MÔ TẢ QUÁ TRÌNH CÀI ĐẶT VÀ CHẠY CODE

I. Cài đặt

1. **Hệ điều hành sử dụng: Window 8 64-bit**
2. **Editor sử dụng: Sublime Text 3**
3. **Cài đặt và thiết lập Virtualenvironment:**

- Cài đặt thư viện virtualenv

Sử dụng virtualenv để tạo 1 môi trường isolated Python và có thể cài đặt các packages vào trong môi trường này.Tham khảo cách cài đặt tại đây: http://python-guide-pt-br.readthedocs.io/en/latest/dev/virtualenvs/

Sau khi cài đặt

```
C:\Users\duyen>virtualenv --version
15.1.0
```

- Tạo môi trường ảo trong thư mục **env** bên trong thư mục **Code**

$ cd Code
$ virtualenv env

```
C:\Users\duyen\Desktop\Code>virtualenv env
New python executable in C:\Users\duyen\Desktop\Code\env\Scripts\python.exe
Installing setuptools, pip, wheel...done.

C:\Users\duyen\Desktop\Code>
```

- Cài đặt packages

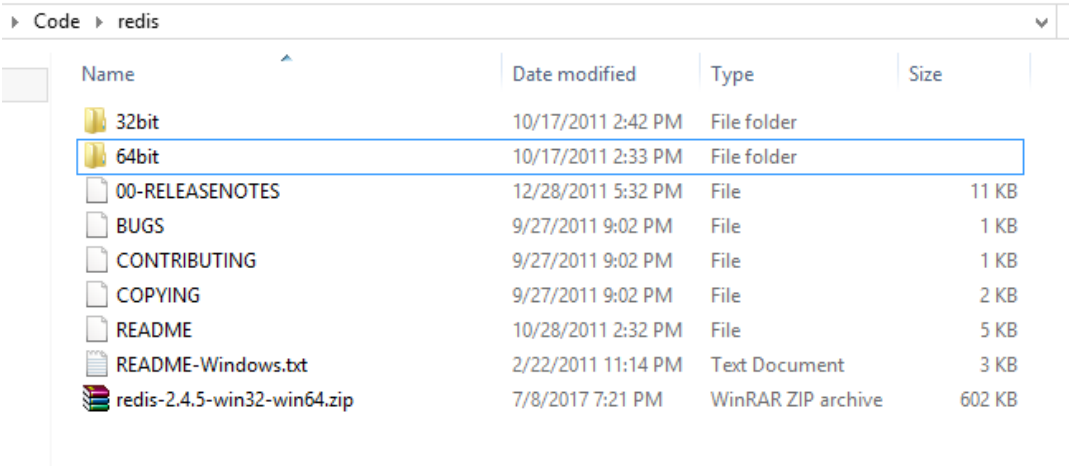
$ pip install -r requirements.txt

```
C:\Users\duyen\Desktop\Code>pip install -r requirements.txt
Collecting amqp==1.4.9 (from -r requirements.txt (line 1))
  Using cached amqp-1.4.9-py2.py3-none-any.whl
Collecting anyjson==0.3.3 (from -r requirements.txt (line 2))
Collecting Babel==2.4.0 (from -r requirements.txt (line 3))
  Using cached Babel-2.4.0-py2.py3-none-any.whl
Collecting backports.ssl-match-hostname==3.5.0.1 (from -r requirements.txt (line
 4))
Requirement already satisfied: beautifulsoup4==4.6.0 in c:\users\duyen\anaconda2
\lib\site-packages (from -r requirements.txt (line 5))
Collecting billiard==3.3.0.23 (from -r requirements.txt (line 6))
  Using cached billiard-3.3.0.23-cp27-none-win_amd64.whl
Requirement already satisfied: bs4==0.0.1 in c:\users\duyen\anaconda2\lib\site-p
ackages (from -r requirements.txt (line 7))
Collecting celery==3.1.25 (from -r requirements.txt (line 8))
  Using cached celery-3.1.25-py2.py3-none-any.whl
Collecting certifi==2017.4.17 (from -r requirements.txt (line 9))
  Using cached certifi-2017.4.17-py2.py3-none-any.whl
Requirement already satisfied: cycler==0.10.0 in c:\users\duyen\anaconda2\lib\si
te-packages (from -r requirements.txt (line 10))
Requirement already satisfied: enum-compat==0.0.2 in c:\users\duyen\anaconda2\li
b\site-packages (from -r requirements.txt (line 11))
Requirement already satisfied: enum34==1.1.6 in c:\users\duyen\anaconda2\lib\sit
e-packages (from -r requirements.txt (line 12))
Requirement already satisfied: eventlet==0.21.0 in c:\users\duyen\anaconda2\lib\
```

