

Project Report

JUN LUO



GitHub: <https://github.com/junluo1/SearchLaptop>

Introduction

In the current amazon website, when users want to search a certain kind of product that have its own specs/attributes, such as products like tv or laptops. For example, Laptops have specs like ssd storage, ram, display size, cpu, gpu, screen resolutions and etc. I find out that the current amazon search engine cannot recognize these specs users put in the query very well.

🔍 Here's an example:

When I searched " laptop **15 inch 8gb ram 256gb ssd core i7**", It's a query that contains display size, ram, ssd and cpu attributes of the laptop. Figure 1 below shows the results I get.

I also circled the specs where the product doesn't meet my requirement. You can clearly see that the top results(exclude the sponsored ones) retrieved performed really bad on meeting my requirements, despite the fact that amazon would rank products based on their business logic.

Digital Educational Resources   Hello, Jun Account & Lists Returns & Orders 

Books Livestreams Shopper Toolkit Health & Household Amazon Basics Coupons Beauty & Personal Care Jun's Amazon.com [Shop gifts for Father's Day](#)

[Sort by: Featured](#)

8gb ram 256gb ssd core i7"

Showing results from All Departments

No results for laptop 15 inch 8gb ram 256gb ssd core i7 in Digital Educational Resources



Sponsored ⓘ

HP ProBook 430 G2 Laptop - Intel Core i5 - 16 GB RAM - 1 TB SSD - WiFi - USB 3.0 Performance Notebook + Windows 10 Pro + Microsoft Office (Renewed)

★★★★★ 8

\$519⁰⁰ \$599.00

✓prime FREE Delivery Wed, Jun 9

Only 9 left in stock - order soon.

Display Size
13.30 inches

Disk Size
1 TB

RAM
8.00 GB

Operating System
Windows 10 Pro



Sponsored ⓘ

2021 HP Laptop 15.6" HD Touch Screen, 11th Gen Intel Core i5-1135G7(Beat i7-1065G7), 12GB RAM, 256GB SSD, 1TB HDD, Backlit Keyboard, HDMI, Wi-Fi, Webcam, Windows 10 | VATTE HDMI...

\$679⁹⁹

FREE Shipping

Only 7 left in stock - order soon.

Display Size
15.6 inches

Disk Size
256 GB

RAM
-

Operating System
Windows 10 Home



HP 15 Laptop, 11th Gen Intel Core i5-1135G7 Processor, 8 GB RAM, 256 GB SSD Storage, 15.6" Full HD IPS Display, Windows 10 Home, HP Fast Charge, Lightweight Design (15-dy2021nr, 2020)

★★★★★ 630

\$709⁰⁰

✓prime FREE Delivery Thu, Jun 10

Only 9 left in stock - order soon.

More Buying Choices

\$520.71 (19 used & new offers)

Display Size
15.6 inches

Disk Size
-

RAM
8 GB

Operating System
Windows 10 Home



New 2020 HP 15.6" HD Touchscreen Laptop Intel Core i7-1065G7 8GB DDR4 RAM 512GB SSD HDMI 802.11b/g/n/ac Windows 10 Silver 15-dy1771ms

★★★★★ 118

\$716⁰⁰

FREE Shipping

Only 5 left in stock - order soon.

More Buying Choices

\$599.99 (13 used & new offers)

Display Size
15.6 inches

Disk Size
512 GB

RAM
8 GB

Operating System
Windows 10



Samsung Notebook 9 15" FHD Intel i7-7500U 3.5GHz 8GB 256GB SSD Webcam Bluetooth Windows 10 Iron Silver

★★★★★ 15

\$742⁹⁸

\$7.01 shipping

Only 2 left in stock - order soon.

Display Size
15 inches

Disk Size
256 GB

RAM
8 GB

Operating System
Windows 10



Lenovo IdeaPad 3 Intel i5-1035G1 Quad Core 12GB RAM 256GB SSD 15.6-inch Touch Screen Laptop

★★★★★ 581

\$588⁰⁰ \$699.00

FREE Shipping
Only 4 left in stock - order soon.

