Google Trends API for Python

In this tutorial, I will demonstrate how to use the Google Trends API for getting the current trending topics on the internet.

Tanu N Prabhu Feb 29, 2020 · 5 min read ★

Image Credits: Google Trends

Introduction

(obviously). With the help of this tutorial, you can get the trending results (and many more) from Google trends website using Python. You don't need to manually search and copy the trending results, the Python API called `pytrends` does the job for you. Before getting started, I want all of you guys to go through the official documentation of the `pytrends` API.

pytrends

Unofficial API for Google Trends Allows simple interface for automating downloading of reports from Google Trends. Main...

pypi.org

Before getting started the whole code for this tutorial is available on my ***GitHub Repository*** which is given below. Please feel free to explore.

Tanu-N-Prabhu/Python

Permalink Dismiss GitHub is home to over 40 million developers working together to host and review code, manage...

github.com

Installation

The first step is to install the library manually. So, open your favorite IDE or notebook start typing the following code. I will use Google Colab because it's my favorite notebook.

```
!pip install pytrends
```

Or, if you are using an IDE, just type the following code

```
pip install pytrends
```

After executing the above code you should get a successful message as shown below:

```
Requirement already satisfied: pytrends in /usr/local/lib/python3.6/dist-packages (4.7.2)
Requirement already satisfied: requests in /usr/local/lib/python3.6/dist-packages (from pytrends) (2.21.0)
Requirement already satisfied: pandas in /usr/local/lib/python3.6/dist-packages (from pytrends) (0.25.3)
Requirement already satisfied: lxml in /usr/local/lib/python3.6/dist-packages (from pytrends) (4.2.6)
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in /usr/local/lib/python3.6/dist-packages (from requests->pytrends) (3.0.4)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.6/dist-packages (from requests->pytrends) (2019.11.28)
Requirement already satisfied: urllib3<1.25,>=1.21.1 in /usr/local/lib/python3.6/dist-packages (from requests->pytrends) (1.24.3)
Requirement already satisfied: idna<2.9,>=2.5 in /usr/local/lib/python3.6/dist-packages (from requests->pytrends) (2.8)
Requirement already satisfied: pytz>=2017.2 in /usr/local/lib/python3.6/dist-packages (from pandas->pytrends) (2018.9)
Requirement already satisfied: python-dateutil>=2.6.1 in /usr/local/lib/python3.6/dist-packages (from pandas->pytrends) (2.6.1)
Requirement already satisfied: numpy>=1.13.3 in /usr/local/lib/python3.6/dist-packages (from pandas->pytrends) (1.17.5)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.6/dist-packages (from python-dateutil>=2.6.1->pandas->pytrends) (1.12.0)
```

Library successfully installed

Implementation

Connecting to Google

You must connect to Google first because, after all, we are requesting the Google trending topics from Google Trends. For this, we need to import the method called `TrendReq` from `pytrends.request` library. Also, I will import the pandas library to store and visualize the data, which you see in the later tutorial.

```
import pandas as pd
from pytrends.request import TrendReq

pytrend = TrendReq()
```

Interest By Region

Let us see the terms which are popular in the region worldwide. I will choose, the term to be searched as "***Taylor Swift***" (I like her so….).