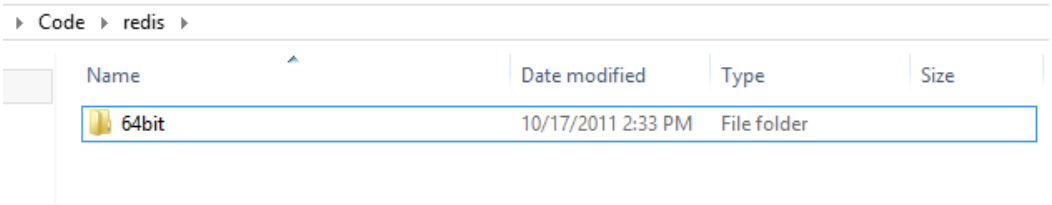
- Cài đặt redis để lưu trữ dữ liệu trên ram (kết quả của quá trình crawl dữ liệu từ Amazon)

Vào trang https://github.com/dmajkic/redis/downloads và tải file **redis-2.4.5-win32-win64.zip**

Tạo thư mục redis trong thư mục Code đặt file để chưa file redis đã tải. Giản nén file **redis-2.4.5-win32-win64.zip**

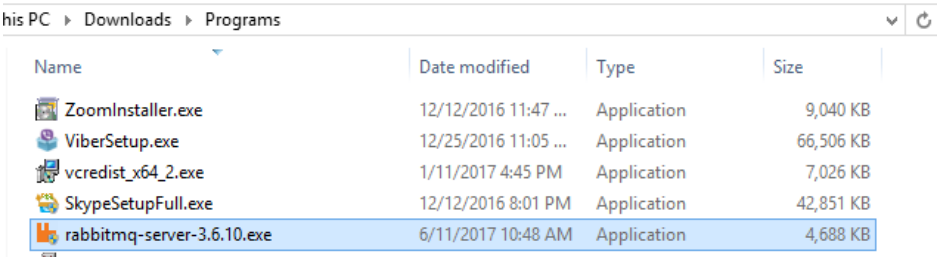


Do hệ điều hành em sử dụng là win 64bit nên em chỉ cần giữ thư mục 64bit còn lại có thể xóa cho gọn



- Cài đặt RabbitMQ để phân tán task tới các celery worker

Vào trang https://www.rabbitmq.com/install-windows.html và tải file **rabbitmq-server-3.6.10.exe**

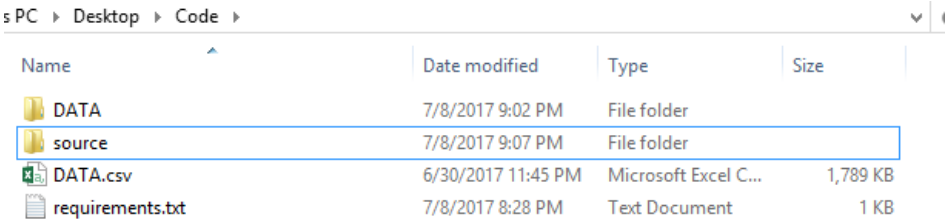


Cài đặt bằng cách click file exe và chạy

II. Thực hiện chạy code

Thư mục Code gồm:

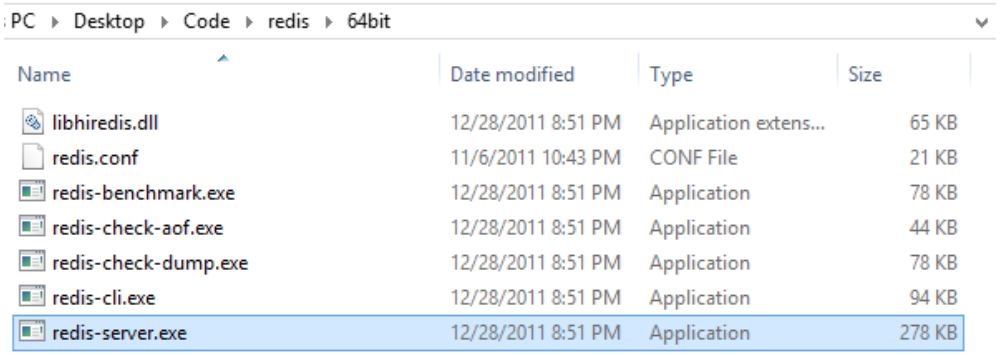
- Thư mục source chứa source code.
- Thư mục DATA chứa các thư mục dữ liệu thu thập tại cái thời gian khác nhau và chưa được tiền xử lý.
- File DATA.csv là file chứa toàn bộ dữ liệu sau khi được tổng hợp và tiền dữ liệu được sử dụng để thực hiện huấn luyện.
- File requirements.txt chứa tên các package cần thiết khi chạy.



