Amazon.MLA-C01.v2026-06-26.q106

Exam Code:	MLA-C01
Exam Name:	AWS Certified Machine Learning Engineer - Associate
Certification Provider:	Amazon
Free Question Number:	106
Version:	v2026-06-26
# of views:	140
# of Questions views:	1060
https://www.freepdfdumps.com/Amazon.MLA-C01.v2026-06-26.q106.html	

NEW QUESTION: 1
A company uses Amazon SageMaker for its ML workloads. The company's ML engineer receives a 50 MB Apache Parquet data file to build a fraud detection model. The file includes several correlated columns that are not required.
What should the ML engineer do to drop the unnecessary columns in the file with the LEAST effort?
A. Download the file to a local workstation. Perform one-hot encoding by using a custom Python script.
B. Create an Apache Spark job that uses a custom processing script on Amazon EMR.
C. Create a SageMaker processing job by calling the SageMaker Python SDK.
D. Create a data flow in SageMaker Data Wrangler. Configure a transform step.
Answer: D (LEAVE A REPLY)
SageMaker Data Wrangler provides a no-code/low-code interface for preparing and transforming data, including dropping unnecessary columns. By creating a data flow and configuring a transform step, the ML engineer can easily remove correlated or unneeded columns from the Parquet file with minimal effort. This approach avoids the need for custom coding or managing additional infrastructure.

NEW QUESTION: 2
An ML engineer trained an ML model on Amazon SageMaker to detect automobile accidents from dosed- circuit TV footage. The ML engineer used SageMaker Data Wrangler to create a training dataset of images of accidents and non-accidents.
The model performed well during training and validation. However, the model is underperforming in production because of variations in the quality of the images from various cameras.
Which solution will improve the model ' s accuracy in the LEAST amount of time?
A. Collect more images from all the cameras. Use Data Wrangler to prepare a new training dataset.
B. Recreate the training dataset by using the Data Wrangler corrupt image transform. Specify the impulse noise option.
C. Recreate the training dataset by using the Data Wrangler enhance image contrast transform. Specify the Gamma contrast option.
D. Recreate the training dataset by using the Data Wrangler resize image transform. Crop all images to the same size.
Answer: B (LEAVE A REPLY)
The model is underperforming in production due to variations in image quality from different cameras. Using the corrupt image transform with the impulse noise option in SageMaker Data Wrangler simulates real-world noise and variations in the training dataset. This approach helps the model

become more robust to inconsistencies in image quality, improving its accuracy in production without the need to collect and process new data, thereby saving time.

NEW QUESTION: 3
A company is creating an application that will recommend products for customers to purchase. The application will make API calls to Amazon Q Business. The company must ensure that responses from Amazon Q Business do not include the name of the company's main competitor.
Which solution will meet this requirement?
A. Configure the competitor's name as a blocked phrase in Amazon Q Business.
B. Configure an Amazon Q Business retriever to exclude the competitor's name.
C. Configure an Amazon Kendra retriever for Amazon Q Business to build indexes that exclude the competitor's name.
D. Configure document attribute boosting in Amazon Q Business to deprioritize the competitor's name.
Answer: A (LEAVE A REPLY)
Amazon Q Business provides built-in guardrails to control and constrain model responses. One such control is the ability to define blocked phrases, which explicitly prevents specified terms from appearing in generated responses.
Configuring the competitor's name as a blocked phrase guarantees that Amazon Q Business will never include that name in responses, fully meeting the requirement. This approach is deterministic and does not rely on ranking or retrieval behavior.
Option B is incorrect because retrievers control what documents are fetched, not how responses are generated.
Option C adds unnecessary complexity and does not guarantee exclusion in generated answers. Option D only deprioritizes content and does not prevent it from appearing.
Therefore, blocking the competitor's name is the correct solution.

NEW QUESTION: 4
An ML engineer is developing a neural network to run on new user data. The dataset has dozens of floating- point features. The dataset is stored as CSV objects in an Amazon S3 bucket. Most objects and columns are missing at least one value. All features are relatively uniform except for a small number of extreme outliers.
The ML engineer wants to use Amazon SageMaker Data Wrangler to handle missing values before passing the dataset to the neural network.
Which solution will provide the MOST complete data?
A. Drop samples that are missing values.
B. Impute missing values with the mean value.
C. Impute missing values with the median value.
D. Drop columns that are missing values.
Answer: (SHOW ANSWER)
The primary goal is to produce the most complete dataset while handling missing values and extreme outliers appropriately. Dropping samples (Option A) or columns (Option D) would reduce data completeness and potentially remove valuable information, which contradicts the requirement. Imputation is therefore the correct approach. Between mean and median imputation, AWS ML best practices recommend using the median when features contain outliers. The mean is sensitive to extreme values and can be skewed significantly, leading to imputed values that are not representative of the typical data distribution.
In contrast, the median is robust to outliers, making it a better statistical estimator for central tendency in such datasets.
Amazon SageMaker Data Wrangler supports median imputation as a built-in transformation, enabling ML engineers to handle missing values consistently across large tabular datasets without custom code. This approach preserves all rows and columns while minimizing distortion caused by extreme values, which is particularly important for neural networks that are sensitive to input distributions.

Therefore, imputing missing values with the median value provides the most complete and statistically appropriate dataset for training.

NEW QUESTION: 5
An ML engineer is setting up an Amazon SageMaker AI pipeline for an ML model. The pipeline must automatically initiate a re-training job if any data drift is detected.
How should the ML engineer set up the pipeline to meet this requirement?
A. Use an AWS Glue crawler and an AWS Glue extract, transform, and load (ETL) job to detect data drift.
Use AWS Glue triggers to automate the retraining job.
B. Use Amazon Managed Service for Apache Flink to detect data drift. Use an AWS Lambda function to automate the re-training job.
C. Use SageMaker Model Monitor to detect data drift. Use an AWS Lambda function to automate the re- training job.
D. Use Amazon Quick Suite (previously known as Amazon QuickSight) anomaly detection to detect data drift. Use an AWS Step Functions workflow to automate the re-training job.
Answer: (SHOW ANSWER)
AWS provides Amazon SageMaker Model Monitor as a native solution for detecting data drift and model quality issues in production ML pipelines. Model Monitor continuously analyzes incoming inference data and compares it with baseline training data to identify schema drift, feature distribution drift, and data quality anomalies.
When drift thresholds are violated, Model Monitor generates CloudWatch metrics and alerts. These alerts can directly trigger an AWS Lambda function, which can then programmatically initiate a SageMaker retraining job or start a SageMaker Pipeline execution. This design is explicitly documented by AWS as the recommended architecture for automated retraining workflows.
Option A is incorrect because AWS Glue is a data integration service and does not provide ML-specific drift detection capabilities.
Option B is incorrect because Apache Flink is designed for stream processing, not ML data drift detection.
Option D is incorrect because Amazon QuickSight anomaly detection is intended for business intelligence metrics, not ML feature drift.
Therefore, using SageMaker Model Monitor with AWS Lambda automation is the correct, AWS-native solution for drift-driven retraining.

NEW QUESTION: 6
An ML engineer has trained an ML model by using Amazon SageMaker AI. The ML engineer determines that the model is overfitting and that the training data contains unnecessary features. The ML engineer must reduce the overfitting and the impact of the unnecessary features.
Which solution will meet these requirements?
A. Apply L1 regularization to the training data. Retrain the model.
B. Use SageMaker Debugger to apply L1 regularization to the running model.
C. Increase the number of training iterations. Retrain the model.
D. Decrease the number of training iterations. Retrain the model.
Answer: (SHOW ANSWER)
Option A is correct because AWS documentation states that regularization helps prevent linear models from overfitting , and specifically that L1 regularization reduces the number of features used in the model by pushing small feature weights to zero . AWS further explains that L1 regularization produces sparse models and reduces noise. That is the exact combination needed in this scenario: the model is overfitting, and the data contains unnecessary features whose impact should be reduced.
AWS guidance on overfitting also says that to reduce model overfitting, you should reduce model flexibility by using feature selection and increasing the amount of regularization . L1 regularization is especially suitable here because it does both in practice: it acts as a regularizer and also effectively performs feature selection by shrinking unhelpful feature coefficients to zero. Even though the option wording says "apply L1

regularization to the training data," the only answer choice that aligns with AWS-documented techniques for both overfitting reduction and unnecessary-feature suppression is L1 regularization with retraining .
The other options do not meet both requirements. SageMaker Debugger monitors and helps analyze training jobs, but it is not the feature that "applies L1 regularization" to a running model. Increasing training iterations can make overfitting worse, not better. Decreasing training iterations may reduce overfitting in some cases, but it does not specifically address the presence of unnecessary features. Therefore, the AWS- aligned choice that best reduces overfitting and the impact of irrelevant features is A .

NEW QUESTION: 7
An ML engineering team is spread across multiple locations. When the lead ML engineer opens an Amazon SageMaker AI notebook, the ML engineer does not see the latest merged notebook made by other team members from a Git repository.
The lead ML engineer must see the latest SageMaker AI notebook updates.
Which solution will meet this requirement?
A. Run the !git pull origin master command.
B. Run the !git commit command.
C. Run the !git push origin master command.
D. Run the !git branch command.
Answer: A (LEAVE A REPLY)
Option A is correct because the issue is that the lead ML engineer's local notebook environment does not yet contain the latest merged changes from the shared Git repository. In Git, the operation used to fetch remote updates and merge them into the local branch is git pull . AWS documentation for SageMaker notebook collaboration explains that Git repositories can be associated with SageMaker notebook instances so distributed teams can share notebooks and collaborate in a source-control environment.
AWS documentation is even more explicit in the SageMaker Git operations guide: to fetch notebooks committed by other users , you should do a pull from the project repository . That is exactly the situation described in the question. Other team members have already merged their notebook changes into the repository, and the lead engineer needs the latest notebook updates in the local SageMaker notebook environment. Therefore, running !git pull origin master is the correct command among the available choices.
The other options do not solve the problem. git commit records local changes in the user's own repository history; it does not download new remote content. git push origin master uploads local commits to the remote repository; it is used when you want to send your own changes outward, not retrieve teammates' updates. git branch only lists or manages branches and does not synchronize notebook content. Because the requirement is to see the latest merged notebook made by others , the necessary operation is a pull from the remote repository. So the best verified answer is A .

NEW QUESTION: 8
A company is planning to use Amazon SageMaker to make classification ratings that are based on images.
The company has 6 ## of training data that is stored on an Amazon FSx for NetApp ONTAP system virtual machine (SVM). The SVM is in the same VPC as SageMaker.
An ML engineer must make the training data accessible for ML models that are in the SageMaker environment.
Which solution will meet these requirements?
A. Mount the FSx for ONTAP file system as a volume to the SageMaker Instance.
B. Create an Amazon S3 bucket. Use Mountpoint for Amazon S3 to link the S3 bucket to the FSx for ONTAP file system.
C. Create a catalog connection from SageMaker Data Wrangler to the FSx for ONTAP file system.
D. Create a direct connection from SageMaker Data Wrangler to the FSx for ONTAP file system.
Answer: A (LEAVE A REPLY)

Amazon FSx for NetApp ONTAP allows mounting the file system as a network-attached storage (NAS) volume. Since the FSx for ONTAP file system and SageMaker instance are in the same VPC, you can directly mount the file system to the SageMaker instance. This approach ensures efficient access to the 6 TB of training data without the need to duplicate or transfer the data, meeting the requirements with minimal complexity and operational overhead.

NEW QUESTION: 9
A company uses AWS CodePipeline to orchestrate a continuous integration and continuous delivery (CI/CD) pipeline for ML models and applications.
Select and order the steps from the following list to describe a CI/CD process for a successful deployment.
Select each step one time. (Select and order FIVE.)
. CodePipeline deploys ML models and applications to production.
CodePipeline detects code changes and starts to build automatically.
. Human approval is provided after testing is successful.
. The company builds and deploys ML models and applications to staging servers for testing.
. The company commits code changes or new training datasets to a Git repository.

Step 1:

Select…
Select…
CodePipeline deploys ML models and applications to production.
CodePipeline detects code changes and starts to build automatically.
Human approval is provided after testing is successful.
The company builds and deploys ML models and applications to staging servers
The company commits code changes or new training datasets to a Git repository.

Step 2:

Select…
Select…
CodePipeline deploys ML models and applications to production.
CodePipeline detects code changes and starts to build automatically.
Human approval is provided after testing is successful.
The company builds and deploys ML models and applications to staging servers
The company commits code changes or new training datasets to a Git repository.

Step 3:

Select…
Select…
CodePipeline deploys ML models and applications to production.
CodePipeline detects code changes and starts to build automatically.
Human approval is provided after testing is successful.
The company builds and deploys ML models and applications to staging servers
The company commits code changes or new training datasets to a Git repository.

Step 4:

Select…
Select…
CodePipeline deploys ML models and applications to production.
CodePipeline detects code changes and starts to build automatically.
Human approval is provided after testing is successful.
The company builds and deploys ML models and applications to staging servers
The company commits code changes or new training datasets to a Git repository.

Step 5:

Select…
Select…
CodePipeline deploys ML models and applications to production.
CodePipeline detects code changes and starts to build automatically.
Human approval is provided after testing is successful.
The company builds and deploys ML models and applications to staging servers
The company commits code changes or new training datasets to a Git repository.

Answer:

Step 1:

Select…
Select…
CodePipeline deploys ML models and applications to production.
CodePipeline detects code changes and starts to build automatically.
Human approval is provided after testing is successful.
The company builds and deploys ML models and applications to staging servers for testing.
The company commits code changes or new training datasets to a Git repository.

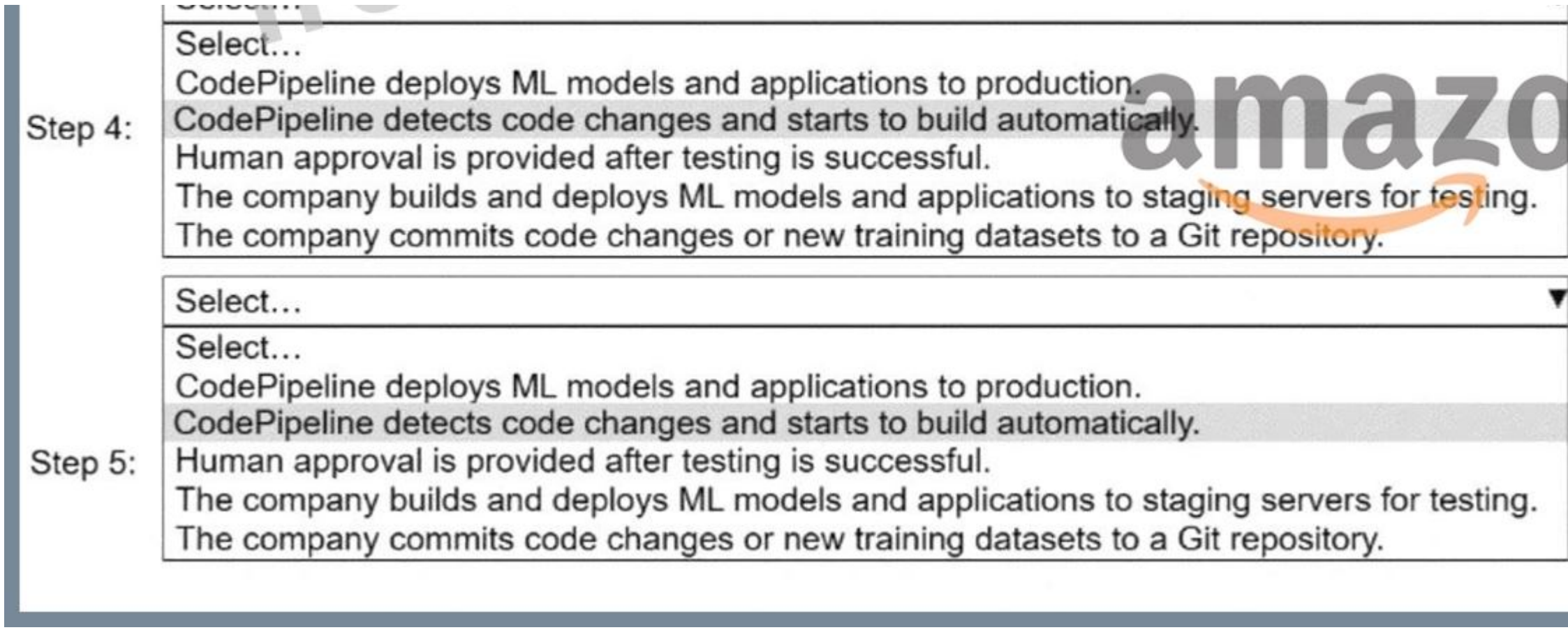
Step 2:

Select…
Select…
CodePipeline deploys ML models and applications to production.
CodePipeline detects code changes and starts to build automatically.
Human approval is provided after testing is successful.
The company builds and deploys ML models and applications to staging servers for testing.
The company commits code changes or new training datasets to a Git repository.

Step 3:

Select…
Select…
CodePipeline deploys ML models and applications to production.
CodePipeline detects code changes and starts to build automatically.
Human approval is provided after testing is successful.
The company builds and deploys ML models and applications to staging servers for testing.
The company commits code changes or new training datasets to a Git repository.

Select…



Explanation:

Step 1:

The company commits code changes or new training datasets to a Git repository.

This is the trigger point. A source code or data change initiates the CI/CD pipeline.

Step 2:

CodePipeline detects code changes and starts to build automatically.

CodePipeline monitors the Git repository (for example, AWS CodeCommit, GitHub, or Bitbucket) and automatically triggers the pipeline when changes are detected.

Step 3:

The company builds and deploys ML models and applications to staging servers for testing.

The pipeline runs build, training, and test stages (often using AWS CodeBuild and SageMaker) and deploys artifacts to a staging or test environment for validation.

Step 4:

Human approval is provided after testing is successful.

A manual approval action is a best practice for ML workflows to ensure governance, compliance, and quality checks before production deployment.

Step 5:

CodePipeline deploys ML models and applications to production.

After approval, the pipeline automatically deploys the validated model or application to the production environment.

NEW QUESTION: 10

A company has implemented a data ingestion pipeline for sales transactions from its ecommerce website. The company uses Amazon Data Firehose to ingest data into Amazon OpenSearch Service. The buffer interval of the Firehose stream is set for 60 seconds. An OpenSearch linear

model generates real-time sales forecasts based on the data and presents the data in an OpenSearch dashboard.
The company needs to optimize the data ingestion pipeline to support sub-second latency for the real-time dashboard.
Which change to the architecture will meet these requirements?
A. Use zero buffering in the Firehose stream. Tune the batch size that is used in the PutRecordBatch operation.
B. Replace the Firehose stream with an AWS DataSync task. Configure the task with enhanced fan-out consumers.
C. Increase the buffer interval of the Firehose stream from 60 seconds to 120 seconds.
D. Replace the Firehose stream with an Amazon Simple Queue Service (Amazon SQS) queue.
Answer: A (LEAVE A REPLY)
The primary requirement in this scenario is achieving sub-second latency for a real-time analytics dashboard powered by Amazon OpenSearch Service. The current architecture uses Amazon Data Firehose, which buffers incoming records based on time or size before delivering them to the destination. A buffer interval of
60 seconds introduces unavoidable latency, making it unsuitable for near-real-time or sub-second use cases.
According to AWS documentation, reducing or eliminating buffering in Firehose is the correct approach when low-latency ingestion is required. Setting the Firehose buffer interval to zero seconds forces Firehose to deliver records as soon as they are received. Additionally, tuning the PutRecordBatch batch size allows efficient ingestion while minimizing delivery delay. This configuration is explicitly recommended for latency-sensitive analytics pipelines.
Option B is incorrect because AWS DataSync is designed for batch-oriented data transfers between storage systems, not real-time streaming. Enhanced fan-out consumers are a feature of Amazon Kinesis Data Streams, not DataSync, making this option invalid.
Option C directly contradicts the requirement. Increasing the buffer interval from 60 seconds to 120 seconds would further increase latency and degrade real-time performance.
Option D is also incorrect because Amazon SQS is a message queueing service, not a streaming ingestion service optimized for indexing data into OpenSearch with minimal latency. Using SQS would add additional processing layers and would not inherently provide sub-second ingestion into OpenSearch.
Therefore, using zero buffering in the Firehose stream and tuning the PutRecordBatch batch size is the only change that aligns with AWS best practices for achieving sub-second latency in real-time analytics pipelines.

