O’reilly AI Conference
Machine Learning at Scale with Kubernetes

Chris Cho and David Sabater Dinter
ML is great!
ML is hard!
Most folks

Lots of pain

Magical AI goodness
Ok, but why?

**DIY**
- Set up from scratch
- Integrate with existing legacy systems
- Eventual need to migrate based on business needs

**HOSTED***
- First five minutes? YES.
- Next five years ... all of the above
- ... plus single cloud only!

* #NotAllHostedSolutions
Haven’t we heard this story before?
Containers & Kubernetes
Kubernetes extensibility
Cloud Native Apps
So...
We need Cloud Native ML!
Composability

Building a model
Portability
Portability

<table>
<thead>
<tr>
<th>Experimentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
</tr>
<tr>
<td>UX</td>
</tr>
<tr>
<td>Tooling</td>
</tr>
<tr>
<td>Framework</td>
</tr>
<tr>
<td>Storage</td>
</tr>
<tr>
<td>Runtime</td>
</tr>
<tr>
<td>Drivers</td>
</tr>
<tr>
<td>OS</td>
</tr>
<tr>
<td>Accelerator</td>
</tr>
<tr>
<td>HW</td>
</tr>
</tbody>
</table>
Portability

Experimentation

Model
UX
Tooling
Framework
Storage
Runtime
Drivers
OS
Accelerator
HW

Data Ingestion → Data analysis → Data transformation → Data validation → Data splitting → Trainer → Building a model → Model validation → Training at scale → Roll-out → Serving → Monitoring → Logging
Multi-Cloud is the reality

81% of enterprises have a multi-cloud strategy

- 51% Hybrid Cloud
- 21% Multiple Public
- 10% Multiple Private
- 9% Single Public
- 5% No Plans
- 4% Single Private

Respondents with 1,000+ Employees

Source: RightScale 2018 State of the Cloud Report
And not just one Cloud!

Companies using almost 5 public and private clouds on average

<table>
<thead>
<tr>
<th>Public + Private Clouds Used</th>
<th>Average All respondents</th>
<th>Median All respondents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Running Applications</td>
<td>3.1</td>
<td>3.0</td>
</tr>
<tr>
<td>Experimenting</td>
<td>1.7</td>
<td>1.0</td>
</tr>
<tr>
<td>Total</td>
<td>4.8</td>
<td>4.0</td>
</tr>
</tbody>
</table>

Source: RightScale 2018 State of the Cloud Report
Portability
Portability
Portability

Training

Experimentation
Portability

Experimentation

Training

Cloud

Data Ingestion -> Data analysis -> Data transformation -> Data validation -> Data splitting

Trainer -> Building a model -> Model validation -> Training at scale

Roll-out -> Serving -> Monitoring -> Logging
Pffft. That doesn’t matter to me.
Wrong.
The way I think about it: every difference between dev/staging/prod will eventually result in an outage.
The way I think about it: every difference between dev/staging/prod will eventually result in an outage.
The way I think about it: every difference between dev/staging/prod will eventually result in an outage.
Your laptop counts.
Portability

Experimentation

- Data Ingestion
- Data analysis
- Data transformation
- Data validation
- Data splitting
- Trainer
- Building a model
- Model validation
- Training at scale
- Roll-out
- Serving
- Monitoring
- Logging

Training

Cloud
Portability

Experimentation

Training

Cloud
Scalability

- More accelerators (GPU, TPU)
- More CPUs
- More disk/networking
- More skillsets (SWEs, data scientists)
- More teams
- More experiments
Composability
Portability
Scalability
You know what’s really good at **composability, portability and scalability**?
Containers & Kubernetes
Containers & Kubernetes

except...
Oh, you want to use ML on Kubernetes?

First, become an expert in:

- Containers
- Packaging
- Kubernetes service endpoints
- Persistent volumes
- Scaling
- Immutable deployments
- GPUs, Drivers & the GPL
- Cloud APIs
- DevOps
- ...

