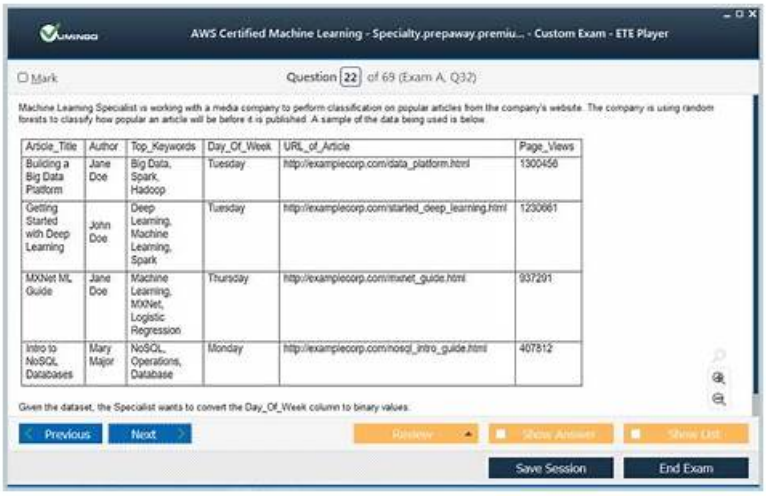
Reliable AWS-Certified-Machine-Learning-Specialty Braindumps Sheet - Pass Guaranteed Quiz Amazon AWS-Certified-Machine-Learning-Specialty First-grade Reliable Test Testking



2025 Latest TestPassed AWS-Certified-Machine-Learning-Specialty PDF Dumps and AWS-Certified-Machine-Learning-Specialty Exam Engine Free Share: https://drive.google.com/open?id=1QOE5hbE1EOGshAMTB9R-PRIy9bgZVTLE

Before you buy our AWS-Certified-Machine-Learning-Specialty study questions you can have a free download and tryout and you can have an understanding of our product by visiting our pages of our product on the website. The content of our AWS-Certified-Machine-Learning-Specialty guide torrent is easy to be mastered and has simplified the important information. Our AWS-Certified-Machine-Learning-Specialty study questions convey more important information with less amount of questions and answers and thus make the learning relaxing and efficient.

Amazon MLS-C01 certification exam is designed to test the knowledge and skills of candidates who are interested in becoming an AWS Certified Machine Learning Specialist. AWS-Certified-Machine-Learning-Specialty exam is intended to validate the candidate's ability to design, implement, deploy, and maintain machine learning solutions for a variety of business use cases. AWS-Certified-Machine-Learning-Specialty Exam is suitable for individuals who have a deep understanding of machine learning algorithms, statistical modeling, data analysis, and data preprocessing techniques.

>> Reliable AWS-Certified-Machine-Learning-Specialty Braindumps Sheet <<

Reliable Amazon AWS-Certified-Machine-Learning-Specialty Test Testking, New AWS-Certified-Machine-Learning-Specialty Exam Simulator

Our professional experts have compiled the AWS-Certified-Machine-Learning-Specialty exam questions carefully and skillfully to let all of our worthy customers understand so that even an average candidate can learn the simplified information on the syllabus contents and grasp it to ace exam by the first attempt. It is the easiest track that can lead you to your ultimate destination with our AWS-Certified-Machine-Learning-Specialty Practice Engine. And as our pass rate of the AWS-Certified-Machine-Learning-Specialty learning guide is high as 98% to 100%, you will pass the exam for sure.

Salary for AWS Certified Machine Learning Specialty Exam certified

- United States: 107,786 USD
- England: 82,930 Pound
- Europe: 97,902 Euro
- India: 712,503 INR

Amazon AWS Certified Machine Learning - Specialty Sample Questions

(Q117-Q122):

NEW QUESTION # 117
An ecommerce company wants to train a large image classification model with 10.000 classes. The company runs multiple model training iterations and needs to minimize operational overhead and cost. The company also needs to avoid loss of work and model retraining.
Which solution will meet these requirements?

- A. Use Amazon EC2 Spot Instances to run the training jobs. Use a Spot Instance interruption notice to save a snapshot of the model to Amazon S3 before an instance is terminated.
- B. Use managed spot training in Amazon SageMaker. Launch the training jobs with checkpointing enabled.
- C. Create the training jobs as AWS Batch jobs that use Amazon EC2 Spot Instances in a managed compute environment.
- D. Use AWS Lambda to run the training jobs. Save model weights to Amazon S3.