NEW QUESTION: 11
A company has an ML model that generates text descriptions based on images that customers upload to the company's website. The images can be up to 50 MB in total size.
An ML engineer decides to store the images in an Amazon S3 bucket. The ML engineer must implement a processing solution that can scale to accommodate changes in demand.
Which solution will meet these requirements with the LEAST operational overhead?
A. Create an Amazon SageMaker batch transform job to process all the images in the S3 bucket.
B. Create an Amazon SageMaker Asynchronous Inference endpoint and a scaling policy. Run a script to make an inference request for each image.
C. Create an Amazon Elastic Kubernetes Service (Amazon EKS) cluster that uses Karpenter for auto scaling. Host the model on the EKS cluster. Run a script to make an inference request for each image.
D. Create an AWS Batch job that uses an Amazon Elastic Container Service (Amazon ECS) cluster.Specify a list of images to process for each AWS Batch job.
Answer: B (LEAVE A REPLY)
SageMaker Asynchronous Inference is designed for processing large payloads, such as images up to 50 MB, and can handle requests that do not

require an immediate response.
It scales automatically based on the demand, minimizing operational overhead while ensuring cost-efficiency.
A script can be used to send inference requests for each image, and the results can be retrieved asynchronously. This approach is ideal for accommodating varying levels of traffic with minimal manual intervention.

NEW QUESTION: 12
A company uses an ML model to recommend videos to users. The model is deployed on Amazon SageMaker AI. The model performed well initially after deployment, but the model's performance has degraded over time.
Which solution can the company use to identify model drift in the future?
A. Create a monitoring job in SageMaker Model Monitor. Then create a baseline from the training dataset.
B. Create a baseline from the training dataset. Then create a monitoring job in SageMaker Model Monitor.
C. Create a baseline by using a built-in rule in SageMaker Clarify. Monitor the drift in Amazon CloudWatch.
D. Retrain the model on new data. Compare the retrained model's performance to the original model's performance.
Answer: B (LEAVE A REPLY)
AWS recommends Amazon SageMaker Model Monitor for detecting data drift and model drift in deployed models. Model Monitor works by comparing live inference data against a baseline, which must first be created from the training dataset.
AWS documentation clearly specifies the required order:
* Create a baseline using training data statistics
* Create a monitoring schedule to compare incoming data against the baseline Option A reverses this order and is therefore incorrect. Option C is incorrect because SageMaker Clarify focuses on bias and explainability, not ongoing drift detection. Option D is reactive and does not provide continuous monitoring.
Model Monitor integrates with Amazon CloudWatch, enabling automated alerts and downstream retraining workflows. This proactive approach allows companies to detect degradation early and maintain model quality.
Therefore, Option B is the correct and AWS-verified answer.

NEW QUESTION: 13
A company has a conversational AI assistant that sends requests through Amazon Bedrock to an Anthropic Claude large language model (LLM). Users report that when they ask similar questions multiple times, they sometimes receive different answers. An ML engineer needs to improve the responses to be more consistent and less random.
Which solution will meet these requirements?
A. Increase the temperature parameter and the top_k parameter.
B. Increase the temperature parameter. Decrease the top_k parameter.
C. Decrease the temperature parameter. Increase the top_k parameter.
D. Decrease the temperature parameter and the top_k parameter.
Answer: D (LEAVE A REPLY)
Thetemperatureparameter controls the randomness in the model's responses. Lowering the temperature makes the model produce more deterministic and consistent answers.
Thetop_kparameter limits the number of tokens considered for generating the next word. Reducing top_k further constrains the model's options, ensuring more predictable responses.
By decreasing both parameters, the responses become more focused and consistent, reducing variability in similar queries.

NEW QUESTION: 14
An ML engineer is using a training job to fine-tune a deep learning model in Amazon SageMaker Studio. The ML engineer previously used the same pre-trained model with a similar dataset. The ML engineer expects vanishing gradient, underutilized GPU, and overfitting problems.
The ML engineer needs to implement a solution to detect these issues and to react in predefined ways when the issues occur. The solution also must provide comprehensive real-time metrics during the training.
Which solution will meet these requirements with the LEAST operational overhead?
A. Use TensorBoard to monitor the training job. Publish the findings to an Amazon Simple Notification Service (Amazon SNS) topic. Create an AWS Lambda function to consume the findings and to initiate the predefined actions.
B. Use Amazon CloudWatch default metrics to gain insights about the training job. Use the metrics to invoke an AWS Lambda function to initiate the predefined actions.
C. Expand the metrics in Amazon CloudWatch to include the gradients in each training step. Use the metrics to invoke an AWS Lambda function to initiate the predefined actions.
D. Use SageMaker Debugger built-in rules to monitor the training job. Configure the rules to initiate the predefined actions.
Answer: (SHOW ANSWER)
SageMaker Debugger provides built-in rules to automatically detect issues like vanishing gradients, underutilized GPU, and overfitting during training jobs. It generates real-time metrics and allows users to define predefined actions that are triggered when specific issues occur. This solution minimizes operational overhead by leveraging the managed monitoring capabilities of SageMaker Debugger without requiring custom setups or extensive manual intervention.

NEW QUESTION: 15
A company has a Retrieval Augmented Generation (RAG) application that uses a vector database to store embeddings of documents. The company must migrate the application to AWS and must implement a solution that provides semantic search of text files. The company has already migrated the text repository to an Amazon S3 bucket.
Which solution will meet these requirements?
A. Use an AWS Batch job to process the files and generate embeddings. Use AWS Glue to store the embeddings. Use SQL queries to perform the semantic searches.
B. Use a custom Amazon SageMaker notebook to run a custom script to generate embeddings. Use SageMaker Feature Store to store the embeddings. Use SQL queries to perform the semantic searches.
C. Use the Amazon Kendra S3 connector to ingest the documents from the S3 bucket into Amazon Kendra. Query Amazon Kendra to perform the semantic searches.
D. Use an Amazon Textract asynchronous job to ingest the documents from the S3 bucket. Query Amazon Textract to perform the semantic searches.
Answer: C (LEAVE A REPLY)
Amazon Kendra is an AI-powered search service designed for semantic search use cases. It allows ingestion of documents from an Amazon S3 bucket using the Amazon Kendra S3 connector. Once the documents are ingested, Kendra enables semantic searches with its built-in capabilities, removing the need to manually generate embeddings or manage a vector database. This approach is efficient, requires minimal operational effort, and meets the requirements for a Retrieval Augmented Generation (RAG) application.

NEW QUESTION: 16
A company regularly receives new training data from the vendor of an ML model. The vendor delivers cleaned and prepared data to the company's Amazon S3 bucket every 3-4 days.

The company has an Amazon SageMaker pipeline to retrain the model. An ML engineer needs to implement a solution to run the pipeline when new data is uploaded to the S3 bucket.
Which solution will meet these requirements with the LEAST operational effort?
A. Create an S3 Lifecycle rule to transfer the data to the SageMaker training instance and to initiate training.
B. Create an AWS Lambda function that scans the S3 bucket. Program the Lambda function to initiate the pipeline when new data is uploaded.
C. Create an Amazon EventBridge rule that has an event pattern that matches the S3 upload. Configure the pipeline as the target of the rule.
D. Use Amazon Managed Workflows for Apache Airflow (Amazon MWAA) to orchestrate the pipeline when new data is uploaded.
Answer: C (LEAVE A REPLY)
UsingAmazon EventBridgewith an event pattern that matches S3 upload events provides an automated, low- effort solution. When new data is uploaded to the S3 bucket, the EventBridge rule triggers the SageMaker pipeline. This approach minimizes operational overhead by eliminating the need for custom scripts or external orchestration tools while seamlessly integrating with the existing S3 and SageMaker setup.

NEW QUESTION: 17
A company is developing an internal cost-estimation tool that uses an ML model in Amazon SageMaker AI.
Users upload high-resolution images to the tool.
The model must process each image and predict the cost of the object in the image. The model also must notify the user when processing is complete.
Which solution will meet these requirements?
A. Store the images in an Amazon S3 bucket. Deploy the model on SageMaker AI. Use batch transform jobs for model inference. Use an Amazon Simple Queue Service (Amazon SQS) queue to notify users.
B. Store the images in an Amazon S3 bucket. Deploy the model on SageMaker AI. Use an asynchronous inference strategy for model inference. Use an Amazon Simple Notification Service (Amazon SNS) topic to notify users.
C. Store the images in an Amazon Elastic File System (Amazon EFS) file system. Deploy the model on SageMaker AI. Use batch transform jobs for model inference. Use an Amazon Simple Queue Service (Amazon SQS) queue to notify users.
D. Store the images in an Amazon Elastic File System (Amazon EFS) file system. Deploy the model on SageMaker AI. Use an asynchronous inference strategy for model inference. Use an Amazon Simple Notification Service (Amazon SNS) topic to notify users.
Answer: (SHOW ANSWER)
This use case involves large payloads (high-resolution images), variable processing time, and a requirement to notify users upon completion. AWS documentation recommends Amazon SageMaker asynchronous inference for exactly this scenario.
Asynchronous inference allows clients to submit inference requests without waiting for a response. The request payload is stored in Amazon S3, and the inference result is written back to S3 when processing is complete. This approach is designed for workloads that cannot meet real-time latency requirements and where payload sizes exceed real-time limits.
SageMaker asynchronous inference integrates natively with Amazon SNS to send notifications when inference completes or fails, which directly satisfies the user notification requirement.
Batch transform jobs are intended for offline, scheduled processing of large datasets and do not provide built- in user notification mechanisms.

Amazon EFS is not required because SageMaker natively integrates with S3 for inference input and output.
AWS explicitly documents asynchronous inference as the preferred solution for large image inference with completion notifications.
Therefore, Option B is the correct and AWS-verified answer.

NEW QUESTION: 18
A company has a large, unstructured dataset. The dataset includes many duplicate records across several key attributes.
Which solution on AWS will detect duplicates in the dataset with the LEAST code development?
A. Use Amazon Mechanical Turk jobs to detect duplicates.
B. Use Amazon QuickSight ML Insights to build a custom deduplication model.
C. Use Amazon SageMaker Data Wrangler to pre-process and detect duplicates.
D. Use the AWS Glue FindMatches transform to detect duplicates.
Answer: D (LEAVE A REPLY)
Scenario:The dataset contains duplicate records that need to be detected with minimal code development.
Why FindMatches in AWS Glue?
* Purpose-Built for Deduplication:The FindMatches transform in AWS Glue is specifically designed to identify duplicate records in structured or semi-structured datasets.
* Machine Learning-Based:It uses ML to identify duplicates based on configurable thresholds and provides flexibility for tuning accuracy.
* Low Code Overhead:Minimal development effort is required as Glue provides an interactive console for configuring and running FindMatches transforms.
Steps to Implement:
* Prepare the Data:Upload the unstructured dataset to an S3 bucket and define a schema if needed.
* Create a Glue Job:
* Use the AWS Glue Studio to create a job and select the FindMatches transform.
* Specify key attributes for deduplication.
* Run and Evaluate:Execute the Glue job, and review the results for duplicates.
* Resolve Duplicates:Export results to an S3 bucket or process them as needed.
References:
* AWS Glue FindMatches Documentation
* FindMatches Transform Example

NEW QUESTION: 19
A financial company receives a high volume of real-time market data streams from an external provider. The streams consist of thousands of JSON records every second.
The company needs to implement a scalable solution on AWS to identify anomalous data points.
Which solution will meet these requirements with the LEAST operational overhead?
A. Ingest real-time data into Amazon Kinesis data streams. Use the built-in RANDOM_CUT_FOREST function in Amazon Managed Service for Apache Flink to process the data streams and to detect data anomalies.
B. Ingest real-time data into Amazon Kinesis data streams. Deploy an Amazon SageMaker endpoint for real-time outlier detection. Create an AWS Lambda function to detect anomalies. Use the data streams to invoke the Lambda function.
C. Ingest real-time data into Apache Kafka on Amazon EC2 instances. Deploy an Amazon SageMaker endpoint for real-time outlier detection. Create an AWS Lambda function to detect anomalies. Use the data streams to invoke the Lambda function.

D. Send real-time data to an Amazon Simple Queue Service (Amazon SQS) FIFO queue. Create an AWS Lambda function to consume the queue messages. Program the Lambda function to start an AWS Glue extract, transform, and load (ETL) job for batch processing and anomaly detection.
Answer: A (LEAVE A REPLY)
This solution is the most efficient and involves the least operational overhead:
Amazon Kinesis data streams efficiently handle real-time ingestion of high-volume streaming data.
Amazon Managed Service for Apache Flink provides a fully managed environment for stream processing with built-in support for RANDOM_CUT_FOREST, an algorithm designed for anomaly detection in real- time streaming data.
This approach eliminates the need for deploying and managing additional infrastructure like SageMaker endpoints, Lambda functions, or external tools, making it the most scalable and operationally simple solution.

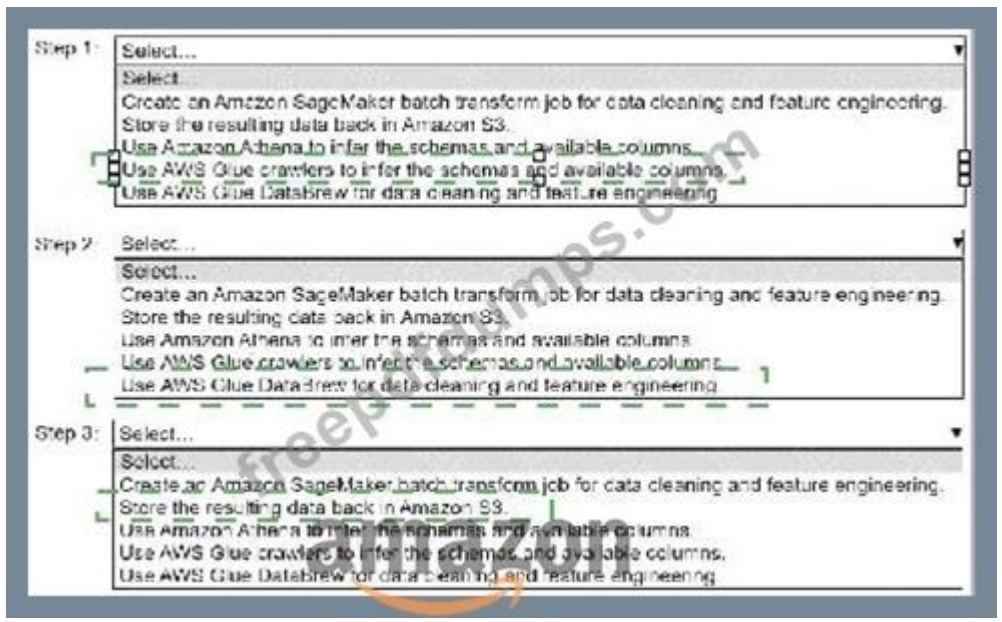
NEW QUESTION: 20
A company is building a deep learning model on Amazon SageMaker. The company uses a large amount of data as the training dataset. The company needs to optimize the model's hyperparameters to minimize the loss function on the validation dataset.
Which hyperparameter tuning strategy will accomplish this goal with the LEAST computation time?
A. Hyperbaric!
B. Grid search
C. Bayesian optimization
D. Random search
Answer: A (LEAVE A REPLY)
Hyperband is a hyperparameter tuning strategy designed to minimize computation time by adaptively allocating resources to promising configurations and terminating underperforming ones early. It efficiently balances exploration and exploitation, making it ideal for large datasets and deep learning models where training can be computationally expensive.

NEW QUESTION: 21
A company runs an ML model on Amazon SageMaker AI. The company uses an automatic process that makes API calls to create training jobs for the model. The company has new compliance rules that prohibit the collection of aggregated metadata from training jobs.
Which solution will prevent SageMaker AI from collecting metadata from the training jobs?
A. Opt out of metadata tracking for any training job that is submitted.
B. Ensure that training jobs are running in a private subnet in a custom VPC.
C. Encrypt the training data with an AWS Key Management Service (AWS KMS) customer managed key.
D. Reconfigure the training jobs to use only AWS Nitro instances.
Answer: A (LEAVE A REPLY)
Amazon SageMaker AI automatically collects aggregated metadata from training jobs to improve service reliability, performance, and operational insights. This metadata can include information such as algorithm usage, instance types, resource utilization, and job configuration details.
However, AWS documentation clearly states that customers can opt out of SageMaker metadata collection to meet regulatory or compliance requirements.
SageMaker provides a supported mechanism to disable metadata tracking at the training job level. By explicitly opting out of metadata tracking when submitting training jobs-either through the AWS Management Console, AWS CLI, or SDK-the service will stop collecting aggregated metadata for those jobs. This option is specifically designed for customers with strict compliance, data residency, or regulatory constraints.
Option B is incorrect because running training jobs in a private subnet within a custom VPC controls network isolation, not service-level telemetry or metadata collection. Metadata collection occurs at the SageMaker service layer and is independent of VPC configuration.

Option C is also incorrect because encrypting training data with a customer-managed AWS KMS key protects data at rest and in transit but does not prevent SageMaker from collecting operational metadata about training jobs.
Option D is incorrect because AWS Nitro instances provide enhanced security and performance isolation at the infrastructure level but have no impact on SageMaker's metadata collection mechanisms.
Therefore, opting out of metadata tracking for training jobs is the only solution that directly addresses the compliance requirement and is explicitly supported by AWS documentation.

NEW QUESTION: 22
A company stores historical data in .csv files in Amazon S3. Only some of the rows and columns in the .csv files are populated. The columns are not labeled. An ML engineer needs to prepare and store the data so that the company can use the data to train ML models.
Select and order the correct steps from the following list to perform this task. Each step should be selected one time or not at all. (Select and order three.)
* Create an Amazon SageMaker batch transform job for data cleaning and feature engineering.
* Store the resulting data back in Amazon S3.
* Use Amazon Athena to infer the schemas and available columns.
* Use AWS Glue crawlers to infer the schemas and available columns.
* Use AWS Glue DataBrew for data cleaning and feature engineering.

Answer:



Explanation:

Step 1: Use AWS Glue crawlers to infer the schemas and available columns.

Step 2: Use AWS Glue DataBrew for data cleaning and feature engineering.

Step 3: Store the resulting data back in Amazon S3.

Step 1: Use AWS Glue Crawlers to Infer Schemas and Available Columns

Why? The data is stored in .csv files with unlabeled columns, and Glue Crawlers can scan the raw data in Amazon S3 to automatically infer the schema, including available columns, data types, and any missing or incomplete entries.

How? Configure AWS Glue Crawlers to point to the S3 bucket containing the .csv files, and run the crawler to extract metadata. The crawler creates a schema in the AWS Glue Data Catalog, which can then be used for subsequent transformations.

Step 2: Use AWS Glue DataBrew for Data Cleaning and Feature Engineering Why? Glue DataBrew is a visual data preparation tool that allows for comprehensive cleaning and transformation of data. It supports imputation of missing values, renaming columns, feature engineering, and more without requiring extensive coding.

How? Use Glue DataBrew to connect to the inferred schema from Step 1 and perform data cleaning and feature engineering tasks like filling in missing rows/columns, renaming unlabeled columns, and creating derived features.

Step 3: Store the Resulting Data Back in Amazon S3

Why? After cleaning and preparing the data, it needs to be saved back to Amazon S3 so that it can be used for training machine learning models.

How? Configure Glue DataBrew to export the cleaned data to a specific S3 bucket location. This ensures the processed data is readily accessible for ML workflows.

Order Summary:

Use AWS Glue crawlers to infer schemas and available columns.

Use AWS Glue DataBrew for data cleaning and feature engineering.

Store the resulting data back in Amazon S3.

This workflow ensures that the data is prepared efficiently for ML model training while leveraging AWS services for automation and scalability.

NEW QUESTION: 23

An ML engineer is using Amazon Quick Suite (previously known as Amazon QuickSight) anomaly detection to detect very high or very low machine operating temperatures compared to normal. The ML engineer sets the Severity parameter to Low and above. The ML engineer sets the Direction parameter to All.

What effect will the ML engineer observe in the anomaly detection results if the ML engineer changes the Direction parameter to Lower than expected?

A. Increased anomaly identification frequency and increased recall

B. Decreased anomaly identification frequency and decreased recall

C. Increased anomaly identification frequency and decreased recall

D. Decreased anomaly identification frequency and increased recall

Answer: B (LEAVE A REPLY)

In Amazon QuickSight anomaly detection, the Direction parameter controls which deviations from the expected baseline are flagged as anomalies. When Direction is set to All, QuickSight detects both higher-than- expected and lower-than-expected values. This results in a larger set of detected anomalies, increasing recall.

When the Direction parameter is changed to Lower than expected, the anomaly detection algorithm filters out all higher-than-expected anomalies and only reports unusually low values. Because the scope of detection is narrowed, the total number of detected anomalies decreases. As a result, fewer true anomalies are identified overall, which reduces recall.

Since fewer data points are evaluated as potential anomalies, the frequency of anomaly identification also decreases. This behavior is consistent with AWS documentation, which explains that directional filtering limits detection to a subset of deviations and therefore reduces overall anomaly counts.

Therefore, changing the Direction from All to Lower than expected leads to decreased anomaly identification frequency and decreased recall.

NEW QUESTION: 24

An ML engineer is setting up a CI/CD pipeline for an ML workflow in Amazon SageMaker AI. The pipeline must automatically retrain, test, and deploy a model whenever new data is uploaded to an Amazon S3 bucket.

New data files are approximately 10 GB in size. The ML engineer also needs to track model versions for auditing.

