

用户推荐数据从BigQuery导出到BigTable

BigQuery是谷歌云的数据仓库产品，主要用来执行数据分析任务，其延时一般在秒级。BigTable是谷歌云的NoSQL数据库产品，主要用来执行高并发实时查询和写入任务，其延时一般在10毫秒以下。结合二者优点，我们可以将针对用户的分析型处理在BigQuery中执行，结果存储在有用户ID字段的表中，然后用Dataflow导出数据到BigTable，存放在以用户ID作为Row Key的表中，之后就可以从客户端对BigTable中的数据做高速读取。下面的例子针对一个简单的内容推荐场景提供相应示例。

BigQuery数据表准备

可以准备一个符合以下Schema的表，存放示例数据。

```
[
  {
    "mode": "NULLABLE",
    "name": "predicted_rating",
    "type": "FLOAT"
  },
  {
    "mode": "NULLABLE",
    "name": "user_id",
    "type": "INTEGER"
  },
  {
    "mode": "NULLABLE",
    "name": "item_id",
    "type": "INTEGER"
  },
  {
    "mode": "NULLABLE",
    "name": "rating_rank",
    "type": "INTEGER"
  }
]
```

```
]
```

本例中的表为`youzhi-lab.movielens.recomm_ranked_filtered_short`，其权限设定为可以公共读取。其中内容如下。

Row	predicted_rating	user_id	item_id	rating_rank
1	4.8766766223891409	1	356	1
2	4.740690997059704	1	318	2
3	4.7170527143988785	1	1234	3
4	4.7052213124968878	1	2905	4
5	4.70433886514265	1	527	5
6	4.7937573047432629	2	356	1
7	4.6538003904722638	2	1240	2
8	4.644783312514762	2	2028	3
9	4.6304927464501473	2	110	4
10	4.5697576169622236	2	1246	5

其中各字段

- `user_id`：用户的ID
- `item_id`：推荐给用户的内容的ID
- `predicted_rating`：此条推荐内容的置信度
- `rating_rank`：此条推荐内容在给此用户的所有推荐内容中的置信度排名

BigTable数据表准备

在谷歌云控制台上创建好BigTable实例。本例的实例名为`bt-quickstart`，然后用谷歌云提供的`cbt`命令行工具创建表和列族。

```
cbt createtable user-recommendations
cbt createfamily user-recommendations cf_recommendations
```

表中不用写入任何数据。稍后作为目的表接收Dataflow写入的用户推荐数据。

运行Dataflow数据导入任务

首先下载代码到本地。

git clone <https://github.com/cloudymoma/DataflowTemplates.git>

创建一个谷歌云存储桶用作零时文件存储。本例为youzhi-lab。

注意：建议BigQuery数据集，BigTable实例，Dataflow临时文件存储桶和Dataflow集群都创建在同一个区域，这样会减少数据跨区域产生的延时和流量费用。

根据实际情况创建以下环境变量。其中高亮部分请用您项目实际内容填写。

```
PROJECT_ID=youzhi-lab
BUCKET=youzhi-lab
BIGTABLE_INSTANCE_ID=bt-quickstart
BIGTABLE_TABLE=user-recommendations
BQ_TABLE=youzhi-lab:movielens.recomm_ranked_filtered_short
```

在本地编译运行。

```
mvn compile exec:java \
-Dexec.mainClass=com.google.cloud.teleport.templates.BigQueryToBigTable \
-Dexec.args=" \
--stagingLocation=gs://$BUCKET/dataflow/pipelines/staging \
--tempLocation=gs://$BUCKET/dataflow/pipelines/temp \
--bigtableProjectId=$PROJECT_ID \
--bigtableInstanceId=$BIGTABLE_INSTANCE_ID \
--bigtableTableId=$BIGTABLE_TABLE \
--bqTable=$BQ_TABLE"
```