More Buying Choices
\$479.00 (50 used & new offers)

Display Size
15.6 inches

Disk Size
256 GB

RAM
12 GB

Operating System
Windows 10



Sponsored

Used Laptop 840 G3 Portable Business Gaming KJD Computer Intel Core i5-6300U 2.4GHz Processor 8GB 256GB SSD 14.0-Inch Display High-Performance Lightweight PC Notebook with Windows 10...

★★★★★ 2

\$346⁰⁰ \$454.00

Save 6% with coupon

prime FREE Delivery Thu, Jun 10

Display Size
14 inches

Disk Size
256 GB

RAM
8.0 GB

Operating System
Windows 10



Apple Macbook Pro MJLQ2LL/A 15-inch Laptop, Intel Core i7 Processor, 16GB RAM, 256GB SSD, Mac OS X (Renewed)

★★★★★ 230

\$682⁹²

FREE Shipping

Display Size
15.4 inches

Disk Size
256 GB

RAM
16 GB

Operating System
OS X 10.10 Yosemite



Newest HP 14" HD Premium Business Laptop PC | 10th Gen Intel Quad-Core i5-1035G1 up to 3.6GHz | 8GB RAM | 256GB SSD | WiFi | HDMI | Card Reader | Bluetooth | Windows 10 | Silver

★★★★★ 191

\$581⁰⁰

FREE Delivery by Sat, Jun 12 for Prime members
Only 2 left in stock - order soon.

More Buying Choices
\$480.00 (6 used & new offers)

Climate Pledge Friendly

See 1 certification

Display Size
14 inches

Disk Size
256 GB

RAM
8 GB

Operating System
Windows 10

Figure 1 search results of query " laptop 15 inch 8gb ram 256gb ssd core i7" in amazon

I assume there wouldn't be so less products that match the keywords in my query. So I tried the manual way of finding a product to prove it, which takes more than 20 seconds, scrolling down and find out one of the checkboxes I need to check, wait for the webpage to refresh then find the next checkbox to check, since the webpage refreshes every time you clicked on a checkbox. Finally, I find out that there's at least 397 products that completely fulfill my requirements. Figure 2 below shows the search results.

Amazon Prime prime

Delivery Day Get it by Tomorrow

Department Electronics Computers & Accessories Computers & Tablets Laptops 2 in 1 Laptops Traditional Laptops

Avg. Customer Review ★★★★★ & Up ★★★★★ & Up ★★★★★ & Up ★★★★★ & Up

Featured Brands Dell Apple LG HP Lenovo ASUS Toshiba

Price Under \$500 \$500 to \$600 \$600+ to \$700

Sort by: Featured

Sponsored

2021 Flagship Dell Inspiron 15 5000 Laptop 15.6" Full HD Display
11th Gen Intel Quad-Core i7-1165G7 8GB RAM 256GB SSD...
\$829⁰⁰
✓prime FREE Delivery Thu, Jun 10
Only 19 left in stock - order soon.

Dell Inspiron 15 5000 Series 5502 Laptop i7-1165G7 Processor, 8GB, 1x8GB, 256GB M.2 PCIe NVMe Solid State Drive, 15.6-inch FHD...
\$715⁰⁰
FREE Shipping
Only 8 left in stock - order soon.
More Buying Choices
\$679.00 (22 used & new offers)

LG gram Laptop 15.6inch IPS Touchscreen, Intel 10th Gen Core i71065G7 CPU, 8GB RAM, 256GB M.2 NVMe SSD, 17 Hours Battery...
\$1,249⁹⁹
✓prime FREE Delivery Thu, Jun 10
Only 3 left in stock - order soon.