Which solution will meet these requirements?

A. Use AWS CodePipeline, Amazon S3, and AWS CodeBuild to retrain and deploy the model automatically and track model versions.

B. Use SageMaker Pipelines with the SageMaker Model Registry to orchestrate model training and version tracking.

C. Use AWS Lambda and Amazon EventBridge to retrain and deploy the model and track versions via logs.

D. Manually retrain and deploy the model using SageMaker notebook instances and track versions with AWS CloudTrail.

Answer: B (LEAVE A REPLY)

AWS documentation identifies SageMaker Pipelines as the native CI/CD service for ML workflows. Pipelines allow engineers to define automated steps for data processing, training, evaluation, and deployment, making them ideal for retraining models when new data arrives in Amazon S3.

For version tracking and auditing, SageMaker Model Registry is explicitly designed to manage model versions, metadata, approval status, and deployment history. This satisfies regulatory and audit requirements without custom tooling.

AWS Lambda is not suitable for handling large datasets (10 GB), and CodeBuild is not ML-aware and lacks built-in model governance. Manual notebook workflows do not meet CI/CD or automation requirements.

AWS best practices strongly recommend SageMaker Pipelines combined with the Model Registry for scalable, auditable, and production-grade ML CI/CD pipelines.

Therefore, Option B is the correct and AWS-verified solution.

NEW QUESTION: 25

A company uses a hybrid cloud environment. A model that is deployed on premises uses data in Amazon S3 to provide customers with a live conversational engine.

The model is using sensitive data. An ML engineer needs to implement a solution to identify and remove the sensitive data.
Which solution will meet these requirements with the LEAST operational overhead?

A. Deploy the model on Amazon SageMaker AI. Create a set of AWS Lambda functions to identify and remove the sensitive data.

B. Deploy the model on an Amazon Elastic Container Service (Amazon ECS) cluster that uses AWS Fargate. Create an AWS Batch job to identify and remove the sensitive data.

C. Use Amazon Macie to identify the sensitive data. Create a set of AWS Lambda functions to remove the sensitive data.

D. Use Amazon Comprehend to identify the sensitive data. Launch Amazon EC2 instances to remove the sensitive data.

Answer: C (LEAVE A REPLY)

The core requirement is to identify and remove sensitive data from content stored in Amazon S3 with minimal operational overhead. Amazon Macie is purpose-built to automatically discover, classify, and protect sensitive data (such as PII) in Amazon S3 using machine learning. Macie continuously scans S3 objects and produces findings that identify sensitive data types and locations without requiring custom ML pipelines or infrastructure.

Once Macie identifies sensitive data, AWS Lambda can be used to automate remediation-such as redacting, masking, or deleting sensitive fields-based on Macie findings. This event-driven approach is serverless, scales automatically, and minimizes operations.

Option A increases overhead by moving the model and building custom detection logic. Option B introduces unnecessary compute orchestration (ECS/Fargate and Batch). Option D uses Amazon Comprehend for entity detection, which is not optimized for broad S3 data discovery and requires EC2 management.

Therefore, using Amazon Macie for detection and Lambda for remediation is the most efficient and AWS- recommended solution.

NEW QUESTION: 26

A company needs to create a central catalog for all the company's ML models. The models are in AWS accounts where the company developed the models initially. The models are hosted in Amazon Elastic Container Registry (Amazon ECR) repositories.
Which solution will meet these requirements?

A. Configure ECR cross-account replication for each existing ECR repository. Ensure that each model is visible in each AWS account.

B. Create a new AWS account with a new ECR repository as the central catalog. Configure ECR cross- account replication between the initial ECR repositories and the central catalog.

C. Use the Amazon SageMaker Model Registry to create a model group for models hosted in Amazon ECR. Create a new AWS account. In the new account, use the SageMaker Model Registry as the central catalog. Attach a cross-account resource policy to each model group in the initial AWS accounts.

D. Use an AWS Glue Data Catalog to store the models. Run an AWS Glue crawler to migrate the models from the ECR repositories to the Data Catalog. Configure cross-account access to the Data Catalog.

Answer: C (LEAVE A REPLY)

The Amazon SageMaker Model Registry is designed to manage and catalog ML models, including those hosted in Amazon ECR. By creating a model group for each model in the SageMaker Model Registry and setting up cross-account resource policies, the company can establish a central catalog in a new AWS account.

This allows all models from the initial accounts to be accessible in a unified, centralized manner for better organization, management, and governance. This solution leverages existing AWS services and ensures scalability and minimal operational overhead.

NEW QUESTION: 27

A hospital is using an ML model to validate x-ray results. The hospital runs a nightly batch inference job. The hospital needs to produce a daily report about model data quality and model performance.

Which solution will meet these requirements?

A. Schedule a monitoring job in Amazon SageMaker Model Monitor. Generate the monitoring results for the model and data.

B. Create an Amazon CloudWatch dashboard that includes the metrics for processing steps in the nightly batch inference job. Compare the baseline resource metrics. Share the dashboard link.

C. Use AWS Glue DataBrew to create a custom recipe job that uses the Numerical Statistics data quality check for the model file. Generate the results.

D. Create a SageMaker AI pipeline that includes a QualityCheck step to run monitoring jobs. Generate the monitoring results for the model and the data.

Answer: (SHOW ANSWER)

Option A is correct because Amazon SageMaker Model Monitor is the AWS service specifically built to monitor data quality and model quality for ML models in production. AWS documentation states that Model Monitor supports continuous monitoring with a batch transform job that runs regularly and also supports on-schedule monitoring for asynchronous batch transform jobs . That aligns directly with the scenario of a hospital running a nightly batch inference job and needing a daily report on both the incoming data and the model's predictive performance.

AWS documentation also separates the two monitoring needs very clearly. Data quality monitoring can be scheduled for batch transform jobs by using DefaultModelMonitor with a BatchTransformInput. Model quality monitoring can also be scheduled for batch transform jobs by using ModelQualityMonitor, which compares predictions against actual ground-truth labels stored in Amazon S3. Since the question explicitly asks for both model data quality and model performance , SageMaker Model Monitor is the documented feature that covers both requirements together.

Option B is not sufficient because CloudWatch dashboards show operational and resource metrics, not the full ML-specific data quality and model quality reports required here. Option C is incorrect because AWS Glue DataBrew is for data preparation and profiling, not model performance monitoring. Option D is partially plausible because SageMaker Pipelines integrates with QualityCheck steps and can run monitoring jobs on demand, but the AWS docs position Model Monitor as the native solution for scheduled monitoring of production batch inference workloads.

Therefore, the best AWS-documented answer is A .

NEW QUESTION: 28

A company runs an Amazon SageMaker domain in a public subnet of a newly created VPC. The network is configured properly, and ML engineers can access the SageMaker domain.

Recently, the company discovered suspicious traffic to the domain from a specific IP address. The company needs to block traffic from the specific IP address.

Which update to the network configuration will meet this requirement?

A. Create a security group inbound rule to deny traffic from the specific IP address. Assign the security group to the domain.

B. Create a network ACL inbound rule to deny traffic from the specific IP address. Assign the rule to the default network Ad for the subnet where the domain is located.

C. Create a shadow variant for the domain. Configure SageMaker Inference Recommender to send traffic from the specific IP address to the shadow endpoint.

D. Create a VPC route table to deny inbound traffic from the specific IP address. Assign the route table to the domain.

Answer: (SHOW ANSWER)

Network ACLs (Access Control Lists) operate at the subnet level and allow for rules to explicitly deny traffic from specific IP addresses. By creating an inbound rule in the network ACL to deny traffic from the suspicious IP address, the company can block traffic to the Amazon SageMaker domain from that IP. This approach works because network ACLs are evaluated before traffic reaches the security groups, making them effective for blocking traffic at the subnet level.

NEW QUESTION: 29
An ML model is deployed in production. The model has performed well and has met its metric thresholds for months.
An ML engineer who is monitoring the model observes a sudden degradation. The performance metrics of the model are now below the thresholds.
What could be the cause of the performance degradation?
A. Lack of training data
B. Drift in production data distribution
C. Compute resource constraints
D. Model overfitting
Answer: B (LEAVE A REPLY)
A sudden drop in model performance after a long period of stability is a classic indicator of data drift.
According to AWS ML documentation, production data distribution drift occurs when the statistical properties of incoming data change over time compared to the training dataset.
This drift can be caused by changes in user behavior, market conditions, seasonal trends, or upstream data pipelines. When drift occurs, the model's learned patterns no longer align with real-world data, leading to degraded accuracy and other performance metrics.
Lack of training data (Option A) and overfitting (Option D) are issues that typically manifest during initial training, not after months of stable production performance. Compute constraints (Option C) may affect latency but do not usually cause sustained accuracy degradation.
AWS recommends using SageMaker Model Monitor to detect such drift and trigger retraining workflows.
Therefore, drift in the production data distribution is the most likely cause of the observed performance degradation.

NEW QUESTION: 30
A company is using an Amazon S3 bucket to collect data that will be used for ML workflows. The company needs to use AWS Glue DataBrew to clean and normalize the data.
Which solution will meet these requirements?
A. Create a DataBrew dataset by using the S3 path. Clean and normalize the data by using a DataBrew profile job.
B. Create a DataBrew dataset by using the S3 path. Clean and normalize the data by using a DataBrew recipe job.
C. Create a DataBrew dataset by using a JDBC driver to connect to the S3 bucket. Use a profile job.
D. Create a DataBrew dataset by using a JDBC driver to connect to the S3 bucket. Use a recipe job.
Answer: B (LEAVE A REPLY)
AWS Glue DataBrew supports datasets sourced directly from Amazon S3 paths. To clean and normalize data, AWS documentation specifies using DataBrew recipes, which define transformation steps such as standardization, deduplication, and formatting.
A profile job is used only for data analysis and statistics generation, not for transformation. A recipe job applies actual transformations to the data and writes the output to Amazon S3.
JDBC connections are used for relational databases, not S3 buckets. Therefore, options C and D are invalid.
AWS clearly documents that the correct workflow is:
Create a DataBrew dataset from an S3 path
Define transformations using a recipe
Run a recipe job to produce cleaned data
Thus, Option B is the correct and AWS-aligned answer.

NEW QUESTION: 31
An ML engineer normalized training data by using min-max normalization in AWS Glue DataBrew. The ML engineer must normalize production inference data in the same way before passing the data to the model.

Which solution will meet this requirement?

A. Apply statistics from a well-known dataset to normalize the production samples.

B. Keep the min-max normalization statistics from the training set and use them to normalize the production samples.

C. Calculate new min-max statistics from a batch of production samples and use them to normalize all production samples.

D. Calculate new min-max statistics from each production sample and use them to normalize all production samples.

Answer: (SHOW ANSWER)

AWS ML best practices state that data preprocessing applied during training must be applied identically during inference. For min-max normalization, this requires reusing the minimum and maximum values calculated from the training dataset.

If production data is normalized using different statistics, the feature distributions will differ from what the model learned, leading to degraded prediction accuracy. AWS documentation explicitly warns against recomputing normalization parameters on inference data.

Options A, C, and D introduce data leakage or inconsistent feature scaling. Option B ensures consistency between training and inference pipelines and preserves model integrity.

Therefore, Option B is the correct and AWS-aligned solution.

NEW QUESTION: 32

A company plans to use Amazon SageMaker AI to build image classification models. The company has 6 TB of training data stored on Amazon FSx for NetApp ONTAP. The file system is in the same VPC as SageMaker AI.

An ML engineer must make the training data accessible to SageMaker AI training jobs.

Which solution will meet these requirements?

A. Mount the FSx for ONTAP file system as a volume to the SageMaker AI instance.

B. Create an Amazon S3 bucket and use Mountpoint for Amazon S3 to link the bucket to FSx for ONTAP.

C. Create a catalog connection from SageMaker Data Wrangler to the FSx for ONTAP file system.

D. Create a direct connection from SageMaker Data Wrangler to the FSx for ONTAP file system.

Answer: (SHOW ANSWER)

Amazon SageMaker supports direct file system access for training jobs through Amazon FSx. AWS documentation states that FSx for NetApp ONTAP file systems can be mounted directly to SageMaker training instances when they are located in the same VPC.

Mounting the FSx file system allows SageMaker to stream large datasets efficiently without copying data into Amazon S3. This is ideal for very large datasets such as 6 TB of image data and avoids unnecessary storage duplication.

Options involving SageMaker Data Wrangler are intended for data preparation and exploration, not large- scale training data access. Mountpoint for Amazon S3 is not required and introduces additional complexity.

Therefore, Option A is the correct and AWS-aligned solution.

NEW QUESTION: 33

An ML engineer wants to deploy a workflow that processes streaming IoT sensor data and periodically retrains ML models. The most recent model versions must be deployed to production.

Which service will meet these requirements?

A. Amazon SageMaker Pipelines

B. Amazon Managed Workflows for Apache Airflow (MWAA)

C. AWS Lambda

D. Apache Spark

Answer: A (LEAVE A REPLY)

Amazon SageMaker Pipelines is purpose-built for orchestrating end-to-end ML workflows, including data ingestion, training, evaluation, and deployment. It supports automation, versioning, and deployment of the latest model versions.

MWAA orchestrates general workflows but lacks ML-native features. Lambda cannot handle long-running ML training. Spark processes data but does not manage ML lifecycle.

Therefore, Option A is the correct AWS-native solution.

NEW QUESTION: 34

A healthcare company wants to detect irregularities in patient vital signs that could indicate early signs of a medical condition. The company has an unlabeled dataset that includes patient health records, medication history, and lifestyle changes.

Which algorithm and hyperparameter should the company use to meet this requirement?

A. Use the Amazon SageMaker AI XGBoost algorithm. Set max_depth to greater than 100 to regulate tree complexity.

B. Use the Amazon SageMaker AI k-means clustering algorithm. Set k to determine the number of clusters.

C. Use the Amazon SageMaker AI DeepAR algorithm. Set epochs to the number of training iterations.

D. Use the Amazon SageMaker AI Random Cut Forest (RCF) algorithm. Set num_trees to greater than 100.

Answer: D (LEAVE A REPLY)

Option D is correct because Amazon SageMaker AI Random Cut Forest (RCF) is documented by AWS as an unsupervised algorithm for detecting anomalous data points within a data set . The question describes a healthcare use case where the company wants to detect irregularities in patient vital signs and only has an unlabeled dataset . That is a direct match for anomaly detection with an unsupervised method rather than a supervised classification or regression algorithm. AWS documentation also notes that anomalies can appear as unexpected spikes, breaks in patterns, or otherwise unusual observations.

RCF is a strong fit because it does not require labeled target outcomes to begin identifying unusual observations. AWS explains that RCF assigns an anomaly score to each data point, where higher scores indicate potentially anomalous records. This is especially suitable for early warning scenarios in healthcare, where the goal is to surface unusual combinations of vitals, medication history, and lifestyle changes for further review.

The hyperparameter in the answer, num_trees , is a valid RCF hyperparameter and controls the number of trees used in the forest. In AWS examples and documentation, this is one of the key parameters used when configuring the algorithm. By contrast, XGBoost is generally used for supervised tasks, k-means is for clustering rather than anomaly scoring, and DeepAR is for forecasting time series rather than general anomaly detection across an unlabeled multidimensional dataset.

Therefore, the fully verified AWS-docs answer is D , because Random Cut Forest is the AWS-native unsupervised anomaly detection algorithm best aligned to this healthcare irregularity-detection requirement.

NEW QUESTION: 35

A credit card company has a fraud detection model in production on an Amazon SageMaker endpoint. The company develops a new version of the model. The company needs to assess the new model ' s performance by using live data and without affecting production end users.

Which solution will meet these requirements?

A. Set up SageMaker Debugger and create a custom rule.
B. Set up blue/green deployments with all-at-once traffic shifting.
C. Set up blue/green deployments with canary traffic shifting.
D. Set up shadow testing with a shadow variant of the new model.
Answer: (SHOW ANSWER)
Shadow testing allows you to send a copy of live production traffic to a shadow variant of the new model while keeping the existing production model unaffected. This enables you to evaluate the performance of the new model in real-time with live data without impacting end users. SageMaker endpoints support this setup by allowing traffic mirroring to the shadow variant, making it an ideal solution for assessing the new model ' s performance.

NEW QUESTION: 36
A company uses Amazon SageMaker AI to create ML models. The data scientists need fine-grained control of ML workflows, DAG visualization, experiment history, and model governance for auditing and compliance.
Which solution will meet these requirements?
A. Use AWS CodePipeline with SageMaker Studio and SageMaker ML Lineage Tracking.
B. Use AWS CodePipeline with SageMaker Experiments.
C. Use SageMaker Pipelines with SageMaker Studio and SageMaker ML Lineage Tracking.
D. Use SageMaker Pipelines with SageMaker Experiments.
Answer: (SHOW ANSWER)
Amazon SageMaker Pipelines provides native orchestration of ML workflows with fine-grained control, DAG-based visualization, and seamless integration with SageMaker Studio. AWS documentation explicitly states that Pipelines is designed for end-to-end ML workflow automation and visualization.
SageMaker ML Lineage Tracking records relationships between datasets, models, training jobs, and endpoints, enabling full auditability and governance, which is essential for compliance.
SageMaker Experiments tracks experiment metrics but does not provide lineage-level governance.
CodePipeline is a general CI/CD service and lacks ML-specific DAG visualization and lineage tracking.
AWS best practices recommend combining SageMaker Pipelines + SageMaker Studio + ML Lineage Tracking for enterprise-grade ML workflow management.
Therefore, Option C is the correct and AWS-verified solution.

NEW QUESTION: 37
An ML engineer is training an ML model to identify medical patients for disease screening. The tabular dataset for training contains 50,000 patient records: 1,000 with the disease and 49,000 without the disease.
The ML engineer splits the dataset into a training dataset, a validation dataset, and a test dataset.
What should the ML engineer do to transform the data and make the data suitable for training?
A. Apply principal component analysis (PCA) to oversample the minority class in the training dataset.
B. Apply Synthetic Minority Oversampling Technique (SMOTE) to generate new synthetic samples of the minority class in the training dataset.
C. Randomly oversample the majority class in the validation dataset.
D. Apply k-means clustering to undersample the minority class in the test dataset.
Answer: B (LEAVE A REPLY)
This dataset shows severe class imbalance, with only 2% of records representing patients with the disease.

AWS ML best practices recommend correcting imbalance only in the training dataset, while keeping validation and test sets representative of real-world distributions.
Synthetic Minority Oversampling Technique (SMOTE) generates synthetic samples of the minority class by interpolating between existing minority examples. This improves the model's ability to learn disease-related patterns without discarding data.
PCA is a dimensionality reduction method, not an oversampling technique. Oversampling the majority class worsens imbalance. Altering the test dataset would invalidate evaluation results.
Therefore, applying SMOTE to the training dataset is the correct approach.

NEW QUESTION: 38
Case Study
A company is building a web-based AI application by using Amazon SageMaker. The application will provide the following capabilities and features: ML experimentation, training, a central model registry, model deployment, and model monitoring.
The application must ensure secure and isolated use of training data during the ML lifecycle. The training data is stored in Amazon S3.
The company needs to use the central model registry to manage different versions of models in the application.
Which action will meet this requirement with the LEAST operational overhead?
A. Create a separate Amazon Elastic Container Registry (Amazon ECR) repository for each model.
B. Use Amazon Elastic Container Registry (Amazon ECR) and unique tags for each model version.
C. Use the SageMaker Model Registry and model groups to catalog the models.
D. Use the SageMaker Model Registry and unique tags for each model version.
Answer: C (LEAVE A REPLY)
Amazon SageMaker Model Registry is a feature designed to manage machine learning (ML) models throughout their lifecycle. It allows users to catalog, version, and deploy models systematically, ensuring efficient model governance and management.
Key Features of SageMaker Model Registry:
Centralized Cataloging: Organizes models into Model Groups, each containing multiple versions.
Version Control: Maintains a history of model iterations, making it easier to track changes.
Metadata Association: Attach metadata such as training metrics and performance evaluations to models.
Approval Status Management: Allows setting statuses like PendingManualApproval or Approved to ensure only vetted models are deployed.
Seamless Deployment: Direct integration with SageMaker deployment capabilities for real-time inference or batch processing.
Implementation Steps:
Create a Model Group: Organize related models into groups to simplify management and versioning.
Register Model Versions: Each model iteration is registered as a version within a specific Model Group.
Set Approval Status: Assign approval statuses to models before deploying them to ensure quality control.
Deploy the Model: Use SageMaker endpoints for deployment once the model is approved.
Benefits:
Centralized Management: Provides a unified platform to manage models efficiently.
Streamlined Deployment: Facilitates smooth transitions from development to production.
Governance and Compliance: Supports metadata association and approval processes.
By leveraging the SageMaker Model Registry, the company can ensure organized management of models, version control, and efficient deployment workflows with minimal operational overhead.
AWS Documentation: SageMaker Model Registry
AWS Blog: Model Registry Features and Usage

NEW QUESTION: 39
An ML engineer is using Amazon SageMaker AI to train an ML model. The ML engineer needs to use SageMaker AI automatic model tuning (AMT) features to tune the model hyperparameters over a large parameter space.
The model has 20 categorical hyperparameters and 7 continuous hyperparameters that can be tuned. The ML engineer needs to run the tuning job a maximum of 1,000 times. The ML engineer must ensure that each parameter trial is built based on the performance of the previous trial.
Which solution will meet these requirements?
A. Define the search space as categorical parameters of 1,000 possible combinations. Use grid search.
B. Define the search space as continuous parameters. Use random search. Set the maximum number of tuning jobs to 1,000.
C. Define the search space as categorical parameters and continuous parameters. Use Bayesian optimization. Set the maximum number of training jobs to 1,000.
D. Define the search space as categorical parameters and continuous parameters. Use grid search. Set the maximum number of tuning jobs to 1,000.
Answer: C (LEAVE A REPLY)
The requirement that each parameter trial is built based on the performance of the previous trial is the defining characteristic of Bayesian optimization. In Amazon SageMaker Automatic Model Tuning, Bayesian optimization uses prior trial results to intelligently select the next set of hyperparameters, making it far more efficient than grid or random search-especially in large, mixed search spaces.
This scenario includes both categorical (20) and continuous (7) hyperparameters and allows up to 1,000 training jobs, which is well within the supported limits of SageMaker AMT. Bayesian optimization natively supports mixed parameter types and is explicitly recommended by AWS for large, high-dimensional search spaces where exhaustive grid search is impractical.
Option A and D (grid search) do not meet the requirement because grid search evaluates combinations independently and does not learn from previous trials. Additionally, grid search becomes computationally infeasible as dimensionality increases.
Option B (random search) also evaluates trials independently and does not leverage previous results, violating the core requirement.
Therefore, defining both categorical and continuous parameters and using Bayesian optimization with a maximum of 1,000 jobs is the correct and AWS-recommended solution.

