Serving Machine Learning Model on Kubeflow



Abhishek Kumar
DATA SCIENTIST | AUTHOR | SPEAKER
@meabhishekkumar



Overview



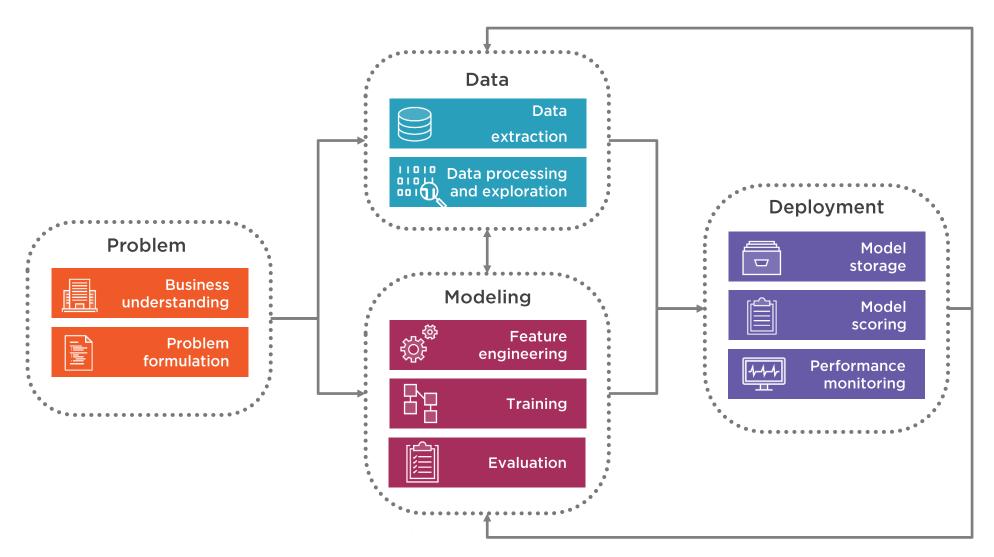
Model serving process and challenges
Kubeflow components for serving
KFServing overview

Demo: Serving machine learning model

- Expose model as API using KFServing
- Pre and post-processing
- Canary release
- Monitoring
- Auto scaling and load testing

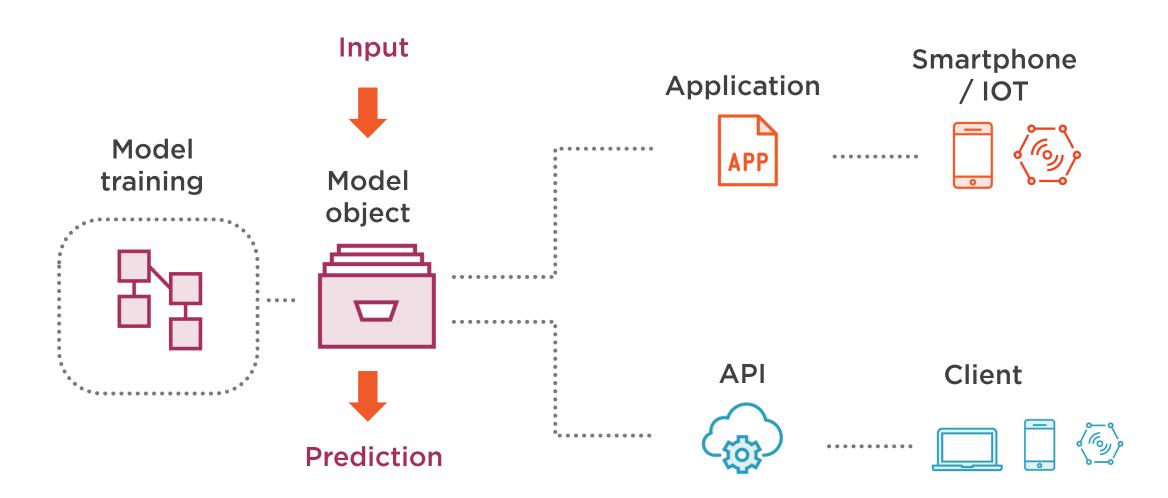


Machine Learning Workflow





Model Serving





Model Serving Challenges



Deploy



Monitoring



Release (Canary, A/B test)



Pre and postprocessing



Scaling



Explanation



Kubeflow Components for Serving

TensorFlow serving

Serve TensorFlow models

TensorFlow batch prediction

Batch prediction for TensorFlow models

NVIDIA TensorRT

NVIDIA inference server

Seldon core serving

Support multiple framework

KFServing

High level abstractions for common frameworks



KFServing

Serverless inference on Kubernetes

Support common and arbitrary frameworks

- TensorFlow, XGBoost, scikit-learn, PyTorch, and ONNX
- Custom

Deployment

- Canary rollouts

Performance monitoring

- Prometheus, Grafana, Elasticsearch

Pre and post-processing

Model explainibility





Model serving using KFServing

- Serve model as API
- Invoke model API for prediction





Pre and post-processing using KFServing

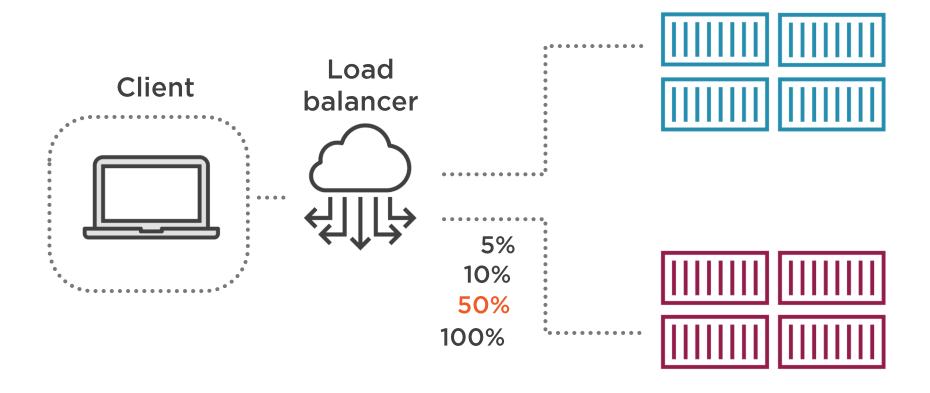




Canary rollout using KFServing



Canary Rollout







Performance monitoring using KFServing, Prometheus, Grafana





Auto scaling and load testing



Summary



Model serving

- Flavors
- Challenges

Model serving in Kubeflow

KFserving

- Expose model as API
- Pre and post processing
- Canary rollout
- Monitoring
- Auto scaling and load testing



Next up: Building Machine Learning Pipeline Using Kubeflow Pipeline