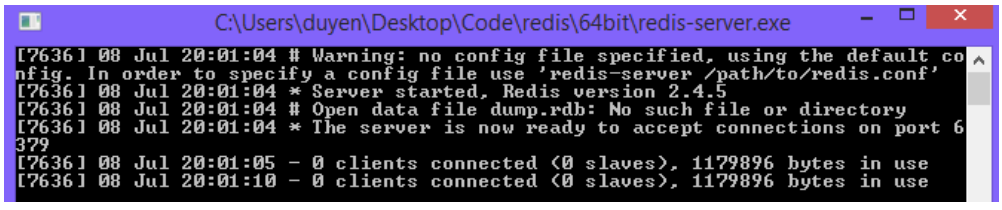
1. Thu thập dữ liệu

- Start redis-server

Vào thư mục **redis/64bit**, click file **redis-server.exe** để start redis-server



Start thành công, giữ nguyên cmd này mà mở cmd khác để start celery worker



- Vào thư mục **Code** sau đó activate môi trường ảo

  $ cd Code

  $ env\Scripts\activate

```
C:\Users\duyen\Desktop\Code>env\Scripts\activate

(env) C:\Users\duyen\Desktop\Code>_
```

- Start celery worker với task **Crawler_Worker** và **concurrency = 15**

```
(env) C:\Users\duyen\Desktop\Code>cd source

(env) C:\Users\duyen\Desktop\Code\source>celery -A Crawler_Worker worker -c 15
```

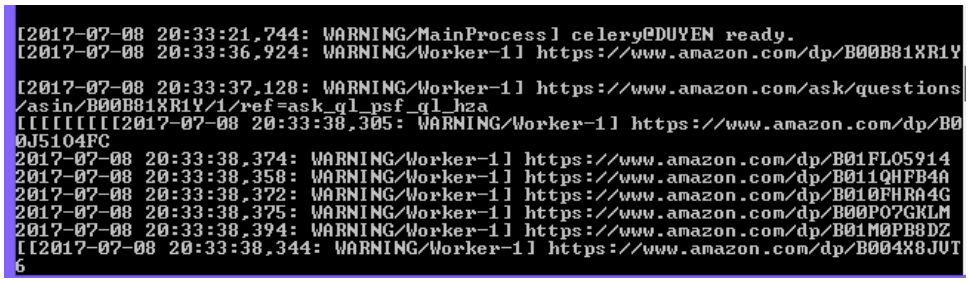
Start thành công

```
(env) C:\Users\duyen\Desktop\Code\source>celery -A Crawler_Worker worker -c 15
[2017-07-08 20:09:44,115: WARNING/MainProcess] c:\users\duyen\anaconda2\lib\site
-packages\celery\apps\worker.py:161: CDeprecationWarning:
Starting from version 3.2 Celery will refuse to accept pickle by default.

The pickle serializer is a security concern as it may give attackers
the ability to execute any command.  It's important to secure
your broker from unauthorized access when using pickle, so we think
that enabling pickle should require a deliberate action and not be
the default choice.

If you depend on pickle then you should set a setting to disable this
warning and to be sure that everything will continue working
when you upgrade to Celery 3.2::

    CELERY_ACCEPT_CONTENT = ['pickle', 'json', 'msgpack', 'yaml']

You must only enable the serializers that you will actually use.


  warnings.warn(CDeprecationWarning(W_PICKLE_DEPRECATED))

 -------------- celery@DUYEN v3.1.25 (Cipater)
---- **** -----
--- * ***  * -- Windows-8.1-6.3.9600
-- * - **** ---
- ** ---------- [config]
- ** ---------- .> app:         Crawler_Worker:0x1daa630
- ** ---------- .> transport:   amqp://guest:**@localhost:5672//
- ** ---------- .> results:     redis://localhost:6379/0
- *** --- * --- .> concurrency: 15 (prefork)
-- ******* ----
--- ***** ----- [queues]
 -------------- .> celery           exchange=celery(direct) key=celery


[2017-07-08 20:10:15,334: WARNING/MainProcess] celery@DUYEN ready.
```