NEW QUESTION: 40
An ML engineer needs to use Amazon SageMaker to fine-tune a large language model (LLM) for text summarization. The ML engineer must follow a low-code no-code (LCNC) approach.
Which solution will meet these requirements?
A. Use SageMaker Studio to fine-tune an LLM that is deployed on Amazon EC2 instances.
B. Use SageMaker Autopilot to fine-tune an LLM that is deployed by a custom API endpoint.
C. Use SageMaker Autopilot to fine-tune an LLM that is deployed on Amazon EC2 instances.
D. Use SageMaker Autopilot to fine-tune an LLM that is deployed by SageMaker JumpStart.
Answer: D (LEAVE A REPLY)
SageMaker JumpStart provides access to pre-trained models, including large language models (LLMs), which can be easily deployed and fine-tuned with a low-code/no-code (LCNC) approach. Using SageMaker Autopilot with JumpStart simplifies the fine-tuning process by automating model optimization and reducing the need for extensive coding, making it the ideal solution for this requirement.

NEW QUESTION: 41
An ML engineer needs to deploy ML models to get inferences from large datasets in an asynchronous manner. The ML engineer also needs to implement scheduled monitoring of data quality for the models and must receive alerts when changes in data quality occur.

Which solution will meet these requirements?
A. Deploy the models by using scheduled AWS Glue jobs. Use Amazon CloudWatch alarms to monitor the data quality and send alerts.
B. Deploy the models by using scheduled AWS Batch jobs. Use AWS CloudTrail to monitor the data quality and send alerts.
C. Deploy the models by using Amazon ECS on AWS Fargate. Use Amazon EventBridge to monitor the data quality and send alerts.
D. Deploy the models by using Amazon SageMaker AI batch transform. Use SageMaker Model Monitor to monitor the data quality and send alerts.
Answer: (SHOW ANSWER)
This requirement combines asynchronous inference on large datasets with automated data quality monitoring and alerting. AWS documentation explicitly recommends Amazon SageMaker batch transform for large- scale, asynchronous inference workloads. Batch transform jobs process large datasets stored in Amazon S3 without requiring a persistent endpoint, making them cost-effective and scalable.
For data quality monitoring, Amazon SageMaker Model Monitor is the AWS-native solution. Model Monitor can be scheduled to analyze inference data, compare it against a baseline, and detect data quality issues such as missing values, schema changes, or statistical drift. When violations occur, Model Monitor emits metrics to Amazon CloudWatch, where alarms can trigger alerts.
Options A, B, and C lack ML-aware data quality monitoring capabilities. AWS Glue and Batch are not designed for model data quality analysis, and CloudTrail tracks API activity-not data quality.
AWS best practices clearly position Batch Transform + Model Monitor as the correct architecture for asynchronous inference with automated monitoring and alerting.
Therefore, Option D is the correct and AWS-verified solution.

NEW QUESTION: 42
A company stores training data as a .csv file in an Amazon S3 bucket. The company must encrypt the data and must control which applications have access to the encryption key.
Which solution will meet these requirements?
A. Create a new SSH access key and use the AWS Encryption CLI to encrypt the file.
B. Create a new API key by using Amazon API Gateway and use it to encrypt the file.
C. Create a new IAM role with permissions for kms:GenerateDataKey and use the role to encrypt the file.
D. Create a new AWS Key Management Service (AWS KMS) key and use the AWS Encryption CLI with the KMS key to encrypt the file.
Answer: (SHOW ANSWER)
AWS Key Management Service (AWS KMS) is the recommended service for encryption and key access control. By creating a customer-managed KMS key, the company can define granular IAM policies that control which applications and roles can use the key.
The AWS Encryption CLI integrates directly with KMS and enables client-side encryption of files before storing them in Amazon S3. This approach ensures data is encrypted at rest and that only authorized principals can decrypt it.
SSH keys and API keys are not designed for data encryption. IAM roles alone do not create or manage encryption keys-they only grant permissions.
AWS documentation explicitly states that KMS customer-managed keys provide centralized key management, auditing, and access control.
Therefore, Option D is the correct and AWS-aligned solution.

NEW QUESTION: 43
A company has trained an ML model that is packaged in a container. The company will integrate the model with an existing Python web application. The company needs to host the model on AWS by using Kubernetes.
The company does not want to manage the control plane and must provision the resources in a repeatable manner. The infrastructure must be provisioned by using Python.

Which solution will meet these requirements?

A. Use AWS CloudFormation to provision Amazon EC2 instances in multiple Availability Zones. Set up a Kubernetes cluster. Host the model container on the Kubernetes cluster.

B. Use the AWS CLI to provision an Amazon Elastic Kubernetes Service (Amazon EKS) cluster. Store the image in an Amazon Elastic Container Registry (Amazon ECR) repository. Host the model container on the EKS cluster.

C. Use the AWS Cloud Development Kit (AWS CDK) to provision an Amazon Elastic Kubernetes Service (Amazon EKS) cluster. Store the image in an Amazon Elastic Container Registry (Amazon ECR) repository. Host the model container on the EKS cluster.

D. Use AWS CloudFormation to provision an Amazon Elastic Kubernetes Service (Amazon EKS) cluster.Store the image in an Amazon Elastic Container Registry (Amazon ECR) repository. Host the model container on the EKS cluster.

Answer: C (LEAVE A REPLY)

Option C is correct because the company needs Kubernetes hosting , does not want to manage the control plane , wants repeatable infrastructure provisioning , and requires that provisioning be done by using Python . Amazon EKS is AWS's managed Kubernetes service, so it satisfies the requirement to avoid managing the Kubernetes control plane directly. The AWS CDK documentation also confirms that Python is a fully supported client language for defining infrastructure as code.

AWS CDK is the best fit because it lets engineers define cloud infrastructure programmatically in Python and deploy it in a repeatable way. The AWS CDK EKS construct library specifically supports defining Amazon EKS clusters and related Kubernetes resources. This makes it a strong match for infrastructure that must be reproducible and expressed in code rather than provisioned manually. Since the model is already packaged in a container, storing the image in Amazon ECR and then deploying it to Amazon EKS follows the normal AWS container workflow.

The other options are less suitable. Option A requires setting up and managing a Kubernetes cluster on EC2, which violates the requirement to avoid control-plane management. Option B uses the AWS CLI, but the question specifically requires infrastructure provisioning by using Python , not command-line provisioning.

Option D uses CloudFormation, which is repeatable infrastructure as code, but the question explicitly says the infrastructure must be provisioned by using Python . AWS CDK uniquely satisfies both the IaC and Python requirements while using managed Kubernetes with EKS.

Therefore, the best verified AWS-docs answer is C .

NEW QUESTION: 44

A company launches a feature that predicts home prices. An ML engineer trained a regression model using the SageMaker AI XGBoost algorithm. The model performs well on training data but underperforms on real- world validation data.

Which solution will improve the validation score with the LEAST implementation effort?

A. Create a larger training dataset with more real-world data and retrain.

B. Increase the num_round hyperparameter.

C. Change the eval_metric from RMSE to Error.

D. Increase the lambda hyperparameter.

Answer: D (LEAVE A REPLY)

This scenario indicates overfitting. AWS documentation for XGBoost recommends increasing the L2 regularization parameter (lambda) to reduce overfitting and improve generalization.

Increasing num_round worsens overfitting. Changing evaluation metrics does not change model behavior.

Collecting more data is effective but requires significant effort.

Regularization is a low-effort, high-impact fix.

Therefore, Option D is correct.

NEW QUESTION: 45
An ML engineer wants to use Amazon SageMaker Data Wrangler to perform preprocessing on a dataset. The ML engineer wants to use the processed dataset to train a classification model. During preprocessing, the ML engineer notices that a text feature has a range of thousands of values that differ only by spelling errors. The ML engineer needs to apply an encoding method so that after preprocessing is complete, the text feature can be used to train the model.
Which solution will meet these requirements?
A. Perform ordinal encoding to represent categories of the feature.
B. Perform similarity encoding to represent categories of the feature.
C. Perform one-hot encoding to represent categories of the feature.
D. Perform target encoding to represent categories of the feature.
Answer: B (LEAVE A REPLY)
The text feature contains high-cardinality categorical values with minor spelling variations, which is a common real-world data quality issue.
Traditional encoding techniques such as ordinal encoding (Option A) or one-hot encoding (Option C) would treat each misspelled value as a completely separate category, resulting in poor generalization and very high dimensionality.
Similarity encoding addresses this problem by grouping or encoding categories based on string similarity.
Categories that differ only slightly-such as spelling errors-are encoded with similar numerical representations. Amazon SageMaker Data Wrangler supports similarity encoding specifically for such noisy categorical features.
Target encoding (Option D) depends on the target label distribution and risks target leakage if not carefully applied.
Therefore, similarity encoding is the most appropriate and AWS-recommended solution.

NEW QUESTION: 46
A company wants to improve its customer retention ML model. The current model has 85% accuracy and a new model shows 87% accuracy in testing. The company wants to validate the new model's performance in production.
Which solution will meet these requirements?
A. Deploy the new model for 4 weeks across all production traffic. Monitor performance metrics and validate improvements.
B. Run A/B testing on both models for 4 weeks. Route 20% of traffic to the new model. Monitor customer retention rates across both variants.
C. Run both models in parallel for 4 weeks. Analyze offline predictions weekly by using historical customer data analysis.
D. Implement alternating deployments for 4 weeks between the current model and the new model. Track performance metrics for comparison.
Answer: B (LEAVE A REPLY)
AWS ML best practices recommend A/B testing to validate model improvements in production while minimizing risk. By routing a controlled portion of live traffic (for example, 20%) to the new model and keeping the majority of traffic on the existing model, the company can directly compare real-world performance using the same data distribution.
This approach allows statistically meaningful comparison of business metrics such as customer retention, rather than relying solely on offline accuracy. It also limits potential negative impact if the new model underperforms in production.
Deploying the new model to 100% of traffic (Option A) introduces unnecessary risk. Offline analysis (Option C) does not reflect live user behavior.
Alternating deployments (Option D) introduces confounding factors such as time-based effects.
Therefore, A/B testing is the correct solution.

NEW QUESTION: 47
A company is developing ML models by using PyTorch and TensorFlow estimators with Amazon SageMaker AI. An ML engineer configures the SageMaker AI estimator and now needs to initiate a training job that uses a training dataset.
Which SageMaker AI SDK method can initiate the training job?
A. fit method
B. create_model method
C. deploy method
D. predict method
Answer: A (LEAVE A REPLY)
In the Amazon SageMaker Python SDK, the fit() method is used to start a training job after an estimator has been configured. AWS documentation explicitly states that once an estimator (such as PyTorch or TensorFlow) is defined with parameters like instance type, framework version, and hyperparameters, the fit() method is responsible for launching the training process.
The fit() method accepts the training data location (commonly an Amazon S3 URI) and initiates the managed training job on SageMaker infrastructure. SageMaker then provisions the required compute resources, stages the data, executes the training script, and stores model artifacts in Amazon S3.
The create_model() method is used after training to create a SageMaker model object from trained artifacts.
The deploy() method deploys a trained model to an endpoint for inference. The predict() method is used only after deployment to request predictions from an endpoint.
AWS documentation clearly separates these lifecycle steps and identifies fit() as the correct method to initiate training.
Therefore, Option A is the correct and AWS-verified answer.

NEW QUESTION: 48
A company is building a near real-time data analytics application to detect anomalies and failures for industrial equipment. The company has thousands of IoT sensors that send data every 60 seconds. When new versions of the application are released, the company wants to ensure that application code bugs do not prevent the application from running.
Which solution will meet these requirements?
A. Use Amazon Managed Service for Apache Flink with the system rollback capability enabled to build the data analytics application.
B. Use Amazon Managed Service for Apache Flink with manual rollback when an error occurs to build the data analytics application.
C. Use Amazon Data Firehose to deliver real-time streaming data programmatically for the data analytics application. Pause the stream when a new version of the application is released and resume the stream after the application is deployed.
D. Use Amazon Data Firehose to deliver data to Amazon EC2 instances across two Availability Zones for the data analytics application.
Answer: A (LEAVE A REPLY)
For near real-time anomaly detection on streaming IoT data, AWS recommends Amazon Managed Service for Apache Flink. Flink is designed for stateful, low-latency stream processing and is well suited for time- series sensor analytics.
A key requirement is application resilience during deployments. Managed Flink supports system rollback, which automatically reverts to the last stable application version if a new deployment fails. This capability ensures uninterrupted processing even if application bugs are introduced,

meeting the company's reliability requirement.
Manual rollback (Option B) introduces operational risk and delays. Amazon Data Firehose (Options C and D) is a delivery service and does not support complex anomaly detection logic or application version rollback.
Therefore, using Managed Flink with system rollback enabled is the correct solution.

NEW QUESTION: 49
A company is creating an ML model to identify defects in a product. The company has gathered a dataset and has stored the dataset in TIFF format in Amazon S3. The dataset contains 200 images in which the most common defects are visible. The dataset also contains 1,800 images in which there is no defect visible.
An ML engineer trains the model and notices poor performance in some classes. The ML engineer identifies a class imbalance problem in the dataset.
What should the ML engineer do to solve this problem?
A. Use a few hundred images and Amazon Rekognition Custom Labels to train a new model.
B. Undersample the 200 images in which the most common defects are visible.
C. Oversample the 200 images in which the most common defects are visible.
D. Use all 2,000 images and Amazon Rekognition Custom Labels to train a new model.
Answer: C (LEAVE A REPLY)
Class imbalance occurs when one class significantly outnumbers another, causing models to bias predictions toward the majority class. In this case, images without defects (1,800) vastly outnumber images with defects (200). AWS ML best practices recommend oversampling the minority class to improve class representation without discarding valuable data.
Oversampling techniques-such as duplicating minority samples or applying data augmentation-help the model better learn defect-related features. This approach preserves all available data and improves recall and precision for underrepresented defect classes.
Option B is incorrect because undersampling the minority class would further worsen imbalance. Option A unnecessarily reduces dataset size. Option D does not address the imbalance problem.
Thus, oversampling defect images is the correct solution.

NEW QUESTION: 50
A company is developing a generative AI conversational interface to assist customers with payments. The company wants to use an ML solution to detect customer intent. The company does not have training data to train a model.
Which solution will meet these requirements?
A. Fine-tune a sequence-to-sequence (seq2seq) algorithm in Amazon SageMaker JumpStart.
B. Use an LLM from Amazon Bedrock with zero-shot learning.
C. Use the Amazon Comprehend DetectEntities API.
D. Run an LLM from Amazon Bedrock on Amazon EC2 instances.
Answer: (SHOW ANSWER)
The key requirement in this scenario is detecting customer intent without having any training data. According to AWS Machine Learning and Generative AI documentation, zero-shot learning is specifically designed for situations where labeled training data is unavailable. Zero-shot learning allows a pre-trained large language model (LLM) to perform tasks it has not been explicitly trained on by leveraging its general knowledge and language understanding.
Amazon Bedrock provides fully managed access to foundation models (FMs) and LLMs that support zero- shot and few-shot learning. By using an LLM from Amazon Bedrock, the company can directly infer customer intent from natural language inputs without building, training, or fine-tuning a

custom model. This approach is ideal for conversational interfaces where rapid deployment and scalability are required.
Option A is incorrect because fine-tuning a sequence-to-sequence (seq2seq) model in Amazon SageMaker JumpStart still requires labeled training data. Since the company explicitly does not have training data, this option does not meet the requirement.
Option C is also incorrect because the Amazon Comprehend DetectEntities API is designed for named entity recognition (NER), such as detecting names, dates, locations, or monetary values. It does not perform intent detection and is not suitable for conversational AI intent classification.
Option D is partially misleading. While it is technically possible to run an LLM on Amazon EC2, this does not inherently solve the problem of intent detection without training data. Additionally, Amazon Bedrock already abstracts infrastructure management, scaling, and model hosting, making direct EC2 deployment unnecessary and less efficient.
Therefore, using an LLM from Amazon Bedrock with zero-shot learning is the most appropriate, scalable, and AWS-recommended solution for intent detection without training data.

NEW QUESTION: 51
A company is developing an ML model to forecast future values based on time series data. The dataset includes historical measurements collected at regular intervals and categorical features. The model needs to predict future values based on past patterns and trends.
Which algorithm and hyperparameters should the company use to develop the model?
A. Use the Amazon SageMaker AI XGBoost algorithm. Set the scale_pos_weight hyperparameter to adjust for class imbalance.
B. Use k-means clustering with k to specify the number of clusters.
C. Use the Amazon SageMaker AI DeepAR algorithm with matching context length and prediction length hyperparameters.
D. Use the Amazon SageMaker AI Random Cut Forest (RCF) algorithm with contamination to set the expected proportion of anomalies.
Answer: (SHOW ANSWER)
The problem is a time series forecasting task with historical data and categorical features. Amazon SageMaker DeepAR is purpose-built for this use case. DeepAR uses recurrent neural networks to learn temporal patterns across multiple related time series and supports categorical covariates.
The context length hyperparameter controls how much historical data the model uses as input, while the prediction length specifies how far into the future the model forecasts. Correctly setting these hyperparameters is critical for capturing trends and seasonality.
XGBoost is a general-purpose tabular algorithm and does not model temporal dependencies natively. k-means is a clustering algorithm. Random Cut Forest is used for anomaly detection, not forecasting.
Therefore, DeepAR with appropriate context and prediction lengths is the correct and AWS-recommended solution.

NEW QUESTION: 52
A company has AWS Glue data processing jobs that are orchestrated by an AWS Glue workflow. The AWS Glue jobs can run on a schedule or can be launched manually.
The company is developing pipelines in Amazon SageMaker Pipelines for ML model development. The pipelines will use the output of the AWS Glue jobs during the data processing phase of model development.
An ML engineer needs to implement a solution that integrates the AWS Glue jobs with the pipelines.
Which solution will meet these requirements with the LEAST operational overhead?
A. Use AWS Step Functions for orchestration of the pipelines and the AWS Glue jobs.
B. Use processing steps in SageMaker Pipelines. Configure inputs that point to the Amazon Resource Names (ARNs) of the AWS Glue jobs.
C. Use Callback steps in SageMaker Pipelines to start the AWS Glue workflow and to stop the pipelines until the AWS Glue jobs finish running.
D. Use Amazon EventBridge to invoke the pipelines and the AWS Glue jobs in the desired order.
Answer: C (LEAVE A REPLY)
Callback steps in Amazon SageMaker Pipelines allow you to integrate external processes, such as AWS Glue jobs, into the pipeline workflow. By

using a Callback step, the SageMaker pipeline can trigger the AWS Glue workflow and pause execution until the Glue jobs complete. This approach provides seamless integration with minimal operational overhead, as it directly ties the pipeline's execution flow to the completion of the AWS Glue jobs without requiring additional orchestration tools or complex setups.

