This the first push :

Command to initiate mysql : mysql -u root -p, it indicates that the user is “root”

After I create a database called “ca2\_tweets”

Enabling loading local with the commandf : sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf

Check the status of the mysql database :

CREATE TABLE ProjectTweets (

`index` INT NOT NULL,

ids BIGINT NOT NULL,

`date` VARCHAR(50) NOT NULL,

flag VARCHAR(50) NOT NULL,

user VARCHAR(50) NOT NULL,

`text` TEXT NOT NULL,

PRIMARY KEY (`index`)

);

SELECT \* FROM ca2\_tweets.ProjectTweets LIMIT 10;

Install locally pandas: sudo apt-get install python3-pandas

sudo apt-get install python3-scikit-learn

Problem using : !pip install python-dotenv

“To install Python packages system-wide, try apt install python3-xyz, where xyz is the package you are trying to install.” So sudo apt-get install python3-dotenv

Check which libraries I have in my jupyter notebook : !pip list

Before using pyspark:  
  
**jps**

**$start-dfs.sh**

**$start-yarn.sh**

**To know the host name : SELECT @@hostname;**

**$stop-dfs.sh**

**$stop-yarn.sh**

**Command to acces to ycsb:**

**cd ycsb-0.17.0**

**check db.properties of ycsb: hduser@ubuntum1:~/ycsb-0.17.0/jdbc-binding/conf$ ls**

**check workloadc with 1.000rows (recordcount=1000): hduser@ubuntum1:~/ycsb-0.17.0/workloads$ nano workloadc**

**check workloada with 10.000 rows(recordcount=10000): hduser@ubuntum1:~/ycsb-0.17.0/workloads$ nano workloada**

**check workload with 487.928 (recordcount=** **487928 : hduser@ubuntum1:~/ycsb-0.17.0/workloads$ nano workloade**

**check workload with 1.000.000 (recordcount=** **1000000 : hduser@ubuntum1:~/ycsb-0.17.0/workloads$ nano workloadd**

Check the number of rows in mysql:  
SELECT COUNT(\*) FROM usertable;

Delete all rows int the table usertable:  
delete from usertable;

Command to use workloadc with 1.000 units in mysql (text file):  
hduser@ubuntum1:~/ycsb-0.17.0$./bin/ycsb.sh load jdbc -P ./jdbc-binding/conf/db.properties -P workloads/workloadc > /home/hduser/outputMySQL\_WORKLOADC\_1000.txt

Command to use workloada with 10.000 units in mysql (text file):  
hduser@ubuntum1:~/ycsb-0.17.0$./bin/ycsb.sh load jdbc -P ./jdbc-binding/conf/db.properties -P workloads/workloada > /home/hduser/outputMySQL\_WORKLOADA\_10000.txt

Command to use workloade with 487.928 units in mysql (text file):  
hduser@ubuntum1:~/ycsb-0.17.0$./bin/ycsb.sh load jdbc -P ./jdbc-binding/conf/db.properties -P workloads/workloade > /home/hduser/outputMySQL\_WORKLOADE\_ 487928.txt

START mongo database: sudo systemctl start mongod

Check status of mongo database: sudo systemctl status mongod

Type: “Mongosh” to interact with the databases;

Command to show databases: show dbs;

To use ycsb database : use ycsb

To show collections: show collections;

To count documents use : db.usertable.countDocuments()

Delete documents form mongo collection: db.usertable.deleteMany({});

Command to use workloada with 1.000 units in mongo :

./bin/ycsb.sh load mongodb -s -P workloads/workloadc > /home/hduser/outputMONGO\_WORKLOADA\_1000.txt

Command to use workloada with 10.000 units in mongo :

./bin/ycsb.sh load mongodb -s -P workloads/workloada > /home/hduser/outputMONGO\_WORKLOADA\_10000.txt

Command to use workloada with 487.928 units in mongo :

./bin/ycsb.sh load mongodb -s -P workloads/workloade > /home/hduser/outputMONGO\_WORKLOADA\_487928.txt

Command to use workloada with 487.928 units in mongo :

./bin/ycsb.sh load mongodb -s -P workloads/workloadd > /home/hduser/outputMONGO\_WORKLOADA\_1000000.txt

./bin/ycsb.sh load mongodb -s -P workloads/workloada > /home/hduser/outputMONGO\_WORKLOADA\_10000.txt

Check whqt I have in Hadoop(hdfs): hdfs dfs -ls /

**hadoop fs -put ./ProjectTweets.csv /ca2\_bd\_ad**

**sudo hadoop fs -put /var/lib/mysql-files/test\_mysql.csv /ca2\_bd\_ad**

**Merge all the files that pyspark created as csv: hdfs dfs -getmerge /db\_to\_forecast\_ca2 /home/hduser/merged\_file.csv**

**check if the file is in Hadoop: change to push**

command to chek disk usage: df -h

A graph with a line

Description automatically generated

When I merged filed from hdfs into 1 csv, in mongo appeared like that:

\_id index ids date flag user text \

263571 664a0fb9e1eba02a48f04b01 index ids date flag user text

527585 664a0fc1e1eba02a48f4524f index ids date flag user text

790636 664a0fc9e1eba02a48f855da index ids date flag user text

1054110 664a0fd0e1eba02a48fc5b0c index ids date flag user text

1317457 664a0fd8e1eba02a48005fc0 index ids date flag user text

1581913 664a0fdfe1eba02a480468c8 index ids date flag user text

timestamp sentiment\_score date\_string

263571 timestamp sentiment\_score date

527585 timestamp sentiment\_score date

790636 timestamp sentiment\_score date

1054110 timestamp sentiment\_score date

1317457 timestamp sentiment\_score date

1581913 timestamp sentiment\_score date