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 257 DISCUSSI...**

Actual exam question from Google's Professional Machine Learning Engineer
Question #: 257
Topic #: 1
[All Professional Machine Learning Engineer Questions]

You recently trained an XGBoost model on tabular data. You plan to expose the model for internal use as an HTTP microservice. After deployment, you expect a small number of incoming requests. You want to productionize the model with the least amount of effort and latency. What should you do?

A. Deploy the model to BigQuery ML by using CREATE MODEL with the BOOSTED_TREE_REGRESSOR statement, and invoke the BigQuery API from the microservice.

B. Build a Flask-based app. Package the app in a custom container on Vertex AI, and deploy it to Vertex AI Endpoints.

C. Build a Flask-based app. Package the app in a Docker image, and deploy it to Google Kubernetes Engine in Autopilot mode.

D. Use a prebuilt XGBoost Vertex container to create a model, and deploy it to Vertex AI Endpoints.

Show Suggested Answer

by 👤 **pikachu007** at *Jan. 13, 2024, 3:30 p.m.*

Comments

Type your comment...

Submit

pikachu007 `Highly Voted` 9 months, 1 week ago

Selected Answer: **D**

Prebuilt Container: It eliminates the need to build and manage a custom container, reducing development time and complexity.
Vertex AI Endpoints: It provides a managed serving infrastructure with low latency and high availability, optimizing performance for predictions.
Minimal Effort: It involves simple steps of creating a Vertex model and deploying it to an endpoint, streamlining the process.

👍 ↩ ⚑ upvoted 6 times

---

b1a8fae `Highly Voted` 9 months ago

Selected Answer: **D**

Bit lost here. I would discard buiding a Flask app since that is the opposite of "minimum effort". Between A and D, I guess a prebuilt container (D) involves less effort, but I am not 100% confident.

👍 ↩ ⚑ upvoted 5 times

AzureDP900 `Most Recent ⊘` 3 months, 2 weeks ago

Option D is correct :
Using a prebuilt XGBoost Vertex container (Option D) is the most straightforward approach. This container is specifically designed for running XGBoost models in production environments and can be easily deployed to Vertex AI Endpoints. This will allow you to expose your model as an HTTP microservice with minimal additional work.

👍 ↩ ⚑ upvoted 1 times
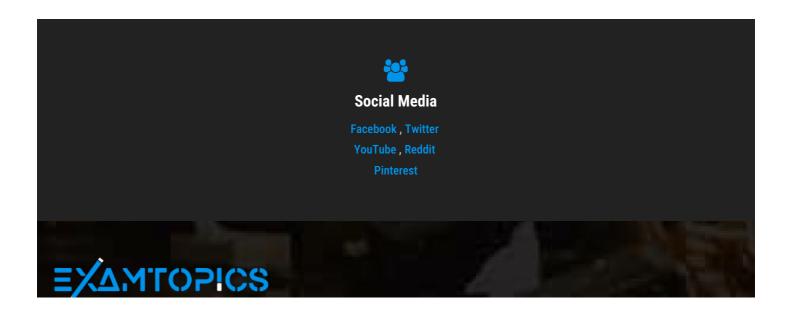
---

fitri001 6 months, 1 week ago
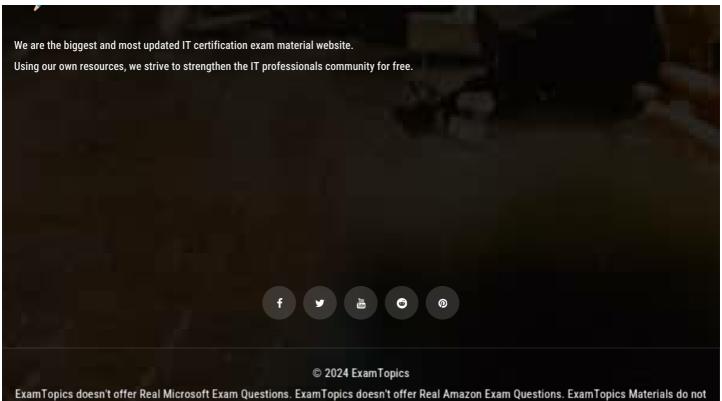
Selected Answer: **D**

Package the Model: Use a library like xgboost-server to create a minimal server for your XGBoost model. This package helps convert your model into a format suitable for serving predictions through an HTTP endpoint.
Deploy to Cloud Functions: Deploy the packaged model server as a Cloud Function on Google Cloud Platform (GCP). Cloud Functions are serverless, lightweight execution environments ideal for event-driven applications like microservices.
Configure Trigger: Set up an HTTP trigger for your Cloud Function, allowing it to be invoked through HTTP requests.

👍 ↩ ⚑ upvoted 2 times

Start Learning for free

We are the biggest and most updated IT certification exam material website.

Using our own resources, we strive to strengthen the IT professionals community for free.