 + \langle w_1, x \rangle + \sum_{i,j>i} x_i x_j \cdot \langle v_i, v_j \rangle$$

aws

Factorization Machines

$$\tilde{y} = w_0 + \langle w_1, x \rangle + \sum_{i,j>i} x_i x_j \cdot \langle v_i, v_j \rangle$$

	Log_loss	F1 Score	**Seconds**
SageMaker	0.494	0.277	820
Other (10 Iter)	0.516	0.190	650
Other (20 Iter)	0.507	0.254	1300
Other (50 Iter)	0.481	0.313	3250



K-Means Clustering

- Cluster data into k groups
- Unsupervised
- Algorithm:
    1. Start with K= k*x cluster centers
    2. Iterate over cluster centers in mini-batches and adjust them based on training data
    3. Reduce K cluster centers to k final clusters by using the same algorithm.

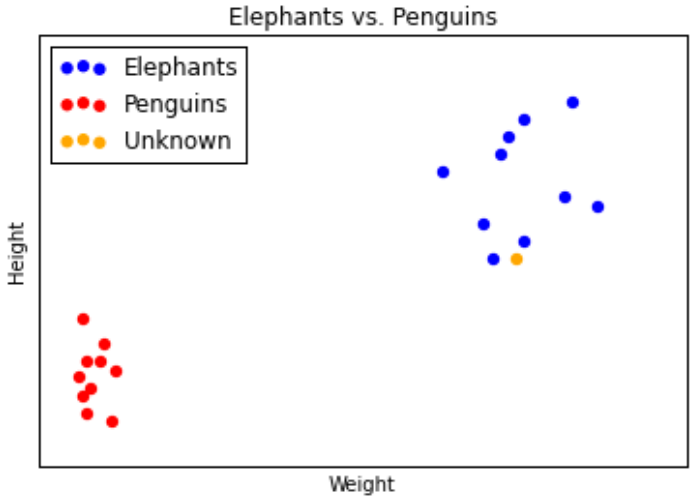
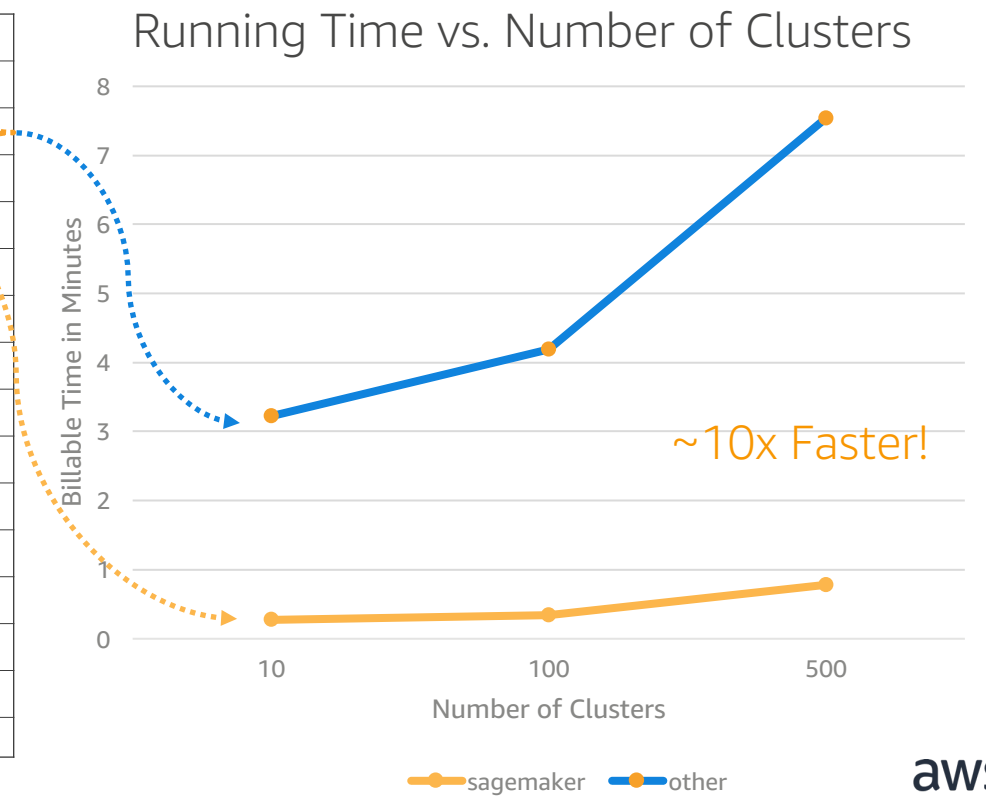


Image: johnloeber.com

aws

K-Means Clustering

	k	SageMaker	Other
Text 1.2GB	10	1.18E3	1.18E3
	100	1.00E3	9.77E2
	500	9.18.E2	9.03E2
Images 9GB	10	3.29E2	3.28E2
	100	2.72E2	2.71E2
	500	2.17E2	Failed
Videos 27GB	10	2.19E2	2.18E2
	100	2.03E2	2.02E2
	500	1.86E2	1.85E2
Advertising 127GB	10	1.72E7	Failed
	100	1.30E7	Failed
	500	1.03E7	Failed
Synthetic 1100GB	10	3.81E7	Failed
	100	3.51E7	Failed
	500	2.81E7	Failed



Principal Component Analysis (PCA)

- Reduce dimensionality while retaining as much information as possible
- Finds new components, sorted by information value.
- Unsupervised
- Regular (for sparse) and Randomized (for large, dense datasets) modes

