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 212 DISCUSSI...

Actual exam question from Google's Professional Machine Learning Engineer

Question #: 212

Topic #: 1

[All Professional Machine Learning Engineer Questions]

You are pre-training a large language model on Google Cloud. This model includes custom TensorFlow operations in the training loop. Model training will use a large batch size, and you expect training to take several weeks. You need to configure a training architecture that minimizes both training time and compute costs. What should you do?

A. Implement 8 workers of a2-megagpu-16g machines by using tf.distribute.MultiWorkerMirroredStrategy.

B. Implement a TPU Pod slice with -accelerator-type=v4-l28 by using tf.distribute.TPUStrategy.

C. Implement 16 workers of c2d-highcpu-32 machines by using tf.distribute.MirroredStrategy.

D. Implement 16 workers of a2-highgpu-8g machines by using tf.distribute.MultiWorkerMirroredStrategy.

Show Suggested Answer

by 👤 pikachu007 at *Jan. 13, 2024, 5:54 a.m.*

Comments

Type your comment...

Submit

⊟ 👤 **baimus** 1 month, 1 week ago

This could be A or D, because they both will perform will with custom Tensorflow operations. A is likely to be better with large batch sizes, which require bigger GPUs, so I went A.

👍 ↩ 🚩 **upvoted 1 times**

⊟ 👤 **AK2020** 2 months, 2 weeks ago

B is not correct as TPUs not suitable for TensorFlow custom operations and C doesn't make any sense. A or D?. I would go with A

👍 ↩ 🚩 **upvoted 3 times**

⊟ 👤 **info_appsatori** 4 months, 1 week ago

Should be A or D. TPU is ok, but TPUs not suitable for TensorFlow custom operations.

👍 ↩ 🚩 **upvoted 2 times**

⊟ 👤 **ccb23cc** 4 months, 1 week ago

B. TPU Acceleration: the question says that uses Tensorflow custom operations in the main loop and Google documentation literatelly says about TPU use: "Models with no custom TensorFlow/PyTorch/JAX operations inside the main training loop"

C. High-CPU Machines: Make no sense because tell you to use a cpu (which does not help us in this case)

So the correct answer is between A and D. However the question says that they are planning to use a large batch size so we need RAM. Therefore we should take the one with more.

Correct answer: Option A

👍 ↩ 🚩 **upvoted 3 times**

⊟ 👤 **fitri001** 6 months, 1 week ago

TPU Acceleration: TPUs are specifically designed for machine learning workloads and offer significant speedups compared to GPUs or CPUs, especially for large models like yours. Utilizing a TPU Pod slice provides access to a collection of interconnected TPUs for efficient parallel training.
tf.distribute.TPUStrategy: This strategy is specifically designed to work with TPUs in TensorFlow. It handles data distribution, model replication, and gradient aggregation across the TPU cores, enabling efficient training with custom TensorFlow operations.

👍 ↩ 🚩 **upvoted 1 times**

  ⊟ 👤 **fitri001** 6 months, 1 week ago

  why not the others?
  A. MultiWorkerMirroredStrategy with GPUs: While GPUs offer some acceleration, TPUs are generally better suited for large language model pre-training due to their architectural optimizations. Additionally, managing 8 workers across separate machines can introduce communication overhead compared to a tightly coupled TPU Pod.
  C. MirroredStrategy with High-CPU Machines: CPU-based training would be significantly slower than TPUs or even GPUs for a large language model. While the high CPU count might seem beneficial for custom operations, the overall training speed would still be limited.
  D. MultiWorkerMirroredStrategy with Multiple High-GPU Machines: Similar to option A, using multiple high-GPU machines with this strategy would incur communication overhead and potentially be less cost-effective compared to a single TPU Pod slice.

  👍 ↩ 🚩 **upvoted 2 times**

⊟ 👤 **BlehMaks** 9 months, 1 week ago

It should be TPU but i'm a bit concerned about this point from Google documentation:
Models with no custom TensorFlow/PyTorch/JAX operations inside the main training loop
https://cloud.google.com/tpu/docs/intro-to-tpu#TPU

👍 ↩ 🚩 **upvoted 2 times**

⊟ 👤 **b1a8fae** 9 months, 1 week ago

B.
NGL quite lost on this one but if the training set is big enough to span over several weeks I would go with the most powerful resource (TPUs) but I might be completely wrong.

👍 ↩ 🚩 **upvoted 3 times**

⊟ 👤 **pikachu007** 9 months, 1 week ago

TPU Advantages:

Highly Specialized: TPUs (Tensor Processing Units) are custom-designed hardware accelerators specifically optimized for machine learning workloads, particularly those involving large batch sizes and matrix-heavy computations, common in large language models.
Exceptional Performance: TPUs can significantly outperform CPUs and GPUs in terms of speed and efficiency for these types of tasks.
Cost-Effective: While TPUs might have a higher hourly cost, their exceptional performance often leads to lower overall costs due to faster training times and reduced resource usage.
TPU Pod Slice:

Scalability: TPU Pod slices allow you to distribute training across multiple TPUv4 chips for even greater performance and scalability.
Custom Operations: The tf.distribute.TPUStrategy ensures compatibility with custom TensorFlow operations,

👍 ↩ ⚑ upvoted 4 times

Start Learning for free