Download WhoisXML API daily data ASAP part 2: An application with trending English words in .com

Posted on June 15, 2021

In the first part of this blog, we demonstrated how to download data from WhoisXML API's daily data feeds right after their publication by using the recently introduced RSS feed as the activator of download. In particular, we showed how to download the list of domains newly registered in the .com top-level domain (TLD).

Now, to make the task a bit more interesting we demonstrate the use of our domain list with a showcase application: we calculate the list of the most frequent English words in the domain names on that day. This can be interesting in various applications. Domainers, for instance, can get information on the newest trends in domain registrations. Journalists and researchers can get a clue on a topic gaining popularity, etc.

We will keep on working in the same environment as in the first part of the blog (Linux system, BASH and Python, at an intermediate level). We just extend the demonstration done in the first part, so we will be working in the "daily_words" subdirectory and assume that everything we did in the previous part is still there and working.

We will use the downloaded lists of newly registered domains in the .com top-level domain to immediately find out which are the most common English words that appear in newly registered domains. To do so, we need to do "word segmentation" (consult, e.g., this page for scientific details). We will be using a Python package called "Word Ninja" which does a good job in finding English words in strings without spaces, and is easy to install:

```
pip3 install wordninja
```

(Remember to do it in the shell where the virtual environment is active, i.e. do not forget to do

```
source ./daily_words_venv/bin/activate
```

before doing this and all that follows.) We wrote a little python program to find English words in all the strings coming from the lines of the standard input and output the top 100. This get_top_words.py reads:

```
#!/usr/bin/env python3

import sys
import re
import wordninja

words_counts = {}

for line in sys.stdin.readlines():
    for word in wordninja.split(line.strip()):
if len(word)>3 and re.search('[a-zA-Z]', word):
    try:
words_counts[word] +=1
    except KeyError:
words_counts[word] = 1
for w in sorted(words_counts, key=words_counts.get, reverse=True)[:100]:
    print(w)
```

Let's now create a subdirectory "top_100_words", and add the following three lines:

```
echo "Now postprocessing"
datehyph=$(echo $available | sed -e s/_/-/g)
./get_top_words.py < ./data/domain_names_new/com/${datehyph}/add.com.csv >
```

to get_com_daily_new.sh developed in the previous part to the end of the branch of downloading fresh data, i.e., before the line with "fi". The complete script (remember that this is the one we run from crontab) thus will read:

```
#!/bin/bash

MY_PLAN=lite
MY_KEY=MY_KEY

PROJECT_DIR="$( cd "$(dirname "$0")" >/dev/null 2>&1 ; pwd -P )"
cd $PROJECT_DIR

source ./daily_words_venv/bin/activate

(rsstail -n1 -u \ "https://${MY_KEY}:${MY_KEY}@newly-registered-domains.who
> rss_output.txt & echo $! >&3) 3>rsstail.pid

sleep 5

last_date=$(cat last_date.txt)
echo "---------------"
date
echo "Last download for day: $last_date"
available=$(tail --lines 1 < rss_output.txt | cut -d " " -f 4)
echo "Last available data for day:" $available
if [[ $last_date == $available ]];then
   echo "We already have data for $available"
else
   echo "Downloading for $available"
   datearg=$(echo $available | sed -e s/_//g)
   cd ./download_whois_data
```

