Web Scraping Google Search Results

Celina Jiang - September 13, 2018

Introduction

There are three different versions of codes where all three of them serve to acquire and store the number of search results obtained from self-generated google queries. To be more specific, the queries are clauses in the form of:

"[subject] [linking verb - to be] [adjective] because [subject] [verb]"

e.g. I am happy because I eat,
They are sad because they work

while the adjectives and verbs come from *adjectives.txt* and *verbs.txt*. The number of search results are then stored in respective results.txt files.

main.py

Description

This program first generates queries in the form of

"[subject] [linking verb - to be] [adjective] because "

e.g. I am happy because

where the adjectives come from *adjectives.txt*. Then it takes the google suggestions for the above query and google their search results respectively. The number of search results are then stored in *results_main.tx*t.

Next, the program generates queries in the form of:

"[subject] [linking verb - to be] [adjective] because [subject] [verb]"

e.g. I am happy because I eat,
They are sad because they work

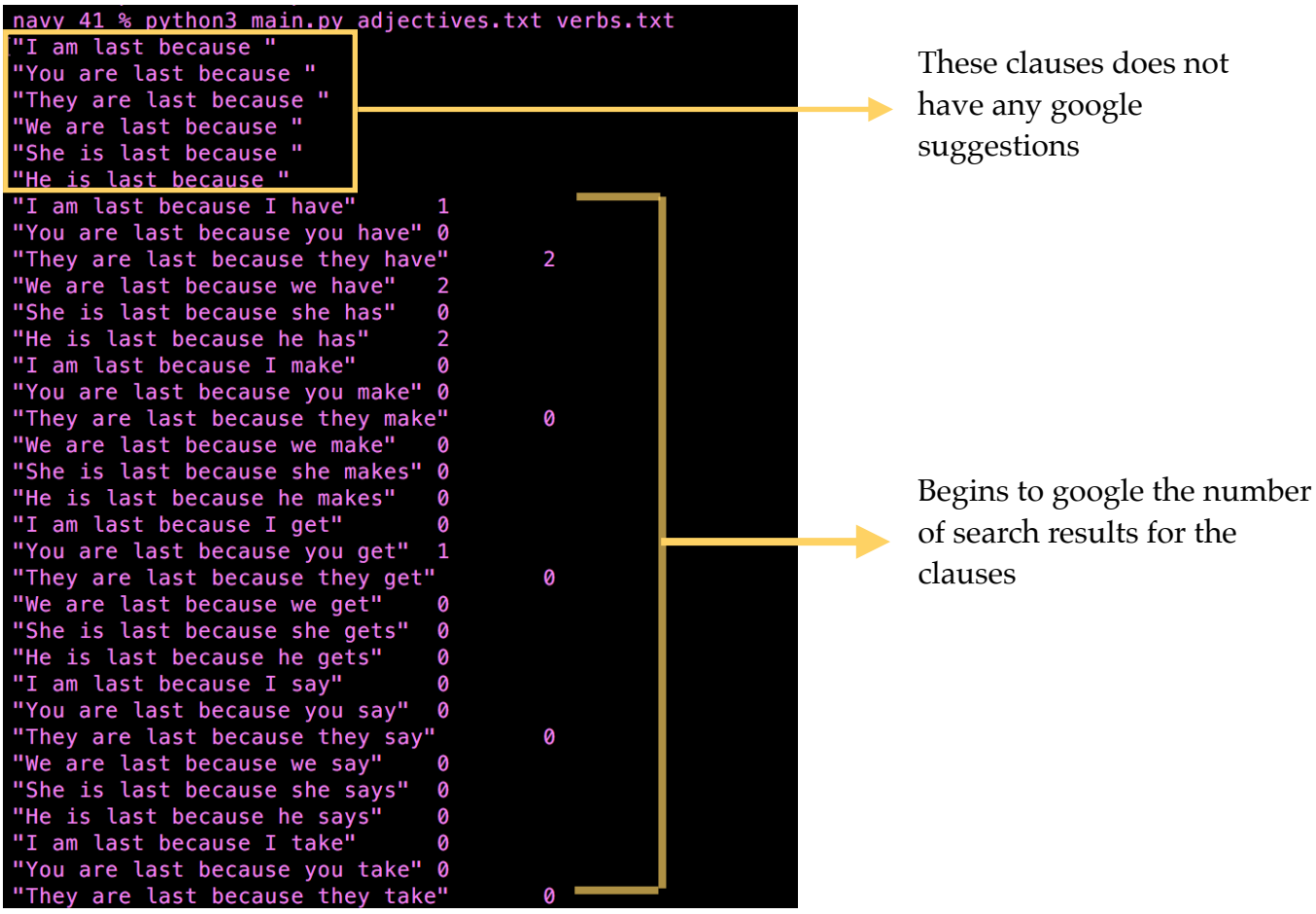
where the adjectives and verbs are taken from *adjectives.txt* and *verbs.txt*. The queries are then being searched on google, and the number of search results generated using the clauses are recorded in *results_main.txt* also.