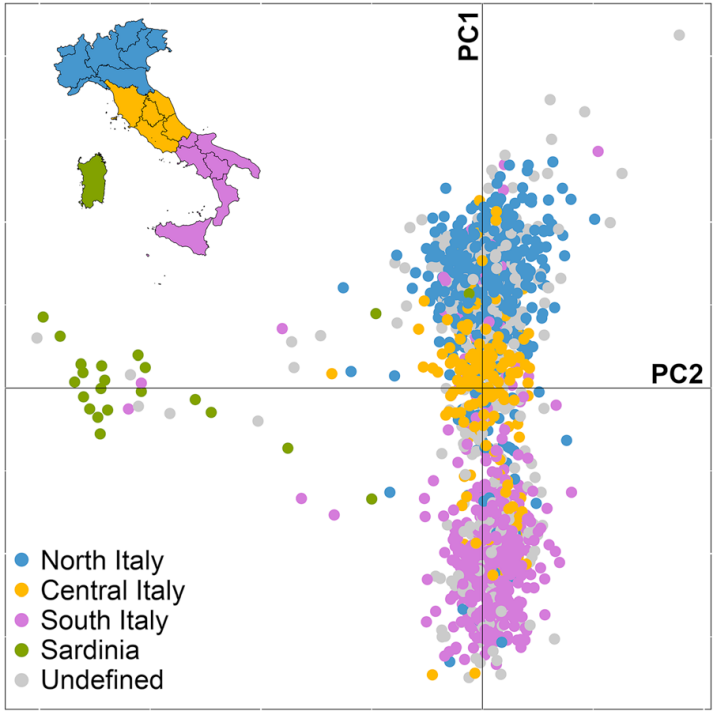
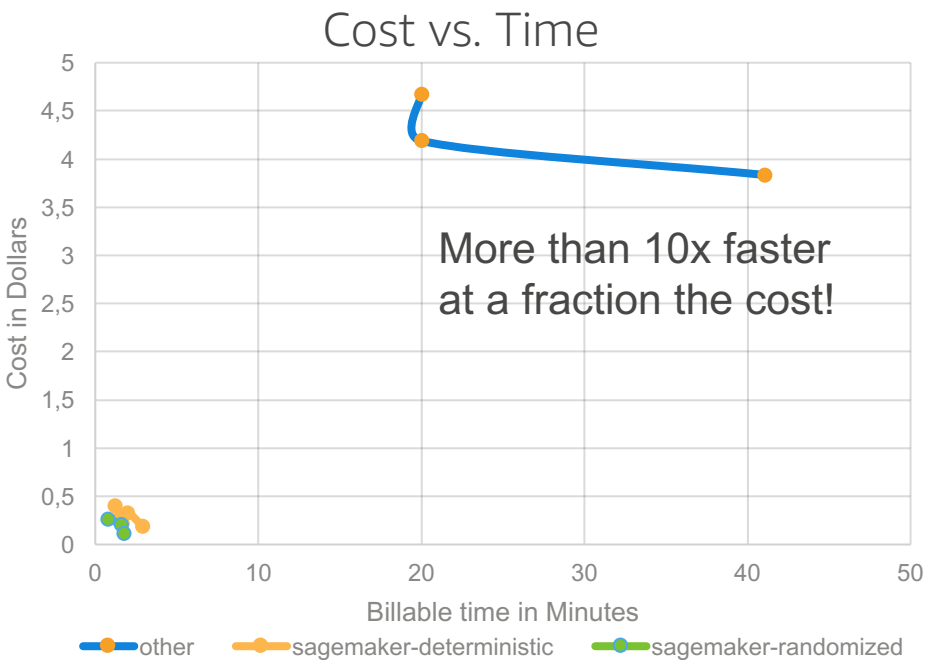
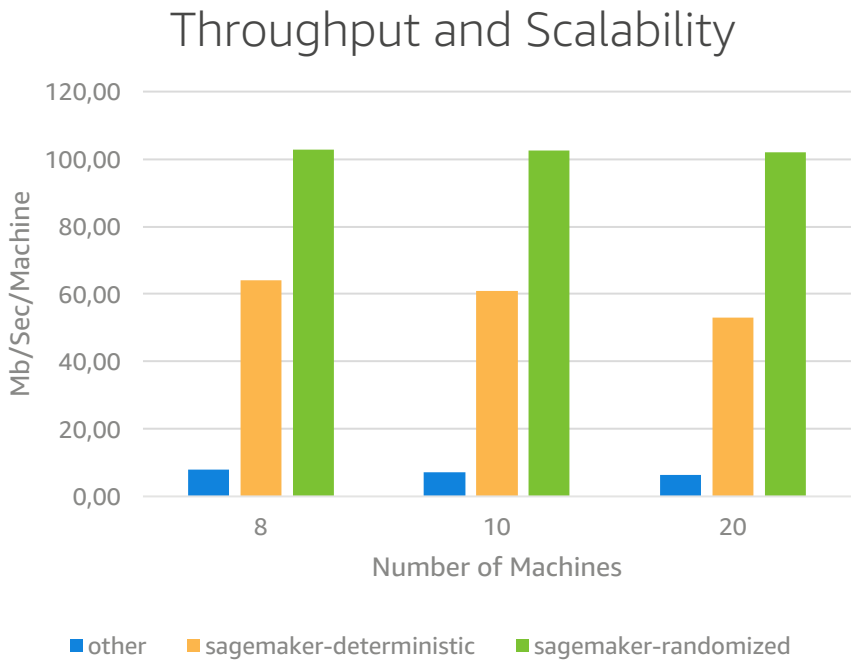


Image: Wikimedia Commons

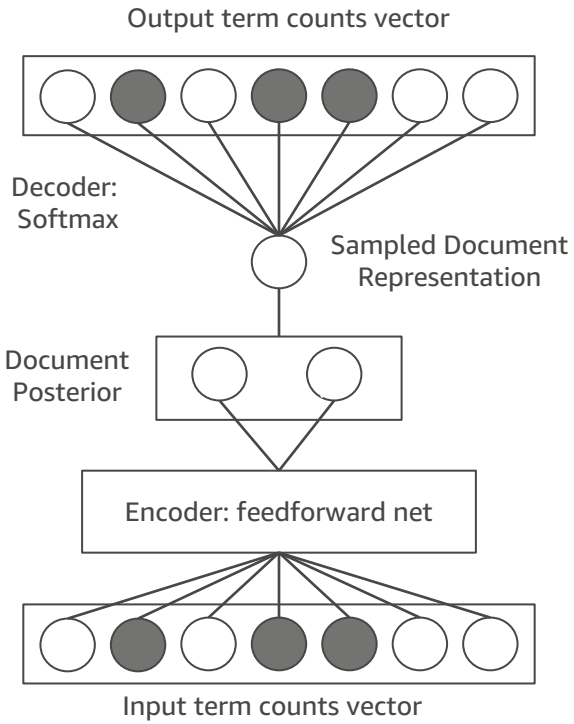
Principal Component Analysis (PCA)

