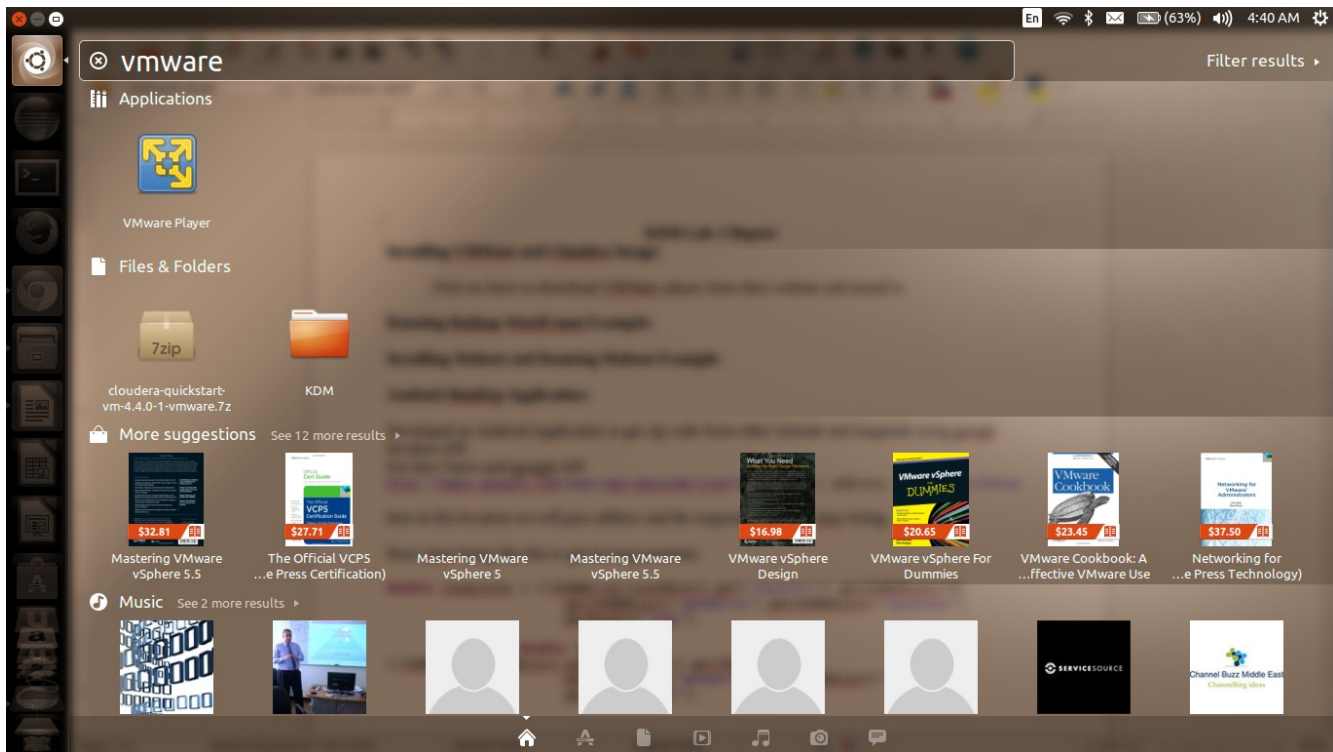


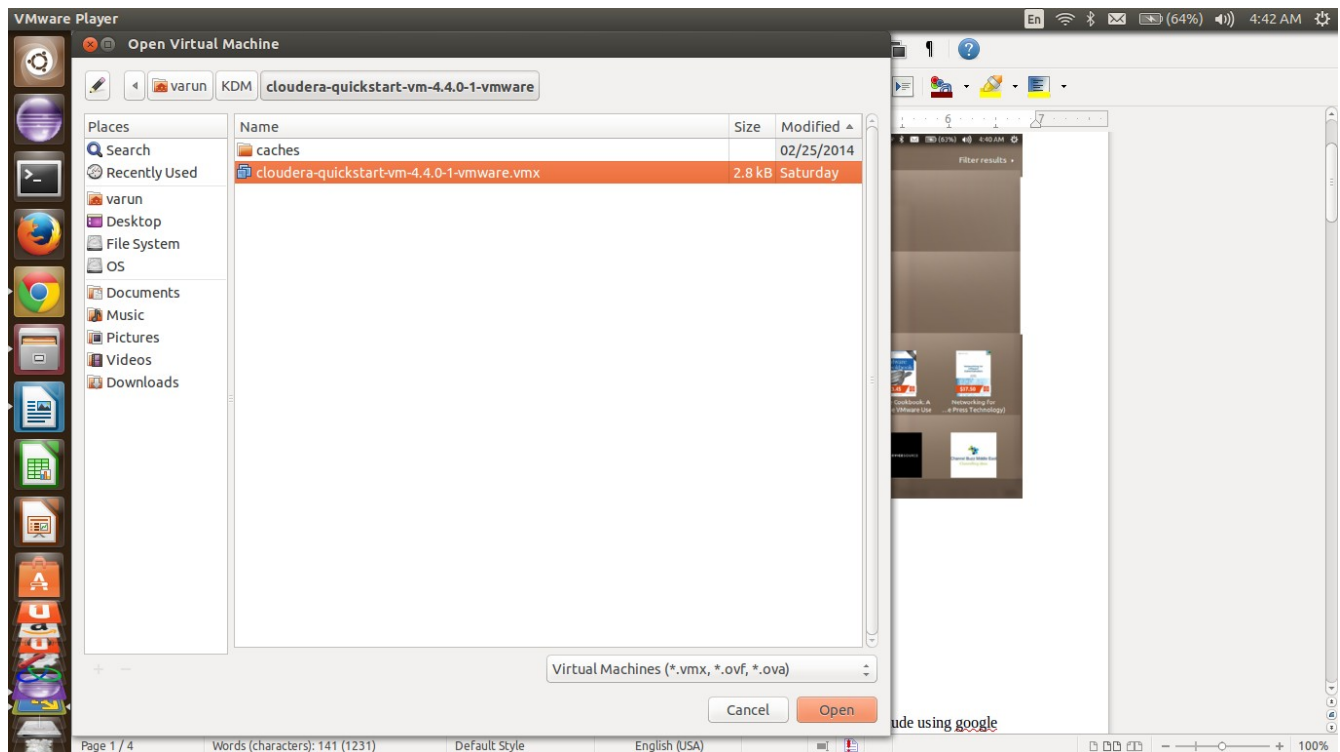
KDM Lab-3 Report

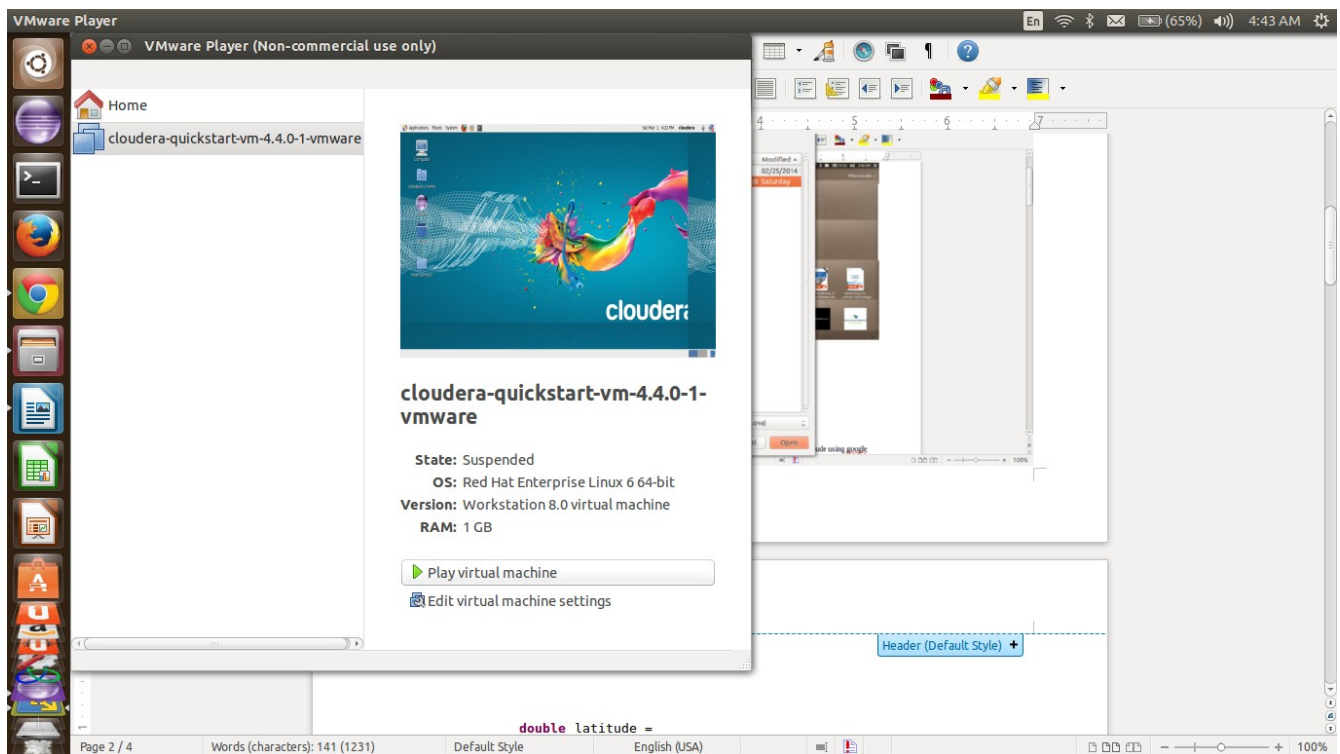
Installing VMWare and Cloudera Image:

First we have to download VMWare player from their website and install it.



After that download cloudera image and open it using vmware player





We can login to cloudera manager with
 username - “cloudera”
 password - “cloudera”

Running Hadoop WordCount Example:

First download HadoopWordcount example jar file and copy it to some location on our virtual machine.

Now go to that location and create a input folder and drop the input data to that folder and now we need to push this input to the hadoop distributed file system this can be done using command