NEW QUESTION: 53
An ML engineer needs to create data ingestion pipelines and ML model deployment pipelines on AWS. All the raw data is stored in Amazon S3 buckets.
Which solution will meet these requirements?
A. Use Amazon Data Firehose to create the data ingestion pipelines. Use Amazon SageMaker Studio Classic to create the model deployment pipelines.
B. Use AWS Glue to create the data ingestion pipelines. Use Amazon SageMaker Studio Classic to create the model deployment pipelines.
C. Use Amazon Redshift ML to create the data ingestion pipelines. Use Amazon SageMaker Studio Classic to create the model deployment pipelines.
D. Use Amazon Athena to create the data ingestion pipelines. Use an Amazon SageMaker notebook to create the model deployment pipelines.
Answer: B (LEAVE A REPLY)
AWS Glue is a serverless data integration service that is well-suited for creating data ingestion pipelines, especially when raw data is stored in Amazon S3. It can clean, transform, and catalog data, making it accessible for downstream ML tasks.
Amazon SageMaker Studio Classic provides a comprehensive environment for building, training, and deploying ML models. It includes built-in tools and capabilities to create efficient model deployment pipelines with minimal setup.
This combination ensures seamless integration of data ingestion and ML model deployment with minimal operational overhead.

NEW QUESTION: 54
A hospital wants to predict patient outcomes for the coming year An ML engineer must improve several existing ML models that currently perform poorly.
Select the correct regularization method from the following list to improve each model Select each regularization method one time, more than one time, or not at all. (Select THREE.)
* L1 regularization
* L2 regularization
* Early stopping

A linear regression model whose coefficients should shrink but not become zero: [▼ L1 regularization / L2 regularization / Early stopping]

A polynomial regression model with irrelevant polynomial terms that should be eliminated: [▼ L1 regularization / L2 regularization / Early stopping]

A logistic regression model that has highly correlated features to eliminate highly redundant predictors: [▼ L1 regularization / L2 regularization / Early stopping]

Answer:

A linear regression model whose coefficients should shrink but not become zero: [▼ L1 regularization / L2 regularization / Early stopping]

A polynomial regression model with irrelevant polynomial terms that should be eliminated: [▼ L1 regularization / L2 regularization / Early stopping]

A logistic regression model that has highly correlated features to eliminate highly redundant predictors: [▼ L1 regularization / L2 regularization / Early stopping]

Explanation:
Linear regression model whose coefficients should shrink but not become zero answer: L2 regularization AWS says L2 produces smaller overall weight values and is the right fit when coefficients should be reduced without being forced to zero.
Polynomial regression model with irrelevant polynomial terms that should be eliminated answer: L1 regularization AWS says L1 reduces the number of features used by pushing small weights to zero , which matches elimination of irrelevant terms.
Logistic regression model that has highly correlated features to eliminate highly redundant predictors L1 regularization This is the nuanced one. AWS says L2 stabilizes weights when there is high correlation between features , but because the question explicitly says eliminate highly redundant predictors , L1 is the better match since it creates sparsity and removes predictors by zeroing coefficients.

NEW QUESTION: 55

A company wants to use Amazon SageMaker AI to host an ML model that runs on CPU for real-time predictions. The model has intermittent traffic during business hours and periods of no traffic after business hours.
Which hosting option will serve inference requests in the MOST cost-effective manner?
A. Deploy the model to a real-time endpoint with scheduled auto scaling.
B. Deploy the model to a SageMaker AI Serverless Inference endpoint with provisioned concurrency during business hours.
C. Deploy the model to an asynchronous inference endpoint with auto scaling to zero.
D. Deploy the model to a real-time endpoint and activate it only during business hours using AWS Lambda.
Answer: B (LEAVE A REPLY)
AWS recommends SageMaker Serverless Inference for workloads with intermittent or unpredictable traffic.
Serverless inference automatically scales compute resources to zero when idle, eliminating costs during periods with no traffic.
For business-hour traffic spikes, provisioned concurrency ensures low-latency responses while still avoiding the cost of continuously running instances. This model is especially cost-effective for CPU-based inference workloads.
Real-time endpoints incur costs even when idle, and asynchronous inference is designed for long-running jobs rather than low-latency predictions.
AWS documentation explicitly states that Serverless Inference is the most cost-effective option for intermittent real-time workloads.
Therefore, Option B is the correct choice.
NEW QUESTION: 56
A company that has hundreds of data scientists is using Amazon SageMaker to create ML models. The models are in model groups in the SageMaker Model Registry.
The data scientists are grouped into three categories: computer vision, natural language processing (NLP), and speech recognition. An ML engineer needs to implement a solution to organize the existing models into these groups to improve model discoverability at scale. The solution must not affect the integrity of the model artifacts and their existing groupings.
Which solution will meet these requirements?
A. Create a custom tag for each of the three categories. Add the tags to the model packages in the SageMaker Model Registry.
B. Create a model group for each category. Move the existing models into these category model groups.
C. Use SageMaker ML Lineage Tracking to automatically identify and tag which model groups should contain the models.
D. Create a Model Registry collection for each of the three categories. Move the existing model groups into the collections.
Answer: (SHOW ANSWER)
Using custom tags allows you to organize and categorize models in the SageMaker Model Registry without altering their existing groupings or affecting the integrity of the model artifacts. Tags are a lightweight and scalable way to improve model discoverability at scale, enabling the data scientists to filter and identify models by category (e.g., computer vision, NLP, speech recognition). This approach meets the requirements efficiently without introducing structural changes to the existing model registry setup.

NEW QUESTION: 57
A company is using Amazon SageMaker AI to develop a credit risk assessment model. During model validation, the company finds that the model achieves 82% accuracy on the validation data. However, the model achieved 99% accuracy on the training data. The company needs to address the model accuracy issue before deployment.
Which solution will meet this requirement?
A. Add more dense layers to increase model complexity. Implement batch normalization. Use early stopping during training.
B. Implement dropout layers. Use L1 or L2 regularization. Perform k-fold cross-validation.
C. Use principal component analysis (PCA) to reduce the feature dimensionality. Decrease model layers.
Implement cross-entropy loss functions.

D. Augment the training dataset. Remove duplicate records from the training dataset. Implement stratified sampling.

Answer: B (LEAVE A REPLY)

The large gap between training accuracy (99%) and validation accuracy (82%) is a textbook case of overfitting. The model has learned patterns that fit the training data extremely well but do not generalize to unseen data.

AWS ML best practices recommend regularization techniques to address overfitting. Dropout layers randomly deactivate neurons during training, preventing the network from relying too heavily on specific paths. L1 and L2 regularization penalize large weights, reducing model complexity and improving generalization. k-fold cross-validation provides a more robust evaluation by training and validating the model across multiple data splits.

Option A increases complexity, which would worsen overfitting. Option C mixes valid ideas (dimensionality reduction) with unrelated changes (loss function choice) and is less targeted. Option D focuses on data quality but does not directly address model variance.

Therefore, implementing dropout, regularization, and k-fold cross-validation is the correct solution.

NEW QUESTION: 58

An ML engineer needs to deploy ML models to get inferences from large datasets in an asynchronous manner. The ML engineer also needs to implement scheduled monitoring of the data quality of the models.

The ML engineer must receive alerts when changes in data quality occur.

Which solution will meet these requirements?

A. Deploy the models by using scheduled AWS Glue jobs. Use Amazon CloudWatch alarms to monitor the data quality and to send alerts.

B. Deploy the models by using scheduled AWS Batch jobs. Use AWS CloudTrail to monitor the data quality and to send alerts.

C. Deploy the models by using Amazon Elastic Container Service (Amazon ECS) on AWS Fargate. Use Amazon EventBridge to monitor the data quality and to send alerts.

D. Deploy the models by using Amazon SageMaker batch transform. Use SageMaker Model Monitor to monitor the data quality and to send alerts.

Answer: D (LEAVE A REPLY)

Amazon SageMaker batch transform is ideal for obtaining inferences from large datasets in an asynchronous manner, as it processes data in batches rather than requiring real-time inputs.

SageMaker Model Monitor allows scheduled monitoring of data quality, detecting shifts in input data characteristics, and generating alerts when changes in data quality occur.

This solution provides a fully managed, efficient way to handle both asynchronous inference and data quality monitoring with minimal operational overhead.

NEW QUESTION: 59

A company has an existing Amazon SageMaker AI model (v1) on a production endpoint. The company develops a new model version (v2) and needs to test v2 in production before substituting v2 for v1.

The company needs to minimize the risk of v2 generating incorrect output in production and must prevent any disruption of production traffic during the change.

Which solution will meet these requirements?

A. Create a second production variant for v2. Assign 1% of the traffic to v2 and 99% to v1. Collect all output of v2 in Amazon S3. If v2 performs as expected, switch all traffic to v2.

B. Create a second production variant for v2. Assign 10% of the traffic to v2 and 90% to v1. Collect all output of v2 in Amazon S3. If v2 performs as expected, switch all traffic to v2.

C. Deploy v2 to a new endpoint. Turn on data capture for the production endpoint. Send 100% of the input data to v2.

D. Deploy v2 into a shadow variant that samples 100% of the inference requests. Collect all output in Amazon S3. If v2 performs as expected,

promote v2 to production.
Answer: D (LEAVE A REPLY)
AWS recommends SageMaker shadow testing as the safest way to validate a new model version using real production traffic without impacting production responses. A shadow variant receives a copy of inference requests but does not return predictions to end users. This completely eliminates the risk of exposing incorrect predictions.
Options A and B are canary deployments. While useful, they still allow v2 to return responses to real users, which violates the requirement to prevent disruption. Option C sends 100% of traffic to v2 externally and is risky.
Shadow variants are explicitly designed to minimize risk while using real data, making them the AWS best practice for pre-production validation.
Therefore, Option D is correct.

NEW QUESTION: 60
A company wants to deploy an Amazon SageMaker AI model that can queue requests. The model needs to handle payloads of up to 1 GB that take up to 1 hour to process. The model must return an inference for each request. The model also must scale down when no requests are available to process.
Which inference option will meet these requirements?
A. Asynchronous inference
B. Batch transform
C. Serverless inference
D. Real-time inference
Answer: (SHOW ANSWER)
Amazon SageMaker Asynchronous Inference is specifically designed for long-running inference requests and large payloads. It supports payload sizes up to 1 GB and processing times of up to 1 hour, while automatically queuing requests.
Asynchronous inference stores results in Amazon S3 and allows clients to retrieve inference outputs after processing completes. It also supports auto scaling down to zero when there are no incoming requests, reducing cost.
Batch transform is intended for offline, bulk inference and does not return per-request results in an asynchronous request-response pattern.
Serverless and real-time inference have strict payload size and timeout limits that do not support 1-hour processing.
Therefore, asynchronous inference is the only SageMaker inference option that meets all stated requirements.

NEW QUESTION: 61
An ML engineer is designing an AI-powered traffic management system. The system must use near real-time inference to predict congestion and prevent collisions.
The system must also use batch processing to perform historical analysis of predictions over several hours to improve the model. The inference endpoints must scale automatically to meet demand.
Which combination of solutions will meet these requirements? (Select TWO.)
A. Use Amazon SageMaker real-time inference endpoints with automatic scaling based on ConcurrentInvocationsPerInstance.
B. Use AWS Lambda with reserved concurrency and SnapStart to connect to SageMaker endpoints.
C. Use an Amazon SageMaker Processing job for batch historical analysis. Schedule the job with Amazon EventBridge.
D. Use Amazon EC2 Auto Scaling to host containers for batch analysis.
E. Use AWS Lambda for historical analysis.
Answer: A,C (LEAVE A REPLY)
For near real-time predictions, AWS documentation recommends Amazon SageMaker real-time inference endpoints. These endpoints support

automatic scaling based on metrics such as ConcurrentInvocationsPerInstance, ensuring low latency and high availability during traffic spikes.
For long-running historical analysis, SageMaker Processing jobs are the appropriate solution. Processing jobs are designed for batch workloads, can run for hours, and integrate cleanly with SageMaker pipelines.
Scheduling them with Amazon EventBridge provides a fully managed, scalable, and serverless solution.
AWS Lambda is unsuitable for multi-hour workloads. EC2 Auto Scaling adds unnecessary infrastructure management overhead.
Therefore, Options A and C together meet all requirements and align with AWS best practices.

NEW QUESTION: 62
A company has trained an ML model in Amazon SageMaker. The company needs to host the model to provide inferences in a production environment.
The model must be highly available and must respond with minimum latency. The size of each request will be between 1 KB and 3 MB. The model will receive unpredictable bursts of requests during the day. The inferences must adapt proportionally to the changes in demand.
How should the company deploy the model into production to meet these requirements?
A. Create a SageMaker real-time inference endpoint. Configure auto scaling. Configure the endpoint to present the existing model.
B. Deploy the model on an Amazon Elastic Container Service (Amazon ECS) cluster. Use ECS scheduled scaling that is based on the CPU of the ECS cluster.
C. Install SageMaker Operator on an Amazon Elastic Kubernetes Service (Amazon EKS) cluster. Deploy the model in Amazon EKS. Set horizontal pod auto scaling to scale replicas based on the memory metric.
D. Use Spot Instances with a Spot Fleet behind an Application Load Balancer (ALB) for inferences. Use the ALBRequestCountPerTarget metric as the metric for auto scaling.
Answer: (SHOW ANSWER)
Amazon SageMaker real-time inference endpoints are designed to provide low-latency predictions in production environments. They offer built-in auto scaling to handle unpredictable bursts of requests, ensuring high availability and responsiveness. This approach is fully managed, reduces operational complexity, and is optimized for the range of request sizes (1 KB to 3 MB) specified in the requirements.

NEW QUESTION: 63
A company collects customer data daily and stores it as compressed files in an Amazon S3 bucket partitioned by date. Each month, analysts process the data, check data quality, and upload results to Amazon QuickSight dashboards.
An ML engineer needs to automatically check data quality before the data is sent to QuickSight, with the LEAST operational overhead.
Which solution will meet these requirements?
A. Run an AWS Glue crawler monthly and use AWS Glue Data Quality rules to check data quality.
B. Run an AWS Glue crawler and create a custom AWS Glue job with PySpark to evaluate data quality.
C. Use AWS Lambda with Python scripts triggered by S3 uploads to evaluate data quality.
D. Send S3 events to Amazon SQS and use Amazon CloudWatch Insights to evaluate data quality.

Answer: A (LEAVE A REPLY)
AWS Glue Data Quality provides managed, declarative data quality checks with minimal configuration.
Combined with Glue crawlers, it enables automatic schema discovery and quality validation without custom code.
Option A uses native AWS services designed for this exact purpose, minimizing operational overhead.
Options B and C require custom code and maintenance. Option D is not designed for data validation.
AWS documentation explicitly recommends Glue Data Quality rules for scalable, automated data quality checks in analytics pipelines.
Therefore, Option A is the correct and AWS-aligned solution.

NEW QUESTION: 64
A company is training a deep learning model to detect abnormalities in images. The company has limited GPU resources and a large hyperparameter space to explore. The company needs to test different configurations and avoid wasting computation time on poorly performing models that show weak validation accuracy in early epochs.
Which hyperparameter optimization strategy should the company use?
A. Grid search across all possible combinations
B. Bayesian optimization with early stopping
C. Manual tuning of each parameter individually
D. Exhaustive search without early stopping
Answer: (SHOW ANSWER)
When GPU resources are limited and the hyperparameter search space is large, AWS documentation strongly recommends Bayesian optimization combined with early stopping. Bayesian optimization uses past evaluation results to intelligently select the next set of hyperparameters to test, focusing exploration on promising regions of the search space rather than testing all combinations.
In Amazon SageMaker, Bayesian optimization is the default and recommended strategy for hyperparameter tuning jobs. It significantly reduces the number of training runs required compared to grid or random search, making it highly cost-efficient for deep learning workloads.
Early stopping further improves efficiency by terminating training jobs that show poor validation performance in early epochs. This prevents wasted GPU time on configurations that are unlikely to perform well. AWS explicitly documents early stopping as a key feature for controlling training cost and duration.
Grid search and exhaustive search are computationally expensive and impractical for large hyperparameter spaces. Manual tuning is slow, error-prone, and does not scale.
By combining Bayesian optimization with early stopping, the company can rapidly converge on high- performing hyperparameter configurations while minimizing resource usage.
Therefore, Option B is the correct and AWS-aligned solution.

NEW QUESTION: 65
A company has a large collection of chat recordings from customer interactions after a product release. An ML engineer needs to create an ML model to analyze the chat data. The ML engineer needs to determine the success of the product by reviewing customer sentiments about the product.
Which action should the ML engineer take to complete the evaluation in the LEAST amount of time?
A. Use Amazon Rekognition to analyze sentiments of the chat conversations.
B. Train a Naive Bayes classifier to analyze sentiments of the chat conversations.
C. Use Amazon Comprehend to analyze sentiments of the chat conversations.
D. Use random forests to classify sentiments of the chat conversations.

Answer: (SHOW ANSWER)
Amazon Comprehend is a fully managed natural language processing (NLP) service that includes a built-in sentiment analysis feature. It can quickly and efficiently analyze text data to determine whether the sentiment is positive, negative, neutral, or mixed. Using Amazon Comprehend requires minimal setup and provides accurate results without the need to train and deploy custom models, making it the fastest and most efficient solution for this task.

NEW QUESTION: 66
An ML engineer needs to implement a solution to host a trained ML model. The rate of requests to the model will be inconsistent throughout the day.
The ML engineer needs a scalable solution that minimizes costs when the model is not in use. The solution also must maintain the model's capacity to respond to requests during times of peak usage.
Which solution will meet these requirements?
A. Deploy the model to an Amazon SageMaker endpoint. Deploy multiple copies of the model to the endpoint. Create an Application Load Balancer to route traffic between the different copies of the model at the endpoint.
B. Deploy the model to an Amazon SageMaker endpoint. Create SageMaker endpoint auto scaling policies that are based on Amazon CloudWatch metrics to adjust the number of instances dynamically.
C. Create AWS Lambda functions that have fixed concurrency to host the model. Configure the Lambda functions to automatically scale based on the number of requests to the model.
D. Deploy the model on an Amazon Elastic Container Service (Amazon ECS) cluster that uses AWS Fargate. Set a static number of tasks to handle requests during times of peak usage.
Answer: (SHOW ANSWER)

NEW QUESTION: 67
An ML engineer is building a model to predict house and apartment prices. The model uses three features:
Square Meters, Price, and Age of Building. The dataset has 10,000 data rows. The data includes data points for one large mansion and one extremely small apartment.
The ML engineer must perform preprocessing on the dataset to ensure that the model produces accurate predictions for the typical house or apartment.
Which solution will meet these requirements?
A. Remove the outliers and perform a log transformation on the Square Meters variable.
B. Keep the outliers and perform normalization on the Square Meters variable.
C. Remove the outliers and perform one-hot encoding on the Square Meters variable.
D. Keep the outliers and perform one-hot encoding on the Square Meters variable.
Answer: (SHOW ANSWER)
In regression problems such as house price prediction, extreme values can significantly distort model learning.
In this dataset, the presence of a large mansion and an extremely small apartment represents clear outliers in the Square Meters feature. According to AWS Machine Learning best practices, outliers can disproportionately influence loss functions (such as mean squared error), leading to poor predictions for the majority of typical data points.
Removing these outliers helps the model focus on learning patterns that apply to the majority of houses and apartments, which aligns with the requirement to produce accurate predictions for typical properties. After removing outliers, applying a log transformation to the Square Meters feature further improves model performance by reducing skewness and stabilizing variance. Log transformations are commonly recommended in

AWS and general ML documentation when numerical features span multiple orders of magnitude.
Option B is incorrect because normalization alone does not address the undue influence of extreme outliers.
Option C and D are incorrect because one-hot encoding is intended for categorical variables, not continuous numerical features such as square meters.
Therefore, removing outliers and applying a log transformation is the most statistically sound preprocessing approach.

NEW QUESTION: 68
An ML engineer has trained a neural network by using stochastic gradient descent (SGD). The neural network performs poorly on the test set. The values for training loss and validation loss remain high and show an oscillating pattern. The values decrease for a few epochs and then increase for a few epochs before repeating the same cycle.
What should the ML engineer do to improve the training process?
A. Introduce early stopping.
B. Increase the size of the test set.
C. Increase the learning rate.
D. Decrease the learning rate.
Answer: D (LEAVE A REPLY)
An oscillating loss pattern during training with stochastic gradient descent (SGD) is a strong indicator that the learning rate is too high. When the learning rate is excessive, the optimizer takes overly large steps during gradient updates, causing the model to repeatedly overshoot the optimal minimum of the loss function. This results in unstable convergence behavior, where training and validation loss decrease briefly and then increase again in a repeating cycle.
AWS Machine Learning documentation and general deep learning best practices recommend reducing the learning rate when training loss and validation loss both remain high and fluctuate rather than steadily decreasing. Lowering the learning rate allows the optimizer to take smaller, more precise steps toward the minimum, leading to smoother convergence and improved generalization on the test dataset.
Option A, early stopping, is used primarily to prevent overfitting when validation loss increases while training loss continues to decrease. In this scenario, both losses remain high and unstable, indicating an optimization issue rather than overfitting.
Option B is incorrect because increasing the test set size does not affect the training dynamics or convergence behavior of the model.
Option C would worsen the problem, as increasing the learning rate would further amplify oscillations and instability.
Therefore, decreasing the learning rate is the correct corrective action to stabilize SGD training and improve model performance.