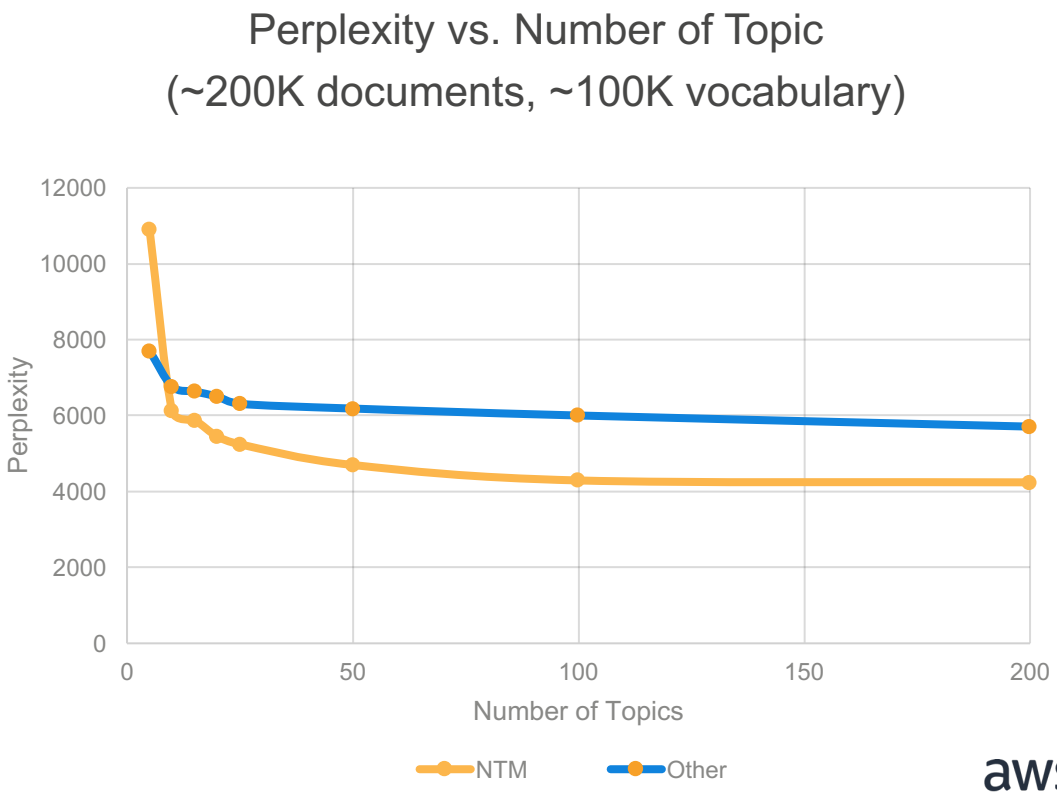




aws

Neural Topic Modeling





aws

Spectral Latent Dirichlet Allocation (LDA)

The New York Times

U.S.

High-Tech Industry, Long Shy of Politics, Is Now Belle of Ball

By LIZETTE ALVAREZ DEC. 26, 1999

Correction Appended

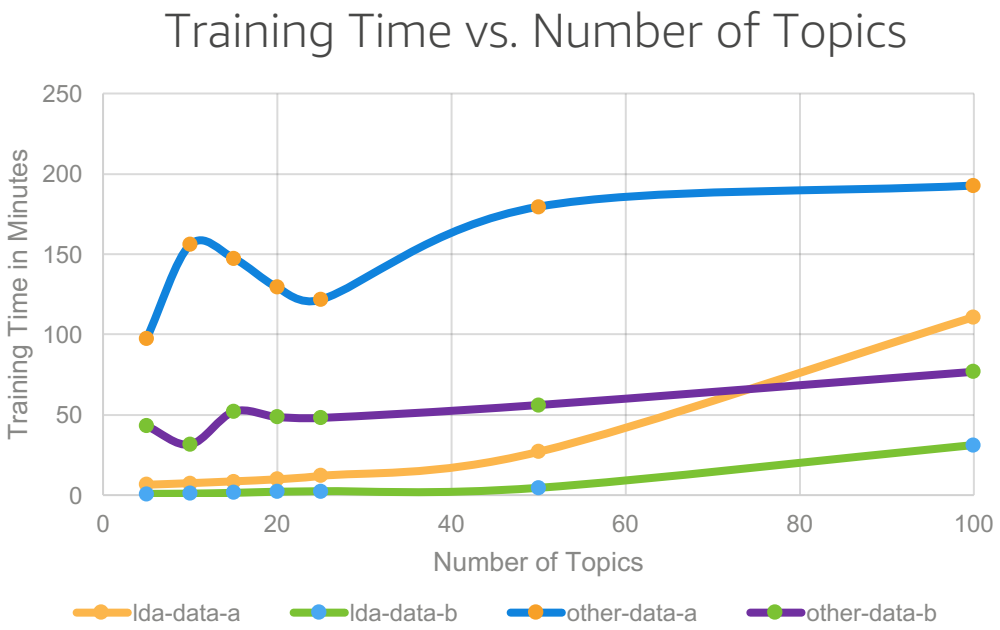
At a time when Congress is bitterly divided and unable to reach consensus on issues like gun control and health care, Democrats and Republicans are happily reaching across party lines to pass legislation backed by high-tech companies.

The high-tech industry, at the same moment, is lavishing new attention on Washington and changing its once-aloof posture toward the federal government.

Republicans and Democrats are both eager to win the loyalties of high-tech companies and executives, knowing that they represent untold jobs, wealth and ultimately votes and campaign contributions.

For its part, the industry has realized that the federal government can do its members as much harm as good. Microsoft, and its battle with the Justice Department, along with a spate of other threatened legal problems, drilled this point home.

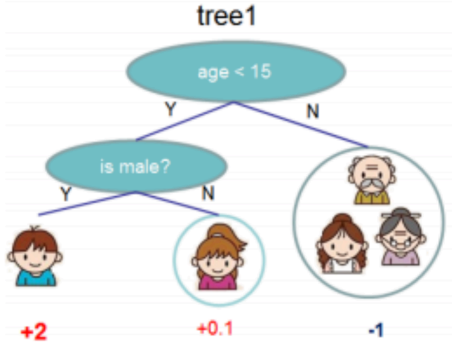
"Microsoft was a poster child for our industry," said Connie Correll, director of communications for the Information Technology Industry Council, a trade organization that represents America Online, Dell and I.B.M., among others.

