

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 144 DISCUSSI...

Actual exam question from Google's Professional Machine Learning Engineer

Question #: 144

Topic #: 1

[\[All Professional Machine Learning Engineer Questions\]](#)

You are developing an image recognition model using PyTorch based on ResNet50 architecture. Your code is working fine on your local laptop on a small subsample. Your full dataset has 200k labeled images. You want to quickly scale your training workload while minimizing cost. You plan to use 4 V100 GPUs. What should you do?

- A. Create a Google Kubernetes Engine cluster with a node pool that has 4 V100 GPUs. Prepare and submit a TFJob operator to this node pool.
- B. Create a Vertex AI Workbench user-managed notebooks instance with 4 V100 GPUs, and use it to train your model.
- C. Package your code with Setuptools, and use a pre-built container. Train your model with Vertex AI using a custom tier that contains the required GPUs.
- D. Configure a Compute Engine VM with all the dependencies that launches the training. Train your model with Vertex AI using a custom tier that contains the required GPUs.



Show Suggested Answer

by [TNT87](#) at Feb. 13, 2023, 11:14 a.m.

### Comments

Type your comment...

Submit

  **John\_Pongthorn** Highly Voted  1 year, 8 months ago

**Selected Answer: C**




Custom trainer , don't overthink 1000%, this is google recommendation.

you don't need Vertex AI Workbench user-managed notebooks, Google Kubernetes Engine, Compute Engine at all , it is a waste of your effort

[https://cloud.google.com/vertex-ai/docs/training/configure-compute#specifying\\_gpus](https://cloud.google.com/vertex-ai/docs/training/configure-compute#specifying_gpus)

You can choose as your want

   upvoted 9 times

  **pawan94** Most Recent  6 months, 1 week ago

Why in the world would you setup a Compute engine VM, when your custom training job on vertex runs "serverless" atleast for the user side don't have to maintain the vm. You literally just have to select the region , machine type and accelerators that's all.

   upvoted 1 times

  **fitri001** 6 months, 2 weeks ago

**Selected Answer: C**

Pre-built container: Utilizing a pre-built PyTorch container image eliminates the need to manage dependencies within your container, saving time and simplifying the process.

Vertex AI custom tier: Vertex AI custom tiers allow you to configure a machine type with the desired GPUs (4 V100 in this case) and pay only for the resources you use. This is more cost-effective than managing a dedicated VM instance.

Setuptools packaging: Packaging your code with tools like Setuptools ensures all necessary libraries and scripts are included within the container, creating a self-contained training environment.

   upvoted 3 times

  **Mickey321** 11 months, 3 weeks ago

**Selected Answer: C**

Using Vertex AI allows you to easily leverage multiple GPUs without managing infrastructure yourself. The custom tier gives you control to specify 4 V100 GPUs.

Packaging with Setuptools and using a pre-built container ensures a consistent and portable environment with all dependencies readily available.

This approach minimizes overhead and cost by relying on Vertex AI's managed service instead of setting up your own Kubernetes cluster or VMs.

   upvoted 1 times

  **PST21** 1 year, 3 months ago

Correct Ans is C. Below mentioned why B is incorrect.

   upvoted 1 times

  **PST21** 1 year, 3 months ago

**Selected Answer: B**

Option B (using a Vertex AI Workbench user-managed notebooks instance with 4 V100 GPUs) is more suitable for interactive data exploration and experimentation rather than large-scale model training. Vertex AI Workbench is designed for collaborative data science, but using it for model training might not be the most efficient approach.

   upvoted 1 times

  **juliet** 1 year, 5 months ago

What is Supetools?

   upvoted 1 times

  **juliet** 1 year, 5 months ago

Found it. Python package that provides a mechanism for packaging, distributing, and installing Python libraries or modules.

   upvoted 1 times

  **M25** 1 year, 5 months ago

**Selected Answer: C**

"Vertex AI provides flexible and scalable hardware and secured infrastructure to train PyTorch based deep learning models with pre-built containers and custom containers. (...) use PyTorch ResNet-50 as the example model and train it on ImageNet validation data (50K images) to measure the training performance for different training strategies":

<https://cloud.google.com/blog/products/ai-machine-learning/efficient-pytorch-training-with-vertex-ai>

   upvoted 1 times

  **M25** 1 year, 5 months ago

There is no indication otherwise why there would be a need for full control over the environment, provided by "user-managed workbooks" within the Vertex AI Workbench [Option B], except for the "plan to use 4 V100 GPUs", but one can do that with "user-managed notebooks" as well.