确认输出没有错误，并且BigTable中接收到了数据。

```
cbt read user-recommendations
```

```
[eugeneyu:~/work/gcp/demos/DataflowDemos]$ cbt read user-recommendations
```

```
-----  
1  
cf_recommendations:item_id1      @ 2020/04/19-14:56:24.090000  
  "356"  
cf_recommendations:item_id2      @ 2020/04/19-14:56:24.123000  
  "318"  
cf_recommendations:item_id3      @ 2020/04/19-14:56:24.124000  
  "1234"  
cf_recommendations:item_id4      @ 2020/04/19-14:56:24.125000  
  "2905"  
cf_recommendations:item_id5      @ 2020/04/19-14:56:24.125000  
  "527"  
cf_recommendations:predicted_rating1 @ 2020/04/19-14:56:24.090000  
  "4.876676622389141"  
cf_recommendations:predicted_rating2 @ 2020/04/19-14:56:24.123000  
  "4.740690997059704"  
cf_recommendations:predicted_rating3 @ 2020/04/19-14:56:24.124000  
  "4.7170527143988785"  
cf_recommendations:predicted_rating4 @ 2020/04/19-14:56:24.125000  
  "4.705221312496888"  
cf_recommendations:predicted_rating5 @ 2020/04/19-14:56:24.125000  
  "4.70433886514265"  
-----  
2  
cf_recommendations:item_id1      @ 2020/04/19-14:56:24.126000  
  "356"  
cf_recommendations:item_id2      @ 2020/04/19-14:56:24.127000  
  "1240"  
cf_recommendations:item_id3      @ 2020/04/19-14:56:24.127000  
  "2028"  
cf_recommendations:item_id4      @ 2020/04/19-14:56:24.128000  
  "110"  
cf_recommendations:item_id5      @ 2020/04/19-14:56:24.129000  
  "1246"  
cf_recommendations:predicted_rating1 @ 2020/04/19-14:56:24.126000  
  "4.793757304743263"  
cf_recommendations:predicted_rating2 @ 2020/04/19-14:56:24.127000  
  "4.653800390472264"  
cf_recommendations:predicted_rating3 @ 2020/04/19-14:56:24.127000  
  "4.644783312514762"  
cf_recommendations:predicted_rating4 @ 2020/04/19-14:56:24.128000  
  "4.630492746450147"  
cf_recommendations:predicted_rating5 @ 2020/04/19-14:56:24.129000  
  "4.569757616962224"
```

之后在提交Dataflow任务在谷歌云上运行。

```
mvn compile exec:java \  
-Dexec.mainClass=com.google.cloud.teleport.templates.BigQueryToBigTable \  
-Dexec.args=" \  
--runner=DataflowRunner \  
--workerMachineType=n1-highmem-4 \  
--numWorkers=2 \  
"
```

```
--region=us-central1 \  
--stagingLocation=gs://$BUCKET/dataflow/pipelines/staging \  
--tempLocation=gs://$BUCKET/dataflow/pipelines/temp \  
--bigtableProjectId=$PROJECT_ID \  
--bigtableInstanceId=$BIGTABLE_INSTANCE_ID \  
--bigtableTableId=$BIGTABLE_TABLE \  
--bqTable=$BQ_TABLE"
```

在谷歌云控制台查看Dataflow任务正确启动并且成功执行完成。确认没有任何错误。

The screenshot displays the Google Cloud Dataflow console interface. The top navigation bar includes the Dataflow logo, a 'Job details' header with a 'BACK TO OLD JOB PAGE' link, and a 'MAX TIME' dropdown. The left sidebar shows 'Jobs' and 'Notebooks' options. The main content area is divided into 'JOB GRAPH' and 'JOB METRICS' tabs. The 'JOB GRAPH' tab shows a vertical sequence of three job steps: 'BigQueryIO.TypedRead' (Succeeded, 5 sec), 'ParDo(Anonymous)' (Succeeded, 1 sec), and 'CloudBigtab...eTransform' (Succeeded, 6 sec). The 'JOB METRICS' tab is currently selected, showing a 'Logs' section with 26 messages. The logs include information about scanning data, worker configuration, and the execution of various operations. On the right side, the 'Job info' panel provides details about the job, including its name, ID, type, status, SDK version, region, and start time.

Job name	bigquerytobigtable-eugeneyu-0421091453-66f3719b
Job ID	2020-04-21_02_18_52-13034690168667684236
Job type	Batch
Job status	✓ Succeeded
SDK version	Apache Beam SDK for Java 2.18.0 A newer version of the SDK family exists and updating is recommended. Learn more
Region	us-central1
Start time	April 21, 2020 at 5:18:53 PM GMT+8

Logs Showing 26 messages

- Scanned up to 4/12/20, 7:59 AM. Scanned 400.2 MB.
- 2020-04-21 17:19:05.856 HKT Worker configuration: n1-highmem-4 in us-central1-c...
- 2020-04-21 17:19:06.885 HKT Executing operation BigQueryIO.TypedRead/Read(BigQueryTableSource)+BigQueryIO.TypedRead/PassThroughThenCleanup/ParMultiDo(Id...
- 2020-04-21 17:19:06.922 HKT Executing operation BigQueryIO.TypedRead/ViewId/Combine.GloballyAsSingletonView/Combine.globally(Singleton)/Combine.perKey(S...
- 2020-04-21 17:19:06.958 HKT Executing operation BigQueryIO.TypedRead/ViewId/Combine.GloballyAsSingletonView/BatchViewOverrides.GroupByWindowHashAsKeyAnd...
- 2020-04-21 17:19:06.969 HKT Starting 2 workers in us-central1-c...

之后在BigTable中查看数据已经更新。

到此，使用简单的Schema从BigQuery同步数据到BigTable成功完成。之后可以根据业务需求修改表的结构和数据，创建您自己的数据同步管道。如有必要，可以对Dataflow代码做简单修改之后编译运行。有更复杂的需求，也可以咨询谷歌云支持工程师。