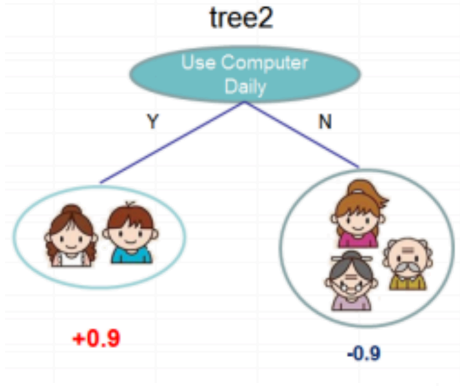


aws

Boosted Decision Trees

- XGBoost gradient boosted trees algorithm
- Combines multiple weak decision tree models
- Supervised
- Binary and multiclass classification
- Libsvm or CSV data





f( ) = 2 + 0.9= 2.9 f( )= -1 - 0.9 = -1.9

Image:Arxiv.org

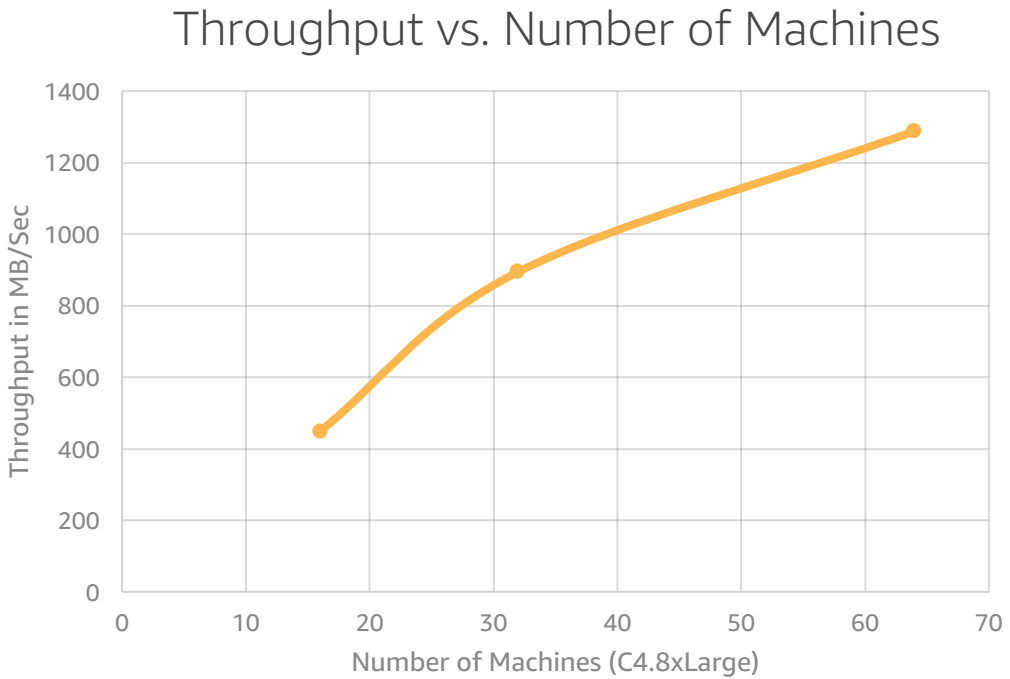
aws

Boosted Decision Trees

XGBoost is one of the most commonly used implementations of boosted decision trees in the world.

It is now available in Amazon SageMaker!

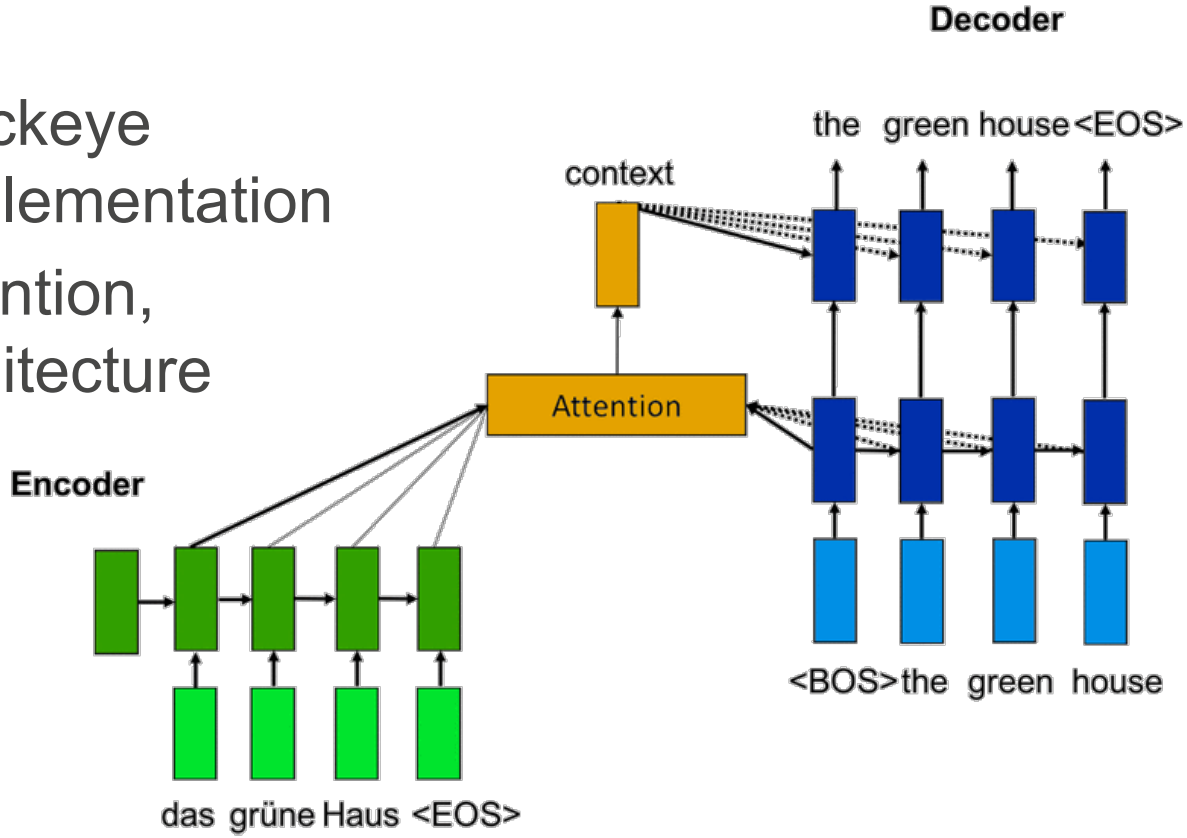
dmlc XGBoost



aws

Sequence to Sequence

- Based on Sockeye Seq2Seq implementation
- Encoder, Attention, Decoder architecture