`hadoop fs -put input input`

```

cloudera@localhost:~
File Edit View Search Terminal Help

[-put <localsrc> ... <dst>]
[-rm [-f] [-r] [-R] [-skipTrash] <src> ...]
[-rmrdir [-ignore-fail-on-non-empty] <dir> ...]
[-setrep [-R] [-w] <rep> <path/file> ...]
[-stat [format] <path> ...]
[-tail [-f] <file>]
[-test [-ezd] <path>]
[-text [-ignoreCrc] <src> ...]
[-touchz <path> ...]
[-usage [cmd ...]]

Generic options supported are
-conf <configuration file>    specify an application configuration file
-D <property=value>           use value for given property
-fs <local|namenode:port>      specify a namenode
-jt <local|jobtracker:port>    specify a job tracker
-files <comma separated list of files> specify comma separated files to be copied to the map reduce cluster
-libjars <comma separated list of jars> specify comma separated jar files to include in the classpath.
-archives <comma separated list of archives> specify comma separated archives to be unarchived on the compute machines.

The general command line syntax is
bin/hadoop command [genericOptions] [commandOptions]

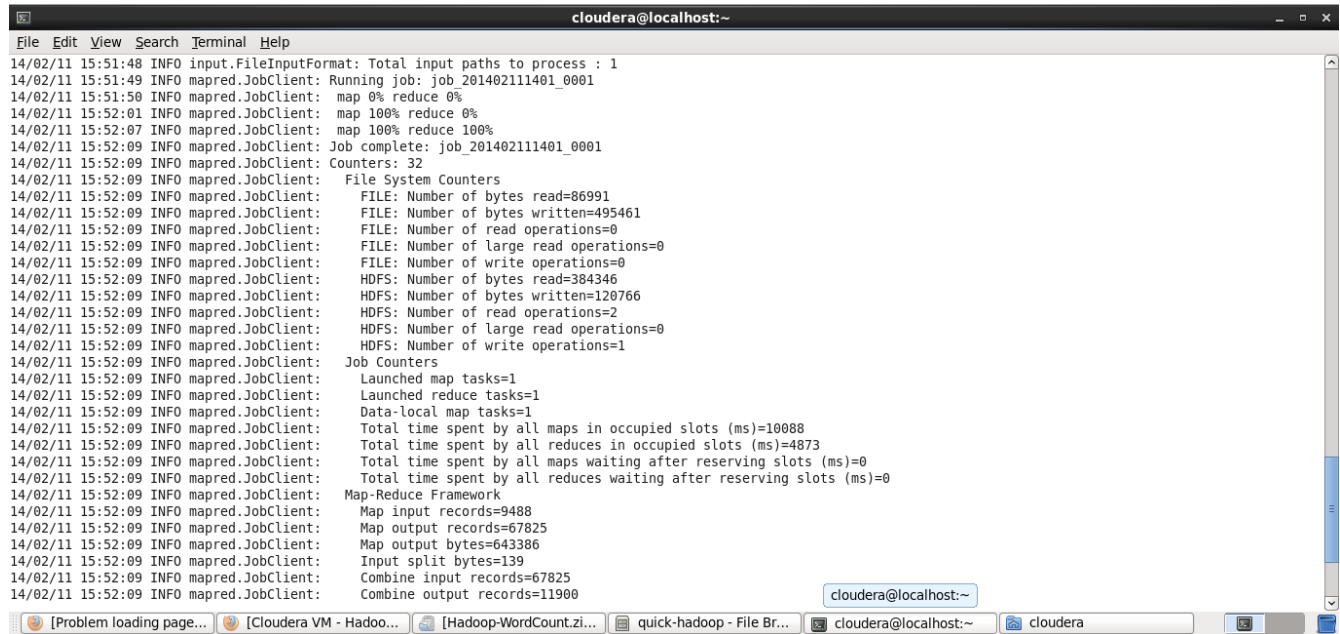
[cloudera@localhost ~]$ hadoop fs -put
-put: Not enough arguments: expected 1 but got 0
Usage: hadoop fs [generic options] -put <localsrc> ... <dst>
[cloudera@localhost ~]$ hadoop fs -put input word_count_input.txt
put: 'input': No such file or directory
[cloudera@localhost ~]$ hadoop fs -put input input
[cloudera@localhost ~]$ hadoop fs -ls
Found 1 items
drwxr-xr-x  - cloudera cloudera          0 2014-02-11 15:43 input
[cloudera@localhost ~]$

```

After pushing the content to hadoop HDFS we can run wordcount example on that input data.