Apple MacBook Pro ME293LL/A 15.4-inch Laptop with Retina Display (OLD VERSION) (Renewed)
\$624⁰⁰
FREE Shipping
Only 18 left in stock - order soon.
More Buying Choices
\$509.73 (11 new offers)

Figure 2 search results of using filter bar for query " laptop 15 inch 8gb ram 256gb ssd core i7" in amazon

😔 Problem Identified:

If the user knows exactly the name of the product they want, it won't be so hard to find the product by typing the name in the query.

Wheres, If the user haven't decided the exact product model they want, instead they just would like to browse the pcs that meet their needs, they have to adjust the filter several times to browse the search results that meet their expectations. This is obviously much more time consuming than just type in the query and do a one-time search, given that every time you checked a checkbox on amazon website, the page will refresh again and it takes a noticeable amount of time between 0.5-1 seconds.

I want to enable users to retrieve content by searching for specific words or phrases, without needing to understand or navigate through the structure of the Web site, such as using filters or selecting categories, departments. This search engine will analyze the query and recognize the specs to accurately filter the results.

Well, to see if this proposal is possible, I tried to search the same query on google shopping. It turns out that google shopping recognize the requirements in the query very well and furthermore, it even automatically checked the checkboxes of the attributes that the search engine detected.

Figure 3 below shows what happened when I type in my query and hit enter key.

The screenshot shows a Google Shopping search results page for the query "laptop 15 inch 8gb ram 256gb ssd core i7". The search bar at the top has the query. Below it, the "Shopping" tab is selected. The left sidebar contains filters for "Buy on Google", "On sale", "Price" (Up to \$500, \$500 - \$700, \$700 - \$1,000, Over \$1,000), "Drive Type" (Solid State Drive, Hard Disk Drive), "Screen Size" (15 inches), and "Installed Memory" (8 GB RAM). The main area displays a grid of six laptop products:

- Lenovo ThinkPad E15 Laptop**: 15.6" Intel Core i7, \$649.35, Lenovo, 4.5 stars (916 reviews)
- Dell Inspiron 15 Business Laptop**: 15.6" Intel Core i7, \$679.00, Dell, 4.5 stars (14,887 reviews)
- Dell Inspiron 15 3000 Laptop**: 15.6" FHD Screen, \$399.99, Dell, 4.5 stars (14,887 reviews)
- Lenovo Ideapad 3 15.6" Touch-Screen Laptop**: 15.6" Intel Core i7, \$449.99, Best Buy, 4.5 stars (9,022 reviews)
- Lenovo IdeaPad Gaming 3i Laptop**: 15.6" Intel Core i7, \$829.99, Lenovo, 4.5 stars (31 reviews)
- Lenovo Legion Y540 Laptop**: 15.6" Intel Core i7, \$1,179.99, Lenovo, 4.5 stars (3,080 reviews)

A sponsored result for "Ace Laptop 1065g" is also visible. The page includes a "Sort by: Default" dropdown and a "Sponsored" label.

Figure 3 search results of query "laptop 15 inch 8gb ram 256gb ssd core i7" in google shopping

Implementation

1 2 3 4 Dataset

I'll use this laptop price dataset to build this project.

<https://www.kaggle.com/muhammetvarl/laptop-price>

Presumptions

The attributes of a laptop are limited, so I can utilize this feature to recognize the attributes one by one. These are the attributes laptops normally would

have based on this dataset I'm using: brand, screen size, screen resolution, OS, computer processor type(including processor speed), Computer Activity Type, RAM capacity, drive capacity, gpu type.

I'll be focusing on the laptop category, so I won't work on too much about recognizing the product name but the attributes in the query. I just assume that amazon has a list of products that got searched frequently. Once it detects what kind of product the query is about, it'll go to the dataset of this product and start to match the requirements.

I'll assume we are search in the laptop dataset.

 The implementation for this project is located in 2 main files: 'searchModel.py' and 'demoserver.py':

Step1 Analyze the dataset

In this step, I first check if there's anything that may affect performance. I noticed that some product name have parenthesis that describe the cpu. Since I plan to extract the unique values for every attribute, this would mess up the corresponding relation. For example, 'core i7' should be a unique value for attribute 'cpu', but it would also appear in the 'product' attribute list. So I deleted such parenthesis ahead.

