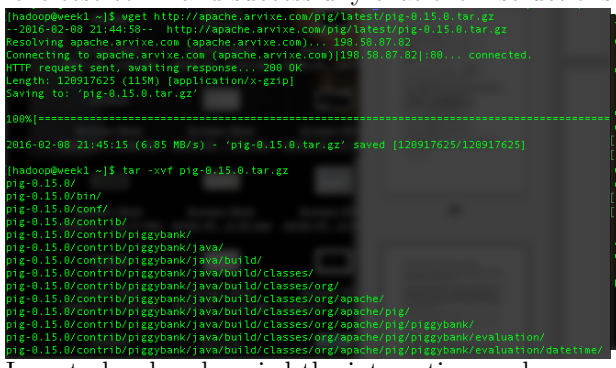
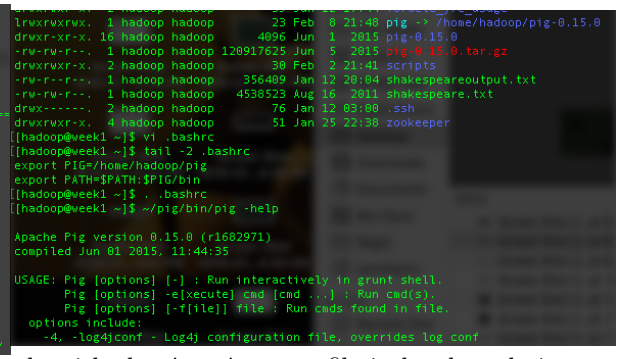
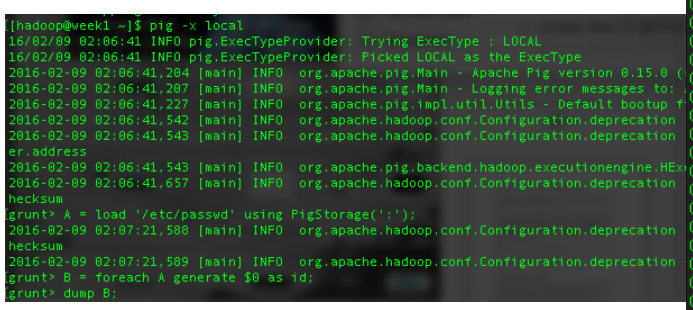
MSDS610 Week 5 Pig Assignment - Nathan Worsham

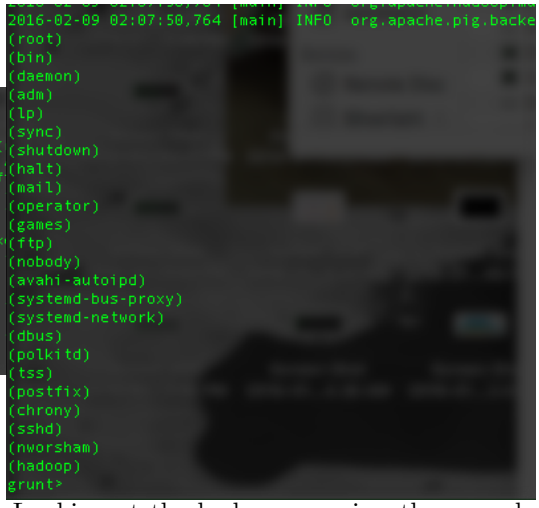
Similar to previous weeks, I began this assignment by powering up my week 1 VM and then geting the download and unpacking the archive. And again similar to previous weeks I installed Pig into the home directory of my hadoop user, created a symbolic link, updated the environment variables for Pig, and then ran the test command successfully that the instructions called for.



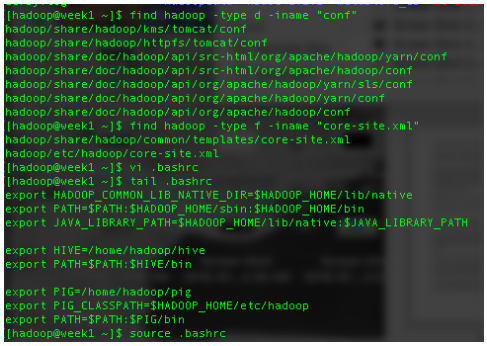


I went ahead and copied the interactive mode example with the `/etc/passwd` file in local mode just to confirm it was working as expected. It did work as expected with the exception that it had some warnings of deprecated features.

