

Contents

1.	TUTORIAL LEARNING OBJECTIVES	1
2.	PREREQUISITES	1
3.	QUICK START	3
4.	EXPLORING ADVANCED FEATURES	5
5.	TROUBLESHOOTING	5
6.	SUMMARY	6
7.	LEGEND	6

1. Tutorial Learning Objectives

튜토리얼의 목표는 Samsung Cloud Platform에서 Kubeflow Mini을 사용하여 AutoML을 실행하는 Instance를 생성하고, AutoML 관련된 고급기능을 알아보는 것입니다.

① Kubeflow Mini는 오픈소스 Kubeflow vanilla version(v1.7)을 제공하는 Samsung Cloud Platform 상품으로, Pipeline과 AutoML의 관련한 튜토리얼은 Kubeflow Mini를 기반으로 진행합니다.

2. Prerequisites

튜토리얼 진행에 필요한 사전요건은 아래와 같습니다.

Kubeflow Mini 또는 AI&MLOps Platform에 대한 이해

YAML 파일 형식 및 사용법에 대한 이해

Python 프로그래밍에 대한 이해

머신러닝(ML)에 대한 이해

- "AI&MLOps Platform에서 Jupyter notebook 생성하기" 튜토리얼 참조

2.1 Consult the Kubeflow Documentation

✓ Kubeflow Documentation (공식문서) 참조를 권장합니다.

→ Kubeflow Documentation의 위치는 아래 순서에 따라 이동합니다.

① www.kubeflow.org로 이동합니다.

② 메인 페이지에서 "Get Started →" 버튼을 클릭합니다.

③ Kubeflow Documentation의 메인 페이지로 이동하였습니다.

→ Kubeflow Documentation에서 아래 순서에 따라 이동합니다.

① "Components"를 선택합니다.

② "Katib"를 선택합니다.

③ 관심분야에 따른 상세 이동 위치를 아래에서 선택합니다.

- Katib를 설정하고 하이퍼 파라미터 튜닝을 수행하는 방법: "Getting Started with Katib"로 이동합니다.
- Katib에서 하이퍼 파라미터 튜닝, 뉴럴 아키텍처 탐색 실험(experiment)을 구성하고 실행하는 방법: "Running an Experiment"로 이동합니다.

✓ 튜토리얼은 Kubeflow(Katib) UI를 통해 experiment 구성을 진행합니다. 따라서, "Katib > Running an Experiment"의 Documentation을 참조하고 있습니다.

➤ Katib은 Kubeflow에 포함되어, Kubernetes 환경에서 머신러닝 모델의 훈련을 최적화하기 위해 하이퍼 파라미터 튜닝을 자동화하는 도구입니다. 하이퍼 파라미터는 머신러닝 모델의 훈련에 영향을 미치는 변수들로, 그 값에 따라 모델의 성능이 달라질 수 있습니다. Katib는 여러 가지 검색 알고리즘을 제공하여 최적의 하이퍼 파라미터 조합을 자동으로 찾아낼 수 있습니다. 또한, Katib은 여러 실험을 동시에 실행하고, 결과를 시각적으로 표시하여 사용자가 쉽게 비교하고 분석할 수 있게 해줍니다. 이로 인해 데이터 사이언티스트가 수동으로 하이퍼 파라미터를 조정하는 시간을 줄이고, 더 나은 모델 성능을 빠르게 달성할 수 있습니다.

➤ 하이퍼 파라미터(Hyper Parameter)는 머신러닝 모델의 학습 과정을 제어하는 데 사용되는 매개변수입니다. 이들은 학습 과정 중에는 변경되지 않지만, 학습 전에 설정되어 모델의 성능과 학습 속도에 큰 영향을 미칠 수 있습니다. 예를 들어, 신경망 모델에서의 레이어 수(Number of layers), 학습률(Learning Rate), 배치 크기(Batch Size) 등은 하이퍼 파라미터의 예입니다.