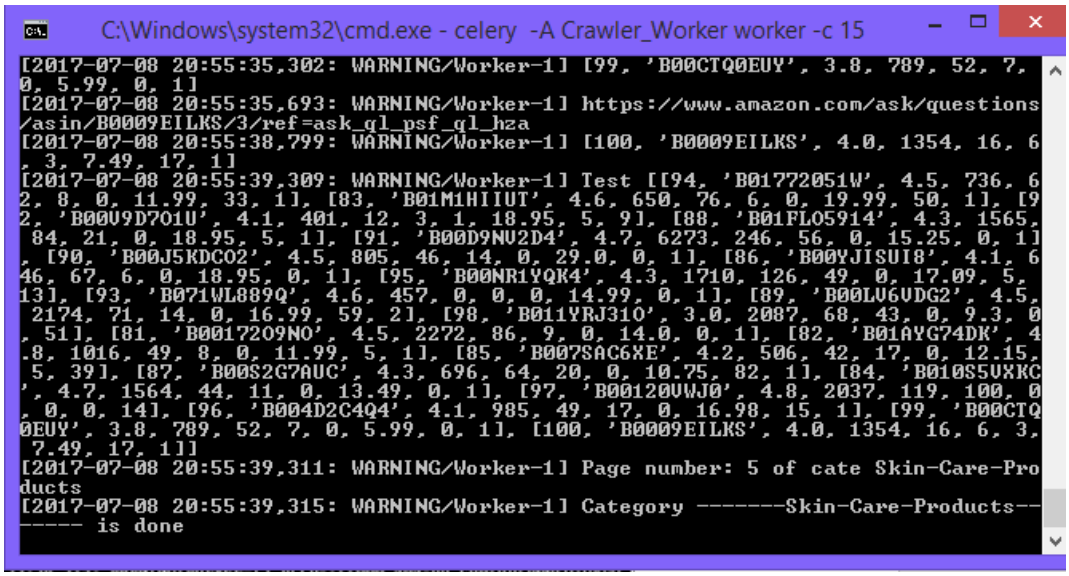
- Mở cmd khác để chạy code thu thập dữ liệu

```
C:\Users\duyen\Desktop\Code>env\Scripts\activate

(env) C:\Users\duyen\Desktop\Code>cd source

(env) C:\Users\duyen\Desktop\Code\source>python Crawler_Worker.py
```

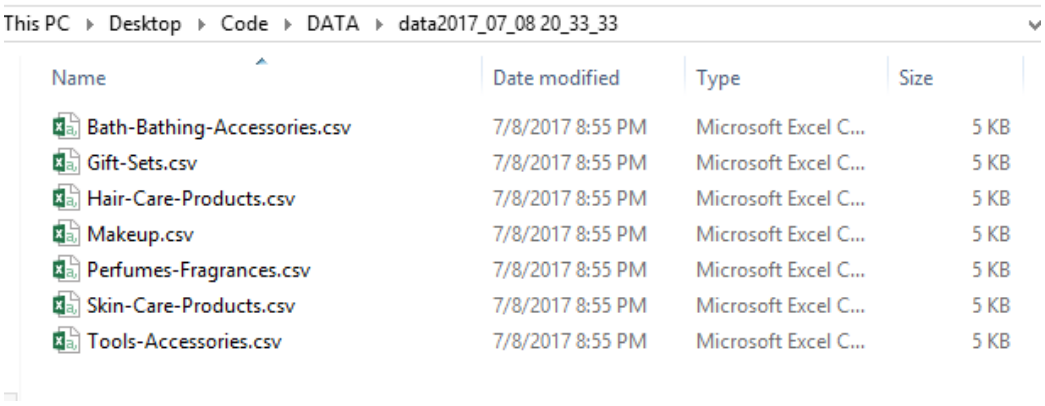
Sau đó quan sát bên cmd celery sẽ thấy bắt đầu crawl dữ liệu