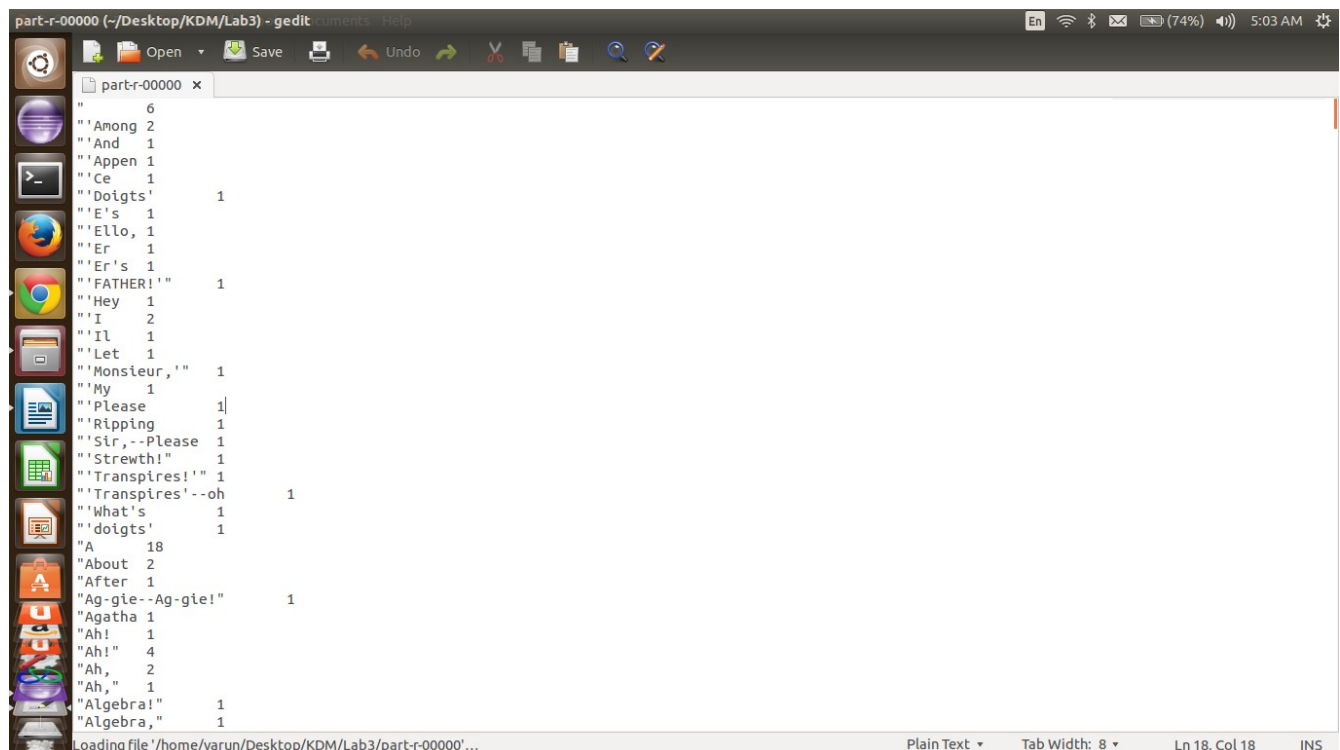
This can be done using command

`hadoop jar wordcount.jar WordCount input output`



```
cloudera@localhost:~  
File Edit View Search Terminal Help  
14/02/11 15:51:48 INFO input.FileInputFormat: Total input paths to process : 1  
14/02/11 15:51:49 INFO mapred.JobClient: Running job: job_201402111401_0001  
14/02/11 15:51:50 INFO mapred.JobClient: map 0% reduce 0%  
14/02/11 15:52:01 INFO mapred.JobClient: map 100% reduce 0%  
14/02/11 15:52:07 INFO mapred.JobClient: map 100% reduce 100%  
14/02/11 15:52:09 INFO mapred.JobClient: Job complete: job_201402111401_0001  
14/02/11 15:52:09 INFO mapred.JobClient: Counters: 32  
14/02/11 15:52:09 INFO mapred.JobClient: File System Counters  
14/02/11 15:52:09 INFO mapred.JobClient: FILE: Number of bytes read=86991  
14/02/11 15:52:09 INFO mapred.JobClient: FILE: Number of bytes written=495461  
14/02/11 15:52:09 INFO mapred.JobClient: FILE: Number of read operations=0  
14/02/11 15:52:09 INFO mapred.JobClient: FILE: Number of large read operations=0  
14/02/11 15:52:09 INFO mapred.JobClient: FILE: Number of write operations=0  
14/02/11 15:52:09 INFO mapred.JobClient: HDFS: Number of bytes read=384346  
14/02/11 15:52:09 INFO mapred.JobClient: HDFS: Number of bytes written=120766  
14/02/11 15:52:09 INFO mapred.JobClient: HDFS: Number of read operations=2  
