Enron Email Analysis:

How to get the Dataset:

To download the datasetset:

Dataset has been uploaded on Google Drive:

https://drive.google.com/open?id=1fy9P52A93Z9tt2qZ43kCq1PxPCSq0FZk

The dataset can be downloaded from the website:

http://www.cs.cmu.edu/~enron/enron_mail_20150507.tar.gz

Download on to the cloudera Virtual Machine by using command:

$ wget http://www.cs.cmu.edu/~enron/enron_mail_20150507.tar.gz

Zipped Code:

Mapper1 and reducer1.py

mapper1.py to go through the unstructured data and with respect to the noted patterns and delimiters it will extract and the track the required informaiton. The output patter of this mapper will be [rank, sender1, sender2, origin, recievers] and this output will go through reducer1.py to get an output of [sender name , number of receivers]

mapper2.py and reducer2.py (output of this is for Hive)

since the dataset is unstructured and cannot to be directly used on Hive to run queries, I have used mapper2 and reducer2 to convert the unstructure data into the a format which can be uploaded to Hive. The reducer2 therefor has an output format of [messageid, sender, receivers, subject, iid]

Upload Enron data to hadoop

1. The dataset is been named as *enron-emails* and stored in a folder called *data*. First, I have renamed the folders to differentiat the sent and the inbox emails and to do so the following shell command is used:

```
$ mkdir data/enron-emails-sent
$ mkdir data/enron-emails-inbox
$ sh shell-scripts/emails-rename.sh data/enron-emails sent data/enron-emails-sent
$ sh shell-scripts/emails-rename.sh data/enron-emails inbox data/enron-emails-inbox
```

2. Uploading the above files to hdfs:

```
$ hdfs dfs -mkdir enron-sent
$ hdfs dfs -mkdir enron-inbox
$ hdfs dfs -put data/enron-emails-sent/* enron-sent
$ hdfs dfs -put data/enron-emails-inbox/* enron-inbox
```

3. To run the mapper and the reducer on hdfs:
   Since I need only the sent folder to solve my problem statement, running the mapper1 and reducer1 only on enron-sent On sent file:

```
$ hdfs dfs jar Hadoop-streaming.jar -input enron-sent -output conns-sent\
  -file mapper1.py -file reducer1.py -mapper -mapper1.py -reducer -reducer1.py
```

   For inbox file:

```
$ hdfs dfs jar Hadoop-streaming.jar -input enron-inbox -output conns-inbox\
  -file mapper1.py -file reducer1.py -mapper -mapper1.py -reducer -reducer1.py
```

   Second mapreducer2 for hive analysis:

```
$ hdfs dfs jar hadoop-streaming.jar -input enron-sent -output n-conns-sent\
  -file mapper2.py -file reducer2.py -mapper mapper2.py -reducer reducer2.py
```

4. Downloading the results:

```
$ hdfs dfs -mkdir result_mr1
$ hdfs dfs -put conns-sent/* result_mr1
$ hdfs dfs -cat result_mr1/*

$ hdfs dfs -mkdir result_mr2
$ hdfs dfs -put n-conns-sent/* result_mr2
$ hdfs dfs -cat result_mr1/*
```

For Hive:

Hive is installed in the cloudera VM provided and can start hive by typing hive in the terminal:

```
$ hive

hive>
```

Note:

Edit the jar file Hadoop-streaming-2.0.0-cdh4.2.1 to hadoop-streaming before using it on the HDFS.