NEW QUESTION: 69
An ML engineer has an Amazon Comprehend custom model in Account A in the us-east-1 Region. The ML engineer needs to copy the model to Account # in the same Region.
Which solution will meet this requirement with the LEAST development effort?
A. Use Amazon S3 to make a copy of the model. Transfer the copy to Account B.
B. Create a resource-based IAM policy. Use the Amazon Comprehend ImportModel API operation to copy the model to Account B.
C. Use AWS DataSync to replicate the model from Account A to Account B.
D. Create an AWS Site-to-Site VPN connection between Account A and Account # to transfer the model.
Answer: B (LEAVE A REPLY)
Amazon Comprehend provides the ImportModel API operation, which allows you to copy a custom model between AWS accounts. By creating a resource-based IAM policy on the model in Account A, you can grant Account B the necessary permissions to access and import the model. This approach requires minimal development effort and is the AWS-recommended method for sharing custom models across accounts.

NEW QUESTION: 70
An ML engineer is preparing a dataset that contains medical records to train an ML model to predict the likelihood of patients developing diseases.
The dataset contains columns for patient ID, age, medical conditions, test results, and a " Disease " target column.
How should the ML engineer configure the data to train the model?
A. Remove the patient ID column.
B. Remove the age column.
C. Remove the medical conditions and test results columns.
D. Remove the " Disease " target column.
Answer: (SHOW ANSWER)
Patient ID is a unique identifier and does not contain predictive information. Including it can cause the model to overfit by memorizing records rather than learning meaningful patterns.
AWS ML best practices recommend removing identifiers that are not causally related to the target variable.
Age, medical conditions, and test results are clinically relevant features and should be retained. The target column must remain for supervised learning.
Therefore, Option A is the correct and AWS-aligned choice.

NEW QUESTION: 71
A company has deployed an XGBoost prediction model in production to predict if a customer is likely to cancel a subscription. The company uses Amazon SageMaker Model Monitor to detect deviations in the F1 score.
During a baseline analysis of model quality, the company recorded a threshold for the F1 score. After several months of no change, the model ' s F1 score decreases significantly.
What could be the reason for the reduced F1 score?
A. Concept drift occurred in the underlying customer data that was used for predictions.
B. The model was not sufficiently complex to capture all the patterns in the original baseline data.
C. The original baseline data had a data quality issue of missing values.
D. Incorrect ground truth labels were provided to Model Monitor during the calculation of the baseline.
Answer: A (LEAVE A REPLY)
Problem Description:
The F1 score, which is a balance of precision and recall, has decreased significantly. This indicates the model ' s predictions are no longer aligned with the real-world data distribution.
Why Concept Drift?
Concept drift occurs when the statistical properties of the target variable or features change over time. For example, customer behaviors or subscription cancellation patterns may have shifted, leading to reduced model accuracy.
Signs of Concept Drift:
Deviation in performance metrics (e.g., F1 score) over time.
Declining prediction accuracy for certain groups or scenarios.
Solution:
Monitor for drift using tools like SageMaker Model Monitor.
Regularly retrain the model with updated data to account for the drift.
Why Not Other Options?:
B: Model complexity is unrelated if the model initially performed well.

C: Data quality issues would have been detected during baseline analysis.
D: Incorrect ground truth labels would have resulted in a consistently poor baseline.
Conclusion: The decrease in F1 score is most likely due to concept drift in the customer data, requiring retraining of the model with new data.

NEW QUESTION: 72
Case study
An ML engineer is developing a fraud detection model on AWS. The training dataset includes transaction logs, customer profiles, and tables from an on-premises MySQL database. The transaction logs and customer profiles are stored in Amazon S3.
The dataset has a class imbalance that affects the learning of the model's algorithm. Additionally, many of the features have interdependencies.
The algorithm is not capturing all the desired underlying patterns in the data.
Which AWS service or feature can aggregate the data from the various data sources?
A. Amazon EMR Spark jobs
B. Amazon Kinesis Data Streams
C. Amazon DynamoDB
D. AWS Lake Formation
Answer: A (LEAVE A REPLY)
* Problem Description:
* The dataset includes multiple data sources:
* Transaction logs and customer profiles in Amazon S3.
* Tables in an on-premises MySQL database.
* There is a class imbalance in the dataset and interdependencies among features that need to be addressed.
* The solution requires data aggregation from diverse sources for centralized processing.
* Why AWS Lake Formation?
* AWS Lake Formation is designed to simplify the process of aggregating, cataloging, and securing data from various sources, including S3, relational databases, and other on-premises systems.
* It integrates with AWS Glue for data ingestion and ETL (Extract, Transform, Load) workflows, making it a robust choice for aggregating data from Amazon S3 and on-premises MySQL databases.
* How It Solves the Problem:
* Data Aggregation: Lake Formation collects data from diverse sources, such as S3 and MySQL, and consolidates it into a centralized data lake.
* Cataloging and Discovery: Automatically crawls and catalogs the data into a searchable catalog, which the ML engineer can query for analysis or modeling.
* Data Transformation: Prepares data using Glue jobs to handle preprocessing tasks such as addressing class imbalance (e.g., oversampling, undersampling) and handling interdependencies among features.
* Security and Governance: Offers fine-grained access control, ensuring secure and compliant data management.
* Steps to Implement Using AWS Lake Formation:
* Step 1: Set up Lake Formation and register data sources, including the S3 bucket and on-premises MySQL database.
* Step 2: Use AWS Glue to create ETL jobs to transform and prepare data for the ML pipeline.
* Step 3: Query and access the consolidated data lake using services such as Athena or SageMaker for further ML processing.
* Why Not Other Options?
* Amazon EMR Spark jobs: While EMR can process large-scale data, it is better suited for complex big data analytics tasks and does not inherently support data aggregation across sources like Lake Formation.

* Amazon Kinesis Data Streams: Kinesis is designed for real-time streaming data, not batch data aggregation across diverse sources.
* Amazon DynamoDB: DynamoDB is a NoSQL database and is not suitable for aggregating data from multiple sources like S3 and MySQL.
Conclusion: AWS Lake Formation is the most suitable service for aggregating data from S3 and on-premises MySQL databases, preparing the data for downstream ML tasks, and addressing challenges like class imbalance and feature interdependencies.
AWS Lake Formation Documentation
AWS Glue for Data Preparation

NEW QUESTION: 73
A travel company wants to create an ML model to recommend the next airport destination for its users. The company has collected millions of data records about user location, recent search history on the company's website, and 2,000 available airports. The data has several categorical features with a target column that is expected to have a high-dimensional sparse matrix.
The company needs to use Amazon SageMaker AI built-in algorithms for the model. An ML engineer converts the categorical features by using one-hot encoding.
Which algorithm should the ML engineer implement to meet these requirements?
A. Use the CatBoost algorithm to recommend the next airport destination.
B. Use the DeepAR forecasting algorithm to recommend the next airport destination.
C. Use the Factorization Machines algorithm to recommend the next airport destination.
D. Use the k-means algorithm to cluster users into groups and map each group to the next airport destination.
Answer: (SHOW ANSWER)
This problem describes a recommendation system with millions of records, many categorical variables, and a high-dimensional sparse feature space created by one-hot encoding. AWS documentation explicitly recommends Amazon SageMaker Factorization Machines (FM) for such use cases.
Factorization Machines are designed to handle sparse datasets efficiently and to model interactions between categorical features without explicitly enumerating all feature combinations. This capability makes FM particularly well-suited for recommendation problems such as predicting user-item interactions, including destination recommendations.
With 2,000 possible airport destinations, the target space is large and sparse. One-hot encoding further increases sparsity. Factorization Machines address this challenge by learning latent factors that capture relationships between features, even when many feature combinations are rarely observed.
Option A (CatBoost) is not an Amazon SageMaker built-in algorithm and therefore does not meet the requirement. Option B (DeepAR) is a time-series forecasting algorithm, not intended for recommendation or classification problems. Option D (k-means) is an unsupervised clustering algorithm and cannot directly predict a specific destination label.
AWS documentation explicitly lists recommendation systems and click prediction as primary use cases for the SageMaker Factorization Machines algorithm.
Therefore, Option C is the correct and AWS-verified choice.

NEW QUESTION: 74
A company is building an Amazon SageMaker AI pipeline for an ML model. The pipeline uses distributed processing and distributed training.
An ML engineer needs to encrypt network communication between instances that run distributed jobs. The ML engineer configures the distributed jobs to run in a private VPC.
What should the ML engineer do to meet the encryption requirement?
A. Enable network isolation.

B. Configure traffic encryption by using security groups.
C. Enable inter-container traffic encryption.
D. Enable VPC flow logs.
Answer: C (LEAVE A REPLY)
In Amazon SageMaker, distributed training and distributed processing jobs often involve multiple instances exchanging data over the network. By default, when these jobs run inside a VPC, network traffic remains private but is not automatically encrypted between instances. When compliance or security requirements mandate encryption of in-transit data, additional configuration is required.
The correct solution is to enable inter-container traffic encryption, which ensures that all network communication between containers running on different instances is encrypted using TLS. Amazon SageMaker provides a built-in feature for this purpose. When inter-container traffic encryption is enabled, SageMaker automatically configures secure communication channels between all nodes participating in a distributed job, including training clusters and processing jobs.
Option A (Network isolation) is incorrect because network isolation prevents containers from making outbound network calls and accessing the internet. It does not encrypt traffic between instances.
Option B (Security groups) is incorrect because security groups control network access and traffic flow, not encryption. They can restrict which instances can communicate, but they do not provide data-in-transit encryption.
Option D (VPC flow logs) is incorrect because VPC flow logs are used for monitoring and auditing network traffic, not for encrypting it.
AWS documentation explicitly states that enabling inter-container traffic encryption is the recommended and supported approach for encrypting data exchanged between instances during distributed SageMaker jobs. This feature aligns with enterprise security best practices and regulatory requirements for protecting sensitive ML training data in transit.
Therefore, Option C is the only solution that directly fulfills the encryption requirement for distributed SageMaker workloads.

NEW QUESTION: 75
A company's ML engineer has deployed an ML model for sentiment analysis to an Amazon SageMaker endpoint. The ML engineer needs to explain to company stakeholders how the model makes predictions.
Which solution will provide an explanation for the model's predictions?
A. Use SageMaker Model Monitor on the deployed model.
B. Use SageMaker Clarify on the deployed model.
C. Show the distribution of inferences from A/# testing in Amazon CloudWatch.
D. Add a shadow endpoint. Analyze prediction differences on samples.
Answer: (SHOW ANSWER)
SageMaker Clarify is designed to provide explainability for ML models. It can analyze feature importance and explain how input features influence the model's predictions. By using Clarify with the deployed SageMaker model, the ML engineer can generate insights and present them to stakeholders to explain the sentiment analysis predictions effectively.

NEW QUESTION: 76
A company ingests sales transaction data using Amazon Data Firehose into Amazon OpenSearch Service. The Firehose buffer interval is set to 60 seconds.
The company needs sub-second latency for a real-time OpenSearch dashboard.
Which architectural change will meet this requirement?
A. Use zero buffering in the Firehose stream and tune the PutRecordBatch batch size.
B. Replace Firehose with AWS DataSync and enhanced fan-out consumers.

C. Increase the Firehose buffer interval to 120 seconds.
D. Replace Firehose with Amazon SQS.
Answer: (SHOW ANSWER)
Amazon Data Firehose supports near real-time delivery by configuring the buffer interval and buffer size.
AWS documentation states that setting the buffer interval to the minimum (as low as 1 second) enables low- latency ingestion.
By using zero or minimal buffering and tuning PutRecordBatch, data is delivered to OpenSearch almost immediately, enabling sub-second dashboard updates.
DataSync and SQS are not designed for real-time streaming analytics. Increasing the buffer interval worsens latency.
AWS explicitly recommends Firehose with minimal buffering for real-time OpenSearch ingestion.
Therefore, Option A is the correct and AWS-verified solution.

NEW QUESTION: 77
A company wants to improve the sustainability of its ML operations.
Which actions will reduce the energy usage and computational resources that are associated with the company ' s training jobs? (Choose two.)
A. Use Amazon SageMaker Debugger to stop training jobs when non-converging conditions are detected.
B. Use Amazon SageMaker Ground Truth for data labeling.
C. Deploy models by using AWS Lambda functions.
D. Use AWS Trainium instances for training.
E. Use PyTorch or TensorFlow with the distributed training option.
Answer: (SHOW ANSWER)
SageMaker Debugger can identify when a training job is not converging or is stuck in a non-productive state.
By stopping these jobs early, unnecessary energy and computational resources are conserved, improving sustainability.
AWS Trainium instances are purpose-built for ML training and are optimized for energy efficiency and cost- effectiveness. They use less energy per training task compared to general-purpose instances, making them a sustainable choice.

NEW QUESTION: 78
An ML engineer needs to use AWS CloudFormation to create an ML model that an Amazon SageMaker endpoint will host.
Which resource should the ML engineer declare in the CloudFormation template to meet this requirement?
A. AWS::SageMaker::Model
B. AWS::SageMaker::Endpoint
C. AWS::SageMaker::NotebookInstance
D. AWS::SageMaker::Pipeline
Answer: A (LEAVE A REPLY)
The AWS::SageMaker::Model resource in AWS CloudFormation is used to create an ML model in Amazon SageMaker. This model can then be

hosted on an endpoint by using the AWS::SageMaker::Endpoint resource. The model resource defines the container or algorithm to use for hosting and the S3 location of the model artifacts.

NEW QUESTION: 79
An ML engineer needs to deploy a trained model based on a genetic algorithm. Predictions can take several minutes, and requests can include up to 100 MB of data.
Which deployment solution will meet these requirements with the LEAST operational overhead?
A. Deploy on EC2 Auto Scaling behind an ALB.
B. Deploy to a SageMaker AI real-time endpoint.
D. Deploy to Amazon ECS on EC2.
C. Deploy to a SageMaker AI Asynchronous Inference endpoint.
Answer: C (LEAVE A REPLY)
SageMaker Asynchronous Inference is designed for long-running inference workloads and large payloads (up to 1 GB). Requests are queued, processed asynchronously, and results are written to Amazon S3.
Real-time endpoints have payload and timeout limits. EC2 and ECS require infrastructure management, increasing operational overhead.
AWS documentation explicitly recommends asynchronous inference for workloads with large inputs and long execution times.
Therefore, Option C is the correct and most efficient solution.

NEW QUESTION: 80
A company has an ML model that is deployed to an Amazon SageMaker AI endpoint for real-time inference.
The company needs to deploy a new model. The company must compare the new model's performance to the currently deployed model's performance before shifting all traffic to the new model.
Which solution will meet these requirements with the LEAST operational effort?
A. Deploy the new model to a separate endpoint. Manually split traffic between the two endpoints.
B. Deploy the new model to a separate endpoint. Use Amazon CloudFront to distribute traffic between the two endpoints.
C. Deploy the new model as a shadow variant on the same endpoint as the current model. Route a portion of live traffic to the shadow model for evaluation.
D. Use AWS Lambda functions with custom logic to route traffic between the current model and the new model.
Answer: (SHOW ANSWER)
AWS recommends shadow testing to evaluate a new model against a production model with minimal operational overhead. Using production variants on a single SageMaker endpoint allows traffic to be routed to multiple models without managing additional endpoints.
With a shadow variant, the new model receives a copy of live traffic but does not affect production responses.
Performance metrics such as latency, accuracy, and error rates can be compared directly against the current model using Amazon CloudWatch metrics. This approach is natively supported by Amazon SageMaker Endpoints.
Options A, B, and D introduce unnecessary complexity by requiring additional endpoints, traffic routing infrastructure, or custom code.
Therefore, deploying the new model as a shadow variant on the same endpoint is the most efficient solution.

NEW QUESTION: 81
An ML engineer uses one ML framework to train multiple ML models. The ML engineer needs to optimize inference costs and host the models on Amazon SageMaker AI.
Which solution will meet these requirements MOST cost-effectively?
A. Create a multi-container inference endpoint for direct invocation.
B. Create a multi-model inference endpoint for all the models.

C. Create a multi-container inference endpoint for sequential invocation.
D. Create multiple single-model inference endpoints for each model.
Answer: (SHOW ANSWER)
Amazon SageMaker multi-model endpoints (MME) are designed to host multiple models behind a single endpoint, dynamically loading models into memory on demand. AWS documentation explicitly recommends MME as the most cost-effective solution when multiple models share the same ML framework and inference container.
With MME, SageMaker loads models from Amazon S3 only when they are invoked and unloads idle models automatically. This dramatically reduces the number of instances required and avoids paying for always-on resources for infrequently used models.
Multi-container endpoints are intended for inference pipelines or ensembles and require all containers to be loaded at startup, which increases cost.
Deploying separate endpoints for each model results in the highest cost due to duplicated infrastructure.
AWS best practices clearly position multi-model endpoints as the optimal choice for reducing inference costs when hosting many models with similar runtime requirements.
Therefore, Option B is the correct and AWS-verified solution.

NEW QUESTION: 82
A company is planning to create several ML prediction models. The training data is stored in Amazon S3. The entire dataset is more than 5 ## in size and consists of CSV, JSON, Apache Parquet, and simple text files.
The data must be processed in several consecutive steps. The steps include complex manipulations that can take hours to finish running. Some of the processing involves natural language processing (NLP) transformations. The entire process must be automated.
Which solution will meet these requirements?
A. Process data at each step by using Amazon SageMaker Data Wrangler. Automate the process by using Data Wrangler jobs.
B. Use Amazon SageMaker notebooks for each data processing step. Automate the process by using Amazon EventBridge.
C. Process data at each step by using AWS Lambda functions. Automate the process by using AWS Step Functions and Amazon EventBridge.
D. Use Amazon SageMaker Pipelines to create a pipeline of data processing steps. Automate the pipeline by using Amazon EventBridge.
Answer: D (LEAVE A REPLY)
Amazon SageMaker Pipelines is designed for creating, automating, and managing end-to-end ML workflows, including complex data preprocessing tasks. It supports handling large datasets and can integrate with custom steps, such as NLP transformations. By combining SageMaker Pipelines with Amazon EventBridge, the entire workflow can be triggered and automated efficiently, meeting the requirements for scalability, automation, and processing complexity.

NEW QUESTION: 83
An ML engineer needs to run intensive model training jobs each month that can take 48-72 hours. The jobs can be interrupted and resumed. The engineer has a fixed budget and needs the most cost-effective compute option.
Which solution will meet these requirements?
A. Purchase Reserved Instances with partial upfront payment.
B. Purchase On-Demand Instances.
C. Purchase SageMaker AI Savings Plans.
D. Purchase Spot Instances that use automated checkpoints.
Answer: D (LEAVE A REPLY)
Amazon EC2 Spot Instances provide unused compute capacity at discounts of up to 90%. AWS documentation strongly recommends Spot Instances for interruptible workloads such as long-running ML training jobs that can resume from checkpoints.

By enabling automatic checkpointing, SageMaker saves model state periodically to Amazon S3. If the Spot Instance is interrupted, training can resume with minimal loss of progress.
Reserved Instances and Savings Plans require long-term commitment and are not as cost-effective for sporadic workloads. On-Demand Instances are the most expensive option.
Therefore, Option D is the most cost-effective solution.

NEW QUESTION: 84
A financial company receives a high volume of real-time market data streams from an external provider. The streams consist of thousands of JSON records per second.
The company needs a scalable AWS solution to identify anomalous data points with the LEAST operational overhead.
Which solution will meet these requirements?
A. Ingest data into Amazon Kinesis Data Streams. Use the built-in RANDOM_CUT_FOREST function in Amazon Managed Service for Apache Flink to detect anomalies.
B. Ingest data into Kinesis Data Streams. Deploy a SageMaker AI endpoint and use AWS Lambda to detect anomalies.
C. Ingest data into Apache Kafka on Amazon EC2 and use SageMaker AI for detection.
D. Send data to Amazon SQS and use AWS Glue ETL jobs for batch anomaly detection.
Answer: A (LEAVE A REPLY)
For real-time anomaly detection on streaming data, AWS recommends using Amazon Managed Service for Apache Flink, which provides serverless, scalable stream processing with built-in ML functions.
Apache Flink includes a native RANDOM_CUT_FOREST (RCF) function for unsupervised anomaly detection. This eliminates the need to deploy, manage, or scale custom ML endpoints. When combined with Amazon Kinesis Data Streams, the solution can process thousands of records per second with very low latency.
Option B introduces multiple services and custom logic, increasing operational overhead. Option C requires managing EC2 infrastructure and Kafka clusters. Option D is batch-oriented and does not meet real-time requirements.
AWS documentation explicitly highlights Kinesis + Managed Flink + RCF as the lowest-overhead solution for real-time anomaly detection.
Therefore, Option A is the correct and AWS-aligned choice.