Oh, you want to use ML on Kubernetes?
Introducing Kubeflow
Introducing Kubeflow 0.1
Make it easy for everyone to develop, deploy and manage portable, distributed ML on Kubernetes.
Portability
Portability

Experimentation

Training

Cloud

Data Ingestion → Data analysis → Data transformation → Data validation → Data splitting

Trainer → Building a model → Model validation → Training at scale

Roll-out → Serving → Monitoring → Logging
Portability

Experimentation

Training

Cloud

Kubeflow

Kubernetes
Portability

- Experimentation
- Training
- Cloud

Kubeflow

Rook
Portability

- Experimentation
- Training
- Cloud

Kubeflow

Rook for Kubernetes
What’s in the box?

- Jupyter notebook
- Multi-architecture, distributed training
- Multi-framework model serving
- Examples and walkthroughs for getting started
- Ksonnet packaging for customizing it yourself!
What’s in the box?

Data ingestion → Data analysis → Data transformation → Data validation → Data splitting →

Trainer → Building a model → Model validation → Training at scale →

Roll-out → Serving → Monitoring → Logging
Introducing Kubeflow 0.1
Introducing Kubeflow 0.1
Introducing Kubeflow 0.2
What’s new in 0.2?

- New and Improved Components (PyTorch Alpha, Katib Alpha)
- Platform Integrations
- Simplified Getting Started Experience
Kubeflow 0.2: Advanced Features
# Katib

<table>
<thead>
<tr>
<th>Project</th>
<th>Metrics</th>
<th>min</th>
<th>max</th>
<th>average</th>
</tr>
</thead>
<tbody>
<tr>
<td>cif10</td>
<td>Validation-accuracy</td>
<td>0.546</td>
<td>0.554</td>
<td>0.55</td>
</tr>
<tr>
<td></td>
<td>time-cost-Min-accuracy</td>
<td>0.491</td>
<td>0.517</td>
<td>0.50</td>
</tr>
<tr>
<td>cif10-pv-test</td>
<td>Validation-accuracy</td>
<td>0.507</td>
<td>0.565</td>
<td>0.54</td>
</tr>
<tr>
<td></td>
<td>time-cost-Min-accuracy</td>
<td>0.65</td>
<td>0.667</td>
<td>0.66</td>
</tr>
<tr>
<td></td>
<td>time-cost-Min-accuracy</td>
<td>0.46</td>
<td>0.549</td>
<td>0.30</td>
</tr>
<tr>
<td>tf-nmt</td>
<td>test ppl</td>
<td>114.67</td>
<td>167.76</td>
<td>164.81</td>
</tr>
<tr>
<td></td>
<td>time-cost-Min</td>
<td>5.567</td>
<td>8.933</td>
<td>6.75</td>
</tr>
<tr>
<td></td>
<td>bleu_test</td>
<td>0.8</td>
<td>2.9</td>
<td>1.95</td>
</tr>
<tr>
<td></td>
<td>bleu_dev</td>
<td>0.4</td>
<td>3.4</td>
<td>1.83</td>
</tr>
<tr>
<td></td>
<td>ppl</td>
<td>126.01</td>
<td>183.72</td>
<td>160.00</td>
</tr>
</tbody>
</table>
# Initialize a ksonnet APP
APP_NAME=my-kubeflow
ks init ${APP_NAME}
cd ${APP_NAME}

# Install Kubeflow components
ks registry add kubeflow github.com/kubeflow/kubeflow/tree/master/kubeflow
ks pkg install kubeflow/core
ks pkg install kubeflow/tf-serving
ks pkg install kubeflow/tf-job
ks pkg install kubeflow/pytorch

# Deploy Kubeflow
NAMESPACE=kubeflow
kubectl create namespace ${NAMESPACE}
ks generate core kubeflow-core --name=kubeflow-core --namespace=${NAMESPACE}
ks apply default -c kubeflow-core
Kubeflow 0.2: Platform Integrations
Platform Integrations