Results

When running *main.py* by using

"python3 main.py adjectives.txt verbs.txt"

the following lines are printed to the terminal while the results from the queries are stored into *results_main.txt*.



```
"We are last because we take"   0
"She is last because she takes" 0
"He is last because he takes"   0
"I am last because I do"        0
"You are last because you do"   0
"They are last because they do" 0
"We are last because we do"     0
"She is last because she does"
WAITING FOR 60 SECONDS BEFORE RETRYING GOOGLE QUERY...
"She is last because she does"
WAITING FOR 120 SECONDS BEFORE RETRYING GOOGLE QUERY...
"She is last because she does"
WAITING FOR 180 SECONDS BEFORE RETRYING GOOGLE QUERY...
"She is last because she does"
WAITING FOR 240 SECONDS BEFORE RETRYING GOOGLE QUERY...
"She is last because she does"
WAITING FOR 300 SECONDS BEFORE RETRYING GOOGLE QUERY...
"She is last because she does"
WAITING FOR 360 SECONDS BEFORE RETRYING GOOGLE QUERY...
"She is last because she does"  0
"He is last because he does"    0
"I am last because I go"        0
"You are last because you go"   0
"They are last because they go" 0
"We are last because we go"     0
"She is last because she goes"  0
"He is last because he goes"    0
"I am last because I think"     0
"You are last because you think"        0
"They are last because they think"      0
"We are last because we think"  0
"She is last because she thinks"        0
"He is last because he thinks"  0
```

Detected by google, waiting for the captcha to go away. Waited for 21 minutes in total. After the wait, searches continue.

Overall, between each wait time, about 35-60 searches could be made in random. And the wait time varies between 21min, 28min and 36min (most of the wait times are 21 min). Hence in an hour, at least 100 searches could be made. I tried to add in random wait times that ranges from 3 seconds to 15 seconds before each google query, but it doesn't improve the number of searches since only less than 20 searches could be made before google detects scraping and blocks me. The corresponding results file, *results_main.txt* looks like:

results_main.txt

```
1  "I am last because I have"  1
2  "You are last because you have" 0
3  "They are last because they have"   2
4  "We are last because we have"   2
5  "She is last because she has"   0
6  "He is last because he has" 2
7  "I am last because I make"  0
8  "You are last because you make" 0
9  "They are last because they make"   0
10 "We are last because we make"   0
11 "She is last because she makes" 0
12 "He is last because he makes"   0
13 "I am last because I get"   0
14 "You are last because you get"  1
15 "They are last because they get"    0
16 "We are last because we get"    0
17 "She is last because she gets"  0
18 "He is last because he gets"    0
19 "I am last because I say"   0
20 "You are last because you say"  0
```

main_multi2.py

Description

This program performs exactly like *main.py* except that it uses multiple local processes to google queries concurrently. The default number of processes was set to 6, and it could be changed by modifying the code on *line 120*. Note that the search results are stored in the file *results_multi2.txt*.

Results

When running *main_multi2.py* by using

"python3 main_multi2.py adjectives.txt verbs.txt"

similar lines as *main.py* are printed to the terminal while the results from the queries are stored into *results_multi2.txt.* The two differences from *main.py* are that main_multi2.py is able to search approximately 120 to 150 clauses before it is blocked by google, but it has a longer wait time — 153 min. Hence, it is less efficient than *main.py.*

main_multi_suggestion.py

Description

This program performs similar to *main_multi2.py* in that it also runs with 6 processes. However, it only searches for recommended results of queries in the form of:

"[subject] [linking verb - to be] [adjective] because "

e.g. I am happy because

or

"[subject] [verb] because "

e.g. I eat because

The default number of processes was set to 6, and it could be changed by modifying the code on *line 121*. Note that the search results are stored in the file *results_multi_suggest.txt.*

Results

When running *main_multi2.py* by using

"python3 main_multi_suggestion.py adjectives.txt verbs.txt"

the following lines are printed to the terminal while the results from the queries are stored into *results_multi_suggest.txt.*

```
"i have because of you" 362
"you have because you ask not"  1
"i say because in french"       0
"i say because in spanish"      1
"you say because in spanish"    6
"we say because"        335,000
"i do because i can"    4,330
"i do because i couldn&#39;t care less" 588,000,000
"i do because i want"   4,740
"we do because we can"  1,850
"i go because you"      8,500
"you go because you must"       106
"i think because i am"  4,820
"i think because for example"   58
"i think because you"   13,500
"you think because you understand one"  15,900
"you think because i am poor"   15,500
"you think because you touched me"      18,900
```

Overall, between each wait time, about 50-100 searches could be made in random. And the wait time for each processes varies between 1min, 28min, 36min and even 153 min. An example is:

```
"we lose because we win analysis"       2
"we lose because we told ourselves"     8
"we serve because we love god"  1,290
"we serve because we care"      8,250
"we serve because we are saved" 783
"i am happy because of you quotes"      95
WAITING FOR 60 SECONDS BEFORE RETRYING GOOGLE QUERY...
WAITING FOR 60 SECONDS BEFORE RETRYING GOOGLE QUERY...
"i am happy because of you"     601
"i am happy because you are in my life" 9
"i am happy because i am alone" 4
"i am happy because i am single"        59
"i am happy because everyone loves me"  659
"i am happy because i choose to be"     311
WAITING FOR 60 SECONDS BEFORE RETRYING GOOGLE QUERY...
WAITING FOR 60 SECONDS BEFORE RETRYING GOOGLE QUERY...
WAITING FOR 60 SECONDS BEFORE RETRYING GOOGLE QUERY...
WAITING FOR 60 SECONDS BEFORE RETRYING GOOGLE QUERY...
WAITING FOR 120 SECONDS BEFORE RETRYING GOOGLE QUERY...
WAITING FOR 120 SECONDS BEFORE RETRYING GOOGLE QUERY...
WAITING FOR 120 SECONDS BEFORE RETRYING GOOGLE QUERY...
```