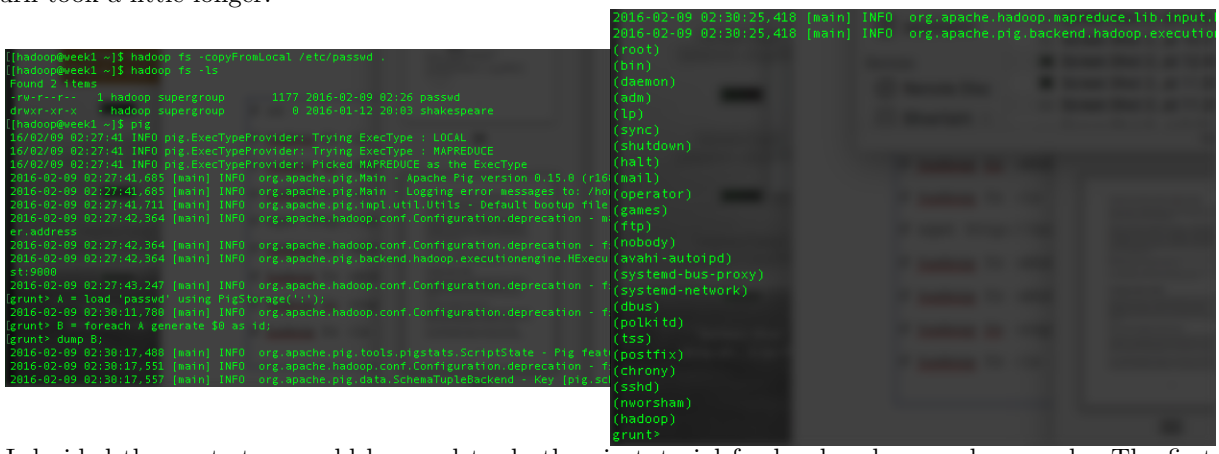




Next I wanted to do the exact same test but in Hadoop. Looking at the hadoop exercise, there needs to be an environment variable set for `PIG_CLASSPATH` which it gives an example as "/mycluster/conf". I went ahead and looked for a conf directory at the root of the hadoop install but could find no such folder, so then I used the find command. None of the results seemed right and after carefully reading I noticed the instructions state "the directory that contains the core-site.xml". So re-running my find command targeting the core-site.xml–a file I was definitely familiar with from past weeks–I found the directory was hadoop/etc/hadoop. So I set my environment variable accordingly.



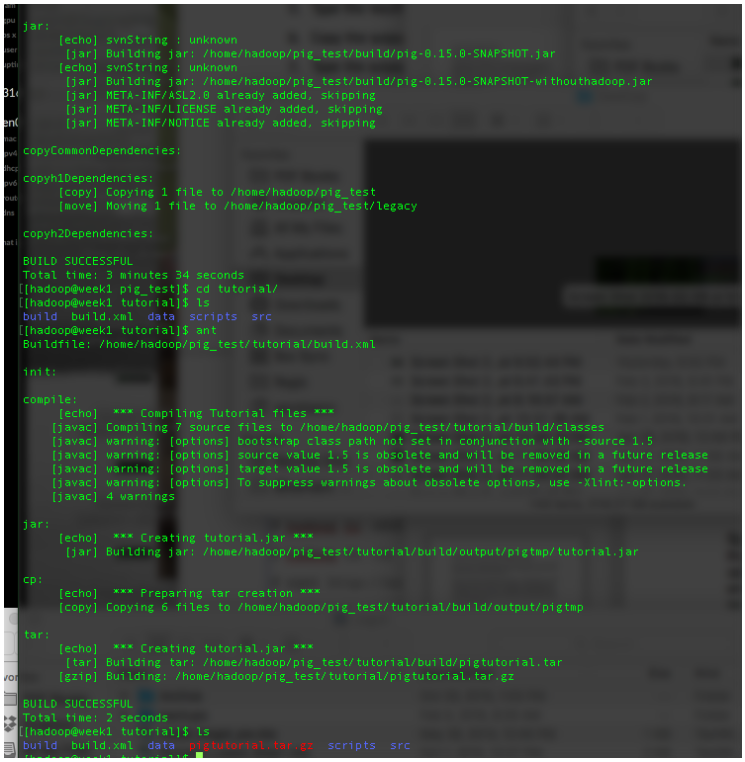
Now I started up HDFS and yarn then ran just simply `pig` since the default mode is mapreduce mode. It started up with again the previous deprecation warnings. I was pleasantly surprised to learn that pig retains a command history that I could access with the up and down arrows from my keyboard. I was equally as pleased that running the same commands in mapreduce mode returned the same output, except that the return took a little longer.