do that with "managed workbooks" as well:  
[https://cloud.google.com/vertex-ai/docs/workbench/notebook-solution#control\\_your\\_hardware\\_and\\_framework\\_from\\_jupyterlab](https://cloud.google.com/vertex-ai/docs/workbench/notebook-solution#control_your_hardware_and_framework_from_jupyterlab)

   upvoted 1 times

  **frangm23** 1 year, 6 months ago

Can someone explain why is B wrong?

   upvoted 2 times



  **andresvelasco** 1 year, 1 month ago

Very likely because of the consideration:

"You want to quickly scale your training workload while minimizing cost"

But I agree with you ... I chose B (notebook) thinking the question was more oriented to quickly achieving an MVP.

   upvoted 1 times

  **TNT87** 1 year, 6 months ago

**Selected Answer: C**

Answer C

Option A involves using Google Kubernetes Engine, which is a platform for deploying, managing, and scaling containerized applications. However, it requires more setup time and knowledge of Kubernetes, which might not be ideal for quickly scaling up training workloads. Furthermore, the use of the TensorFlow Job operator seems inappropriate for a PyTorch-based model.

   upvoted 1 times

  **wlts** 1 year, 7 months ago

Select C



   upvoted 1 times

  **wlts** 1 year, 7 months ago

The TFJob operator is designed for TensorFlow workloads, not PyTorch. So option A is out.

Vertex AI Workbench is primarily designed for interactive work with Jupyter Notebooks and not optimized for large-scale, long-running model training. Moreover, it may not provide the same level of cost optimization as Vertex AI Training, which automatically provisions and manages resources, and can scale down when not in use. So option B also out.

   upvoted 1 times

  **TNT87** 1 year, 8 months ago

C. Package your code with Setuptools, and use a pre-built container. Train your model with Vertex AI using a custom tier that contains the required GPUs.

The recommended approach to scale the training workload while minimizing cost would be to package the code with Setuptools and use a pre-built container, then train the model with Vertex AI using a custom tier that contains the required GPUs. This approach allows for quick and easy scaling of the training workload while minimizing infrastructure management costs.

   upvoted 1 times

  **FherRO** 1 year, 8 months ago

**Selected Answer: A**



Vote for A, as you need to scale

   upvoted 1 times

  **alelamb** 1 year, 8 months ago

It clearly says a Pytorch model, you cannot use a TFJob

   upvoted 2 times

  **Scipione\_** 1 year, 8 months ago

**Selected Answer: B**



It's B according to me, since VertexAI Notebook has all dependencies for PyTorch that is the fastest solution

   upvoted 2 times

  **tavva\_prudhvi** 1 year, 3 months ago

It involves using a managed notebooks instance, which might have limitations in terms of customizability and flexibility compared to a containerized approach.

   upvoted 1 times

  **TNT87** 1 year, 8 months ago

**Selected Answer: A**

Google Kubernetes Engine (GKE) is a powerful and easy-to-use platform for deploying and managing containerized applications. It allows you to create a cluster of virtual machines that are pre-configured with the necessary dependencies and resources to run your machine learning workloads. By creating a GKE cluster with a node pool that has 4 V100 GPUs, you can take advantage of the powerful processing capabilities of these GPUs to train your model quickly and efficiently.

you can take advantage of the powerful processing capabilities of these GPUs to train your model quickly and efficiently.

You can then use the Kubernetes Framework such as TFJob operator to submit the job of training your model, which will automatically distribute the workload across the available GPUs.

References:

Google Kubernetes Engine

TFJob operator



Vertex AI

   upvoted 2 times

  **alelamb** 1 year, 8 months ago

It clearly says a Pytorch model, you cannot use a TFjob

   upvoted 1 times

  **TNT87** 1 year, 8 months ago

Answer C

   upvoted 2 times

**Start Learning for free**



## Social Media

[Facebook](#) , [Twitter](#)

[YouTube](#) , [Reddit](#)

[Pinterest](#)



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.



© 2024 ExamTopics

ExamTopics doesn't offer Real Microsoft Exam Questions. ExamTopics doesn't offer Real Amazon Exam Questions. ExamTopics Materials do not contain actual questions and answers from Cisco's Certification Exams.

CFA Institute does not endorse, promote or warrant the accuracy or quality of ExamTopics. CFA® and Chartered Financial Analyst® are registered trademarks owned by CFA Institute.