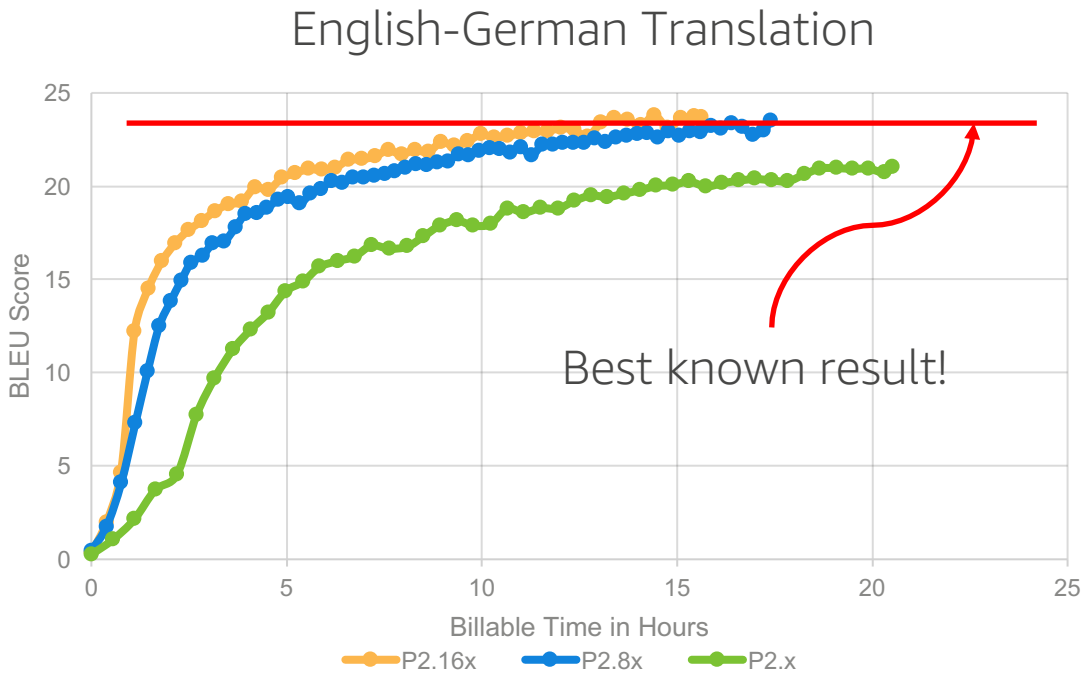
See: https://aws.amazon.com/blogs/ai/train-neural-machine-translation-models-with-sockeye/

aws

Sequence to Sequence

Based on Sockeye and Apache incubated MxNet, Multi-GPU, and can be used for Neural Machine Translation.

Supports both RNN/CNN as encoder/decoder



aws

Image Classification

- Build your own “Rekognition” service!
- Currently based on ResNet
- Configurable number of layers
- Full training and transfer learning
- Data formats:
    - Apache MXNet RecordIO
    - .jpg or .png

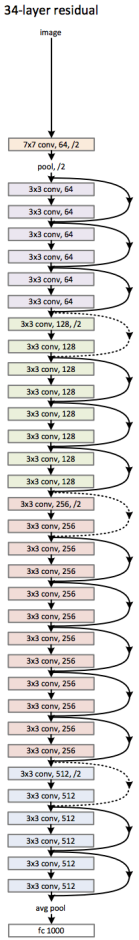
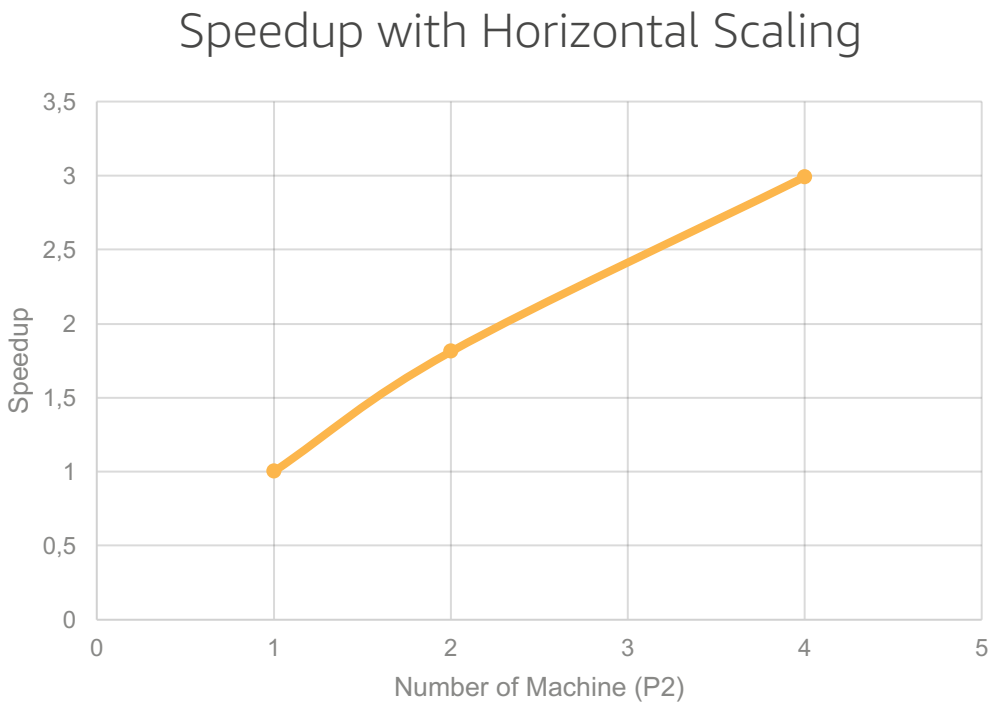


Image Classification

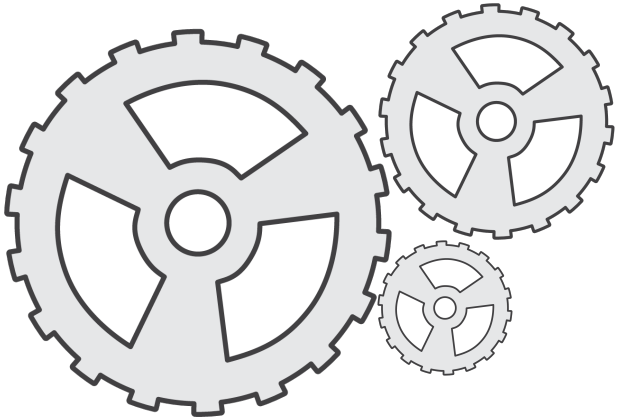
Implementation in MxNet of ResNet.
Other networks such as DenseNet and Inception will be added in the future.