I decided the next step would be good to do the pig tutorial for local and mapreduce mode. The first instruction was to make sure that `JAVA_HOME` was set, which I checked and it was, but then it also asked for another environment variable–`PIG_HOME`–to be set to the same as the previous environment variable `PIG` which I went ahead and did. Next I needed to run the "ant" command in the tutorial directory, a command I had never heard of. I had to install the command first, looks like from its description it is a "Build tool for java". However after running the command, I did not receive the expected output, I received a couple of errors.

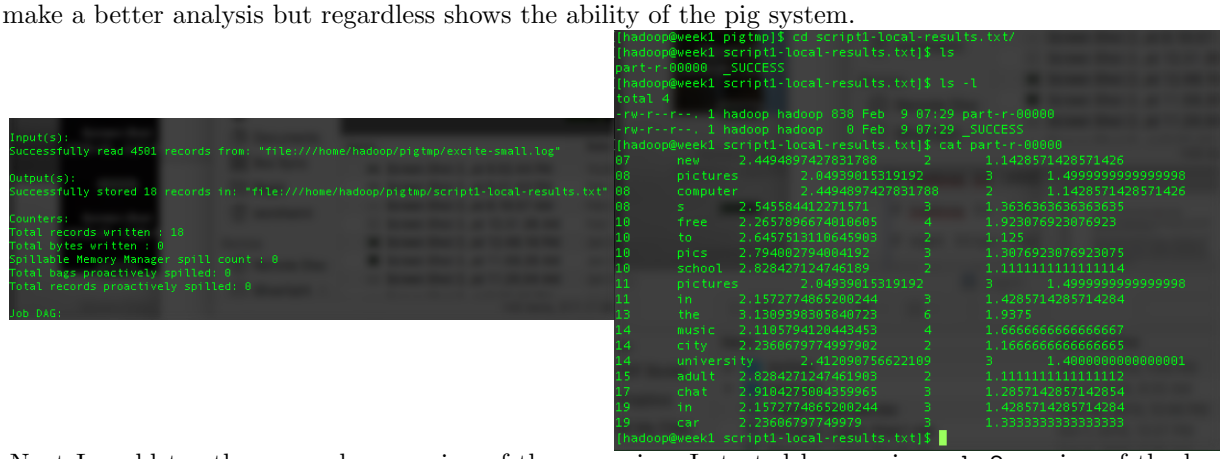
```
[echo]  *** Compiling Tutorial files ***
    [javac] /home/hadoop/pig-0.15.0/tutorial/build.xml:66: warning: 'includeantruntime'
    was not set, defaulting to build.sysclasspath=last; set to false for repeatable builds
    [javac] Compiling 7 source files to /home/hadoop/pig-0.15.0/tutorial/build/classes

BUILD FAILED
/home/hadoop/pig-0.15.0/tutorial/build.xml:66: /home/hadoop/pig-0.15.0/build/ivy/lib/
Pig does not exist.
```