➤ 하이퍼 파라미터 튜닝(Hyper Parameter Tuning)은 최적의 모델 성능을 달성하기 위해 여러 하이퍼 파라미터 값을 시험하여 그 중에서 최적의 값을 찾는 과정입니다. 자동화된 방법들(예: 그리드 검색, 랜덤 검색, 베이지안 최적화)을 사용하여, 가능한 많은 조합들 중에서 가장 좋은 성능을 나타내는 하이퍼 파라미터의 조합을 찾아내는 것이 목표입니다. (동의어: Hyperparameter optimization, HPO)

➤ Neural Architecture Search (뉴럴 아키텍처 탐색, NAS)는 최적의 신경망 구조를 자동으로 탐색하는 알고리즘입니다. 전통적인 머신러닝 및 딥러닝 모델에서는 연구자나 엔지니어가 신경망의 구조나 레이어를 수동으로 디자인했습니다. 그러나 NAS를 통해, 알고리즘이 주어진 자원과 제한 사항 내에서 가장 효과적인 신경망 구조를 자동으로 찾아내고, 이 과정은 하이퍼 파라미터 튜닝과 유사하지만, 신경망의 전체 구조와 디자인에 집중됩니다.

2.2 Secure Necessary Permissions

- ✓ 튜토리얼 진행을 위한 Kubeflow Mini 사용자 권한을 확인합니다.
- ✓ 필요시 Kubeflow Mini 관리자를 통해 권한을 확보합니다.

3. Quick Start

Kubeflow Mini에서 AutoML Instance를 생성하여 experiment로 실행하는 과정을 학습합니다.

 오픈소스 Kubeflow는 속성이 다른 Pipeline과 AutoML의 실행(Run)을 동일한 명칭의 experiment(실험)를 통해 관리합니다. 따라서 Kubeflow 대시보드의 사이드바(Sidebar) 항목에는 "Experiments (AutoML)"와 "Experiments (KFP)"가 연이어 표시됩니다. 하지만 experiment의 명칭만 동일하게 사용할 뿐, 실행을 구성하는 요소는 서로 상이합니다. 예를 들어, AutoML을 구성하는 Workflow 솔루션은 Pipeline을 구성하는 Workflow 솔루션과 동일하지 않습니다.

3.1 Preliminary Checks

- ✓ AutoML Instance 생성에 충분한 리소스를 확보합니다.

① Kubeflow Mini 리소스 부족 시 Kubernetes Engine의 node pool을 추가하여 리소스를 확보할 수 있습니다.

3.2 Creating a New AutoML Instance

- ✓ Kubeflow.org에서 제공하는 Random Search Algorithm 예제를 이용하여 AutoML Instance 생성하겠습니다.

➔ Random Search Algorithm 예제 파일 다운로드 순서를 따릅니다.

① <https://github.com/kubeflow/katib> 로 이동합니다.

② "examples/v1beta1" 디렉토리로 이동합니다.

③ "hp-tuning" 디렉토리로 이동합니다.

④ "random.yaml" 파일을 다운로드합니다.

이 YAML 파일은 Random Search Algorithm을 포함하는 "experiments.yaml" 파일입니다.

➤ "experiments.yaml"은 AutoML workflows를 정의하는 YAML파일입니다.

➔ Kubeflow의 "Experiments(AutoML)" 페이지로 이동합니다.

① Kubeflow Mini 대시보드 사이드바에서 "Experiments(AutoML)"을 클릭합니다.

② "Experiments" 페이지로 이동하였습니다.

➔ Random Search Algorithm 예제 파일 붙여넣기는 아래 순서를 따릅니다.

① 편집기를 사용하여 "random.yaml" 파일의 namespace 항목을 사용자의 (Kubernetes) namespace로 수정하고 클립보드에 복사합니다.

② 혹은, 붙여넣기 후에 namespace 항목을 직접 수정할 수 있도록, 사용자의 (Kubernetes) namespace를 사전에 확인합니다.