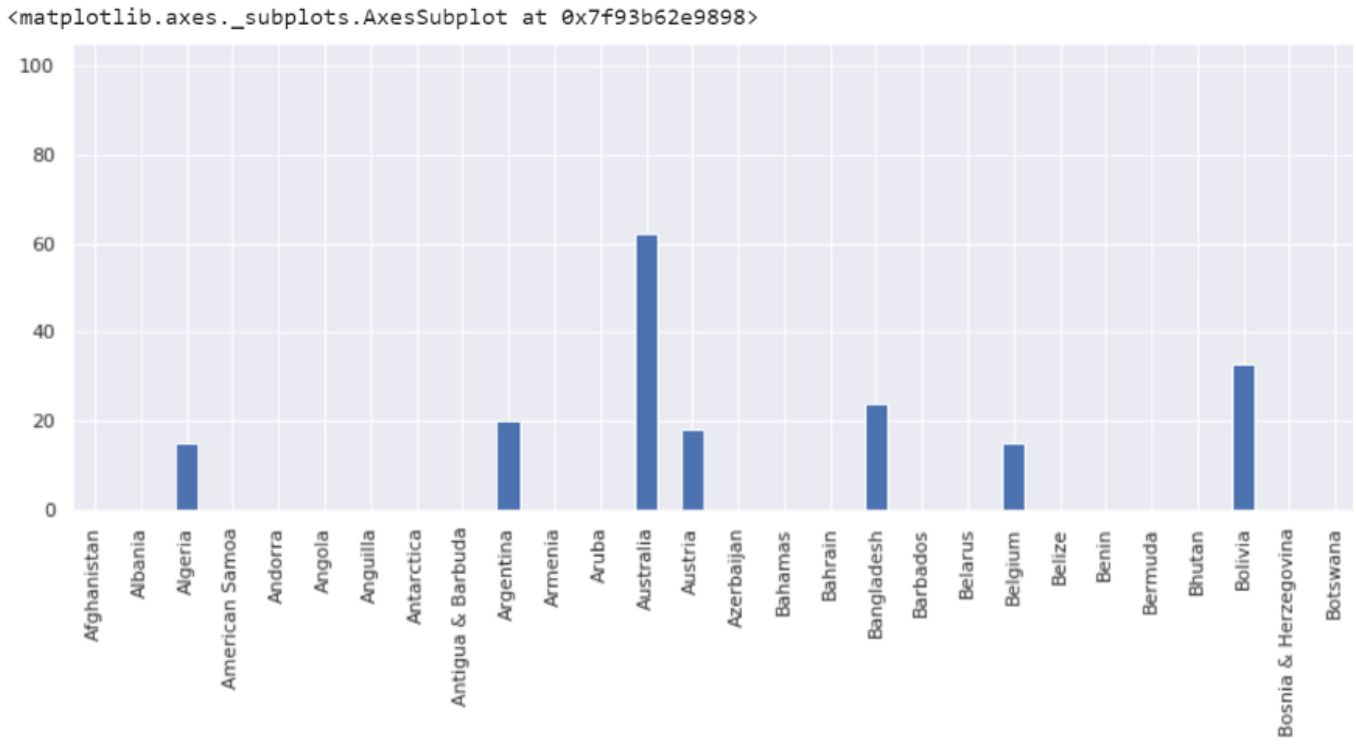
```
pytrend.build_payload(kw_list=['Taylor Swift'])

# Interest by Region
df = pytrend.interest_by_region()
df.head(10)
```

geoName	Taylor Swift
Afghanistan	0
Albania	0
Algeria	15
American Samoa	0
Andorra	0
Angola	0
Anguilla	0
Antarctica	0
Antigua & Barbuda	0
Argentina	20

Interest by region

Now you might be thinking what are the values, what do they denote? The values are calculated on a scale from 0 to 100, where 100 is the location with the most popularity as a fraction of total searches in that location, a value of 50 indicates a location which is half as popular. A value of 0 indicates a location where there was not enough data for this term. **Source** → Google Trends.

```
df.reset_index().plot(x='geoName', y='Taylor Swift', figsize=(120,
10), kind ='bar')
```



Bar plot

Also, you use the parameter `resolution = 'COUNTRY_NAME'` to filter the results.

Daily Search Trends

Now let us get the top daily search trends worldwide. To do this we have to use the `trending_searches()` method. If you want to search worldwide just don't pass any

```
# Get Google Hot Trends data
df = pytrend.trending_searches(pn='united_states')
df.head()
```

	0
0	Sir John Tenniel
1	Apple stock
2	Five Finger Death Punch F8
3	Leap year
4	Bad Bunny
5	Britt McHenry
6	Bobby Ryan
7	Kobe crash photos
8	Michael Turk
9	Barry Sanders

Trending searches

Make sure you enter the country name in lowercase `pn = "canada"` . Also, you can compare the above results with the google trend's result. To get today's trending topics just use:

```
df = pytrend.today_searches(pn='US')
```

Top Charts

Let was see what was trending in 2019. With the help of `top_charts` method we can get the top trending searches yearly.

```
df.head()
```