14/02/11 15:52:09 INFO mapred.JobClient: HDFS: Number of large read operations=0  
14/02/11 15:52:09 INFO mapred.JobClient: HDFS: Number of write operations=1  
14/02/11 15:52:09 INFO mapred.JobClient: Job Counters  
14/02/11 15:52:09 INFO mapred.JobClient: Launched map tasks=1  
14/02/11 15:52:09 INFO mapred.JobClient: Launched reduce tasks=1  
14/02/11 15:52:09 INFO mapred.JobClient: Data-local map tasks=1  
14/02/11 15:52:09 INFO mapred.JobClient: Total time spent by all maps in occupied slots (ms)=10088  
14/02/11 15:52:09 INFO mapred.JobClient: Total time spent by all reduces in occupied slots (ms)=4873  
14/02/11 15:52:09 INFO mapred.JobClient: Total time spent by all maps waiting after reserving slots (ms)=0  
14/02/11 15:52:09 INFO mapred.JobClient: Total time spent by all reduces waiting after reserving slots (ms)=0  
14/02/11 15:52:09 INFO mapred.JobClient: Map-Reduce Framework  
14/02/11 15:52:09 INFO mapred.JobClient: Map input records=9488  
14/02/11 15:52:09 INFO mapred.JobClient: Map output records=67825  
14/02/11 15:52:09 INFO mapred.JobClient: Map output bytes=643386  
14/02/11 15:52:09 INFO mapred.JobClient: Input split bytes=139  
14/02/11 15:52:09 INFO mapred.JobClient: Combine input records=67825  
14/02/11 15:52:09 INFO mapred.JobClient: Combine output records=11900
```

