EXAMTOPICS

- Expert Verified, Online, **Free.**

☰ MENU   🔍

← **Google Discussions**

Exam Professional Machine Learning Engineer All Questions

View all questions & answers for the Professional Machine Learning Engineer exam

Go to Exam

📄 **EXAM PROFESSIONAL MACHINE LEARNING ENGINEER TOPIC 1 QUESTION 80 DISCUSSIO..**

Actual exam question from Google's Professional Machine Learning Engineer

Question #: 80

Topic #: 1

[All Professional Machine Learning Engineer Questions]

You work for an online travel agency that also sells advertising placements on its website to other companies. You have been asked to predict the most relevant web banner that a user should see next. Security is important to your company. The model latency requirements are 300ms@p99, the inventory is thousands of web banners, and your exploratory analysis has shown that navigation context is a good predictor. You want to Implement the simplest solution. How should you configure the prediction pipeline?

A. Embed the client on the website, and then deploy the model on AI Platform Prediction.

B. Embed the client on the website, deploy the gateway on App Engine, deploy the database on Firestore for writing and for reading the user's navigation context, and then deploy the model on AI Platform Prediction.

C. Embed the client on the website, deploy the gateway on App Engine, deploy the database on Cloud Bigtable for writing and for reading the user's navigation context, and then deploy the model on AI Platform Prediction.

D. Embed the client on the website, deploy the gateway on App Engine, deploy the database on Memorystore for writing and for reading the user's navigation context, and then deploy the model on Google Kubernetes Engine.

Show Suggested Answer

by 👤 ares81 at *Dec. 14, 2022, 10:24 a.m.*

Comments

Type your comment...

Submit

☐ 👤 **hiromi** `Highly Voted 👍` 1 year, 10 months ago

`Selected Answer: C`

C (same question 49)
keywords
the inventory is thousands of web banners -> Bigtable
You want to Implement the simplest solution -> AI Platform Prediction

👍 ↩ 🚩 upvoted 8 times

    ☐ 👤 **tavva_prudhvi** 1 year, 2 months ago

    Yes, but in that question Option B doesnt have a database.Firestore can handle thousands of web banners, right?

    👍 ↩ 🚩 upvoted 1 times

☐ 👤 **e707** `Highly Voted 👍` 1 year, 5 months ago

`Selected Answer: B`

Here are some of the reasons why C is not as simple as B:

Cloud Bigtable is a more complex database to set up and manage than Firestore.
Cloud Bigtable is not as secure as Firestore.
Cloud Bigtable is not as well-integrated with other Google Cloud services as Firestore.
Therefore, B is the simpler solution that meets all of the requirements.

👍 ↩ 🚩 upvoted 6 times

☐ 👤 **192malba192** `Most Recent ⊘` 2 months, 2 weeks ago

go for B

👍 ↩ 🚩 upvoted 1 times

☐ 👤 **pinimichele01** 6 months, 2 weeks ago

`Selected Answer: B`

see e707

👍 ↩ 🚩 upvoted 1 times

☐ 👤 **ludovikush** 6 months, 2 weeks ago

`Selected Answer: C`

as Hiromi said

👍 ↩ 🚩 upvoted 1 times

☐ 👤 **ludovikush** 7 months, 4 weeks ago

`Selected Answer: B`

I would opt for B as we have requirement of retrieval latency

👍 ↩ 🚩 upvoted 1 times

☐ 👤 **Mickey321** 11 months, 1 week ago

`Selected Answer: B`

Embed the client on the website, deploy the gateway on App Engine, and then deploy the model on AI Platform Prediction.

👍 ↩ 🚩 upvoted 1 times

☐ 👤 **Krish6488** 11 months, 2 weeks ago

`Selected Answer: B`

I would go with Firestore as throughput or latency requirement provided in the question are possible with Firestore and bigTable may be an overkill. Had the scenario involved super large volumes of data, CBT would have taken precedence

👍 ↩ 🚩 upvoted 1 times

☐ 👤 **andresvelasco** 1 year, 1 month ago

`Selected Answer: B`

I think B, based on "the simplest solution" consideration.