Transfer learning: begin with a model already trained on ImageNet!



Your Own Algorithms

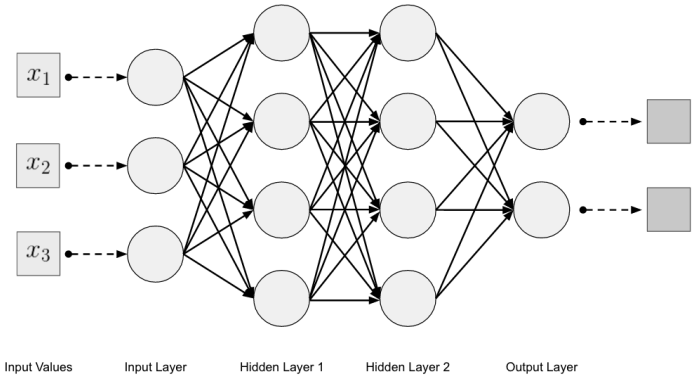
- Bring your own algorithm!
- Just wrap it into a Docker container
  - One container for training
  - One container for inference
- Combine SageMaker containers with your own
- Documented example on GitHub



aws

Deep Neural Networks

- Train your own Deep Neural Networks!
- TensorFlow and Apache MXNet supported
- You provide training/inference scripts with your DNN
- SageMaker does the rest



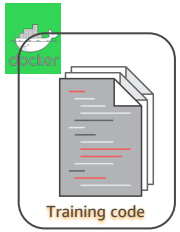
Using SageMaker with Apache Spark

- Apache Spark library for Amazon SageMaker provided
- Both Python and Scala
- Makes org.apache.spark.sql.DataFrame objects available to SageMaker
- Training and Inference supported

aws

Algorithms

② Amazon SageMaker: 10x better algorithms



- Matrix Factorization
- Regression
- Principal Component Analysis
- K-Means Clustering
- Gradient Boosted Trees
- And More!

Amazon provided Algorithms

mxnet TensorFlow™

Bring Your Own Script
(SageMaker builds the Container)

APACHE Spark SageMakerEstimators in Apache Spark

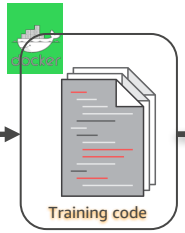
Bring Your Own Algorithm (You build the Container)

aws

ML Training Service

Secured

Fetch Training



Training code

Save Model Artifacts

Save Inference Image

- Matrix Factorization
- Regression
- Principal Component Analysis
- K-Means Clustering
- Gradient Boosted Trees
- And More!

Amazon provided Algorithms

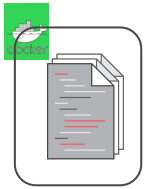
mxnet

TensorFlow

Bring Your Own Script
(SageMaker builds the Container)

Apache Spark

SageMakerEstimators
in Apache Spark



Bring Your Own Algorithm (You build the Container)

CPU GPU

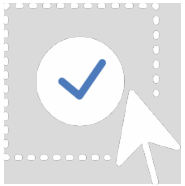
HPO

Fully managed

aws

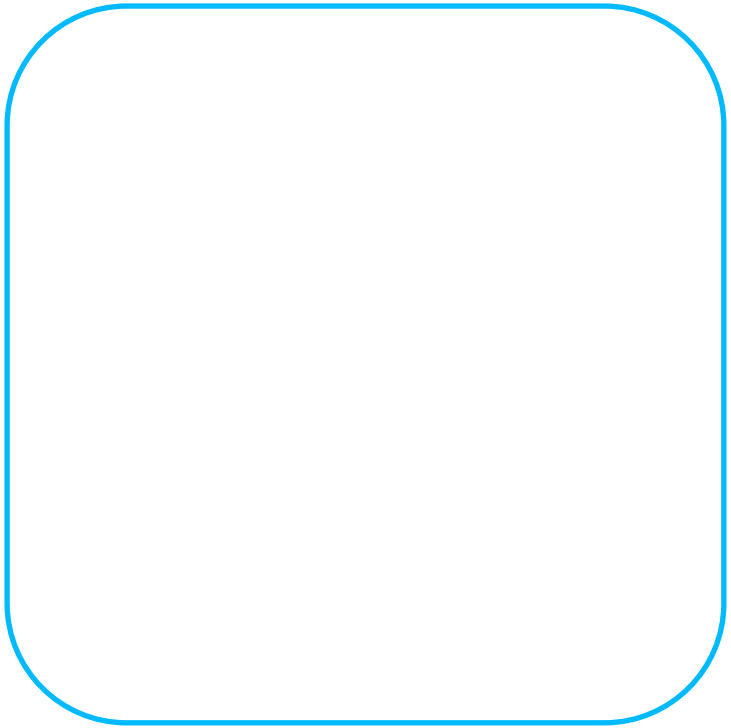
Model Deployment

4



Versions of the same inference code saved in inference containers. Prod is the primary one, 50% of the traffic must be served there!