I was able to fix the includeanttime warning by following a Stackoverflow.com (2011) thread and adding `includeantruntime="false"` inside of the `<javac>` options. The final error though was about a build directory not existing at the root level of pig. Taking a look at that area, indeed I saw no build directory but I did see a "build.xml" file just like there was in the tutorial directory. Given this info I assumed I would first need to run `ant` at the root level but was a bit worried that might break something as the installation of pig I chose was the already built version and not the source. I went ahead and copied the entire pig directory to a test directory and ran the `ant` command on the test instance. Sure enough it built the missing directories and running `ant` inside of the tutorial directory now worked and I now had my pigtutorial.tar.gz file.



```
jar:
     [echo] svnString : unknown
      [jar] Building jar: /home/hadoop/pig_test/build/pig-0.15.0-SNAPSHOT.jar
     [echo] svnString : unknown
      [jar] Building jar: /home/hadoop/pig_test/build/pig-0.15.0-SNAPSHOT-withouthadoop.jar
      [jar] META-INF/ASL2.0 already added, skipping
      [jar] META-INF/LICENSE already added, skipping
      [jar] META-INF/NOTICE already added, skipping

copyCommonDependencies:

copyh1Dependencies:
     [copy] Copying 1 file to /home/hadoop/pig_test
     [move] Moving 1 file to /home/hadoop/pig_test/legacy

copyh2Dependencies:

BUILD SUCCESSFUL
Total time: 3 minutes 34 seconds
[[hadoop@week1 pig_test]$ cd tutorial/
[[hadoop@week1 tutorial]$ ls
build  build.xml  data  scripts  src
[[hadoop@week1 tutorial]$ ant
Buildfile: /home/hadoop/pig_test/tutorial/build.xml

init:

compile:
     [echo]  *** Compiling Tutorial files ***
    [javac] Compiling 7 source files to /home/hadoop/pig_test/tutorial/build/classes
    [javac] warning: [options] bootstrap class path not set in conjunction with -source 1.5
    [javac] warning: [options] source value 1.5 is obsolete and will be removed in a future release
    [javac] warning: [options] target value 1.5 is obsolete and will be removed in a future release
    [javac] warning: [options] To suppress warnings about obsolete options, use -Xlint:-options.
    [javac] 4 warnings

jar:
     [echo]  *** Creating tutorial.jar ***
      [jar] Building jar: /home/hadoop/pig_test/tutorial/build/output/pigtmp/tutorial.jar

cp:
     [echo]  *** Preparing tar creation ***
     [copy] Copying 6 files to /home/hadoop/pig_test/tutorial/build/output/pigtmp

tar:
     [echo]  *** Creating tutorial.jar ***
      [tar] Building tar: /home/hadoop/pig_test/tutorial/build/pigtutorial.tar
     [gzip] Building: /home/hadoop/pig_test/tutorial/pigtutorial.tar.gz

BUILD SUCCESSFUL
Total time: 2 seconds
[[hadoop@week1 tutorial]$ ls
build  build.xml  data  pigtutorial.tar.gz  scripts  src
```

After extracting the tar file, I had the remaining files I needed. First running the local test–`script1-local.pig`–which appears to read the "excite-small.log", a log file of the Excite search engine and find the popular query phrases as they relate to times of the day. However the output from the local mode ended up with only single popular words, not phrases, and it seems to include stop words that probably should have been filtered out to make a better analysis but regardless shows the ability of the pig system.



```
Input(s):
Successfully read 4501 records from: "file:///home/hadoop/pigtmp/excite-small.log"

Output(s):
Successfully stored 18 records in: "file:///home/hadoop/pigtmp/script1-local-results.txt"

Counters:
Total records written : 18
Total bytes written : 0
Spillable Memory Manager spill count : 0
Total bags proactively spilled: 0
Total records proactively spilled: 0

Job DAG:
```

```
[hadoop@week1 pigtmp]$ cd script1-local-results.txt/
[hadoop@week1 script1-local-results.txt]$ ls
part-r-00000  _SUCCESS
[hadoop@week1 script1-local-results.txt]$ ls -l
total 4
-rw-r--r--. 1 hadoop hadoop 838 Feb  9 07:29 part-r-00000
-rw-r--r--. 1 hadoop hadoop   0 Feb  9 07:29 _SUCCESS
[hadoop@week1 script1-local-results.txt]$ cat part-r-00000
07      new     2.4494897427831788      2       1.1428571428571426
08      pictures        2.04939015319192        3       1.4999999999999998
06      computer        2.4494897427831788      2       1.1428571428571426
08      s       2.545584412271571       3       1.3636363636363635
10      free    2.2657896674010605      4       1.923076923076923
10      to      2.6457513110645903      2       1.125
10      pics    2.794802794004192       3       1.3076923076923075
10      school  2.828427124746189       2       1.1111111111111114
11      pictures        2.04939015319192        3       1.4999999999999998
11      in      2.1572774865200244      3       1.4285714285714284
13      the     3.1309398305840723      6       1.9375
14      music   2.1105794120443453      4       1.6666666666666667
14      city    2.2360679774997902      2       1.1666666666666665
14      university      2.412090756622109       3       1.4000000000000001
15      adult   2.8284271247461903      2       1.1111111111111112
17      chat    2.9104275004359965      3       1.2857142857142854
19      in      2.1572774865200244      3       1.4285714285714284
19      car     2.23606797749979        3       1.3333333333333333
[hadoop@week1 script1-local-results.txt]$
```

