EXAMTOPICS

- Expert Verified, Online, **Free.**

☰ MENU   🔍

← **Google Discussions**

Exam Professional Machine Learning Engineer All Questions

View all questions & answers for the Professional Machine Learning Engineer exam

Go to Exam

📄 **EXAM PROFESSIONAL MACHINE LEARNING ENGINEER TOPIC 1 QUESTION 186 DISCUSSI...**

Actual exam question from Google's Professional Machine Learning Engineer

Question #: 186

Topic #: 1

[All Professional Machine Learning Engineer Questions]

You have been tasked with deploying prototype code to production. The feature engineering code is in PySpark and runs on Dataproc Serverless. The model training is executed by using a Vertex AI custom training job. The two steps are not connected, and the model training must currently be run manually after the feature engineering step finishes. You need to create a scalable and maintainable production process that runs end-to-end and tracks the connections between steps. What should you do?

A. Create a Vertex AI Workbench notebook. Use the notebook to submit the Dataproc Serverless feature engineering job. Use the same notebook to submit the custom model training job. Run the notebook cells sequentially to tie the steps together end-to-end.

B. Create a Vertex AI Workbench notebook. Initiate an Apache Spark context in the notebook and run the PySpark feature engineering code. Use the same notebook to run the custom model training job in TensorFlow. Run the notebook cells sequentially to tie the steps together end-to-end.

C. Use the Kubeflow pipelines SDK to write code that specifies two components:
- The first is a Dataproc Serverless component that launches the feature engineering job
- The second is a custom component wrapped in the create_custom_training_job_from_component utility that launches the custom model training job

Create a Vertex AI Pipelines job to link and run both components

D. Use the Kubeflow pipelines SDK to write code that specifies two components
- The first component initiates an Apache Spark context that runs the PySpark feature engineering code
- The second component runs the TensorFlow custom model training code

Create a Vertex AI Pipelines job to link and run both components.

by 👤 **vale_76_na_xxx** at *Jan. 8, 2024, 8:32 p.m.*

Comments

Type your comment...

Submit

👤 **Akel123** 5 months, 1 week ago

Selected Answer: C

The first is a Dataproc Serverless component that launches the feature engineering job
The second is a custom component wrapped in the create_custom_training_job_from_component utility that launches the custom model training job
Create a Vertex AI Pipelines job to link and run both components

👍 🔙 🚩 upvoted 1 times

👤 **fitri001** 6 months, 2 weeks ago

Selected Answer: C

The first is a Dataproc Serverless component that launches the feature engineering job
The second is a custom component wrapped in the create_custom_training_job_from_component utility that launches the custom model training job
Create a Vertex AI Pipelines job to link and run both components

👍 🔙 🚩 upvoted 1 times

> 👤 **fitri001** 6 months, 2 weeks ago
>
> A. Vertex AI Workbench notebook: While notebooks are a good way to prototype workflows, they are not ideal for production due to limitations in scalability and version control. Running everything sequentially also doesn't allow for potential parallelization of tasks.
> B. Apache Spark context in notebook: Similar to A, notebooks are not ideal for production. Additionally, running the model training with TensorFlow within the notebook ties the process to a specific framework, making it less flexible.
> D. Kubeflow pipelines with Spark context: This option gets close, but it's unnecessary to initiate a Spark context within the first component. Dataproc Serverless already handles the Spark environment for running PySpark code.
>
> 👍 🔙 🚩 upvoted 2 times

👤 **CHARLIE2108** 9 months ago

Selected Answer: C

I went with C

👍 🔙 🚩 upvoted 1 times

👤 **kalle_balle** 9 months, 3 weeks ago

Selected Answer: C

Vote C

👍 🔙 🚩 upvoted 1 times

👤 **36bdc1e** 9 months, 3 weeks ago

C
The best option for creating a scalable and maintainable production process that runs end-to-end and tracks the connections between steps, using prototype code to production, feature engineering code in PySpark that runs on Dataproc Serverless, and model training that is executed by using a Vertex AI custom training job, is to use the Kubeflow pipelines SDK to write code that specifies two components. The first is a Dataproc Serverless component that launches the feature engineering job. The second is a custom component wrapped in the create_custom_training_job_from_component utility that launches the custom model training job. This option allows you to leverage the power and simplicity of Kubeflow pipelines to orchestrate and automate your machine learning workflows on Vertex AI. Kubeflow pipelines is a platform that can build, deploy, and manage machine learning pipelines on Kubernetes.

👍 🔙 🚩 upvoted 1 times

👤 **pikachu007** 9 months, 3 weeks ago

Selected Answer: C

By using Kubeflow Pipelines, you establish a structured, scalable, and maintainable production process for end-to-end model development and deployment, ensuring proper orchestration, tracking, and integration with the chosen services.

👍 🔙 🚩 upvoted 3 times

vale_76_na_xxx 10 months ago

I go for C

👍 ↩ 🏳 upvoted 1 times

Start Learning for free