EXAM PROFESSIONAL MACHINE LEARNING ENGINEER TOPIC 1 QUESTION 268 DISCUSSI...

Actual exam question from Google's Professional Machine Learning Engineer
Question #: 268
Topic #: 1
[All Professional Machine Learning Engineer Questions]

You want to migrate a scikit-learn classifier model to TensorFlow. You plan to train the TensorFlow classifier model using the same training set that was used to train the scikit-learn model, and then compare the performances using a common test set. You want to use the Vertex AI Python SDK to manually log the evaluation metrics of each model and compare them based on their F1 scores and confusion matrices. How should you log the metrics?

A. Use the aiplatform.log_classification_metrics function to log the F1 score, and use the aiplatform.log_metrics function to log the confusion matrix.

B. Use the aiplatform.log_classification_metrics function to log the F1 score and the confusion matrix.

C. Use the aiplatform.log_metrics function to log the F1 score and the confusion matrix.

D. Use the aiplatform.log_metrics function to log the F1 score: and use the aiplatform.log_classification_metrics function to log the confusion matrix.

Show Suggested Answer

by pikachu007 at *Jan. 13, 2024, 4:17 p.m.*

Comments

Type your comment...

Submit

YangG 1 week, 2 days ago

Selected Answer: D

d

upvoted 1 times

bobjr 4 months, 2 weeks ago

Selected Answer: D

https://cloud.google.com/vertex-ai/docs/experiments/log-data#classification-metrics

log_classification_metrics -> only the confusion matrix, not the F1scores
log_metrics -> any number you want -> you can use it to store a F1 scores

upvoted 3 times

fitri001 5 months, 4 weeks ago

Selected Answer: B

aiplatform.log_classification_metrics is specifically designed for logging classification metrics, which includes F1 score and confusion matrix.
aiplatform.log_metrics is a more generic function for logging any kind of metric, but it wouldn't capture the rich structure of a confusion matrix.
Therefore, using aiplatform.log_classification_metrics allows you to log both F1 score and confusion matrix in a single call, simplifying your code and ensuring proper handling of these classification-specific metrics.

upvoted 3 times

pinimichele01 5 months, 3 weeks ago

https://cloud.google.com/python/docs/reference/aiplatform/latest/google.cloud.aiplatform#google_cloud_aiplatform_log_classification_metrics

upvoted 1 times

fitri001 5 months, 4 weeks ago

While aiplatform.log_metrics can handle numeric values like F1 score, it wouldn't capture the complexity of a confusion matrix. Confusion matrix is a two-dimensional table and requires specific handling for proper logging.expand_more
aiplatform.log_classification_metrics is designed for classification tasks and understands the structure of both F1 score and confusion matrix, allowing them to be logged efficiently in a single function call.

upvoted 1 times

fitri001 5 months, 4 weeks ago

Therefore, using separate functions like log_metrics for F1 score and log_classification_metrics for confusion matrix would be inefficient and might not capture the matrix structure accurately.

upvoted 1 times

tardigradum 2 months, 1 week ago

Hi fitri001. You are usually right but, I this particular case, I think D is the right answer.

As you can see here in the link I provide you below, it "Currently support confusion matrix and ROC curve."

Link:

https://cloud.google.com/python/docs/reference/aiplatform/latest/google.cloud.aiplatform#google_cloud_aiplatform_log_classification_metrics

upvoted 1 times

gscharly 6 months ago

Selected Answer: D

According to docs, log_classification_metrics supports confusion matrix and ROC curve. Not sure if it means that it only supports those... Assuming those are the only ones supported, I would got with D

upvoted 2 times

gscharly 6 months ago

forgot to add the link:
https://cloud.google.com/python/docs/reference/aiplatform/latest/google.cloud.aiplatform#google_cloud_aiplatform_log_classification_metrics

upvoted 2 times

omermahgoub 6 months, 1 week ago

Selected Answer: B

aiplatform.log_classification_metrics to log metrics relevant to classification tasks, including F1 score and confusion matrix

aiplatform.log_classification_metrics to log metrics relevant to classification tasks, including F1 score and confusion matrix.

upvoted 1 times

pinimichele01 5 months, 3 weeks ago

link??

i find only:
https://cloud.google.com/python/docs/reference/aiplatform/latest/google.cloud.aiplatform#google_cloud_aiplatform_log_classification_metrics

so D NOT B

upvoted 1 times

Yan_X 7 months, 2 weeks ago

Selected Answer: B

The aiplatform.log_classification_metrics function is designed to log classification metrics, including the F1 score and the confusion matrix. It takes the following arguments:

predictions: The predicted labels.
labels: The true labels.
weight: The weight of each sample.
logger: The logger to use.

The aiplatform.log_metrics function is designed to log general metrics, such as accuracy, loss, and precision. It takes the following arguments:

metric: The metric to log.
value: The value of the metric.
step: The step at which the metric was logged.
logger: The logger to use.

upvoted 1 times

daidai75 9 months ago

Selected Answer: B

Actually, the F1 score is calculated by the Precision and recall metrics. The the log_classification_metrics is OK for both confusion matrix and F1 score

upvoted 2 times

b1a8fae 9 months ago

Selected Answer: D

I go with D.
log_classification_metrics currently support confusion matrix and ROC curve.
https://cloud.google.com/python/docs/reference/aiplatform/latest/google.cloud.aiplatform#google_cloud_aiplatform_log_classification_metrics
Because it is not explicitly mentioned in the docs of log_classification_metrics, I assume F1 Score must be logged with log_metrics.
https://cloud.google.com/python/docs/reference/aiplatform/latest/google.cloud.aiplatform#google_cloud_aiplatform_log_metrics
(if accuracy and recall are logged in the example, probably F1 is done the same way)

upvoted 4 times

pikachu007 9 months, 1 week ago

Selected Answer: B

Option A: It's incorrect because aiplatform.log_metrics is a more general function that doesn't provide the same specialized structure for classification metrics.
Option C: While technically possible to log both metrics using aiplatform.log_metrics, it's less optimal as it requires manual formatting and might not be as easily interpreted by Vertex AI's visualization tools.
Option D: This is incorrect as it suggests using aiplatform.log_classification_metrics for the confusion matrix, but that function doesn't support logging confusion matrices directly.

upvoted 1 times

b1a8fae 9 months ago

Option B also suggests sing aiplatform.log_classification_metrics for the confusion matrix. Which is supported, btw.
https://cloud.google.com/python/docs/reference/aiplatform/latest/google.cloud.aiplatform#google_cloud_aiplatform_log_classification_metrics

upvoted 4 times

Start Learning for free

Social Media

Facebook , Twitter
YouTube , Reddit
Pinterest

We are the biggest and most updated IT certification exam material website.
Using our own resources, we strive to strengthen the IT professionals community for free.