Next I could try the mapreduce version of the excercise. I started by copying a `bz2` version of the log file to the HDFS. Looking at the size of the file compared to the local version, it is considerably bigger–10 mb versus 204 kb. I had already taken care of the environment variable `PIG_CLASSPATH` but now I needed to make `HADOOP_CONF_DIR` equivallent to the same path.

```
[hadoop@week1 ~]$ cd pigtmp/
[hadoop@week1 pigtmp]$ ls
excite.log.bz2  excite-small.log  script1-hadoop.pig  script1-local.pig  scrip
[hadoop@week1 pigtmp]$ hadoop fs -copyFromLocal ~/pigtmp/excite.log.bz2 .
[hadoop@week1 pigtmp]$ ls -l
total 10400
-rw-r--r--. 1 hadoop hadoop 10408717 Feb  9 07:01 excite.log.bz2
-rw-r--r--. 1 hadoop hadoop   208348 Feb  9 07:01 excite-small.log
-rw-r--r--. 1 hadoop hadoop     3835 Feb  9 07:01 script1-hadoop.pig
-rw-r--r--. 1 hadoop hadoop     3820 Feb  9 07:01 script1-local.pig
drwxrwxr-x. 2 hadoop hadoop       84 Feb  9 07:29 script1-local-results.txt
-rw-r--r--. 1 hadoop hadoop     3489 Feb  9 07:01 script2-hadoop.pig
-rw-r--r--. 1 hadoop hadoop     3480 Feb  9 07:01 script2-local.pig
-rw-r--r--. 1 hadoop hadoop    10783 Feb  9 07:01 tutorial.jar
[hadoop@week1 pigtmp]$ ls -lh
total 11M
-rw-r--r--. 1 hadoop hadoop  10M Feb  9 07:01 excite.log.bz2
-rw-r--r--. 1 hadoop hadoop 204K Feb  9 07:01 excite-small.log
-rw-r--r--. 1 hadoop hadoop 3.8K Feb  9 07:01 script1-hadoop.pig
-rw-r--r--. 1 hadoop hadoop 3.8K Feb  9 07:01 script1-local.pig
drwxrwxr-x. 2 hadoop hadoop   84 Feb  9 07:29 script1-local-results.txt
-rw-r--r--. 1 hadoop hadoop 3.5K Feb  9 07:01 script2-hadoop.pig
-rw-r--r--. 1 hadoop hadoop 3.4K Feb  9 07:01 script2-local.pig
-rw-r--r--. 1 hadoop hadoop  11K Feb  9 07:01 tutorial.jar
```

```
[hadoop@week1 pigtmp]$ vi ~/.bashrc
[hadoop@week1 pigtmp]$ tail ~/.bashrc
export JAVA_LIBRARY_PATH=$HADOOP_HOME/lib/native:$JAVA_LIBRARY_PATH

export HIVE=/home/hadoop/hive
export PATH=$PATH:$HIVE/bin

export PIG=/home/hadoop/pig
export PIG_HOME=/home/hadoop/pig
export PIG_CLASSPATH=$HADOOP_HOME/etc/hadoop
export HADOOP_CONF_DIR=$PIG_CLASSPATH
export PATH=$PATH:$PIG/bin
[hadoop@week1 pigtmp]$
```

I was now ready to run the mapreduce version of the script. As expected it took considerably longer considering the much bigger file to analyze and also being within the HDFS.

```
Input(s):
Successfully read 944954 records (31338918 bytes) from: "hdfs://localhost:9000/user/hadoop/excite.log.bz2"

Output(s):
Successfully stored 13530 records (206471324 bytes) in: "hdfs://localhost:9000/user/hadoop/script1-hadoop-results"
```