NEW QUESTION: 85
An ML engineer needs to use AWS services to identify and extract meaningful unique keywords from documents.
Which solution will meet these requirements with the LEAST operational overhead?
A. Use the Natural Language Toolkit (NLTK) library on Amazon EC2 instances for text pre-processing.
Use the Latent Dirichlet Allocation (LDA) algorithm to identify and extract relevant keywords.
B. Use Amazon SageMaker and the BlazingText algorithm. Apply custom pre-processing steps for stemming and removal of stop words. Calculate term frequency-inverse document frequency (TF-IDF) scores to identify and extract relevant keywords.
C. Store the documents in an Amazon S3 bucket. Create AWS Lambda functions to process the documents and to run Python scripts for stemming and removal of stop words. Use bigram and trigram techniques to identify and extract relevant keywords.
D. Use Amazon Comprehend custom entity recognition and key phrase extraction to identify and extract relevant keywords.
Answer: D (LEAVE A REPLY)
Amazon Comprehend provides pre-built functionality for key phrase extraction and can identify meaningful keywords from documents with minimal setup or operational overhead. It eliminates the need for manual preprocessing, stemming, or stop-word removal and does not require custom model development or infrastructure management. This makes it the most efficient and low-maintenance solution for the task.

NEW QUESTION: 86
A company ' s ML engineer is creating a classification model. The ML engineer explores the dataset and notices a column named day_of_week. The column contains the following values: Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, and Sunday.
Which technique should the ML engineer use to convert this column's data to binary values?
A. Binary encoding
B. Label encoding
C. One-hot encoding
D. Tokenization
Answer: (SHOW ANSWER)
The day_of_week feature is a categorical variable with a small, fixed number of unique values and no inherent ordinal relationship. AWS machine learning best practices strongly recommend one-hot encoding for this type of categorical data when preparing features for classification models.
One-hot encoding converts each unique category into a separate binary feature (0 or 1). For example,
"Monday" becomes a column where Monday = 1 and all other days = 0. This ensures that the ML model does not incorrectly assume a numeric or ordered relationship between categories.
Option B (label encoding) assigns integer values to categories (e.g., Monday = 1, Tuesday = 2). AWS documentation cautions against this approach for nominal data because models may incorrectly infer ordinal meaning, leading to biased or inaccurate predictions.
Option A (binary encoding) is typically used for high-cardinality categorical features to reduce dimensionality. With only seven categories, AWS recommends one-hot encoding for clarity and interpretability.
Option D (tokenization) is used for text processing, such as NLP tasks, and is not appropriate for structured categorical features.
AWS SageMaker feature engineering guidelines emphasize that one-hot encoding is the preferred method for low-cardinality categorical variables in classification models, especially when using algorithms such as logistic regression, neural networks, and tree-based models.
Therefore, Option C is the correct and AWS-aligned choice.

NEW QUESTION: 87
A gaming company needs to deploy a natural language processing (NLP) model to moderate a chat forum in a game. The workload experiences heavy usage during evenings and weekends but minimal activity during other hours.
Which solution will meet these requirements MOST cost-effectively?
A. Use an Amazon SageMaker AI batch transform job with fixed capacity.
B. Use Amazon SageMaker Serverless Inference.
C. Use a single Amazon EC2 GPU instance with reserved capacity.
D. Use Amazon SageMaker Asynchronous Inference.
Answer: B (LEAVE A REPLY)
The key requirements in this scenario are variable traffic patterns and cost efficiency. The workload has unpredictable spikes during evenings and weekends, followed by long periods of low or no usage. According to AWS Machine Learning documentation, Amazon SageMaker Serverless Inference is specifically designed for such use cases.
SageMaker Serverless Inference automatically provisions, scales, and shuts down compute resources based on incoming inference requests. Customers are billed only for the compute time used during inference, not for idle resources. This makes it highly cost-effective for workloads with intermittent or spiky traffic, such as real- time chat moderation in gaming environments.
Option A is incorrect because batch transform jobs are intended for offline, large-scale inference and require fixed capacity during job execution. They are not suitable for real-time NLP moderation.
Option C is also incorrect because reserving an EC2 GPU instance incurs continuous costs regardless of utilization. This would be inefficient given

the long idle periods described in the scenario.
Option D, SageMaker Asynchronous Inference, is designed for workloads with long processing times or large payloads and still requires endpoint provisioning. While it can handle traffic spikes, it does not scale down to zero in the same cost-efficient manner as Serverless Inference.
Therefore, Amazon SageMaker Serverless Inference is the most cost-effective and operationally efficient solution for deploying an NLP moderation model with highly variable usage patterns.

NEW QUESTION: 88
A company has significantly increased the amount of data that is stored as .csv files in an Amazon S3 bucket.
Data transformation scripts and queries are now taking much longer than they used to take.
An ML engineer must implement a solution to optimize the data for query performance.
Which solution will meet this requirement with the LEAST operational overhead?
A. Configure an AWS Lambda function to split the .csv files into smaller objects in the S3 bucket.
B. Configure an AWS Glue job to drop columns that have string type values and to save the results to the S3 bucket.
C. Configure an AWS Glue extract, transform, and load (ETL) job to convert the .csv files to Apache Parquet format.
D. Configure an Amazon EMR cluster to process the data that is in the S3 bucket.
Answer: C (LEAVE A REPLY)
AWS documentation strongly recommends using columnar storage formats to optimize analytical query performance on large datasets stored in Amazon S3. Apache Parquet is a columnar, compressed, and splittable file format that significantly improves query speed and reduces I/O compared to row-based formats such as CSV.
By using AWS Glue to convert CSV files into Parquet format, the company can achieve faster query execution with minimal operational overhead. Glue is fully managed, serverless, and integrates natively with S3, Amazon Athena, and Amazon Redshift Spectrum.
Option A does not improve query efficiency; splitting files still leaves the data in an inefficient row-based format. Option B may reduce data size but does not address the fundamental inefficiency of CSV for analytics. Option D introduces significant operational overhead because Amazon EMR requires cluster provisioning, scaling, and maintenance.
Therefore, converting CSV files to Apache Parquet using AWS Glue ETL is the most efficient and low- maintenance solution.

NEW QUESTION: 89
An ML engineer is tuning an image classification model that performs poorly on one of two classes. The poorly performing class represents an extremely small fraction of the training dataset.
Which solution will improve the model's performance?
A. Optimize for accuracy. Use image augmentation on the less common images.
B. Optimize for F1 score. Use image augmentation on the less common images.
C. Optimize for accuracy. Use SMOTE to generate synthetic images.
D. Optimize for F1 score. Use SMOTE to generate synthetic images.
Answer: B (LEAVE A REPLY)
This scenario describes a severely imbalanced classification problem. In such cases, accuracy is a misleading metric, because the model can achieve high accuracy by predicting only the majority class.
AWS ML best practices recommend using F1 score (or precision/recall) when evaluating imbalanced datasets.
The F1 score balances false positives and false negatives, making it ideal for assessing minority-class performance.
For image data, image augmentation (rotations, flips, crops, color jitter) is the preferred technique to increase minority-class representation. SMOTE is designed for tabular data and is not suitable for image pixel data.
Therefore, the correct solution is to optimize for F1 score and apply image augmentation.

Thus, Option B is the correct and AWS-aligned answer.

NEW QUESTION: 90

A recommendation model uses ML and calls an Amazon SageMaker AI endpoint to get recommendations. An ML engineer must ensure that the model stays available during an expected increase in user traffic.
Which solution will meet these requirements?

A. Configure auto scaling on the SageMaker AI endpoint.

B. Create a new SageMaker AI endpoint. Deploy the model to the new endpoint.

C. Use SageMaker Neo to optimize the model for inference.

D. Attach an Auto Scaling group to the SageMaker AI endpoint.

Answer: A (LEAVE A REPLY)

Option A is correct because Amazon SageMaker AI supports automatic scaling for hosted models , and AWS documentation states that auto scaling dynamically adjusts the number of instances provisioned for a model in response to workload changes. When traffic increases, auto scaling brings more instances online; when traffic decreases, it removes unneeded capacity. This directly matches the requirement to keep the recommendation endpoint available during an expected increase in user traffic.

AWS also documents that SageMaker integrates with Application Auto Scaling , and you can scale SageMaker endpoint variants by using target tracking , step scaling , and scheduled scaling policies. That means the service is purpose-built for this scenario: maintaining availability and capacity as inference demand changes over time. Since the model is already being served from a SageMaker endpoint, the most direct and AWS-recommended solution is to configure endpoint auto scaling rather than redesigning the deployment.

The other options do not address the requirement as effectively. Creating a new endpoint does not automatically handle fluctuating traffic and adds unnecessary operational complexity. SageMaker Neo optimizes models for inference performance on supported hardware, but it does not provide elastic capacity management. Attaching an Auto Scaling group to a SageMaker endpoint is incorrect because SageMaker endpoints do not use EC2 Auto Scaling groups directly; scaling is handled through SageMaker's integration with Application Auto Scaling. Therefore, to maintain availability during traffic spikes, the best AWS- documented answer is A: configure auto scaling on the SageMaker AI endpoint .

NEW QUESTION: 91

A logistics company has installed in-vehicle cameras for basic monitoring of its drivers. The company wants to improve driver safety by identifying distractions that could lead to accidents.
Which solution will meet this requirement with the LEAST operational effort?

A. Use Amazon Rekognition eye gaze direction detection to monitor driver behavior and identify distractions.

B. Use Amazon SageMaker AI to customize an AI model to monitor driver behavior and identify distractions.

C. Integrate a third-party driver monitoring system with Amazon Rekognition to monitor driver behavior and identify distractions.

D. Use Amazon Comprehend to analyze text-based driver feedback and identify distractions.

Answer: (SHOW ANSWER)

Option A is correct because Amazon Rekognition provides a built-in EyeDirection attribute that indicates the direction a person's eyes are gazing, expressed through pitch and yaw. AWS documentation describes EyeDirection as showing where the eyes are looking independently of head pose. For a driver-monitoring use case, this is directly useful for identifying whether a driver is looking away from the road and may be distracted.

AWS also announced that Rekognition's eye gaze direction detection is intended to support safety use cases and to help identify where users focus their attention. That makes it a strong match for monitoring in-vehicle cameras for distraction detection. Because Rekognition is a fully managed service with ready-to-use computer vision capabilities, it requires far less operational effort than collecting data, labeling it, training a custom model in SageMaker, and maintaining that model over time.

The other options are less appropriate. Option B could work technically, but building and maintaining a custom SageMaker model would require significantly more engineering and operational work than using a managed Rekognition feature. Option C introduces unnecessary complexity by adding a third-party system when AWS already provides a directly relevant managed capability. Option D is unrelated because Amazon Comprehend analyzes text, while the input here is camera footage. Since the question asks for the solution with the least operational effort , the best AWS-aligned answer is to use the managed Rekognition eye gaze detection capability. Therefore, the fully verified answer is A .

NEW QUESTION: 92
An ML engineer is training an XGBoost regression model in Amazon SageMaker AI. The ML engineer conducts several rounds of hyperparameter tuning with random grid search. After these rounds of tuning, the error rate on the test hold-out dataset is much larger than the error rate on the training dataset.
The ML engineer needs to make changes before running the hyperparameter grid search again.
Which changes will improve the model ' s performance? (Select TWO.)
A. Increase the model complexity by increasing the number of features in the dataset.
B. Decrease the model complexity by reducing the number of features in the dataset.
C. Decrease the model complexity by reducing the number of samples in the dataset.
D. Increase the value of the L2 regularization parameter.
E. Decrease the value of the L2 regularization parameter.
Answer: B,D (LEAVE A REPLY)
The scenario describes a classic overfitting problem: the XGBoost model performs well on the training dataset but poorly on the test hold-out dataset. According to AWS Machine Learning and XGBoost documentation, overfitting occurs when a model is too complex and learns noise and patterns specific to the training data rather than generalizable relationships.
One effective way to address overfitting is to reduce model complexity. Option B, reducing the number of features, simplifies the hypothesis space and lowers the risk of fitting spurious correlations. Feature reduction is a recommended best practice when the model shows a large generalization gap between training and test error.
Another effective method is to increase regularization. Option D, increasing the L2 regularization parameter (lambda in XGBoost), penalizes large weights and discourages overly complex trees. AWS documentation explicitly notes that L2 regularization helps improve generalization by smoothing model parameters and reducing variance.
Option A would worsen overfitting by increasing complexity. Option C is incorrect because reducing the number of training samples generally increases overfitting risk. Option E would decrease regularization strength and further degrade test performance.
Therefore, reducing feature complexity and increasing L2 regularization are the correct changes.

NEW QUESTION: 93
A company must install a custom script on any newly created Amazon SageMaker AI notebook instances.
Which solution will meet this requirement with the LEAST operational overhead?

A. Create a lifecycle configuration script to install the custom script when a new SageMaker AI notebook is created. Attach the lifecycle configuration to every new SageMaker AI notebook as part of the creation steps.

C. Create a custom package index repository. Use AWS CodeArtifact to manage the installation of the custom script. Set up AWS PrivateLink endpoints to connect CodeArtifact to the SageMaker AI instance. Install the script.

B. Create a custom Amazon Elastic Container Registry (Amazon ECR) image that contains the custom script. Push the ECR image to a Docker registry. Attach the Docker image to a SageMaker Studio domain. Select the kernel to run as part of the SageMaker AI notebook.

D. Store the custom script in Amazon S3. Create an AWS Lambda function to install the custom script on new SageMaker AI notebooks. Configure Amazon EventBridge to invoke the Lambda function when a new SageMaker AI notebook is initialized.

Answer: A (LEAVE A REPLY)

AWS recommends lifecycle configuration scripts as the simplest and most direct way to customize Amazon SageMaker Notebook Instances at creation time. Lifecycle configurations run automatically when a notebook instance is created or started, allowing scripts, packages, and system dependencies to be installed without manual intervention.

This approach is fully supported, requires no additional infrastructure, and integrates directly with the notebook creation workflow. The script can be reused across notebooks, ensuring consistency.

Options B, C, and D introduce unnecessary complexity, such as container management, private package repositories, or event-driven orchestration.

Therefore, lifecycle configuration scripts provide the least operational overhead solution.

NEW QUESTION: 94

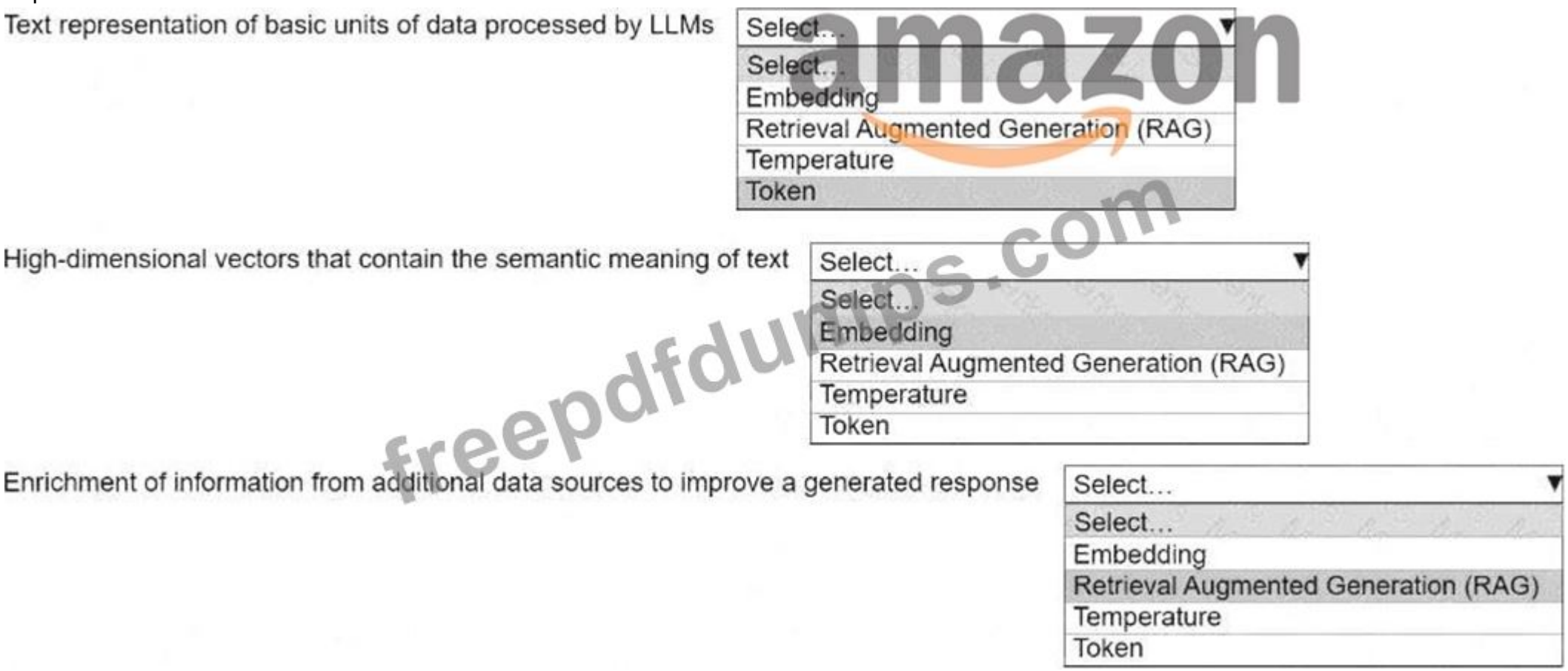
An ML engineer is building a generative AI application on Amazon Bedrock by using large language models (LLMs).

Select the correct generative AI term from the following list for each description. Each term should be selected one time or not at all. (Select three.)

* Embedding
* Retrieval Augmented Generation (RAG)
* Temperature
* Token

Answer:

Explanation:



Text representation of basic units of data processed by LLMs: Token

High-dimensional vectors that contain the semantic meaning of text: Embedding Enrichment of information from additional data sources to improve a generated response: Retrieval Augmented Generation (RAG) Comprehensive Detailed Explanation Token:

Description: A token represents the smallest unit of text (e.g., a word or part of a word) that an LLM processes. For example, " running " might be split into two tokens: " run " and " ing. " Why? Tokens are the fundamental building blocks for LLM input and output processing, ensuring that the model can understand and generate text efficiently.

Embedding:

Description: High-dimensional vectors that encode the semantic meaning of text. These vectors are representations of words, sentences, or even paragraphs in a way that reflects their relationships and meaning.

Why? Embeddings are essential for enabling similarity search, clustering, or any task requiring semantic understanding. They allow the model to " understand " text contextually.

Retrieval Augmented Generation (RAG):
Description: A technique where information is enriched or retrieved from external data sources (e.g., knowledge bases or document stores) to improve the accuracy and relevance of a model ' s generated responses.
Why? RAG enhances the generative capabilities of LLMs by grounding their responses in factual and up-to- date information, reducing hallucinations in generated text.
By matching these terms to their respective descriptions, the ML engineer can effectively leverage these concepts to build robust and contextually aware generative AI applications on Amazon Bedrock.

NEW QUESTION: 95
A company needs to ingest data from data sources into Amazon SageMaker Data Wrangler. The data sources are Amazon S3, Amazon Redshift, and Snowflake. The ingested data must always be up to date with the latest changes in the source systems.
Which solution will meet these requirements?
A. Use direct connections to import data from the data sources into Data Wrangler.
B. Use cataloged connections to import data from the data sources into Data Wrangler.
C. Use AWS Glue to extract data from the data sources. Use AWS Glue also to import the data directly into Data Wrangler.
D. Use AWS Lambda to extract data from the data sources. Use Lambda also to import the data directly into Data Wrangler.
Answer: (SHOW ANSWER)
Amazon SageMaker Data Wrangler supports both direct and cataloged connections. To ensure data is always up to date, AWS recommends using cataloged connections backed by AWS Glue Data Catalog.
Cataloged connections allow Data Wrangler to reference the source systems dynamically, ensuring that each import reflects the latest data changes without manual reconfiguration. This approach supports Amazon S3, Amazon Redshift, and Snowflake and integrates securely using managed credentials.
Direct connections are point-in-time imports and do not automatically reflect schema or data updates. Glue and Lambda-based solutions introduce unnecessary complexity and operational overhead.
Therefore, using cataloged connections in Data Wrangler is the correct solution.

NEW QUESTION: 96
Case Study
A company is building a web-based AI application by using Amazon SageMaker. The application will provide the following capabilities and features: ML experimentation, training, a central model registry, model deployment, and model monitoring.
The application must ensure secure and isolated use of training data during the ML lifecycle. The training data is stored in Amazon S3.
The company needs to run an on-demand workflow to monitor bias drift for models that are deployed to real- time endpoints from the application.
Which action will meet this requirement?
A. Configure the application to invoke an AWS Lambda function that runs a SageMaker Clarify job.
B. Invoke an AWS Lambda function to pull the sagemaker-model-monitor-analyzer built-in SageMaker image.
C. Use AWS Glue Data Quality to monitor bias.
D. Use SageMaker notebooks to compare the bias.
Answer: A (LEAVE A REPLY)
Monitoring bias drift in deployed machine learning models is crucial to ensure fairness and accuracy over time. Amazon SageMaker Clarify provides tools to detect bias in ML models, both during training and after deployment. To monitor bias drift for models deployed to real-time endpoints, an effective approach involves orchestrating SageMaker Clarify jobs using AWS Lambda functions.