③ "Experiments" 페이지 우측 상단의 "+ New Experiment" 버튼을 클릭합니다.

④ "Create an Experiment" 페이지로 이동하였습니다.

⑤ 페이지 하단 중앙의 "Edit" 텍스트를 클릭합니다.

⑥ "Edit YAML" 팝업 페이지가 나타납니다.

⑦ 클립보드 저장 내용을 "Edit YAML" 팝업 페이지의 텍스트 상자에 붙여 넣습니다.

⑧ 혹은, 붙여넣기 후에 namespace 항목을 사용자의 (Kubernetes) namespace로 직접 수정합니다.

⑨ "Create" 버튼을 클릭합니다.

⑩ "Experiments" 페이지로 이동하였습니다.

✓ "Create an Experiment" 페이지에서 experiment 구성을 입력하는 두 가지 방법이 있습니다:

- YAML 파일 붙여넣기: experiment의 전체 구성을 포함하는 YAML 파일을 붙여 넣는 방법입니다. 튜토리얼에서는 이 방법으로 진행하였습니다.

- 사용자의 직접입력: experiment 구성 값을 단계별로 수동으로 입력할 수 있으며, 구성이 단순하거나 YAML 파일의 공유나 저장이 불필요한 경우 사용합니다

➤ Kubeflow Mini는 AutoML의 experiment에 대하여 실시간으로 experiment 진행 상태(status), 로그(log), 결과(trial performance) 등을 확인할 수 있는 모니터링 대시보드를 제공합니다.

➔ "Experiments(AutoML)" 페이지에서 아래의 순서를 따라 experiment를 모니터링합니다.

- ① Kubeflow Mini 대시보드 사이드바에서 "Experiments(AutoML)"을 클릭합니다.
- ② experiment 목록에서 "Name" 필드를 클릭합니다.
- ③ "Experiment details" 페이지로 이동하였습니다.
- ④ experiment 상세정보가 대시보드에 표시되고, 관련 정보를 확인할 수 있습니다.

4. Exploring Advanced Features

✓ 하이퍼 파라미터 튜닝, 뉴럴 아키텍처 탐색, 그리고 Kubeflow 프레임워크 내에서 Katib의 심화기능(HPO, NAS)은 Kubeflow Documentation을 통해 확인할 수 있습니다. (<https://www.kubeflow.org/docs/started/>)

✓ 튜토리얼에서 활용한 YAML 파일을 kubeflow-katib SDK를 이용하여 생성하는 기능은 <https://pypi.org/project/kubeflow-katib/>에서 확인할 수 있습니다.

5. Troubleshooting

Kubeflow Mini 사용오류를 최소화하는 방안은,

- ✓ 다양한 오픈소스 환경에 따른 컴포넌트 간 버전관리를 철저히 하는 것
- ✓ 사용자와 자원의 권한/인증에 대한 정확한 정보를 유지하는 것
- ✓ 사용자 가이드 등 공식문서를 사전에 참조하여 적용하는 것입니다.

6. Summary

- ✓ AutoML은 데이터 사이언티스트 뿐만 아니라 머신러닝 툴킷의 전체 범위에서도 중요도가 높은 핵심기능을 제공합니다. AutoML을 활용하면 머신러닝 솔루션을 더 쉽게 사용할 수 있으며, 표준화된 experiment 실행 결과를 조직이 공유하는 이점도 가져옵니다.
- ✓ AI&MLOps Platform 또는 Kubeflow Mini에서 AutoML을 Jupyter Notebook과 같은 도구와 결합하여 활용함으로써, 모델 선택을 가속화하고, 최적의 하이퍼 파라미터 튜닝 보장 및 다양한 데이터셋에서 일관된 결과를 기대할 수 있습니다.

7. Legend

[✓] Tip

[→] 따라하기

[✎] Note

[i] Notice

[>] Definition or Terminology

[⚠] Warning

[★] See User Guide or Documentation

[•] 순서가 없는 리스트

[①~⑮] 순서가 있는 리스트