Report on ProjectA： Twitter Analysis

Jianwen Liu

1.Introduction

Building data pipeline is a crucial work for data science. As the development of massive data in social network, how to building data pipeline with massive social network data is becoming a necessary skill for data scientist. With the Jetstream, we built a data pipeline to deal with the Twitter dataset and update it in the MongoDB database.

2. Implementation and description

2.1. Creating VM

Within the Jetstream, we create a project called ProjectA, and select the image: 'I535-I435-B669 Project A' and style: m1.tiny (1 CPU, 2 GB memory) as follows:

Then I comment the lines in the VI for MongoDB configuration file:

```
#security:
# authorization: enabled
```

2.2. Building pipeline

First, I Extract the tools from their zipped and tarred package:

```
tar –zxf I535-TwitterProjectCode.tar.gz

cd I535-TwitterProjectCode
```

Then I change the configuration file in VI editor by replacing "username" with my login name:

```
# $Id: build.properties

# @author: Yuan Luo

# Configuration properties for building I535-TwitterProjectCode
```

```
project.base.dir=/home/jianwenl/Project/I535-TwitterProjectCode

java.home=/usr/bin
```

Thus, we successfully created the Java software.

2.3. Running Data Pipeline

we move users_10000.txt data file from project directory to the /I535-TwitterPorjectCode directory:

```
mv users_10000.txt I535-TwitterProjectCode/
```

reformat it and add header to the tsv:

```
user_id user_name       friend_count    follower_count  status_count    favorite_count  account_age     user_location
100008949       esttrellitta    264     44      6853    0       28 Dec 2009 18:01:42 GMT        El Paso,Tx.
100009841       ChelseaBex      152     50      394     0       28 Dec 2009 18:05:43 GMT
100012792       ErinPattisonn   984     666     5003    0       28 Dec 2009 18:19:39 GMT        under your bed.
100013967       TUBeautifulRosa 323     251     1269    0       28 Dec 2009 18:24:51 GMT        on  Twitter ....... a
```

we import the tsv to mongoDB:

```
./bin/import_mongodb.sh projectA profile tsv user_10000.tsv
```

2.4. Updates in MongoDB-Option One

Now, in the mongoDB, since the constraint on google geo coding, we manually insert geo data. I choose Option 1 and do steps as follows:

Requirement 1. run the command at least 5 times with various options (update a single document, update documents that match query criteria, etc.)

1) update one document with a specific ObjectId, setting user name to “jwen”

```
> db.profile.update({"_id" : ObjectId("5c895e097275df776c0f499d")},{$set: {"user_name" : "jwen"}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
```

2) update one document with a specific ObjectId, setting both friend count and favorite count.

```
> db.profile.update({"_id" : ObjectId("5c895e097275df776c0f499d")},{$set: {"friend_count" : 300,"favorite_cou
nt":1}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.profile.find({user_location:/tx/})
{ "_id" : ObjectId("5c895e097275df776c0f499d"), "user_id" : 113206951, "user_name" : "jwen", "friend_count" :
 300, "follower_count" : 465, "status_count" : 35840, "favorite_count" : 1, "account_age" : "11 Feb 2010 01:5
5:55 GMT", "user_location" : "pimpin pens, tx" }
{ "_id" : ObjectId("5c895e097275df776c0f4ee0"), "user_id" : 117641301, "user_name" : "sincereself", "friend_c
ount" : 84, "follower_count" : 69, "status_count" : 1104, "favorite_count" : 0, "account_age" : "26 Feb 2010
04:26:43 GMT", "user_location" : "Harker heights tx 76548" }
```

3) update multiple documents with a condition that favorite account is 0, setting both follower count and and multi to true.

```
> db.profile.update({"favorite_count":0},{$inc : {"follower_count":2}},{multi:true})
WriteResult({ "nMatched" : 9999, "nUpserted" : 0, "nModified" : 9999 })
```

4) update one document with a specific ObjectId, insert both latitude and longitude initiated with zero.

```
> db.profile.update({"_id" : ObjectId("5c895e097275df776c0f38dd")},{$set : {"lat":0,"long":0}},{upsert:false,
multi:true})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.profile.find().limit(1)
{ "_id" : ObjectId("5c895e097275df776c0f38dd"), "user_id" : 100008949, "user_name" : "esttrellitta", "friend_
count" : 264, "follower_count" : 52, "status_count" : 6853, "favorite_count" : 0, "account_age" : "28 Dec 200
9 18:01:42 GMT", "user_location" : "El Paso,Tx.", "lat" : 0, "long" : 0 }
```

5) update multiple documents with a condition that follower count is 3, insert both latitude and longitude initiated with zero.

```
> db.profile.update({"follower_count":3},{$set : {"lat":0,"long":0}},{upsert:false,multi:true})
WriteResult({ "nMatched" : 11, "nUpserted" : 0, "nModified" : 11 })
```

Requirement 2. update at least 25 documents in the database (locations and coordinates do not have to be real)