The results where much bigger than the local script test–13530 lines versus 18–and this time did have phrases in addition to single words. The results seemed to cover all hours of the day.

```
[hadoop@week1 pigtmp]$ hadoop fs -ls
Found 4 items
-rw-r--r--   1 hadoop supergroup   10408717 2016-02-09 12:58 excite.log.bz2
-rw-r--r--   1 hadoop supergroup       1177 2016-02-09 02:26 passwd
drwxr-xr-x   - hadoop supergroup          0 2016-02-09 13:12 script1-hadoop-results
drwxr-xr-x   - hadoop supergroup          0 2016-01-12 20:03 shakespeare
[hadoop@week1 pigtmp]$ hadoop fs -ls script1-hadoop-results
Found 2 items
-rw-r--r--   1 hadoop supergroup          0 2016-02-09 13:12 script1-hadoop-results/_SUCCESS
-rw-r--r--   1 hadoop supergroup     659954 2016-02-09 13:12 script1-hadoop-results/part-r-00000
[hadoop@week1 pigtmp]$ hadoop fs -tail -20 script1-hadoop-results/part-r-00000
-tail: Illegal option -20
Usage: hadoop fs [generic options] -tail [-f] <file>
[hadoop@week1 pigtmp]$ hadoop fs -tail script1-hadoop-results/part-r-00000
foto    2.691946385511004       3       1.3571428571428568
23      shauna  2.7714348384599967      4       1.625
23      bosch   2.7940027940041911      3       1.3076923076923077
23      hottest 2.7950849718747373      3       1.3333333333333333
23      alamak  2.8185890875479 6       9       2.782608695652174
23      chilton 2.8284271247461 9       3       1.2222222222222223
23      red hot 2.8284271247461 9       2       1.1111111111111112
23      jerk off        2.8284271247461 9       2       1.1111111111111112
23      live video      2.8284271247461 9       2       1.1111111111111112
23      brothels        2.8284271247461 9       2       1.1111111111111112
23      bubbles 2.8284271247461 9       2       1.1111111111111112
23      ping    2.8284271247461 9       5       1.8888888888888893
23      orgasmic        2.8284271247461 9       3       1.2222222222222223
23      highway patrol  2.8284271247461 9       2       1.1111111111111112
23      jenni cam       3.0     2       1.0999999999999999
23      cheryl bachman  3.162277660168379       2       1.090909090909091
23      wallpapers      3.2888995827048717      6       1.9333333333333331
23      agp     3.316624790355399       2       1.0833333333333335
23      bachman 3.4641016151377557      2       1.0769230769230766
23      kimberley       3.6055512754639096      2       1.0714285714285712
23      jerk    3.6055512754639096      2       1.0714285714285712
[hadoop@week1 pigtmp]$ hadoop fs -less script1-hadoop-results/part-r-00000
-less: Unknown command
[hadoop@week1 pigtmp]$ hadoop fs -cat script1-hadoop-results/part-r-00000|more
00      and shareware   2.112885636821291       3       1.5294117647058825
00      vcd     2.1213203435596424      3       1.5
00      bluebird        2.1555530241167826      4       1.9285714285714282
00      cute    2.1650635094610955      4       1.8571428571428574
00      chested 2.182820625326997       4       1.75
00      diablo cheats   2.197401062294143       3       1.4705882352941178
00      psygnosis       2.23606797749979        2       1.1666666666666667
00      vacancy 2.2360679774997902      2       1.1666666666666665
00      pennywise       2.2360679774997902      2       1.1666666666666665
00      morisette       2.2360679774997902      2       1.1666666666666665
00      labyrinth       2.2360679774997902      2       1.1666666666666665
```