Amazon ECR

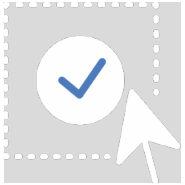


Amazon SageMaker

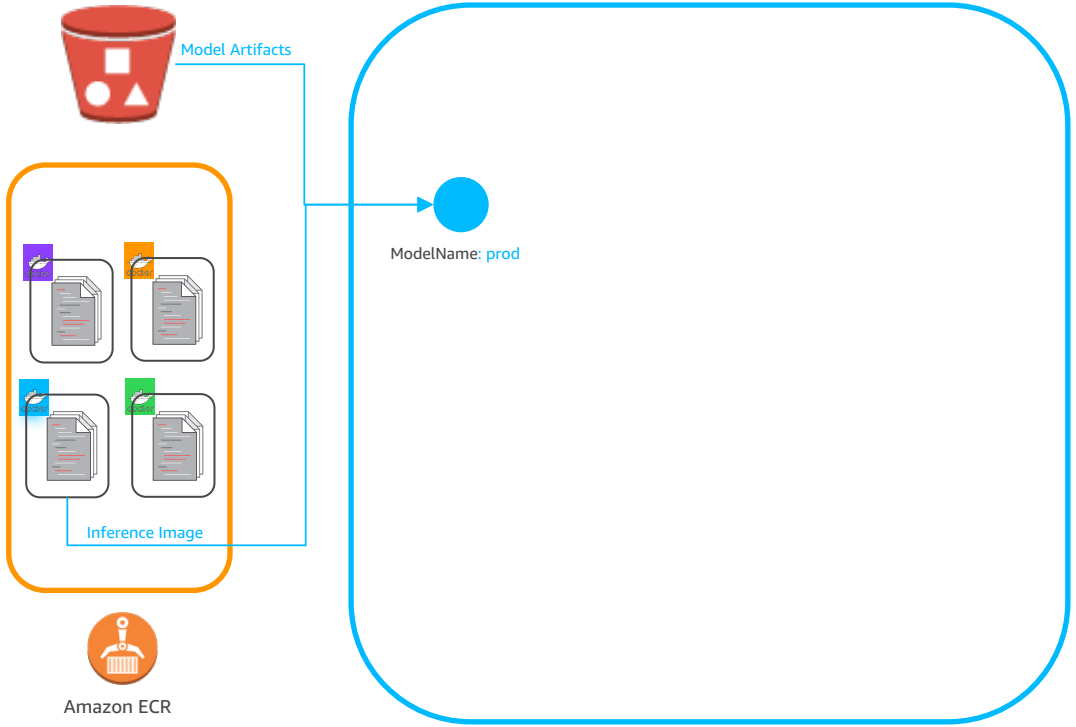
aws

Model Deployment

4



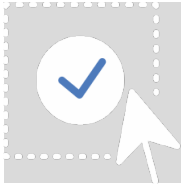
Versions of the same inference code saved in inference containers. Prod is the primary one, 50% of the traffic must be served there!



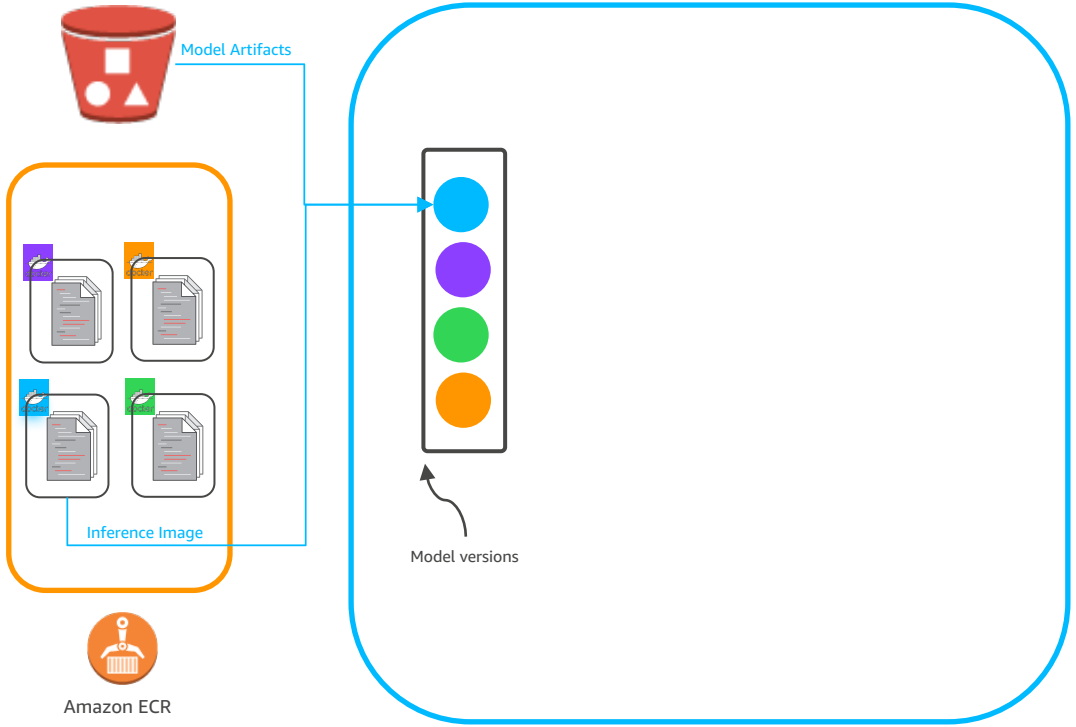
Create a Model

Model Deployment

4



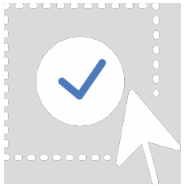
Versions of the same inference code saved in inference containers. Prod is the primary one, 50% of the traffic must be served there!



Create versions of a Model

Model Deployment

4

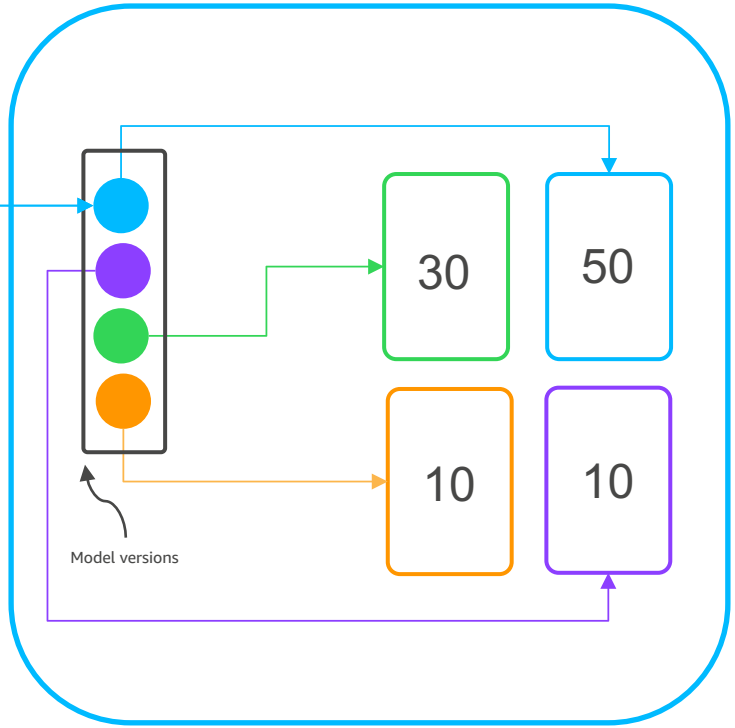


Versions of the same inference code saved in inference containers. Prod is the primary one, 50% of the traffic must be served there!