Answer: B

Explanation:
Amazon SageMaker managed spot training allows for cost-effective training by utilizing Spot Instances, which are lower-cost EC2 instances that can be interrupted when demand is high. By enabling checkpointing in SageMaker, the company can save intermediate model states to Amazon S3, allowing training to resume from the last checkpoint if interrupted. This solution minimizes operational overhead by automating the checkpointing process and resuming work after interruptions, reducing the need for retraining from scratch.
This setup provides a reliable and cost-efficient approach to training large models with minimal operational overhead and risk of data loss.

NEW QUESTION # 118
A manufacturing company wants to create a machine learning (ML) model to predict when equipment is likely to fail. A data science team already constructed a deep learning model by using TensorFlow and a custom Python script in a local environment. The company wants to use Amazon SageMaker to train the model.
Which TensorFlow estimator configuration will train the model MOST cost-effectively?

- A. Turn on SageMaker Training Compiler by adding compiler_config=TrainingCompilerConfig() as a parameter. Pass the script to the estimator in the call to the TensorFlow fit() method.
- B. Turn on SageMaker Training Compiler by adding compiler_config=TrainingCompilerConfig() as a parameter. Set the MaxWaitTimeInSeconds parameter to be equal to the MaxRuntimeInSeconds parameter. Pass the script to the estimator in the call to the TensorFlow fit() method.
- C. Adjust the training script to use distributed data parallelism. Specify appropriate values for the distribution parameter. Pass the script to the estimator in the call to the TensorFlow fit() method.
- D. Turn on SageMaker Training Compiler by adding compiler_config=TrainingCompilerConfig() as a parameter. Turn on managed spot training by setting the use_spot_instances parameter to True. Pass the script to the estimator in the call to the TensorFlow fit() method.

Answer: D

Explanation:
The TensorFlow estimator configuration that will train the model most cost-effectively is to turn on SageMaker Training Compiler by adding compiler_config=TrainingCompilerConfig() as a parameter, turn on managed spot training by setting the use_spot_instances parameter to True, and pass the script to the estimator in the call to the TensorFlow fit() method. This configuration will optimize the model for the target hardware platform, reduce the training cost by using Amazon EC2 Spot Instances, and use the custom Python script without any modification.
SageMaker Training Compiler is a feature of Amazon SageMaker that enables you to optimize your TensorFlow, PyTorch, and MXNet models for inference on a variety of target hardware platforms.
SageMaker Training Compiler can improve the inference performance and reduce the inference cost of your models by applying various compilation techniques, such as operator fusion, quantization, pruning, and graph optimization. You can enable SageMaker Training Compiler by adding compiler_config=TrainingCompilerConfig() as a parameter to the TensorFlow estimator constructor1.
Managed spot training is another feature of Amazon SageMaker that enables you to use Amazon EC2 Spot Instances for training your machine learning models. Amazon EC2 Spot Instances let you take advantage of unused EC2 capacity in the AWS Cloud. Spot Instances are available at up to a 90% discount compared to On- Demand prices. You can use Spot Instances for various fault-tolerant and flexible applications. You can enable managed spot training by setting the use_spot_instances parameter to True and specifying the max_wait and max_run parameters in the TensorFlow estimator constructor2.

The TensorFlow estimator is a class in the SageMaker Python SDK that allows you to train and deploy TensorFlow models on SageMaker. You can use the TensorFlow estimator to run your own Python script on SageMaker, without any modification. You can pass the script to the estimator in the call to the TensorFlow fit() method, along with the location of your input data. The fit() method starts a SageMaker training job and runs your script as the entry point in the training containers3.
The other options are either less cost-effective or more complex to implement. Adjusting the training script to use distributed data parallelism would require modifying the script and specifying appropriate values for the distribution parameter, which could increase the development time and complexity. Setting the MaxWaitTimeInSeconds parameter to be equal to the MaxRuntimeInSeconds parameter would not reduce the cost, as it would only specify the maximum duration of the training job, regardless of the instance type.
References:
* 1: Optimize TensorFlow, PyTorch, and MXNet models for deployment using Amazon SageMaker Training Compiler | AWS Machine Learning Blog
* 2: Managed Spot Training: Save Up to 90% On Your Amazon SageMaker Training Jobs | AWS Machine Learning Blog
* 3: sagemaker.tensorflow - sagemaker 2.66.0 documentation

NEW QUESTION # 119
An automotive company uses computer vision in its autonomous cars. The company trained its object detection models successfully by using transfer learning from a convolutional neural network (CNN). The company trained the models by using PyTorch through the Amazon SageMaker SDK.
The vehicles have limited hardware and compute power. The company wants to optimize the model to reduce memory, battery, and hardware consumption without a significant sacrifice in accuracy.
Which solution will improve the computational efficiency of the models?

