Nicolas Escobar (escobarn)
Harsh Reddy (hagandav)

Cloud Computing - Project 8 Report

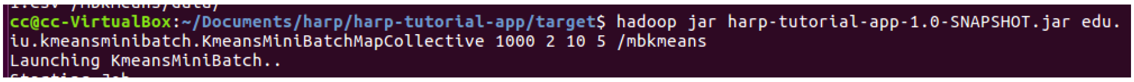
Bonus Credits

1. **Perform experiments on various (small, medium, large, etc) datasets**

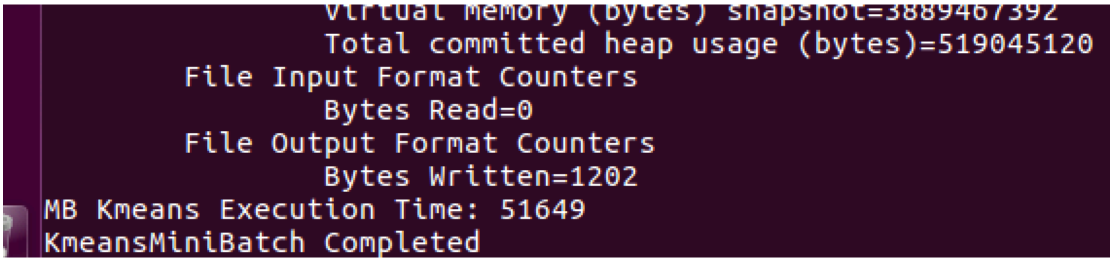
   We executed the following experiments:

   Experiment #1 (Small data set, small batch size)

   - Input data: two files with 4,000 rows each
   - Infrastructure: Local VM
   - Mappers: 2
   - Iterations: 10
   - Centroids: 5
   - Batch size: 1,000
   - Execution time: 51,649 ms.

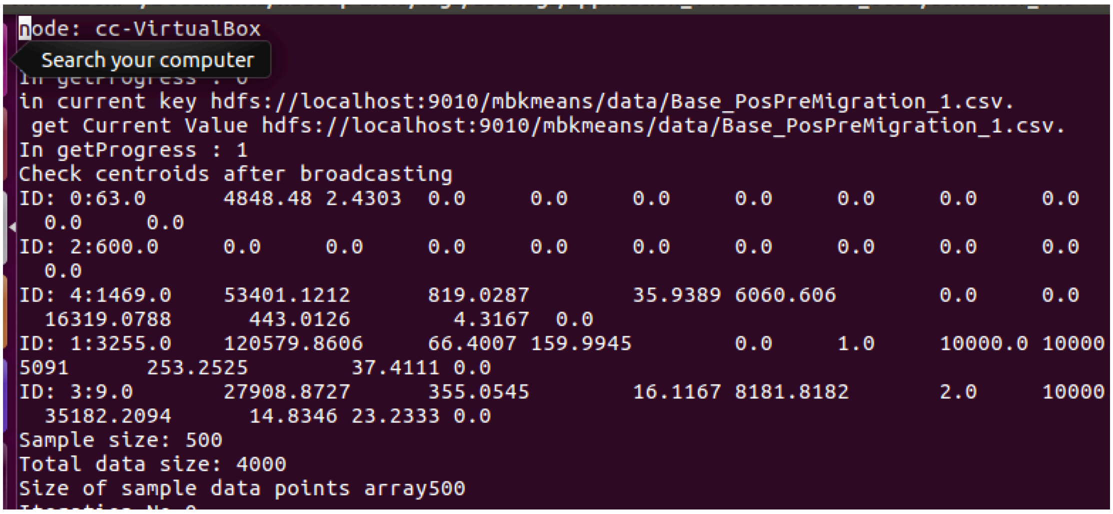


```
cc@cc-VirtualBox:~/Documents/harp/harp-tutorial-app/target$ hadoop jar harp-tutorial-app-1.0-SNAPSHOT.jar edu.
iu.kmeansminibatch.KmeansMiniBatchMapCollective 1000 2 10 5 /mbkmeans
Launching KmeansMiniBatch..
```



```
                Virtual memory (bytes) snapshot=3889467392
                Total committed heap usage (bytes)=519045120
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=1202
MB Kmeans Execution Time: 51649
KmeansMiniBatch Completed
```

Log file execution sample:



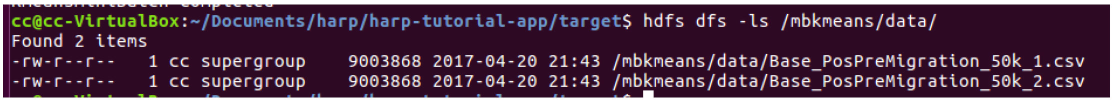
```
node: cc-VirtualBox
In getProgress : 0
in current key hdfs://localhost:9010/mbkmeans/data/Base_PosPreMigration_1.csv.
 get Current Value hdfs://localhost:9010/mbkmeans/data/Base_PosPreMigration_1.csv.
In getProgress : 1
Check centroids after broadcasting
ID: 0:63.0      4848.48 2.4303  0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0     0.0
ID: 2:600.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0
ID: 4:1469.0    53401.1212      819.0287        35.9389 6060.606        0.0     0.0
  16319.0788      443.0126        4.3167  0.0
ID: 1:3255.0    120579.8606     66.4007 159.9945        0.0     1.0     10000.0 10000
5091      253.2525        37.4111 0.0
ID: 3:9.0       27908.8727      355.0545        16.1167 8181.8182       2.0     10000
  35182.2094      14.8346 23.2333 0.0
Sample size: 500
Total data size: 4000
Size of sample data points array500
```

Experiment #2 (Large data set, small batch size)

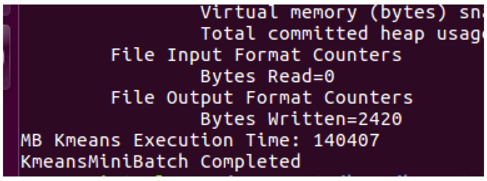
- Input data: two files with over 50,000 rows each
- Infrastructure: Local VM
- Mappers: 2
- Iterations: 10
- Centroids: 5
- Batch size: 5,000
- Execution time: 51,426 ms.

Experiment #3 (Large data set, big batch size)

- Input data: two files with over 50,000 rows each
- Infrastructure: Local VM
- Mappers: 2
- Iterations: 10
- Centroids: 5
- Batch size: 80,000
- Execution time: 140,407 ms.

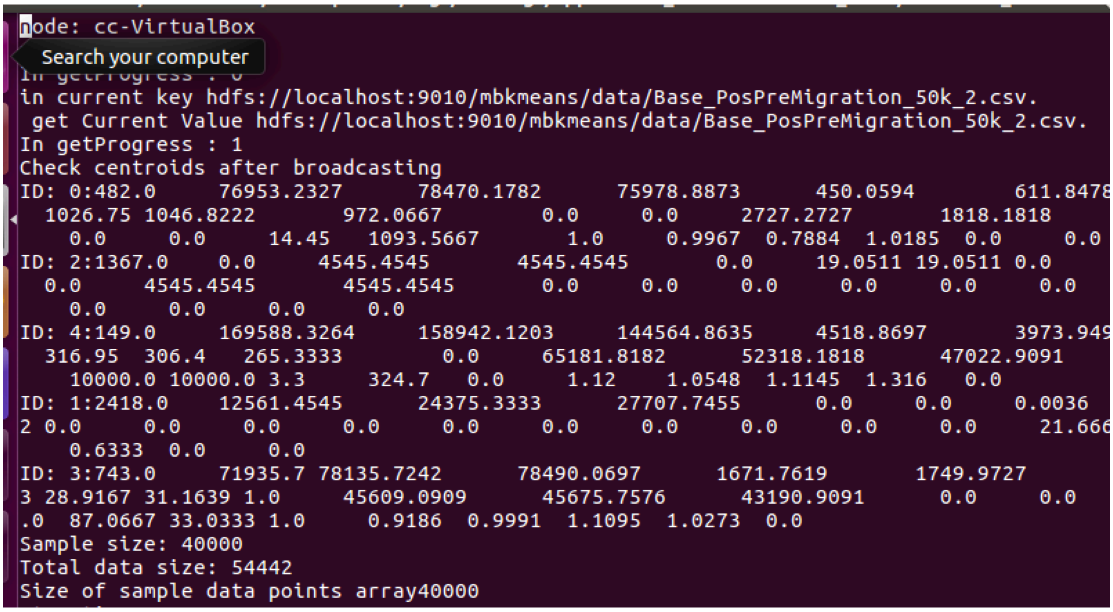


```
cc@cc-VirtualBox:~/Documents/harp/harp-tutorial-app/target$ hdfs dfs -ls /mbkmeans/data/
Found 2 items
-rw-r--r--   1 cc supergroup    9003868 2017-04-20 21:43 /mbkmeans/data/Base_PosPreMigration_50k_1.csv
-rw-r--r--   1 cc supergroup    9003868 2017-04-20 21:43 /mbkmeans/data/Base_PosPreMigration_50k_2.csv
```



```
                Virtual memory (bytes) sn
                Total committed heap usag
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=2420
MB Kmeans Execution Time: 140407
KmeansMiniBatch Completed
```