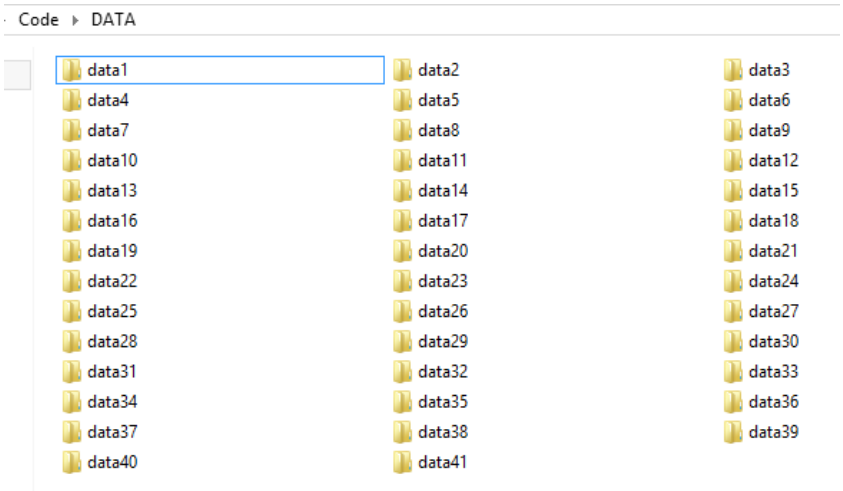
Hoàn tất thu thập dữ liệu tại 1 thời điểm



Sau khi hoàn tất crawl trong thư mục **Code** đã được tạo thư mục **DATA** chứa thư mục con data<time> chứa dữ liệu đã crawl



Em đã đánh số các thư mục 1-41 từ thứ tự thu thập dữ liệu theo thời gian thu thập trong thư mục **DATA**



2. Tổng hợp dữ liệu và tiền xử lý dữ liệu

- Chạy file **Pre_Data.py** để thực hiện quá trình tổng hợp dữ liệu và tiền xử lý.

```
(env) C:\Users\duyen\Desktop\Code\source>python Pre_Data.py
****************** count_rank_freq Folder 1******************
******************Folder 2******************
****************** count_rank_freq Folder 2******************
```

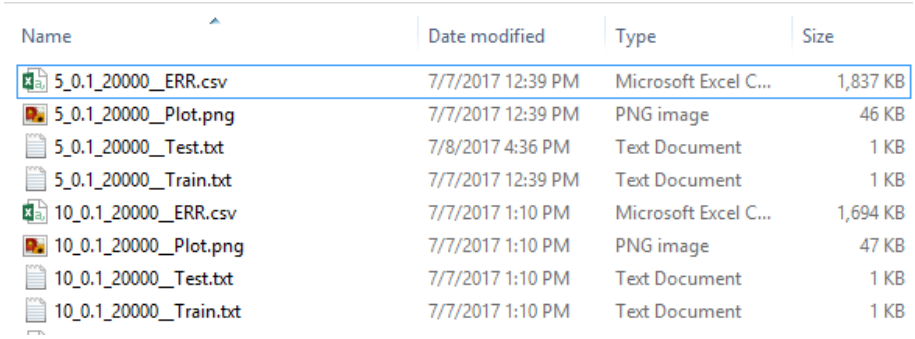
Sau khi hoàn tất file tổng hợp DATA.csv sẽ được tạo.

3. Huấn luyện

- Chạy file **Train_Data.py** để thực hiện huấn luyện

```
(env) C:\Users\duyen\Desktop\Code\source>python Train_Data.py
Epoch: 0,training err: 73.590476, val err: 74.028571,patience: 20000
Epoch: 100,training err: 35.314286, val err: 36.228571,patience: 19999
```

Sau khi hoàn tất training thư mục **RESULT** sẽ được tạo chứa cái file kết quả độ lỗi tập test (*_TEST.txt), độ lỗi tập train và val (*_Train.txt), ghi lại tất cả độ lỗi trên tập train và val trong quá trình train (*_ERR.csv) và xuất ra file đồ thị biểu diễn sự biến thiên của train error và val error (*_Plot.png). Tên file đặt dựa vào các tham số huấn luyện



Name	Date modified	Type	Size
5_0.1_20000__ERR.csv	7/7/2017 12:39 PM	Microsoft Excel C...	1,837 KB
5_0.1_20000__Plot.png	7/7/2017 12:39 PM	PNG image	46 KB
5_0.1_20000__Test.txt	7/8/2017 4:36 PM	Text Document	1 KB
5_0.1_20000__Train.txt	7/7/2017 12:39 PM	Text Document	1 KB
10_0.1_20000__ERR.csv	7/7/2017 1:10 PM	Microsoft Excel C...	1,694 KB
10_0.1_20000__Plot.png	7/7/2017 1:10 PM	PNG image	47 KB
10_0.1_20000__Test.txt	7/7/2017 1:10 PM	Text Document	1 KB
10_0.1_20000__Train.txt	7/7/2017 1:10 PM	Text Document	1 KB