Google Discussions

EXAM PROFESSIONAL DATA ENGINEER TOPIC 1 QUESTION 67 DISCUSSION

Actual exam question from Google's Professional Data Engineer
Question #: 67
Topic #: 1
[All Professional Data Engineer Questions]

You are developing an application that uses a recommendation engine on Google Cloud. Your solution should display new videos to customers based on past views. Your solution needs to generate labels for the entities in videos that the customer has viewed. Your design must be able to provide very fast filtering suggestions based on data from other customer preferences on several TB of data. What should you do?

A. Build and train a complex classification model with Spark MLlib to generate labels and filter the results. Deploy the models using Cloud Dataproc. Call the model from your application.

B. Build and train a classification model with Spark MLlib to generate labels. Build and train a second classification model with Spark MLlib to filter results to match customer preferences. Deploy the models using Cloud Dataproc. Call the models from your application.

C. Build an application that calls the Cloud Video Intelligence API to generate labels. Store data in Cloud Bigtable, and filter the predicted labels to match the user's viewing history to generate preferences.

D. Build an application that calls the Cloud Video Intelligence API to generate labels. Store data in Cloud SQL, and join and filter the predicted labels to match the user's viewing history to generate preferences.

Show Suggested Answer

by [deleted] at *March 21, 2020, 4:25 p.m.*

Comments

Type your comment...

Submit

[Removed] Highly Voted 4 years, 1 month ago

Answer: C
A & B - Need to build your own model, so discarded as options C D can do the job here using Cloud Video Intelligence API.
BigTable is better option. So C is correct

upvoted 36 times

jin0 1 year, 2 months ago

I don't understand why Vision API should be a answer for labeling? there is no information about input data. isn't it?

upvoted 1 times

jin0 1 year, 2 months ago

Is there any notice that has to reject own model in question..?

upvoted 1 times

[Removed] Highly Voted 4 years, 1 month ago

Answer: C
Description: Why to build own model, Video API with Bigtable is best solution

upvoted 14 times

Mathew106 Most Recent 9 months, 2 weeks ago

Selected Answer: C

I don't even know if MLLib has out-of-the-box Computer Vision models. Developing this in Dataproc would be a nightmare.

Using the computer vision API on the other hand makes perfect sense.

The fact that the filtering must happen very fast and that this is a customer facing application points to BigTable so that there is very little latency and high availability. BigTable is eventually consistent but that doesn't really matter for this application.

CloudSQL will ensure strong consistency which we don't really need but it is slower and supports max 64 TB. The description mentions multiple TBs. Not really sure what several means here, but Cloud SQL doesn't have a high cap.

upvoted 2 times

euro202 10 months ago

Selected Answer: C

We need a model that extracts labels from videos, so Vision API could be used.
Then we need a DB very fast and that can handle several TB of data, so BigTable is the best choice.
Answer is C.

upvoted 1 times

samdhimal 1 year, 3 months ago

Option C is the correct choice because it utilizes the Cloud Video Intelligence API to generate labels for the entities in the videos, which would save time and resources compared to building and training a model from scratch. Additionally, by storing the data in Cloud Bigtable, it allows for fast and efficient filtering of the predicted labels based on the user's viewing history and preferences. This is a more efficient and cost-effective approach than storing the data in Cloud SQL and performing joins and filters.

upvoted 2 times

AzureDP900 1 year, 4 months ago

Answer is C
Build an application that calls the Cloud Video Intelligence API to generate labels. Store data in Cloud Bigtable, and filter the predicted labels to match the user's viewing history to generate preferences.

1. Rather than building a new model - it is better to use Google provide APIs, here - Google Video Intelligence. So option A and B rules out
2. Between SQL and Bigtable - Bigtable is the better option as Bigtable support row-key filtering. Joining the filters is not required.

Reference:
https://cloud.google.com/video-intelligence/docs/feature-label-detection

upvoted 1 times

MaxNRG 2 years, 4 months ago

Selected Answer: C

C.
The cloud video intelillence api does the label generation without the need of building any model, A and B are excluded.

Now, the bbdd most suitable for this is bigtable and not SQL (this big joins would be anything but fast).
https://cloud.google.com/video-intelligence/docs/feature-label-detection

upvoted 2 times

sumanshu 2 years, 10 months ago

Vote for C

upvoted 4 times

timolo 3 years, 1 month ago

Answer: C
Reference https://cloud.google.com/video-intelligence/docs/feature-label-detection

upvoted 2 times

daghayeghi 3 years, 1 month ago

answer C:
If we presume that use label of video as a rowkey, Bigtable will be the best option. because it can store several TB, but Cloud SQL is limited to 30TB.

upvoted 7 times

NamitSehgal 3 years, 4 months ago

Answer: C

upvoted 3 times

Alasmindas 3 years, 5 months ago

Option C is the correct answer.
1. Rather than building a new model - it is better to use Google provide APIs, here - Google Video Intelligence.
So option A and B rules out
2) Between SQL and Bigtable - Bigtable is the better option as Bigtable support row-key filtering. Joining the filters is not required.

upvoted 7 times

SureshKotla 3 years, 7 months ago

Answer is D : BigTable doesnt support JOIN and not built for transactions - https://cloud.google.com/bigtable/docs/overview

upvoted 2 times

Surjit24 3 years, 6 months ago

There are no joins but filtering based on condition.

upvoted 4 times

karthik89 3 years, 2 months ago

but the requirement involves join as well, it is stated in the problem.

upvoted 2 times

sumanshu 2 years, 10 months ago

Where? Though it's mention - " very fast filtering suggestions" - which means something like dictionary in python OR Key: Value (which is Bigtable)

upvoted 1 times

sraakesh95 2 years, 3 months ago

I think "based on other customer preferences" from the questions requires a join before a filter is applied for collaborative filtering

upvoted 1 times

Deepakd 2 years, 1 month ago

Recommendation based on other customer"s views cannot be achieved through simple joins. A class pf machine learning algorithms called collaborative filtering is required for that. You need big table to run these algorithms.

upvoted 1 times

haroldbenites 3 years, 8 months ago

Correct C

upvoted 2 times

dg63 3 years, 10 months ago

I doubt if C can be an answer. Will Bigtable allow filtering on labels?

upvoted 2 times

tprashanth 3 years, 9 months ago

Yes, if its part of the rowkey

upvoted 3 times

Rajuuu 3 years, 10 months ago

Answer is C.

upvoted 4 times

Ganshank 4 years ago

C.
The recommendation requires filtering based on several TB of data, therefore BigTable is the recommended option vs Cloud SQL which is limited to 10TB.

upvoted 7 times