Implementation Steps:
* Set Up Data Capture:
* Enable data capture on the SageMaker endpoint to record input data and model predictions. This captured data serves as the basis for bias analysis.
* Develop a Lambda Function:
* Create an AWS Lambda function configured to initiate a SageMaker Clarify job. This function will process the captured data to assess bias metrics.
* Schedule or Trigger the Lambda Function:
* Configure the Lambda function to run on-demand or at scheduled intervals using Amazon CloudWatch Events or EventBridge. This setup allows for regular bias monitoring as per the application's requirements.
* Analyze and Respond to Results:
* After each Clarify job completes, review the generated bias reports. If bias drift is detected, take appropriate actions, such as retraining the model or adjusting data preprocessing steps.
Advantages of This Approach:
* Automation:Utilizing AWS Lambda for orchestrating Clarify jobs enables automated and scalable bias monitoring without manual intervention.
* Cost-Effectiveness:AWS Lambda's serverless nature ensures that you only pay for the compute time consumed during the execution of the function, optimizing resource usage.
* Flexibility:The solution can be tailored to specific monitoring needs, allowing for adjustments in monitoring frequency and analysis parameters.
By implementing this solution, the company can effectively monitor bias drift in real-time, ensuring that the AI application maintains fairness and accuracy throughout its lifecycle.
References:
* Bias drift for models in production - Amazon SageMaker
* Schedule Bias Drift Monitoring Jobs - Amazon SageMaker

NEW QUESTION: 97
An ML engineer is using an Amazon SageMaker AI shadow test to evaluate a new model that is hosted on a SageMaker AI endpoint. The shadow test requires significant GPU resources for high performance. The production variant currently runs on a less powerful instance type.
The ML engineer needs to configure the shadow test to use a higher performance instance type for a shadow variant. The solution must not affect the instance type of the production variant.
Which solution will meet these requirements?
A. Modify the existing ProductionVariant configuration in the endpoint to include a ShadowProductionVariants list. Specify the larger instance type for the shadow variant.
B. Create a new endpoint configuration with two ProductionVariant definitions. Configure one definition for the existing production variant and one definition for the shadow variant with the larger instance type. Use the UpdateEndpoint action to apply the new configuration.
C. Create a separate SageMaker AI endpoint for the shadow variant that uses the larger instance type.
Create an AWS Lambda function that routes a portion of the traffic to the shadow endpoint. Assign the Lambda function to the original endpoint.
D. Use the CreateEndpointConfig action to define a new configuration. Specify the existing production variant in the configuration and add a separate ShadowProductionVariants list. Specify the larger instance type for the shadow variant. Use the CreateEndpoint action and pass the new configuration to the endpoint.
Answer: D (LEAVE A REPLY)
Amazon SageMaker AI shadow testing enables ML engineers to evaluate new model versions by sending a copy of live production traffic to a

shadow variant without affecting production inference responses. AWS documentation specifies that shadow variants are configured separately from production variants and can use different instance types, including higher-performance GPU instances.
The correct approach is to create a new endpoint configuration using the CreateEndpointConfig API. This configuration includes the existing production variant and a separate ShadowProductionVariants list. The shadow variant can be assigned a larger instance type to meet GPU performance requirements while leaving the production variant unchanged. After creating the configuration, the engineer deploys it using the CreateEndpoint action.
Option A is incorrect because production variant configurations cannot be directly modified to include shadow variants. Option B is incorrect because shadow variants are not defined as standard production variants; defining two production variants would route traffic differently and could affect production behavior. Option C introduces unnecessary complexity and deviates from SageMaker's built-in shadow testing functionality.
AWS explicitly documents that shadow variants are designed to isolate testing resources, support different instance types, and ensure zero impact on production inference. Therefore, Option D is the correct and AWS- recommended solution.

NEW QUESTION: 98
A company needs to run a batch data-processing job on Amazon EC2 instances. The job will run during the weekend and will take 90 minutes to finish running. The processing can handle interruptions. The company will run the job every weekend for the next 6 months.
Which EC2 instance purchasing option will meet these requirements MOST cost-effectively?
A. Spot Instances
B. Reserved Instances
C. On-Demand Instances
D. Dedicated Instances
Answer: A (LEAVE A REPLY)
Scenario:The company needs to run a batch job for 90 minutes every weekend over the next 6 months. The processing can handle interruptions, and cost-effectiveness is a priority.
Why Spot Instances?
* Cost-Effective:Spot Instances provide up to 90% savings compared to On-Demand Instances, making them the most cost-effective option for batch processing.
* Interruption Tolerance:Since the processing can tolerate interruptions, Spot Instances are suitable for this workload.
* Batch-Friendly:Spot Instances can be requested for specific durations or automatically re-requested in case of interruptions.
Steps to Implement:
* Create a Spot Instance Request:
* Use the EC2 console or CLI to request Spot Instances with desired instance type and duration.
* Use Auto Scaling:Configure Spot Instances with an Auto Scaling group to handle instance interruptions and ensure job completion.
* Run the Batch Job:Use tools like AWS Batch or custom scripts to manage the processing.
Comparison with Other Options:
* Reserved Instances:Suitable for predictable, continuous workloads, but less cost-effective for a job that runs only once a week.
* On-Demand Instances:More expensive and unnecessary given the tolerance for interruptions.
* Dedicated Instances:Best for isolation and compliance but significantly more costly.
References:
* Amazon EC2 Spot Instances
* Best Practices for Using Spot Instances
* AWS Batch for Spot Instances

NEW QUESTION: 99
An ML engineer needs to use an ML model to predict the price of apartments in a specific location.
Which metric should the ML engineer use to evaluate the model ' s performance?
A. Accuracy
B. Area Under the ROC Curve (AUC)
C. F1 score
D. Mean absolute error (MAE)
Answer: (SHOW ANSWER)
When predicting continuous variables, such as apartment prices, it ' s essential to evaluate the model ' s performance using appropriate regression metrics. The Mean Absolute Error (MAE) is a widely used metric for this purpose.
Understanding Mean Absolute Error (MAE):
MAE measures the average magnitude of errors in a set of predictions, without considering their direction. It calculates the average absolute difference between predicted values and actual values, providing a straightforward interpretation of prediction accuracy.

Formula: $MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$ Where:

- n = Number of data points
- y_i = Actual value
- $\hat{y}_i$ = Predicted value

Advantages of MAE:
Interpretability: MAE is expressed in the same units as the target variable, making it easy to understand.
Robustness to Outliers: Unlike metrics that square the errors (e.g., Mean Squared Error), MAE does not disproportionately penalize larger errors, making it more robust to outliers.
Comparison with Other Metrics:
Accuracy, AUC, F1 Score: These metrics are designed for classification tasks, where the goal is to predict discrete labels. They are not suitable for regression problems involving continuous target variables.
Mean Squared Error (MSE): While MSE also measures prediction errors, it squares the differences, giving more weight to larger errors. This can be useful in certain contexts but may be sensitive to outliers.
Conclusion:
For evaluating the performance of a model predicting apartment prices-a continuous variable-MAE is an appropriate and effective metric. It provides a clear indication of the average prediction error in the same units as the target variable, facilitating straightforward interpretation and comparison.
References:
Regression Metrics - GeeksforGeeks
Evaluation Metrics for Your Regression Model - Analytics Vidhya
Regression Metrics for Machine Learning - Machine Learning Mastery

NEW QUESTION: 100
An ML engineer at a credit card company built and deployed an ML model by using Amazon SageMaker AI.
The model was trained on transaction data that contained very few fraudulent transactions. After deployment, the model is underperforming.
What should the ML engineer do to improve the model's performance?
A. Retrain the model with a different SageMaker built-in algorithm.
B. Use random undersampling to reduce the majority class and retrain the model.

C. Use Synthetic Minority Oversampling Technique (SMOTE) to generate synthetic minority samples and retrain the model.
D. Use random oversampling to duplicate minority samples and retrain the model.
Answer: (SHOW ANSWER)
This is a classic class imbalance problem, where fraudulent transactions (minority class) are severely underrepresented. AWS documentation for SageMaker Data Wrangler recommends SMOTE (Synthetic Minority Oversampling Technique) as an effective approach for improving model performance in such scenarios.
SMOTE generates synthetic minority samples by interpolating between existing minority class examples.
This improves the model's ability to learn decision boundaries without simply duplicating data, which can cause overfitting.
Random undersampling removes valuable majority class data, reducing overall model robustness. Random oversampling duplicates data and increases overfitting risk. Changing algorithms does not address the root cause.
AWS best practices highlight SMOTE as the preferred technique for fraud detection and other highly imbalanced datasets.
Therefore, Option C is the correct and AWS-verified answer.

NEW QUESTION: 101
An ML engineer needs to use an ML model to predict the price of apartments in a specific location.
Which metric should the ML engineer use to evaluate the model's performance?
A. Accuracy
B. Area Under the ROC Curve (AUC)
C. F1 score
D. Mean absolute error (MAE)
Answer: D (LEAVE A REPLY)
Predicting apartment prices is a regression problem, where the target variable is continuous. AWS documentation states that classification metrics such as accuracy, AUC, and F1 score are not appropriate for regression tasks.
Mean Absolute Error (MAE) measures the average absolute difference between predicted values and actual values. MAE is easy to interpret because it is expressed in the same units as the target variable (for example, dollars), making it especially useful for business-facing problems like price prediction.
AWS best practices recommend MAE for evaluating regression models when understanding average prediction error magnitude is important and when robustness to outliers is desired.
Therefore, Option D is the correct and AWS-aligned answer.

NEW QUESTION: 102
A company needs to combine data from multiple sources. The company must use Amazon Redshift Serverless to query an AWS Glue Data Catalog database and underlying data that is stored in an Amazon S3 bucket.
Select and order the correct steps from the following list to meet these requirements. Select each step one time or not at all. (Select and order three.)
* Attach the IAM role to the Redshift cluster.
* Attach the IAM role to the Redshift namespace.
* Create an external database in Amazon Redshift to point to the Data Catalog schema.
* Create an external schema in Amazon Redshift to point to the Data Catalog database.
* Create an IAM role for Amazon Redshift to use to access only the S3 bucket that contains underlying data.
* Create an IAM role for Amazon Redshift to use to access the Data Catalog and the S3 bucket that contains underlying data.

Step 1:

- Attach the IAM role to the Redshift cluster.
- Attach the IAM role to the Redshift namespace.
- Create an external database in Amazon Redshift to point to the Data Catalog schema.
- Create an external schema in Amazon Redshift to point to the Data Catalog database.
- Create an IAM role for Amazon Redshift to use to access only the S3 bucket that contains underlying data.
- Create an IAM role for Amazon Redshift to use to access the Data Catalog and the S3 bucket that contains underlying data.

Step 2:

- Attach the IAM role to the Redshift cluster.
- Attach the IAM role to the Redshift namespace.
- Create an external database in Amazon Redshift to point to the Data Catalog schema.
- Create an external schema in Amazon Redshift to point to the Data Catalog database.
- Create an IAM role for Amazon Redshift to use to access only the S3 bucket that contains underlying data.
- Create an IAM role for Amazon Redshift to use to access the Data Catalog and the S3 bucket that contains underlying data.

Step 3:

- Attach the IAM role to the Redshift cluster.
- Attach the IAM role to the Redshift namespace.
- Create an external database in Amazon Redshift to point to the Data Catalog schema.
- Create an external schema in Amazon Redshift to point to the Data Catalog database.
- Create an IAM role for Amazon Redshift to use to access only the S3 bucket that contains underlying data.
- Create an IAM role for Amazon Redshift to use to access the Data Catalog and the S3 bucket that contains underlying data.

Answer:

Step 1:

Attach the IAM role to the Redshift cluster.
Attach the IAM role to the Redshift namespace.
Create an external database in Amazon Redshift to point to the Data Catalog schema.
Create an external schema in Amazon Redshift to point to the Data Catalog database.
Create an IAM role for Amazon Redshift to use to access only the S3 bucket that contains underlying data.
Create an IAM role for Amazon Redshift to use to access the Data Catalog and the S3 bucket that contains underlying data.

Step 2:

Attach the IAM role to the Redshift cluster.
Attach the IAM role to the Redshift namespace.
Create an external database in Amazon Redshift to point to the Data Catalog schema.
Create an external schema in Amazon Redshift to point to the Data Catalog database.
Create an IAM role for Amazon Redshift to use to access only the S3 bucket that contains underlying data.
Create an IAM role for Amazon Redshift to use to access the Data Catalog and the S3 bucket that contains underlying data.

Step 3:

Attach the IAM role to the Redshift cluster.
Attach the IAM role to the Redshift namespace.
Create an external database in Amazon Redshift to point to the Data Catalog schema.
Create an external schema in Amazon Redshift to point to the Data Catalog database.
Create an IAM role for Amazon Redshift to use to access only the S3 bucket that contains underlying data.
Create an IAM role for Amazon Redshift to use to access the Data Catalog and the S3 bucket that contains underlying data.

Explanation:
Step 1
Create an IAM role for Amazon Redshift to use to access the Data Catalog and the S3 bucket that contains underlying data.
This role must include:
Permissions for AWS Glue Data Catalog (e.g., glue:GetDatabase, glue:GetTables) Permissions for the Amazon S3 bucket that stores the underlying data Step 2 Attach the IAM role to the Redshift namespace.
Redshift Serverless uses a namespace, not a cluster, so the role must be associated with the namespace to allow Redshift to assume it when querying external data.
Step 3
Create an external schema in Amazon Redshift to point to the Data Catalog database.

The external schema maps Redshift to the Glue Data Catalog database so Redshift can query the tables stored in S3.

NEW QUESTION: 103
A music streaming company constantly streams song ratings from an application to an Amazon S3 bucket.
The company wants to use the ratings as an input for training and inference of an Amazon SageMaker AI model.
The company has an AWS Glue Data Catalog that is configured with the S3 bucket as the source. An ML engineer needs to implement a solution to create a repository for this data. The solution must ensure that the data stays synchronized during batch training and real-time inference.
Which solution will meet these requirements?
A. Use the Generate Data Insights function in SageMaker Data Wrangler.
B. Ingest data into SageMaker Feature Store from the S3 bucket. Apply tags and indexes.
C. Use AWS Lake Formation. Apply tag-based control on the data.
D. Use Amazon Athena. Create tables by using CREATE TABLE AS SELECT (CTAS) queries to group data.
Answer: (SHOW ANSWER)

NEW QUESTION: 104
An ML engineer is collecting data to train a classification ML model by using Amazon SageMaker AI. The target column can have two possible values: Class A or Class B. The ML engineer wants to ensure that the number of samples for both Class A and Class B are balanced, without losing any existing training data. The ML engineer must test the balance of the training data.
Which solution will meet this requirement?
A. Use SageMaker Clarify to check for class imbalance (CI). If the value is equal to 0, then use random undersampling in SageMaker Data Wrangler to balance the classes.
B. Use SageMaker Clarify to check for class imbalance (CI). If the value is greater than 0, then use synthetic minority oversampling technique (SMOTE) in SageMaker Data Wrangler to balance the classes.
C. Use SageMaker JumpStart to generate a class imbalance (CI) report. If the value is greater than 0, then use random undersampling in SageMaker Studio to balance the classes.
D. Use SageMaker JumpStart to generate a class imbalance (CI) report. If the value is equal to 0, then use synthetic minority oversampling technique (SMOTE) in SageMaker Studio to balance the classes.
Answer: B (LEAVE A REPLY)
The requirement has two key constraints: detect class imbalance and balance classes without losing any existing data. AWS provides Amazon SageMaker Clarify as the native tool to detect pre-training bias, including class imbalance (CI). CI measures differences in label distributions between classes, and a CI value greater than 0 indicates imbalance.
Once imbalance is detected, the engineer must rebalance the dataset without discarding data. Random undersampling would remove samples from the majority class, violating the requirement. Instead, oversampling is required. SMOTE (Synthetic Minority Oversampling Technique) creates synthetic samples for the minority class, preserving all original data while improving class balance.
Amazon SageMaker Data Wrangler natively supports SMOTE, making it the correct AWS-managed tool for this preprocessing task.
Options C and D are incorrect because SageMaker JumpStart is used for pretrained models and solutions, not for bias detection reporting. Option A is incorrect because it uses undersampling and misinterprets CI = 0 (which actually indicates no imbalance).
Therefore, detecting imbalance with SageMaker Clarify and correcting it using SMOTE in Data Wrangler is the correct solution.

NEW QUESTION: 105
A company has an ML model in Amazon SageMaker AI. An ML engineer needs to implement a monitoring solution to automatically detect changes

in the input data distribution of model features.
Which solution will meet this requirement with the LEAST operational overhead?
A. Configure SageMaker Model Monitor. Establish a data quality baseline. Ensure that the emit_metrics option is enabled in the baseline constraints file. Configure an Amazon CloudWatch alarm to notify the company about changes in specific metrics that are related to data quality.
B. Configure SageMaker Model Monitor. Establish a model quality baseline. Ensure that the comparison_method option is set to Robust in the baseline constraints file. Configure an Amazon CloudWatch alarm to notify the company about changes in model quality metrics.
C. Use SageMaker Debugger with custom rules to track shifts in feature distributions. Configure Amazon CloudWatch alarms to notify the company when the rules detect significant changes.
D. Use Amazon CloudWatch to directly observe the SageMaker AI endpoint ' s performance metrics.Manually analyze the CloudWatch logs for indicators of data drift or shifts in feature distribution.
Answer: (SHOW ANSWER)
Option A is correct because the requirement is to detect changes in the input data distribution of model features , which is a data quality / data drift monitoring problem. AWS documentation states that Amazon SageMaker Model Monitor uses rules to detect data drift and alerts you when it happens. The documented workflow is to enable data capture, create a baseline from training data, and then run monitoring jobs that compare incoming inference data against that baseline. That directly matches the need to automatically detect changes in feature distributions.
AWS also documents that Model Monitor can emit metrics to Amazon CloudWatch , and those metrics can be used with CloudWatch alarms to notify teams when data quality drifts beyond acceptable thresholds.
That makes Option A the lowest-operational-overhead solution because it uses SageMaker's built-in monitoring capability plus managed alerting, rather than requiring custom drift logic. The inclusion of emit_metrics and CloudWatch alarming is consistent with the SageMaker monitoring pattern for automated notification.
The other options are weaker. Option B is for model quality monitoring, which focuses on prediction performance against ground truth, not shifts in the input feature distribution. Option C uses SageMaker Debugger, which is aimed at training-time debugging and custom rule analysis rather than managed production data drift monitoring. Option D relies on manual log analysis and endpoint performance metrics, which does not directly solve feature-distribution drift detection and adds more operational effort. Therefore, the best AWS-documented answer is A .

NEW QUESTION: 106
An ML engineer wants to run a training job on Amazon SageMaker AI. The training job will train a neural network by using multiple GPUs. The training dataset is stored in Parquet format.
The ML engineer discovered that the Parquet dataset contains files too large to fit into the memory of the SageMaker AI training instances.
Which solution will fix the memory problem?
A. Attach an Amazon Elastic Block Store (Amazon EBS) Provisioned IOPS SSD volume to the instance.
Store the files in the EBS volume.
B. Repartition the Parquet files by using Apache Spark on Amazon EMR. Use the repartitioned files for the training job.
C. Change the instance type to Memory Optimized instances with sufficient memory for the training job.
D. Use the SageMaker AI distributed data parallelism (SMDDP) library with multiple instances to split the memory usage.
Answer: B (LEAVE A REPLY)
The issue is caused by oversized Parquet files that cannot be efficiently read into memory during training. The most effective and scalable solution is to repartition the dataset into smaller Parquet files.
AWS best practices for large-scale ML training recommend optimizing data layout, not simply increasing memory. By using Apache Spark on Amazon EMR, the ML engineer can repartition the Parquet files into smaller chunks that can be streamed and processed efficiently by SageMaker training jobs.
Attaching EBS volumes (Option A) increases storage capacity but does not solve in-memory constraints.

Changing to memory-optimized instances (Option C) increases cost and does not address long-term scalability. SMDDP (Option D) distributes gradients and computation, not dataset file sizes.
Therefore, repartitioning the Parquet files is the correct solution.

Valid MLA-C01 Dumps shared by Actual4test.com for Helping Passing MLA-C01 Exam! Actual4test.com now offer the **newest MLA-C01 exam dumps**, the Actual4test.com MLA-C01 exam **questions have been updated** and **answers have been corrected** get the **newest** Actual4test.com MLA-C01 dumps with Test Engine here: https://www.actual4test.com/MLA-C01_examcollection.html (**243** Q&As Dumps, **30%OFF Special Discount: Freepdfdumps**)