After completion of map reduce function we will get output in output folder we need to get that output to our virtual machine filesystem and can check the output.



```
part-r-00000 x  
" 6  
" 'Among 2  
" 'And 1  
" 'Appen 1  
" 'Ce 1  
" 'Doigts' 1  
" 'E's 1  
" 'Ello, 1  
" 'Er 1  
" 'Er's 1  
" 'FATHER!' 1  
" 'Hey 1  
" 'I 2  
" 'Il 1  
" 'Let 1  
" 'Monsieur,' 1  
" 'My 1  
" 'Please 1  
" 'Ripping 1  
" 'Sir,--Please 1  
" 'Strewth!' 1  
" 'Transpires!' 1  
" 'Transpires'--oh 1  
" 'What's 1  
" 'doigts' 1  
" 'A 18  
" 'About 2  
" 'After 1  
" 'Ag-gie--Ag-gie!' 1  
" 'Agatha 1  
" 'Ah! 1  
" 'Ah!' 4  
" 'Ah, 2  
" 'Ah," 1  
" 'Algebra!' 1  
" 'Algebra," 1
```

Installing Maven ,Mahout and Running Mahout Example:

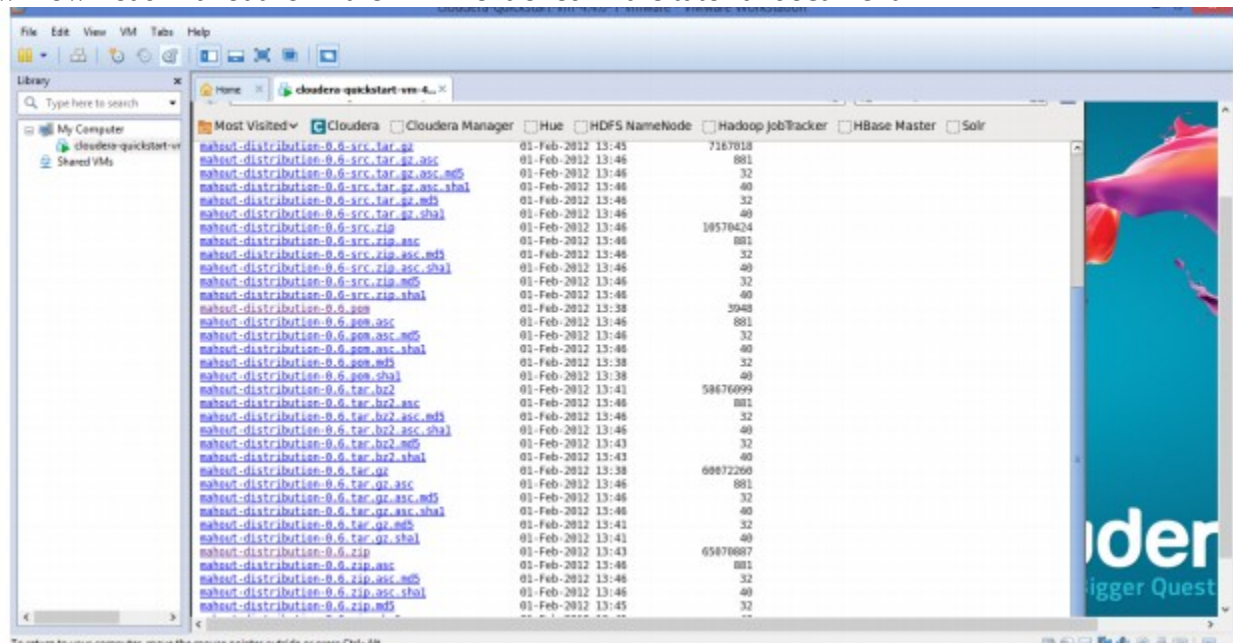
we can download Maven from command line by typing

```
wget http://www.apache.org/dist/maven/binaries/apache-maven-3.0.5-bin.tar.gz
```