Furthermore it is interesting to compare the results at different hours of the day, example the noon hour ngram results seem to be "cleaner" than the 11 and 12 am results.

```
cat: Unable to write to output stream.
[hadoop@week1 pigtmp]$ hadoop fs -cat script1-hadoop-results/part-r-00000|wc -l
13530
[hadoop@week1 pigtmp]$ ls
excite.log.bz2  excite-small.log  script1-hadoop.pig  script1-local.pig  script1-local-resu
[hadoop@week1 pigtmp]$ cat script1-local-results.txt/part-r-00000 |wc -l
18
[hadoop@week1 pigtmp]$ hadoop fs -cat script1-hadoop-results/part-r-00000|egrep "^12"|more
12      mama    2.0000000000000004      4       1.9999999999999996
12      sets    2.0000000000000004      3       1.6666666666666665
12      university college       2.0000000000000004      3       1.6666666666666665
12      behavioral      2.0000000000000004      4       1.9999999999999998
12      bug     2.0036623359733645      4       1.9444444444444446
12      advantage       2.0036623359733645      4       1.9444444444444446
12      mental  2.0043741974907974      7       3.217391304347826
12      3d studio       2.004607239461597       4       1.9375
12      guard   2.0062576200186775      7       2.7499999999999996
12      martial arts    2.010235005197005       5       2.2857142857142856
12      oil     2.010255242141953 5     19      8.916666666666668
12      touche  2.011435198964418       4       1.9000000000000001
12      universities in 2.0139959720120832      4       2.1000000000000005
12      of california   2.014203131157181       9       4.0476190476190474
12      softball        2.0165150423000915      6       2.4375
12      radio   2.0165872059002656      50      26.999999999999996
12      architects      2.0175288189295504      4       1.875
12      jr      2.017590488204153       9       4.2
12      wrestlers       2.0180183819089375      6       2.842105263157895
12      winfrey 2.018018381988938       6       2.8421052631578947
12      dave    2.0182702288973022      12      5.2608695652173925
12      cancer  2.022509811780486       28      12.0
12      auto insurance  2.0225995873897267      4       1.8571428571428568
12      msds    2.0225995873897267      4       1.8571428571428568
12      literary        2.0251582216668433      5       2.230769230769231
12      endangered      2.0272943000433674      7       3.1764705882352935
12      whales  2.027708610709328       6       2.6111111111111107
12      fm      2.028630468520778       8       3.625000000000004
12      fun     2.0302294357371142      16      7.124999999999999
12      jobline 2.0302589045518786      4       1.8333333333333335
12      wisconsin department    2.0302589045518789      4       1.8333333333333333
```

Looking through the script1-hadoop.pig script, at first it would seem that "pig latin" is its own language but the more I look through the script the more I see that that the UDFs (user defined funchtions) are really where the heavy lifting is done. Looking into what exactly a UDF is, looks like it is a way to provide pig with custom processing and can be written in Java, JavaScript, Python, or Ruby (pig.apache.org, 2012). This made me think, why don't users then share the functions they have written but looks like pig has that as well, called piggybank. Currenlty piggybank only supports Java written functions. Pig latin looks like it is pretty in-depth but the fact that the script had to use a UDF to simply make the field lowercase–`org.apache.pig.tutorial.ToLower(query)`–makes it seem like it is missing some pretty basic built-in functions.

Not sure if the assignment required me to do the second set of scripts but out of interest I ran them. Again the local script only had minimal results, 3 versus 5343 for the hadoop script. This time the output was ngrams and their occurrence at the midnight hour and then the noon hour so you can see the difference.

```
[hadoop@week1 pigtmp]$ cd script2-local-results.txt/
[hadoop@week1 script2-local-results.txt]$ ls
part-r-00000  _SUCCESS
[hadoop@week1 script2-local-results.txt]$ ls -l
total 4
-rw-r--r--. 1 hadoop hadoop 30 Feb 10 09:52 part-r-00000
-rw-r--r--. 1 hadoop hadoop  0 Feb 10 09:52 _SUCCESS
[hadoop@week1 script2-local-results.txt]$ cat part-r-00000
and     2       1
steel   1       1
ukraine 1       1
```

```
population      1       9
postscript      1       5
prediction      1       1
prime time      1       1
production      1       6
properties      1       3
protection      3       9
psychology      3       13
publishers      1       8
publishing      1       16
puff daddy      2       12
queensland      2       1
rap lyrics      1       1
real audio      2       8
recordable      1       1
recruiters      1       6
recruiting      2       7
reform act      1       1
relocation      1       3
republican      2       3
resistance      1       1
restaurant      2       8
rewritable      1       1
richardson      1       2
rio grande      1       1
rome italy      1       1
room rooms      1       1
royal bank      1       2
s homepage      1       5
sacramento      2       4
saddleback      1       1
salt water      1       1
santa cruz      1       3
scanmaster      1       3
school and      1       2
schoolgirl      1       3
```

References

pig.apache.org, 2016. Retrieved from http://pig.apache.org/docs/r0.14.0/start.html
Stackoverflow.com, 2011. Retrieve from http://stackoverflow.com/questions/5103384/ant-warning-includeantruntime-was-not-set
pig.apache.org, 2016. Retrieved from https://pig.apache.org/docs/r0.10.0/udf.html#piggybank