Team Members:

Moonis Javed (50208261, moonisja)
Surabhi Singh Ludu (50207139, sludu)

Activity 1:

Step 1: Use the following command to run the vignette file

```
spark-submit vignette.py
```

Activity 2:

Step 1: Use the following command to start hadoop on virtual machine

```
Start-hadoop.sh
```

Step 2: Place “la.lexicon.csv” in the current working directory in Virtual Machine where you’ll run your code.

Step 3: Use following command to put input files into the hadoop file system

```
hdfs dfs -put <input_folder_name>
```

Step 4: Use this command to run the co-occurrence with two-gram and three-gram

```
spark-submit cooccurence.py <input_folder_name> <output_folder_name> <value of n>
```

[eg: spark_submit cooccurence.py sample_input sample_output 2] (for two gram)
[eg: spark_submit cooccurence.py sample_input sample_output 3] (for three gran)

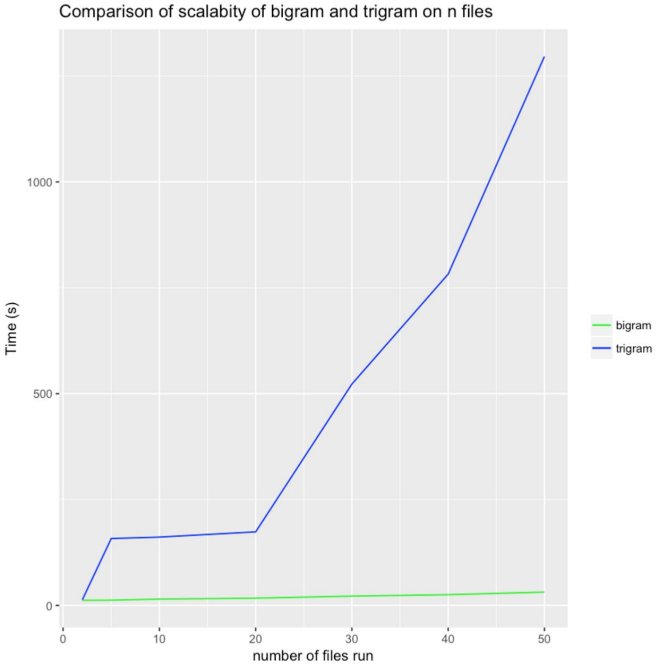
Step 5: Use this command to get the output folder

```
hdfs dfs -get <output_folder_name>
```

Step 6: Repeat step 1-3 multiple times for two-gram and three-gram for different number of files from 2 to 50 (50 was the maximum number of files my system could support with Three-gram), and record values

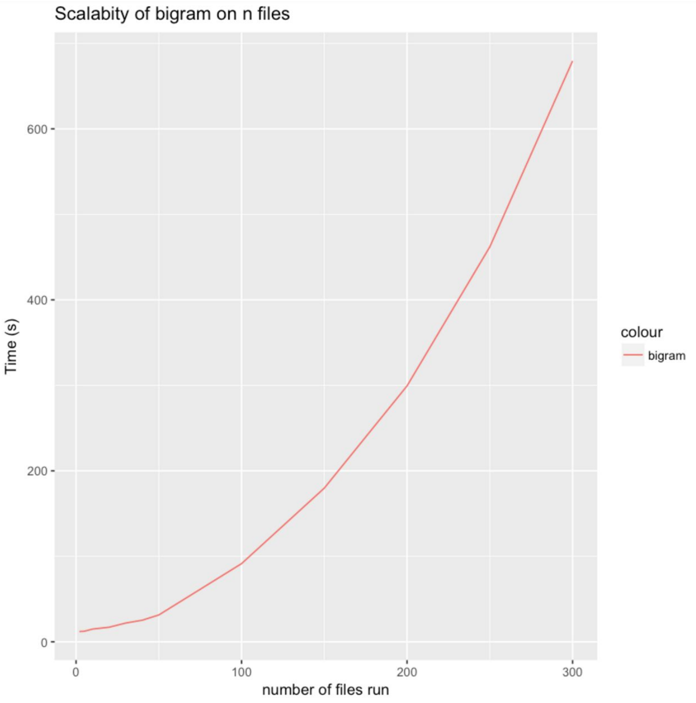
Step 7: Put the recorded values in a data frame in “Lab5_PlotForFeaturedActivity.ipynb” and run the file to plot graph.

Inference



From this plot we can infer that tri-gram co-occurrence is not scalable with increasing number of files, as, the run time is increasing exponentially, and goes as high as 31 minutes with only 50 files.

Bi-gram co-occurrence on the other hand is quite scalable, we were able to process 300 files even with a run time of 11 minutes approximately.


Scalabity of bigram on n files
Time (s)
number of files run
colour
bigram