👍 ↩ 🚩 upvoted 1 times

☐ 👤 **tavva_prudhvi** 1 year, 2 months ago

`Selected Answer: B`

the primary requirement mentioned in the original question is to implement the simplest solution. Firestore is a fully managed, serverless NoSQL database that can also handle thousands of web banners and dynamically changing user browsing history. It is designed for real-time data synchronization and can quickly update the most relevant web banner as the user

browses different pages of the website.

While Cloud Bigtable offers high performance and scalability, it is more complex to manage and is better suited for large-scale, high-throughput workloads. Firestore, on the other hand, is easier to implement and maintain, making it a more suitable choice for the simplest solution in this scenario.

👍 ↩ 🚩 upvoted 2 times

☐ 👤 **[Removed]** 1 year, 3 months ago

Selected Answer: C

The answer is C for the following reason:

If you need:
- Submillisecond retrieval latency on a limited amount of quickly changing data, retrieved by a few thousand clients, use Memorystore.
- Millisecond retrieval latency on slowly changing data where storage scales automatically, use Datastore.
- Millisecond retrieval latency on dynamically changing data, using a store that can scale linearly with heavy reads and writes, use Bigtable.
Source: https://cloud.google.com/architecture/minimizing-predictive-serving-latency-in-machine-learning#choosing_a_nosql_database

C is better than B because 1) the inventory is thousands of web banners and 2) we expect the user to compare many travel destinations, dates, hotels, and tariffs during their search process. It means the user's browsing history is dynamically changing, and we need to identify "the most relevant web banner that a user should see next" => we will be dynamically changing the ad as the user browses different pages of the website.

👍 ↩ 🚩 upvoted 2 times

   ☐ 👤 **andresvelasco** 1 year, 1 month ago

   BTW, the storage solution does not mention web banners, just browsing history.
   but what about the "simplest solution" consideration? that wold point into the Datastore direction.
   It is true however that the guide you mention recommends firestore for " slowly changing data ", which I wonder why? I expect Firectore to be able to perfectly handle many updates per second, few updates per user per second.

   👍 ↩ 🚩 upvoted 1 times

☐ 👤 **M25** 1 year, 5 months ago

Selected Answer: C

Went with C

👍 ↩ 🚩 upvoted 1 times

☐ 👤 **lucaluca1982** 1 year, 6 months ago

Selected Answer: B

B for me

👍 ↩ 🚩 upvoted 1 times

☐ 👤 **ares81** 1 year, 9 months ago

Selected Answer: B

B, for me.

👍 ↩ 🚩 upvoted 2 times

☐ 👤 **kn29** 1 year, 10 months ago

I think C because of latency requirements.
Cloud BigTable has high latency feature from https://cloud.google.com/bigtable

👍 ↩ 🚩 upvoted 3 times

   ☐ 👤 **tavva_prudhvi** 1 year, 2 months ago

   correct that Cloud Bigtable can provide better latency compared to Firestore, especially when dealing with very large datasets and high-throughput workloads. However, it's important to consider the trade-offs and the specific use case.

   For the given scenario, the latency requirements are 300ms@p99, which Firestore can handle effectively for thousands of web banners and dynamically changing user browsing history. Firestore is designed for real-time data synchronization and can quickly update the most relevant web banner as the user browses different pages on the website.

   While Cloud Bigtable can offer improved latency, it comes with added complexity in terms of management and configuration. If the primary goal is to implement the simplest solution while meeting the latency requirements, Firestore remains a more suitable choice for this use case.

   👍 ↩ 🚩 upvoted 1 times

☐ 👤 **ares81** 1 year, 10 months ago

I need a DB to store the banners, so no A. We're talking of thousands of banners, so no C. Memorystore calls Redis, and other solutions, so no D. The answer is B, for me.

👍 ↩ 🚩 upvoted 1 times

Start Learning for free

EXAMTOPICS

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.