This will download Maven to our virtual machine.

Now we need to extract and export HOME and PATH of Maven folder.

Now Download Mahout form the link mentioned in the tutorial document



install mahout using following command

```
mvn -f mahout-distribution-0.6.pom -DskipTests install
```

```
File Edit View Search Terminal Help
drwxrwxr-x 3 cloudera cloudera 4096 Feb 11 16:40 integration
drwxrwxr-x 2 cloudera cloudera 4096 Feb 11 16:40 lib
-rw-r--r-- 1 cloudera cloudera 39588 Feb 1 2012 LICENSE.txt
-rw-r--r-- 1 cloudera cloudera 1662876 Feb 1 2012 mahout-core-0.6.jar
-rw-r--r-- 1 cloudera cloudera 11190212 Feb 1 2012 mahout-core-0.6-job.jar
-rwxrwxr-w 1 cloudera cloudera 3948 Feb 11 16:34 mahout-distribution-0.6.pom
-rw-r--r-- 1 cloudera cloudera 379461 Feb 1 2012 mahout-examples-0.6.jar
-rw-r--r-- 1 cloudera cloudera 23593299 Feb 1 2012 mahout-examples-0.6-job.jar
-rw-r--r-- 1 cloudera cloudera 284781 Feb 1 2012 mahout-integration-0.6.jar
-rw-r--r-- 1 cloudera cloudera 288914 Feb 1 2012 mahout-math-0.6.jar
drwxrwxr-x 3 cloudera cloudera 4096 Feb 11 16:40 math
-rw-r--r-- 1 cloudera cloudera 1888 Feb 1 2012 NOTICE.txt
-rw-r--r-- 1 cloudera cloudera 1200 Feb 1 2012 README.txt
[cloudera@localhost mahout-distribution-0.6]$ mvn -f mahout-distribution-0.6.pom -DskipTests install
[INFO] Scanning for projects...
Downloading: http://repo.maven.apache.org/maven2/org/apache/mahout/mahout/0.6/mahout-0.6.pom
Downloaded: http://repo.maven.apache.org/maven2/org/apache/mahout/mahout/0.6/mahout-0.6.pom (32 KB at 48.6 KB/sec)
Downloading: http://repo.maven.apache.org/maven2/org/apache/apache/9/apache-9.pom
Downloaded: http://repo.maven.apache.org/maven2/org/apache/apache/9/apache-9.pom (15 KB at 45.6 KB/sec)
[INFO]
[INFO] -----
[INFO] Building Mahout Release Package 0.6
[INFO] -----
Downloading: http://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-assembly-plugin/2.2/maven-assembly-plugin-2.2.pom
Downloaded: http://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-assembly-plugin/2.2/maven-assembly-plugin-2.2.pom (17 KB at 66.8 KB/sec)
Downloading: http://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-plugins/18/maven-plugins-18.pom
Downloaded: http://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-plugins/18/maven-plugins-18.pom (13 KB at 73.8 KB/sec)
Downloading: http://repo.maven.apache.org/maven2/org/apache/maven/maven-parent/16/maven-parent-16.pom
Downloaded: http://repo.maven.apache.org/maven2/org/apache/maven/maven-parent/16/maven-parent-16.pom (23 KB at 70.6 KB/sec)
Downloading: http://repo.maven.apache.org/maven2/org/apache/apache/7/apache-7.pom
Downloaded: http://repo.maven.apache.org/maven2/org/apache/apache/7/apache-7.pom (15 KB at 79.6 KB/sec)
Downloading: http://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-assembly-plugin/2.2/maven-assembly-plugin-2.2.jar
Downloaded: http://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-assembly-plugin/2.2/maven-assembly-plugin-2.2.jar (214 KB at 202.2 KB/sec)
```

