# Lazada Crawler

## Lazada\_spider\_shop\_api.py

* Use API, fast
* **Input**

Put in Input folder. Name as Shop.csv. File contains 3 columns: Country (sg or id), Name and Page

* **Output**

<shopname>.csv (shopee format)

* **Other Issue**

Api not very stable, put maxitems = 400 but return less than 400

* **Command to start crawling**

scrapy crawl lazada\_shop\_api

## Lazada\_spider\_category.py

* No API, relatively slow
* **Input**

Category.csv in input folder, 3 columns: Country (sg or id), Name, Page

* **Output**

Define by user

* **Command to start crawling**

scrapy crawl lazada\_category –o <filename>.csv

## Lazada\_spider\_shop.py

* No API, relatively slow
* **Input**

Same as lazada\_spider\_shop\_api.py

* **Output**

Define by user

* **Command to start crawling**

scrapy crawl lazada\_shop –o <filename>.csv

# Lazada\_crawler\_api

## Crawler.py

* Use API and request
* **Input**

Put the input file in Input folder. Name it as Shop.csv. The file contains 3 columns: Country (sg, id or my), Name, and Page. User can also choose to input seller name, country and pages directly.

* **Output**

<shopname>.csv, <shopname>.xlsx (shopee format)

* **Other Issue**

Sometimes API connection will be blocked.

# Lazada Crawl Number of SKU

* **Input**

User input country name (sg, id, my or ph)

* **Output**

<country name>\_result.csv

* **Other Issue**

Remember to put the category that you do not want to crawl in the excluding list

# Qoo10\_crawler

## Qoo10\_spider\_category.py

* **Input**

Category.csv Name, after gdmc\_cd, is a number, Page, need to scroll down to show content (in the future can use selenium)

* **Output**

Defined by user

* **Command to start crawling**

scrapy crawl qoo10\_category –o <name>.csv

* **Other Issue**

Please do not use it, as it always lose connection and cannot scroll down to view more skus.

## Qoo10\_spider\_shop.py

* **Command to start crawling**

scrapy crawl qoo10\_shop –o <name>.csv

* **Other Issue**

Please do not use it, as it cannot jump to next page. It may need to use selenium instead.

# Wish\_crawler

## Wish\_spider\_scrapy.py

Please do not use it, not complete.

## Wish\_spider\_selenium.py

* Sometimes connection is blocked, very slow
* No country-specific website and it needs to scroll down and login
* **Input**

Shop.csv in Input folder with only one column: Name

* **Output**

<shopname>.xlsx and <shopname>\_shopee\_format.xlsx, temp.csv

* **Other Issue**

Content is crawled from website javascript. In future, script may need to be changed when website javascript changes.

# Ebay-crawler

## Crawler.py

* Use API, slow
* **Input**

User input shop name, country name (full name) and number of pages to crawl

* **Output**

<shopname>\_raw.csv, <shopname>\_shopee\_format .xlsx

# Aliexpress-crawler

## Crawler.py

* **Input**

User input country name and decide whether to crawl number of reviews or not.

* **Output**

<shop name>\_temp.csv, <shop name>.xlsx, <shop name>\_shopee\_format.xlsx

* **Other Issue**

The site would randomly require login. For login, it may require you manually slide a sliding bar or input captcha

# Price-comparison

* Use Jupyter, sklearn and pandas
* Need to execute the notebook cell by cell
* First match lazada and shopee products of same shop using tf-idf cosine similarity
* After that compare their price and generate statistics
* Use Lazada shipping fee for skus in different weight to guess Lazada pricing scheme
* **Input**

Shopee and Lazada data (use crawler) of the shop

* **Output**

Single csv with matched Shopee and Lazada sku data and analytics graph.

* **Command to Start Notebook**

Open cmd, navigate to the folder, input “jupyter notebook –port=8001”

# Comment-analysis

* Use Jupyter, sklearn and pandas
* Sentimental Analysis
  + Need to annotate data first
  + **Input**

Comment Data

* + **Output**

Trained Model with accuracy statistics (can choose different model based on accuracy)

* Topic Analysis
  + Similar to Sentimental Analysis, just the data label change from sentiment to more specific topics.