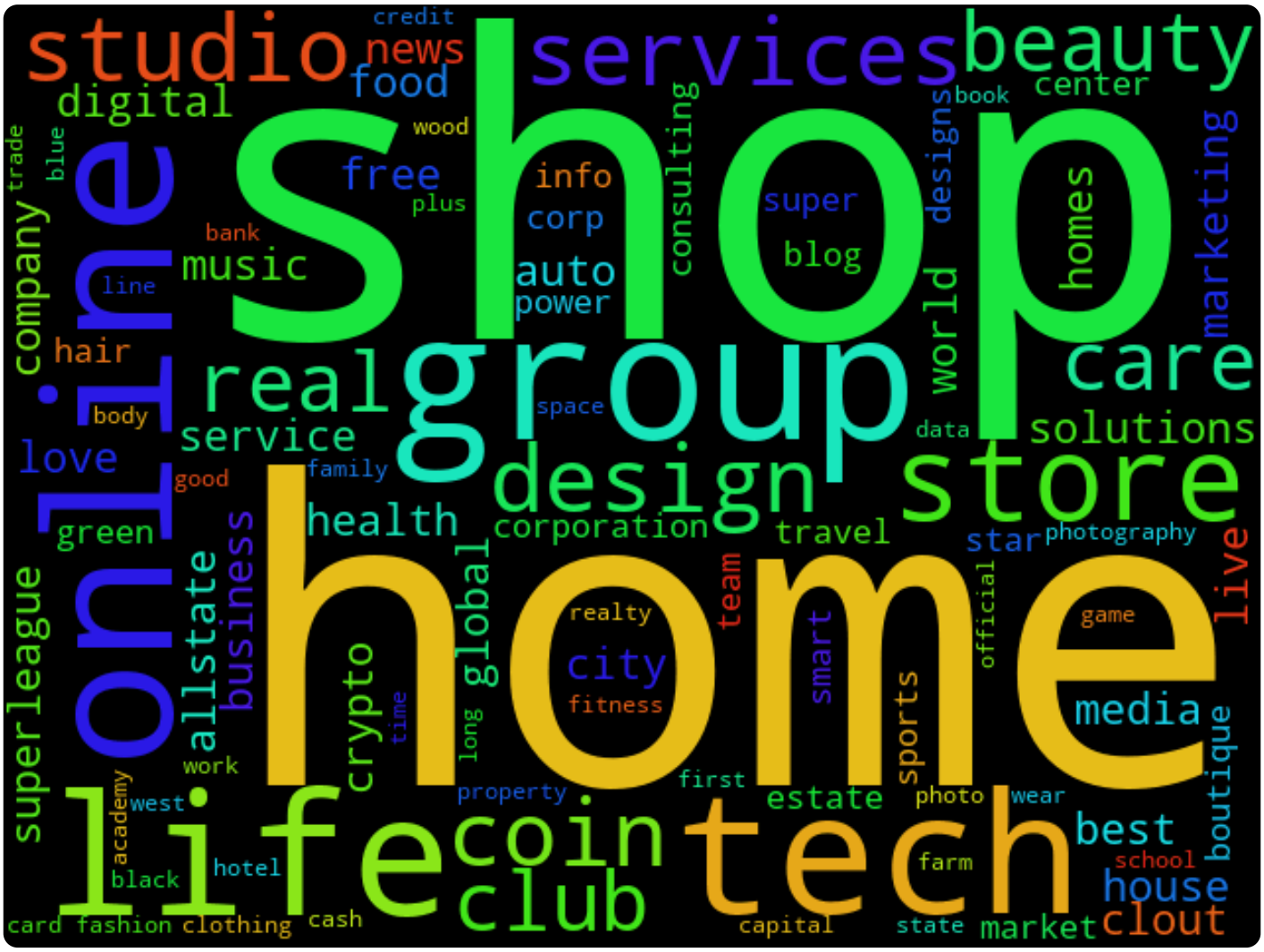
```
    ./download_whois_data.py --feed domain_names_new \
       --tlds com \
       --dataformats csv \
       --startdate $datearg \
       --plan $MY_PLAN \
       --password $MY_KEY
       --output-dir ../data
    cd ..
    echo $available > last_date.txt
    echo "Now postprocessing"
    datehyph=$(echo $available | sed -e s/_/-/g)
    ./get_top_words.py < ./data/domain_names_new/com/${datehyph}/add.com.csv
       > top_100_words_daily/com_top_100_words_${available}.txt
fi

kill $(cat rsstail.pid)
```