After completing installation go to examples and run the navi-base classification

Category	Feature 1	Feature 2	Feature 3	Feature 4	Feature 5	Feature 6	Feature 7	Feature 8	Feature 9	Feature 10	Feature 11	Feature 12	Feature 13	Feature 14	Feature 15	Feature 16	Feature 17	Feature 18	Total
f	= sci.electronics	0	1	2	256	3	0	0	5	0	50	8	5	8	1	40	0	0	394
g	= comp.os.ms-windows.misc	1	0	6	2	342	0	0	0	0	12	7	1	4	9	3	0	2	390
h	= misc.forsale	2	32	102	0	0	0	26	29	0	4	0	0	13	7	7	6	0	251
i	= talk.religion.misc	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
j	= alt.atheism	0	0	3	9	4	0	0	284	0	11	6	3	1	0	66	0	1	395
k	= comp.windows.x	1	2	12	1	0	2	0	0	0	338	0	0	0	3	3	1	0	376
l	= talk.politics.mideast	0	0	0	25	18	13	0	0	0	291	27	2	0	3	11	0	0	392
m	= comp.sys.ibm.pc.hardware	0	0	0	12	7	12	0	0	0	4	330	4	2	4	5	0	0	385
n	= comp.sys.mac.hardware	0	0	1	4	0	2	0	0	3	0	0	0	369	0	0	9	0	394
o	= sci.space	0	0	0	2	0	3	0	0	0	1	0	0	381	9	0	0	1	398
p	= rec.motorcycles	0	0	0	6	0	11	0	0	0	3	0	1	10	359	1	0	1	396
q	= rec.autos	0	0	0	14	8	7	0	0	11	0	11	10	7	0	3	297	0	389
r	= comp.graphics	4	109	11	3	0	1	0	0	1	0	1	18	8	6	1	98	10	310
s	= talk.politics.misc	1	3	1	6	12	0	9	0	0	0	2	0	4	4	2	7	0	396
t	= sci.med																		

14/02/11 17:12:54 INFO bayes.TestClassifier: Classified instances from rec.sport.hockey.txt

examples

Android MashUp Application:

Developed an Android Application to get zip code from either latitude and longitude using google location API.

For this I have used google API

[http://maps.google.com/maps/api/geocode/json?address=" + address + "&sensor=false](http://maps.google.com/maps/api/geocode/json?address=)

here in this location we will pass address and the response will be in Json String.

Parse Json String like this to get lat long positions.

```
double longitude = ((JSONArray)jsonObject.get("results")).getJSONObject(0)
    .getJSONObject("geometry").getJSONObject("location")
    .getDouble("lng");

double latitude =
((JSONArray)jsonObject.get("results")).getJSONObject(0)
    .getJSONObject("geometry").getJSONObject("location")
    .getDouble("lat");
```


and to get cheapest GasStationn details for that given location for this service I have used <http://www.mshd.net/api/gasprices/>+ zipcode API

39.030408

-94.5744349

OR

5311 Harrison
StUniversity of
Missouri-Kansas City,
Volker CampusUnited

Get ZipCode Cheapest GasStation

Price :3.13
Brand :Shell
Address :3901 Tomahawk Rd, Prairie Village,
KS 66208

After clicking on the GasStation we will show directions to that GasStation from current location.
Using Google maps.