We further chose follower_account that is greater than 800, adding latitude and longitude and initiating them to zero to the documents. It turns out to have 1587 documents having this condition.

```
> db.profile.update({"follower_count":{$gte:800}},{$set : {"lat":0,"long":0}},{upsert:false,multi:true})
WriteResult({ "nMatched" : 1587, "nUpserted" : 0, "nModified" : 1587 })
```

we further choose documents whose geo location is in TX (Texas) and update their latitudes and longitudes. We firstly use dp.profile.find() to choose those with location TX, and add latitude and longitude (we choose latitude and longitude of the state of Texas for all the documents here identically).

```
> db.profile.update({"follower_count":{$gte:800}},{$set : {"lat":0,"long":0}},{upsert:false,multi:true})
WriteResult({ "nMatched" : 1587, "nUpserted" : 0, "nModified" : 1587 })
> db.profile.update({user_location: /tx/ },{$set : {"lat":31.169,"long":-99.68}},{upsert:false,multi:true})
WriteResult({ "nMatched" : 7, "nUpserted" : 0, "nModified" : 7 })
> db.profile.find({user_location:/tx/})
{ "_id" : ObjectId("5c895e097275df776c0f499d"), "user_id" : 113206951, "user_name" : "jwen", "friend_count" :
 300, "follower_count" : 465, "status_count" : 35840, "favorite_count" : 1, "account_age" : "11 Feb 2010 01:5
5:55 GMT", "user_location" : "pimpin pens, tx", "lat" : 31.169, "long" : -99.68 }
{ "_id" : ObjectId("5c895e097275df776c0f4ee0"), "user_id" : 117641301, "user_name" : "sincereself", "friend_c
ount" : 84, "follower_count" : 71, "status_count" : 1104, "favorite_count" : 0, "account_age" : "26 Feb 2010
04:26:43 GMT", "user_location" : "Harker heights tx 76548", "lat" : 31.169, "long" : -99.68 }
{ "_id" : ObjectId("5c895e097275df776c0f526f"), "user_id" : 120869102, "user_name" : "mysadityass", "friend_c
ount" : 473, "follower_count" : 617, "status_count" : 12361, "favorite_count" : 0, "account_age" : "7 Mar 201
0 20:41:00 GMT", "user_location" : "dallas tx", "lat" : 31.169, "long" : -99.68 }
{ "_id" : ObjectId("5c895e097275df776c0f5662"), "user_id" : 124317883, "user_name" : "Afrikcaribbean", "frien
d_count" : 97, "follower_count" : 66, "status_count" : 719, "favorite_count" : 0, "account_age" : "19 Mar 201
0 00:52:58 GMT", "user_location" : "Houston tx", "lat" : 31.169, "long" : -99.68 }
{ "_id" : ObjectId("5c895e097275df776c0f5780"), "user_id" : 125243466, "user_name" : "cutd903", "friend_count
" : 404, "follower_count" : 323, "status_count" : 2149, "favorite_count" : 0, "account_age" : "22 Mar 2010 05
:15:47 GMT", "user_location" : "dekalb tx", "lat" : 31.169, "long" : -99.68 }
{ "_id" : ObjectId("5c895e097275df776c0f5b94"), "user_id" : 128622757, "user_name" : "BrownBeauty91", "friend
_count" : 98, "follower_count" : 57, "status_count" : 589, "favorite_count" : 0, "account_age" : "1 Apr 2010
17:47:27 GMT", "user_location" : "tyler,tx", "lat" : 31.169, "long" : -99.68 }
{ "_id" : ObjectId("5c895e097275df776c0f5e29"), "user_id" : 130983911, "user_name" : "Jaih101", "friend_count
" : 130, "follower_count" : 40, "status_count" : 353, "favorite_count" : 0, "account_age" : "8 Apr 2010 22:59
:57 GMT", "user_location" : "Houston, tx", "lat" : 31.169, "long" : -99.68 }
```

In addition, I also updated the documents on location CA (for California) with an identical coordinate.

```
> db.profile.update({user_location: /ca/ },{$set : {"lat":36.77,"long":119.41}},{upsert:false,multi:true})
WriteResult({ "nMatched" : 317, "nUpserted" : 0, "nModified" : 317 })
```

Requirement 3. export your dataset into a csv so that your updated locations and coordinates are in the file

Finally, we use the command:

scp username@ipadress:/home/username/ProjectA/I535-TwitterProjectCode/res.csv /Users/jianwenl

in local terminal to transfer the csv back.

3.Conclusion

By building the data pipeline and update the database in the Mongodb running in jetstream, we find out that using VM for twitter analysis is a good choice. By directly updating data in MongoDB, we can change the data, regardless of individual or multiple documents. We can also use Mongodb update to edit data based on conditions. In general, the Jetstream with Mongodb is a great practice tool for twitter analysis and the analysis show there are more Twitter user in California than in Texas according to this file.