# User Model Generator from Email Corpus

This document describes the needed tools, their installations, and the algorithms implemented in this library for user model generator from Enron Corpus.

## Installing and importing Enron corpus

1. Install PyDev to Eclipse Kepler (doesn't work on Eclipse Indigo)
2. Import "email modeler" project to Eclipse
3. Download Enron corpus from *https://www.cs.cmu.edu/~./enron/*
   1. unzips into the directory *maildir*
   2. move it to wherever you want it stored, say, *corpusdir*
4. pip install pymongo==2.8 #install version 2.8 of pymongo, NOT the latest version!
5. Go to Eclipse preferences, and configure Python Interpreter with "Auto-Quick" option
6. Add the directory where pymongo is stored to the paths for Python interpreter.

e.g., /Users/su11111/anaconda/lib/python2.7/site-packages/pymongo

to know where it is installed, just do "pip install pymongo==2.8" again:-)

1. Run importScript.py to load the whole MongoDB (start MongoDB by typing *mongod* on command line or do "brew info mongo", and it will tell you what command to use.

Add the .../*corpusdir/maildir* as the argument for ImportScript.py

1. Download robomongo (visual interface for MongoDB) - used v0.85

Click "create" to create a connection to MongoDB

you can review the DB content now

1. Install igraph #for viewing the output and running the algorithms (this is a bit of work...see below)
   1. brew install Caskroom/cask/xquartz brew install py2cairo
   2. Now try: brew install igraph Well, if this doesn't work and gives weird errors such as proxy errors, then: brew tap homebrew/science brew install igraph
   3. pip install python-igraph
   4. Now: try "pip install pygraphviz", if it didn't work, then...
   5. brew install graphviz, and then retry
2. Useful trivia but do read this!
   1. brew info py2cairo will give you details of the installation such as where the stuff is located. It is useful to include this in the python path in eclipse

## Running the Algorithms

Below is a list of the algorithms and their use.

|  |  |
| --- | --- |
| **Algorithm** | **Description** |
| importScript.py | Import the full Enron corpus into MongoDB |
| importScript.py\* | Import the Enron corpus parts based on different dates. |
| SetupGraphOptimized.py | Topic Modeling Algorithm with direct calls from Python |
| SetupGraphOptimizedKQL.py | Topic Modeling Algorithm with KQL calls |

## KQL Dependency

The SetupGraphOptimizedKQL.py is dependent on KQL implementation, and there fore, kql\_engine-1.0-SNAPSHOT-jar-with-dependencies.jar generated by the KQL project is copied in here.