Model Artifacts

Inference Image

Amazon ECR



Amazon SageMaker

InstanceType: c3.4xlarge

InitialInstanceCount: 3

ModelName: prod

VariantName: primary

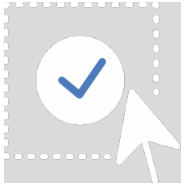
InitialVariantWeight: 50

ProductionVariant

Create weighted ProductionVariants

Model Deployment

4

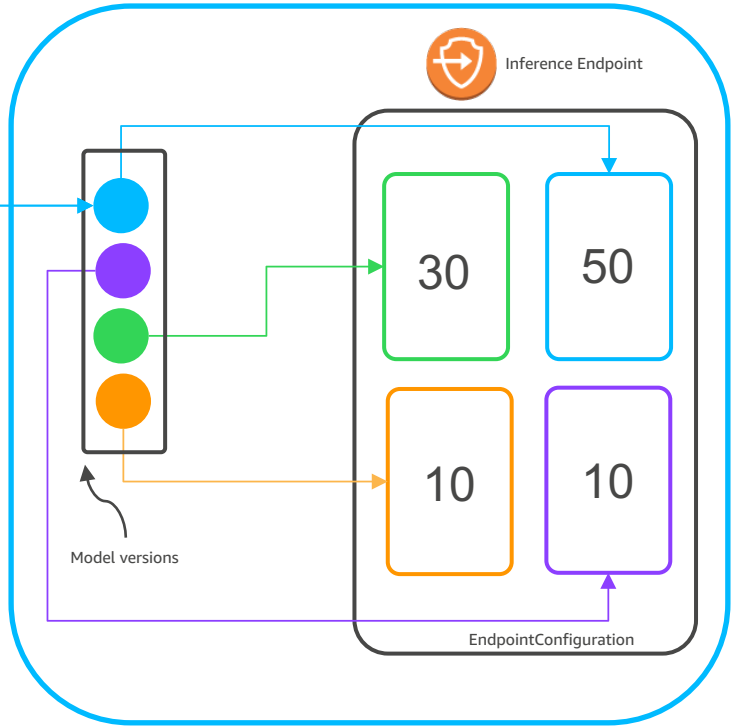


Versions of the same inference code saved in inference containers. Prod is the primary one, 50% of the traffic must be served there!

Model Artifacts

Inference Image

Amazon ECR



Amazon SageMaker

InstanceType: c3.4xlarge

InitialInstanceCount: 3

ModelName: prod

VariantName: primary

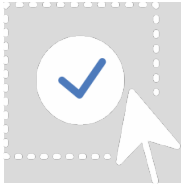
InitialVariantWeight: 50

ProductionVariant

Create an Endpoint from one EndpointConfiguration

Model Deployment

4

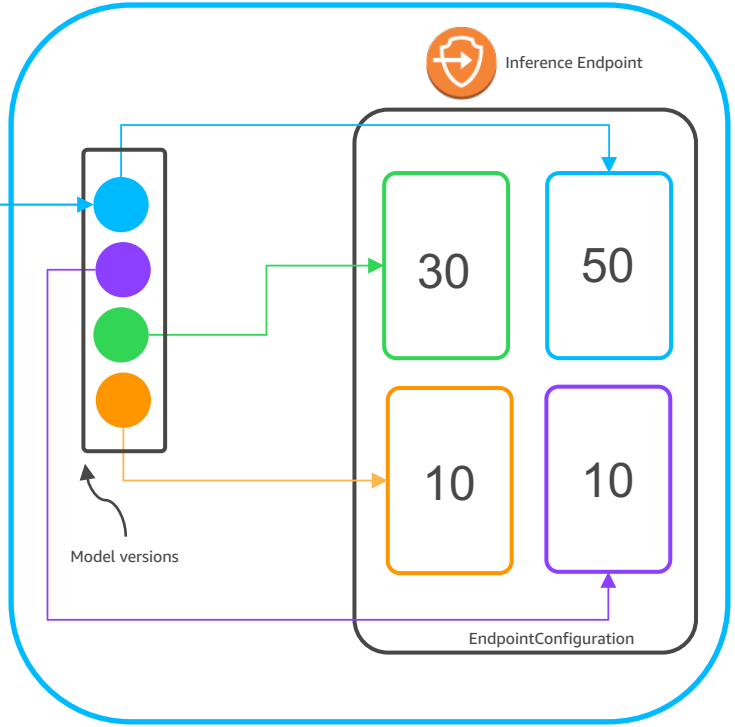


Versions of the same inference code saved in inference containers. Prod is the primary one, 50% of the traffic must be served there!

Model Artifacts

Inference Image

Amazon ECR



Amazon SageMaker

InstanceType: c3.4xlarge

InitialInstanceCount: 3

ModelName: prod

VariantName: primary

InitialVariantWeight: 50

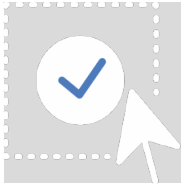
ProductionVariant

One-Click!

Amazon Provided Algorithms

Model Deployment

4



- ✓ Auto-Scaling Inference APIs
- ✓ A/B Testing (more to come)
- ✓ Low Latency & High Throughput
- ✓ Bring Your Own Model
- ✓ Python SDK

Amazon SageMaker

aws

Amazon SageMaker—Your Turn

- Getting started with Amazon SageMaker: https://aws.amazon.com/sagemaker/
- Use the Amazon SageMaker SDK:
  - For Python: https://github.com/aws/sagemaker-python-sdk
  - For Spark: https://github.com/aws/sagemaker-spark
- SageMaker Examples: https://github.com/awslabs/amazon-sagemaker-examples
- Let us know what you build!

aws

Thank you!

Constantin Gonzalez
glez@amazon.de
@zalez