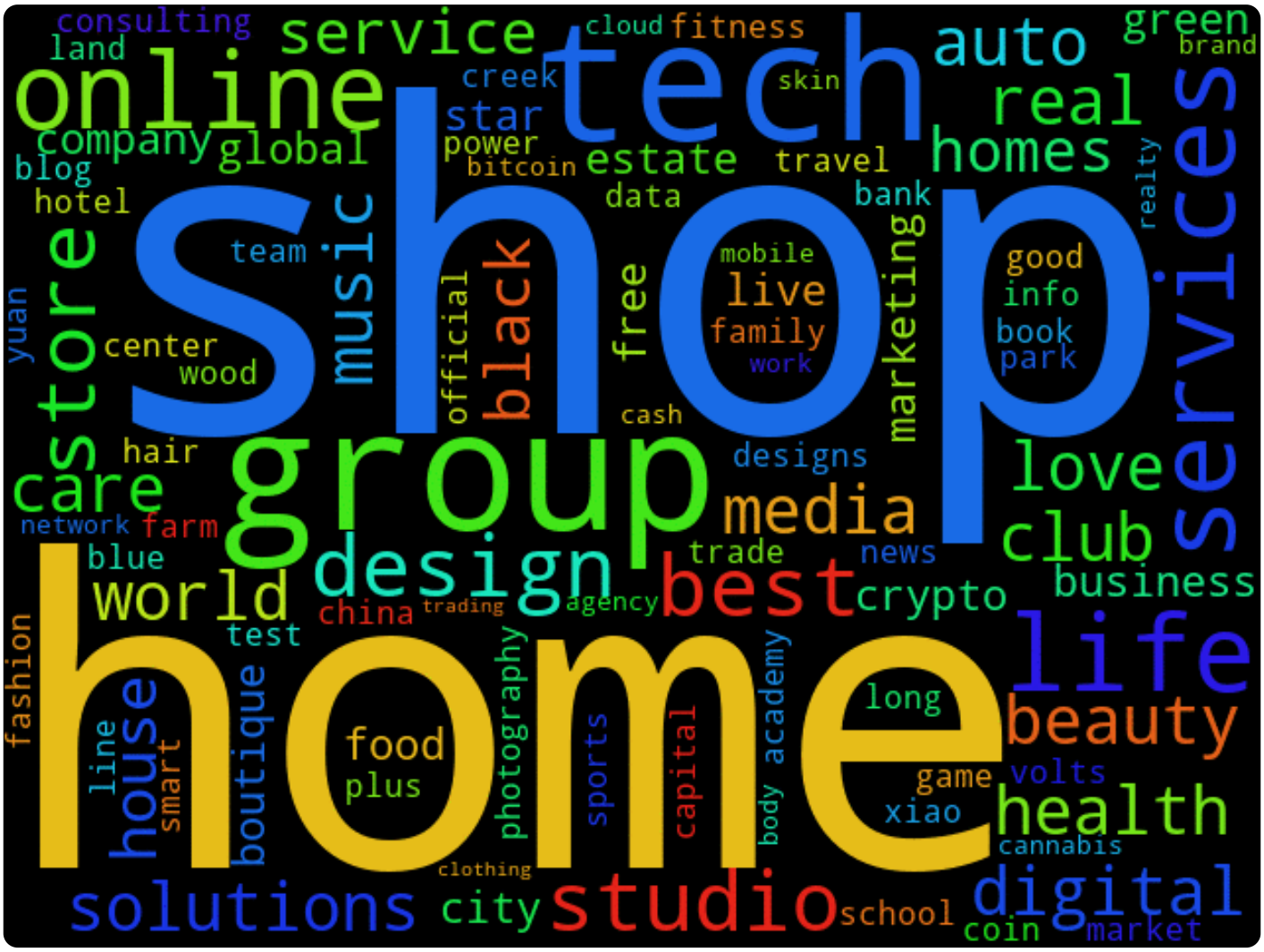
From now on, immediately after the new domain list has been published, we will find the list of the top 100 words in the files of top_100_words_daily, e.g. those for 2021-04020 will be in the file top_100_words_daily/com_top_100_words_2021_04_20.txt. We conclude our demonstration by creating a word cloud of these data:

```
pip3 install wordcloud
wordcloud_cli --width 640 --height 480 \
  --no_collocations --no_normalize_plurals \
  --text top_100_words_daily/com_top_100_words_2021_04_20.txt \
  --imagefile top_100_words_daily/com_top_100_words_2021_04_20_wordcloud.gi
```

The result is like this:



To visualize the trends, may even create an animation from these daily clouds, like the one below for 1 to 14 April 2021:


consulting
land
service
cloud
fitness
green
brand
online
creek
skin
auto
tech
real
company
global
star
power
bitcoin
estate
travel
homes
blog
hotel
data
bank
realty
shop
store
team
music
official
black
free
mobile
live
family
work
marketing
good
info
book
park
services
yuan
center
wood
cash
hair
designs
love
care
group
media
network
farm
club
blue
design
trade
news
business
world
china
trading
agency
best
crypto
test
life
fashion
home
long
beauty
line
house
smart
boutique
food
plus
photography
sports
capital
body
academy
game
volts
xiao
health
cannabis
clothing
solutions
city
studio
school
digital
coin
market