Katib
A Distributed General AutoML Platform Based on Kubernetes

Jinan Zhou¹, Andrey Velichkevich¹, Kirill Prosvirov¹, Anubhav Garg¹,
Yuji Oshima², Debo Dutta¹

¹: Cisco Systems, USA
²: NTT Software Innovation Center, Japan
CONTENTS

01 Introduction
  • What is Katib
  • Why develop Katib

02 AutoML Workflows
  • Hyperparameter Tuning
  • Neural Architecture Search

03 System Design
  • Design Principles
  • System Diagram

04 Demo
Part I: Introduction

• What is Katib
• Why develop Katib
Introduction

What is AutoML?
Automated Machine Learning (AutoML) is the automation of the end to end process of applying machine learning to real-world problems. The primary challenge is designing the model for a specific task by Hyperparameter Tuning or Neural Architecture Search.

What is Kubernetes?
Kubernetes (k8s) is an open-source system for automating deployment, scaling, and management of containerized applications.

What is Katib?
Katib is a general platform to run AutoML tasks in distributed systems. It is built on k8s and is part of Kubeflow, an open source machine learning toolkit for k8s.
Why Develop Katib?

To tackle the most central problems in deploying AutoML workflows.

Problems in AutoML
- Diversity of AutoML Algorithms
- Expensive Computing Cost

Solutions by Katib
- General AutoML System
- Distributed Training
Part II: AutoML Workflows

- Hyperparameter Tuning
- Neural Architecture Search
AutoML Workflows

Hyperparameter Tuning (HPT)

- Find an optimal hyperparameter $\hat{p}$ so that $f(\cdot)$ can be minimized or maximized.
- $f(\cdot)$ is usually regarded as a black box.
- Commonly used algorithms:
  - Bayesian Optimization
  - Hyperband
Neural Architecture Search (NAS)

A neural network is represented by a directed acyclic graph (DAG) $G = \{V, E\}$, where a vertex $V_i$ denotes the latent representation in $i^{th}$ layer and a directed edge $E_k = (V_i, V_j)$ denotes an operation $O_k$ whose input is $V_j$ and output is given to $V_i$. The value of the vertex $V_i$ depends on all the incoming edges:

$$V_i = g(\{ o_k(V_j)|(V_i, V_j) \in E \})$$

$g(\cdot)$ is a function to combine all the inputs. It can be weighted sum, concatenation, etc., depending on the algorithm.
Neural Architecture Search (NAS)

- Needs a Model Constructor:
  - In HPT, the model is already specified. In NAS, however, we need to find the model.
  - Interpret the representations given by the suggestion.

- Different searching strategies
  - Search for a network
  - Search for a cell

- Different evolving strategies
  - Evolve by generation
  - Evolve by modification
All the AutoML Workflows run in an iterative manner.

Integrate all model representation interpretation functions needed by different NAS algorithms into Model Manager.

HPT and NAS workflows can be naturally unified in one system.
Part III: System Design

- System Design Principles
- System Diagram
Design Principles

Compatible
Universally compatible with all AutoML algorithms.

Scalable
Can be deployed at large scale. Complete support for parallel training.

User friendly
Powerful CLI for experts and intuitive GUI for novice.

Extendible
Easy to be customized for the user’s own propose
System Design

- StudyJob
- CLI / GUI
- User Interface
- Static Kubernetes Pods
- Dynamic Kubernetes Pods
- Data Flow
- Module Spawning

System Diagram:
- StudyJob Controller
- Vizier Core Manager
- Suggestions:
  - EnvelopeNet
  - Reinforcement Learning
  - Bayesian Optimization
  - Hyperband
- Metrics Collector
- Model Manager
- Trial 0
- Trial 1
- Trial 2
- MySQL Database
- Persistent Volume
Part IV: Demo
https://www.youtube.com/watch?v=WAK37UW7spo
THANK YOU FOR LISTENING

Please feel free to ask any questions!

Katib
A Distributed General AutoML Platform Based on Kubernetes