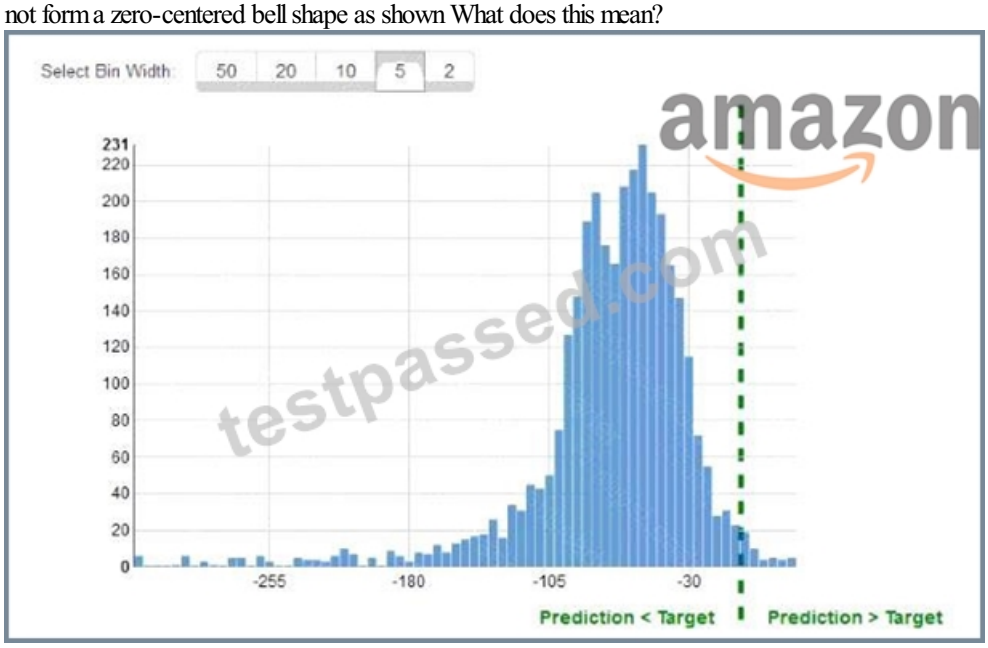
- A. Use Amazon SageMaker Model Monitor to gain visibility into the ModelLatency metric and OverheadLatency metric of the model after the company deploys the model. Increase the model learning rate. Run a new training job.
- B. Use Amazon SageMaker Debugger to gain visibility into the training weights, gradients, biases, and activation outputs. Compute the filter ranks based on the training information. Apply pruning to remove the low-ranking filters. Set the new weights based on the pruned set of filters. Run a new training job with the pruned model.
- C. Use Amazon CloudWatch metrics to gain visibility into the SageMaker training weights, gradients, biases, and activation outputs. Compute the filter ranks based on the training information. Apply pruning to remove the low-ranking filters. Set new weights based on the pruned set of filters. Run a new training job with the pruned model.
- D. Use Amazon SageMaker Ground Truth to build and run data labeling workflows. Collect a larger labeled dataset with the labelling workflows. Run a new training job that uses the new labeled data with previous training data.

Answer: B

Explanation:
The solution C will improve the computational efficiency of the models because it uses Amazon SageMaker Debugger and pruning, which are techniques that can reduce the size and complexity of the convolutional neural network (CNN) models. The solution C involves the following steps:
Use Amazon SageMaker Debugger to gain visibility into the training weights, gradients, biases, and activation outputs. Amazon SageMaker Debugger is a service that can capture and analyze the tensors that are emitted during the training process of machine learning models. Amazon SageMaker Debugger can provide insights into the model performance, quality, and convergence. Amazon SageMaker Debugger can also help to identify and diagnose issues such as overfitting, underfitting, vanishing gradients, and exploding gradients1.
Compute the filter ranks based on the training information. Filter ranking is a technique that can measure the importance of each filter in a convolutional layer based on some criterion, such as the average percentage of zero activations or the L1-norm of the filter weights. Filter ranking can help to identify the filters that have little or no contribution to the model output, and thus can be removed without affecting the model accuracy2.
Apply pruning to remove the low-ranking filters. Pruning is a technique that can reduce the size and complexity of a neural network by removing the redundant or irrelevant parts of the network, such as neurons, connections, or filters. Pruning can help to improve the computational efficiency, memory usage, and inference speed of the model, as well as to prevent overfitting and improve generalization3.
Set the new weights based on the pruned set of filters. After pruning, the model will have a smaller and simpler architecture, with fewer filters in each convolutional layer. The new weights of the model can be set based on the pruned set of filters, either by initializing them randomly or by fine-tuning them from the original weights4.
Run a new training job with the pruned model. The pruned model can be trained again with the same or a different dataset, using the same or a different framework or algorithm. The new training job can use the same or a different configuration of Amazon SageMaker, such as the instance type, the hyperparameters, or the data ingestion mode. The new training job can also use Amazon SageMaker Debugger to monitor and analyze the training process and the model quality5.