- Cloud Native Security
  - Data Exfiltration prevention
  - Identity Aware Proxy

- Cloud-specific Infrastructure
  - Accelerators (GPUs, TPUs)
  - Preemptible VMs

- Kubernetes extensions
  - Node Pools
  - Auto scaling clusters
TPU Acceleration

Images / Second

<table>
<thead>
<tr>
<th></th>
<th>Images per Second</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-TPU</td>
<td>0.92</td>
</tr>
<tr>
<td>TPU</td>
<td>1.71</td>
</tr>
</tbody>
</table>

(Data Courtesy of RiseML)

Total Dollars

<table>
<thead>
<tr>
<th></th>
<th>Cost to Reach 75% Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-TPU</td>
<td>$129</td>
</tr>
<tr>
<td>TPU</td>
<td>$55</td>
</tr>
</tbody>
</table>

(Data Courtesy of RiseML)
Kubeflow 0.2: Simplified Setup
Old and Busted

ks registry add kubeflow kubeflow

ks pkg install kubeflow/argo
ks pkg install kubeflow/core
ks pkg install kubeflow/tf-serving

ks generate kubeflow-core kubeflow-core

ks apply default
export KUBEFLOW_VERSION=0.2.2
curl https://raw.githubusercontent.com/kubeflow/kubeflow/${KUBEFLOW_VERSION}/scripts/gke/deploy.sh
./deploy.sh
Introducing Kubeflow 0.2
Introducing Kubeflow 0.2
Introducing Kubeflow 0.3
Kubeflow 0.3: Improved tooling
Kubeflow 0.3!

Some of the new features...

- Streamline and Extensible Deployment using kfctl
- Multi-Cloud Management - Minikube, microk8s, and GKE
- New StudyJob operators - No code to submit HP tuning jobs
- Improved multi framework support  Consistent APIs across supported frameworks
  - PyTorch v1alpha2 provides a consistent API for PyTorch and TensorFlow Jobs
  - Gang-scheduling support using Kube Arbitrator
- More examples!
  - Financial time series, xgboost, and Object detection with GPU serving and batch prediction
- Blog with full update coming soon at www.kubeflow.org!
Pytorch ML pipeline in Kubeflow
Reminder... What you DIDN’T see

- Bespoke solutions
- Cloud-specific, non-portable tech
- Forking of Kubernetes APIs
Momentum!

Commits Since Launch

- 1000+ commits
- 100+ Community contributors
- 20+ Companies contributing, including:
  - Alibaba Cloud
  - Canonical
  - Cisco
  - Datawire
  - Dell
  - GitHub
  - Google
  - Heptio
  - Huawei
  - Intel
  - IBM
  - Microsoft
  - Momentum
  - Netflix
  - Red Hat
  - Seldon
  - Uber
  - Weaveworks
Momentum!

Where can I see more Kubeflow?

- [Intro to Kubeflow Meetup on Oct 11](#)
- Kubecon Shanghai
It’s a whole new world

- Data science will touch EVERY industry.
- We can’t ask people to become a PhD in statistics though.
- How do WE help everyone take advantage of this transformation?
We’re just getting started!

Kubeflow 0.4! Our roadmap:

- Better integration with Notebook
- New PM community working on the 0.4 roadmap coming soon!*  
- **You tell us!** (Or better yet, help!)

* 3~4 weeks
Kubeflow is open!

Open community
Open design
Open source
Open to ideas
Come Help!

**website:** https://kubeflow.org

**github:** https://github.com/kubeflow/kubeflow

**slack:** kubeflow (http://kubeflow.slack.com)

**twitter:** @kubeflow

chrischo@google.com davidsabater@google.com
Thank you!
Using data to save lives

https://www.youtube.com/watch?v=udSjBbGwJEg