Log file execution sample:



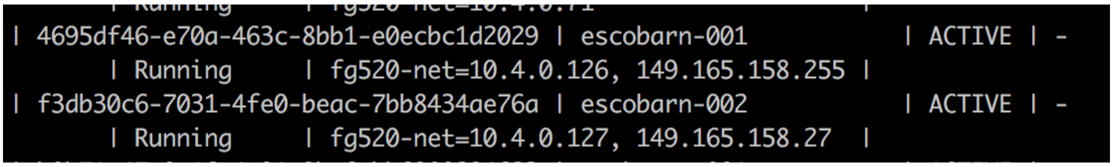
```
node: cc-VirtualBox
In getProgress : 0
in current key hdfs://localhost:9010/mbkmeans/data/Base_PosPreMigration_50k_2.csv.
 get Current Value hdfs://localhost:9010/mbkmeans/data/Base_PosPreMigration_50k_2.csv.
In getProgress : 1
Check centroids after broadcasting
ID: 0:482.0     76953.2327      78470.1782      75978.8873      450.0594        611.8478
  1026.75 1046.8222       972.0067        0.0     0.0     2727.2727       1818.1818
    0.0     0.0     14.45   1093.5667       1.0     0.9967  0.7884  1.0185  0.0     0.0
ID: 2:1367.0    0.0     4545.4545       4545.4545       0.0     19.0511 19.0511 0.0
  0.0     4545.4545       4545.4545       0.0     0.0     0.0     0.0     0.0     0.0
    0.0     0.0     0.0     0.0
ID: 4:149.0     169588.3264     158942.1203     144564.8635     4518.8697       3973.949
  316.95  306.4   265.3333        0.0     65181.8182      52318.1818      47022.9091
    10000.0 10000.0 3.3     324.7   0.0     1.12    1.0548  1.1145  1.316   0.0
ID: 1:2418.0    12561.4545      24375.3333      27707.7455      0.0     0.0     0.0036
2 0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     21.666
    0.6333  0.0     0.0
ID: 3:743.0     71935.7 78135.7242      78490.0697      1671.7619       1749.9727
3 28.9167 31.1639 1.0     45609.0909      45675.7576      43190.9091      0.0     0.0
.0  87.0667 33.0333 1.0     0.9186  0.9991  1.1095  1.0273  0.0
Sample size: 40000
Total data size: 54442
Size of sample data points array40000
```

2. **Test your algorithm on at least 2 nodes on FutureSystem.**

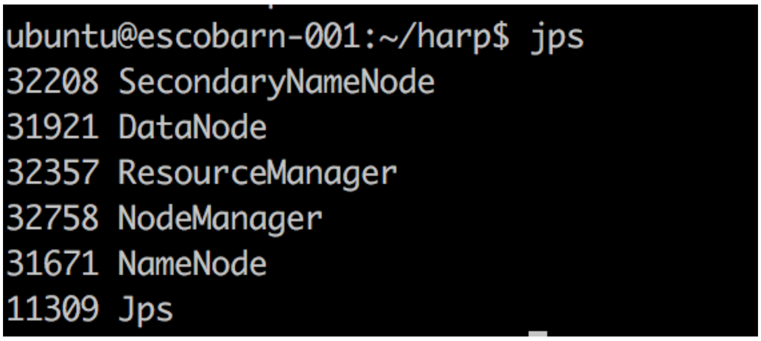
We configured two nodes on FutureSystem which are:

- escobarn-001 → 149.165.158.255
- escobarn-002 → 149.165.158.27

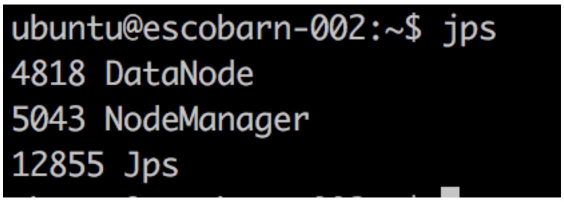
These nodes are running hadoop-2.6.5 with java 1.8 (openjdk 1.8.0_111) and harp. We followed the instructions from https://dsc-spidal.github.io/harp/docs/getting-started-cluster/ to configure and execute the cluster.



```
| 4695df46-e70a-463c-8bb1-e0ecbc1d2029 | escobarn-001           | ACTIVE | -
       | Running      | fg520-net=10.4.0.126, 149.165.158.255 |
| f3db30c6-7031-4fe0-beac-7bb8434ae76a | escobarn-002           | ACTIVE | -
       | Running      | fg520-net=10.4.0.127, 149.165.158.27  |
```



```
ubuntu@escobarn-001:~/harp$ jps
32208 SecondaryNameNode
31921 DataNode
32357 ResourceManager
32758 NodeManager
31671 NameNode
11309 Jps
```