The other options are not suitable because:
Option A: Using Amazon CloudWatch metrics to gain visibility into the SageMaker training weights, gradients, biases, and activation outputs will not be as effective as using Amazon SageMaker Debugger.
Amazon CloudWatch is a service that can monitor and observe the operational health and performance of AWS resources and applications. Amazon CloudWatch can provide metrics, alarms, dashboards, and logs for various AWS services, including Amazon SageMaker. However, Amazon CloudWatch does not provide the same level of granularity and detail as Amazon SageMaker Debugger for the tensors that are emitted during the training process of machine learning models. Amazon CloudWatch metrics are mainly focused on the resource utilization and the training progress, not on the model performance, quality, and convergence6.
Option B: Using Amazon SageMaker Ground Truth to build and run data labeling workflows and collecting a larger labeled dataset with the labeling workflows will not improve the computational efficiency of the models. Amazon SageMaker Ground Truth is a service that can create high-quality training datasets for machine learning by using human labelers. A larger labeled dataset can help to improve the model accuracy and generalization, but it will not reduce the memory, battery, and hardware consumption of the model. Moreover, a larger labeled dataset may increase the training time and cost of the model7.
Option D: Using Amazon SageMaker Model Monitor to gain visibility into the ModelLatency metric and OverheadLatency metric of the model after the company deploys the model and increasing the model learning rate will not improve the computational efficiency of the models. Amazon SageMaker Model Monitor is a service that can monitor and analyze the quality and performance of machine learning models that are deployed on Amazon SageMaker endpoints. The ModelLatency metric and the OverheadLatency metric can measure the inference latency of the model and the endpoint, respectively. However, these metrics do not provide any information about the training weights, gradients, biases, and activation outputs of the model, which are needed for pruning. Moreover, increasing the model learning rate will not reduce the size and complexity of the model, but it may affect the model convergence and accuracy.
1: Amazon SageMaker Debugger
2: Pruning Convolutional Neural Networks for Resource Efficient Inference
3: Pruning Neural Networks: A Survey
4: Learning both Weights and Connections for Efficient Neural Networks
5: Amazon SageMaker Training Jobs
6: Amazon CloudWatch Metrics for Amazon SageMaker
7: Amazon SageMaker Ground Truth
Amazon SageMaker Model Monitor

NEW QUESTION # 120
While reviewing the histogram for residuals on regression evaluation data a Machine Learning Specialist notices that the residuals do not form a zero-centered bell shape as shown What does this mean?



- A. There are too many variables in the model
- B. The model is predicting its target values perfectly.
- C. The dataset cannot be accurately represented using the regression model
- D. The model might have prediction errors over a range of target values.

Answer: B

NEW QUESTION # 121
A logistics company needs a forecast model to predict next month's inventory requirements for a single item in 10 warehouses. A machine learning specialist uses Amazon Forecast to develop a forecast model from 3 years of monthly dat a. There is no missing data. The specialist selects the DeepAR+ algorithm to train a predictor. The predictor means absolute percentage error (MAPE) is much larger than the MAPE produced by the current human forecasters.
Which changes to the CreatePredictor API call could improve the MAPE? (Choose two.)

- A. Set PerformAutoML to true.
- B. Set FeaturizationMethodName to filling.
- C. Set ForecastFrequency to W for weekly.
- D. Set ForecastHorizon to 4.
- E. Set PerformHPO to true.

Answer: C,E

NEW QUESTION # 122
.....

Specialty Vce

- Pass AWS-Certified-Machine-Learning-Specialty Exam with Professional Reliable AWS-Certified-Machine-Learning-Specialty Braindumps Sheet by www.practicevce.com □ Open □ www.practicevce.com □ and search for ☀ AWS-Certified-Machine-Learning-Specialty □☀□ to download exam materials for free □AWS-Certified-Machine-Learning-Specialty Relevant Exam Dumps
- study.stcs.edu.np, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, myportal.utt.edu.tt, course.cost-ernst.eu, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, www.stes.tyc.edu.tw, Disposable vapes

BONUS!!! Download part of TestPassed AWS-Certified-Machine-Learning-Specialty dumps for free: https://drive.google.com/open?id=1QOE5hbE1EOGshAMTB9R-PRIy9bgZVTLE