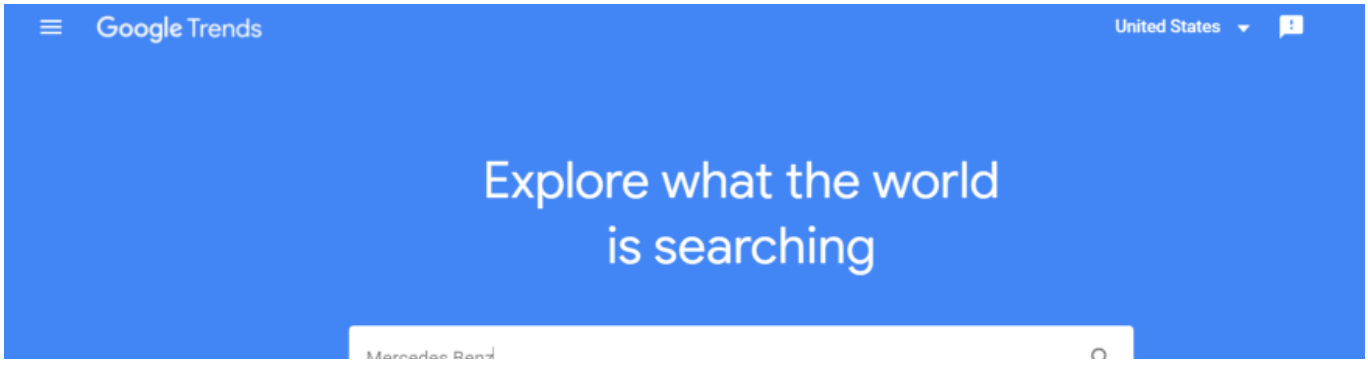
	title	exploreQuery
0	India vs South Africa	
1	Cameron Boyce	
2	Copa America	
3	Bangladesh vs India	
4	iPhone 11	
5	Game of Thrones	
6	Avengers: Endgame	
7	Joker	
8	Notre Dame	
9	ICC Cricket World Cup	

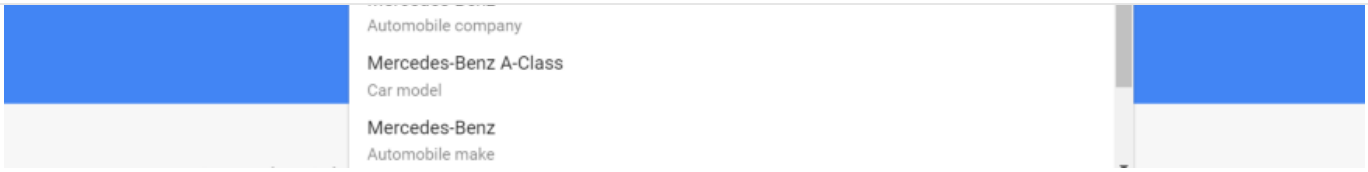
Trending titles last year (2019)

To compare the results just visit Google Trends. We can specify the year and the country that we want to see the trending searches.

Google Keyword Suggestions

Let us see how can we obtain google's keyword suggestion. If you don't know what I'm talking about. The below image explains things more clear.





Keyword Suggestions

```
# Get Google Keyword Suggestions

keywords = pytrend.suggestions(keyword='Mercedes Benz')
df = pd.DataFrame(keywords)
df.drop(columns= 'mid')   # This column makes no sense
```

	title	type
0	Mercedes-Benz	Automobile company
1	Mercedes-Benz	Automobile make
2	Mercedes-Benz A-Class	Car model
3	Mercedes-Benz G-Class	Topic
4	Mercedes-Benz C-Class	Car model

Keyword Suggestions

Related Queries

It's a common thing that when a user searches for a topic, they would also search for something related. These are called related queries. Let us see what are the related queries for the topic "***Coronavirus***". Always remember when you want to change the topic name just run the following code again with the new name as the parameter.

```
pytrend.build_payload(kw_list=['Coronavirus'])
```

```
# Related Queries, returns a dictionary of dataframes

related_queries = pytrend.related_queries()
related_queries.values()
```

```
0         virus coronavirus    100
1                     virus     99
2                     china     89
3         china coronavirus     88
4                    corona     87
5      coronavirus symptoms     81
6                  symptoms     81
7          news coronavirus     69
8              corona virus     55
9        coronavirus update     49
10       coronavirus italia     48
11         coronavirus 2020     37
12           el coronavirus     34
13                    wuhan     34
14       coronavirus wuhan     34
15       coronavirus death     31
16     what is coronavirus     31
17          coronavirus map     29
18          coronavirus usa     29
19       coronavirus cases     29
20     sintomas coronavirus     29
21           coronavirus uk     24
22  symptoms of coronavirus     23
```

Related queries to coronavirus

Similarly, you can also search for the related topics just run the below code to do so:

```
# Related Topics, returns a dictionary of dataframes

related_topic = pytrend.related_topics()
related_topic.values()
```

this is a short tutorial there is a lot to learn. Alright see you in my next tutorial, have a good day!!!

Programming Google Google Trends Python API