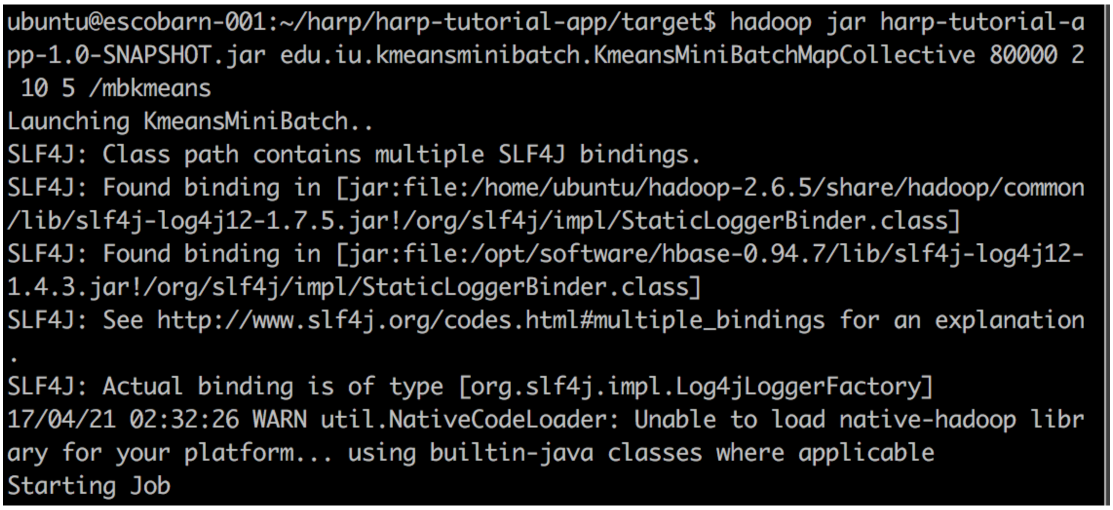
```
ubuntu@escobarn-002:~$ jps
4818 DataNode
5043 NodeManager
12855 Jps
```

We have tested our algorithm against a large data set on the FutureSystems nodes and obtained the following results:

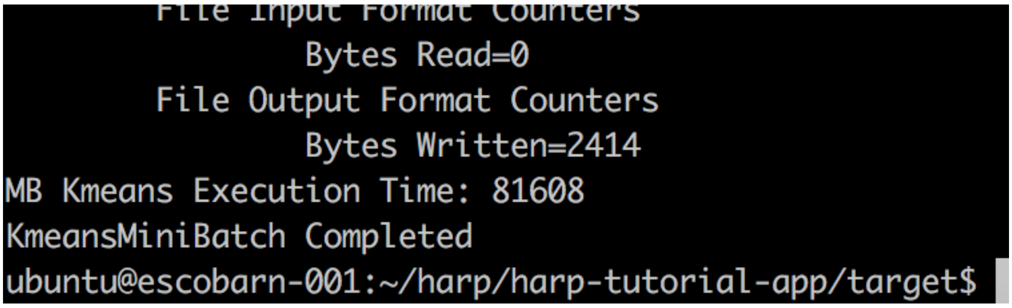
- Input data: two files with over 50,000 rows each
- Infrastructure: Future Systems Nodes running Ubuntu
- Mappers: 2
- Iterations: 10
- Centroids: 5
- Batch size: 80,000
- Execution time: 81,608 ms.

Screenshots of execution logs can be found below:

Sample execution logs



```
ubuntu@escobarn-001:~/harp/harp-tutorial-app/target$ hadoop jar harp-tutorial-a
pp-1.0-SNAPSHOT.jar edu.iu.kmeansminibatch.KmeansMiniBatchMapCollective 80000 2
 10 5 /mbkmeans
Launching KmeansMiniBatch..
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/ubuntu/hadoop-2.6.5/share/hadoop/common
/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/opt/software/hbase-0.94.7/lib/slf4j-log4j12-
1.4.3.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation
.
SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFactory]
17/04/21 02:32:26 WARN util.NativeCodeLoader: Unable to load native-hadoop libr
ary for your platform... using builtin-java classes where applicable
Starting Job
```



```
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=2414
MB Kmeans Execution Time: 81608
KmeansMiniBatch Completed
ubuntu@escobarn-001:~/harp/harp-tutorial-app/target$
```