68	HP	250 G6	Notebook
69	HP	Stream 14-AX040wm	Notebook
70	Lenovo	V310-15ISK (i5-7200U/4GB/1TB/FHD/W10)	Notebook
71	Asus	FX753VE-GC093 (i7-7700HQ/12GB/1TB/GeForce)	Gaming
72	Microsoft	Surface Laptop	Ultrabook
73	Dell	Inspiron 5370	Ultrabook

parenthesis following the product name in the third column

I used pandas module to analyze the dataset and get a list of unique values(lower case) for every attribute a laptop has. This will be used to analyze

the query.

For attribute 'Company', the unique list looks like this:

```
['Apple' 'HP' 'Acer' 'Asus' 'Dell' 'Lenovo' 'Chuwi' 'MSI' 'Microsoft'  
 'Toshiba' 'Huawei' 'Xiaomi' 'Vero' 'Razer' 'Mediacom' 'Samsung' 'Google'  
 'Fujitsu' 'LG']
```

'analyzeData' function in 'searchModel.py' does the above. It created a unique value list for each of the following attributes: "Company", "Product", "TypeName", "Inches", "ScreenResolution", "Cpu", "Ram", "Memory", "Gpu", "OpSys"

Step2 Analyze the query

I wrote a 'detectAttr' function to analyze the query, It takes 2 params, one is the query, the other is the returned value from **Step1** which is a matrix a unique values.

The function would iterate through every word in the query, match it with the value list of every attribute and returns a dictionary recording the corresponding relation of every word and its corresponding attribute.

Something like this (for query 'lenovo, 1920x1080'):

```
[{'Company': 'lenovo'}, {'ScreenResolution': '1920x1080'}]
```

Here I'll call the 'matchTitleScore' function:

1. This function compare the word with list of unique values of one attribute and calculate the similarity score for every value with the word. In the end, calculate and return the average score that the word compare to the entire list. The score is calculate by 'fuzz.token_set_ratio' method from 'fuzzywuzzy' module, it handle fuzziness to tolerant any spelling mistakes in the query.
2. During the above process, If the term got a 100 score match, 100 is return right away. Meaning that we got a complete match, this word is definitely describing the very attribute.

After match the word with every list, we got a list a scores shows how similar this word is to a list, then I pick the attribute corresponding to the max score, store the title of the attribute as the key and the word as corresponding value

into a dictionary, so that we know this word belongs to which attribute. If more than one word belong to the same attribute, I concatenate it with the current string. (such as query 'core i7', both 'core' and 'i7' would be matched to 'cpu' and finally stored in the dictionary as 'cpu': 'core i7')

To this step, we've already screened out the words that are the attribute title itself, some words like 'cpu' for example, would not be recorded in the corresponding relation dictionary and used as query text in the next step. During matching process, It won't get any match with the value list. Well, some mistake may happen, some product names have 'cpu' letters in it, but in general this doesn't happen a lot and I'll overlook the tiny problem here.

Step3 Search

From last step, I got a dictionary of which value to search in which field, In the elastic search object, I insert the csv dataset using 'csv.DictReader'. I choose to use bool search to combine the match queries so that it follows 'more-matches-is-better' and would add up the score for each match query. So I structure the query doc based on the given corresponding relation dictionary of attribute titles and query words.

For query 'lenovo 1920x1080', the dictionary and final query doc are as follows:

```
{'Company': 'lenovo', 'ScreenResolution': '1920x1080'}  
{'query': {'bool': {'should': [{('match': {'Company': 'lenovo'})}, {('match': {'ScreenResolution': '1920x1080'})}]}}}
```

Step4 Show results

The 'demoserver.py' file contains the python flask app and would render the result in the end. I defined that the retrieved size would be 30. I removed the duplicates so the results showed may be less. Since there's no such date like review or rating to effect the ranking, any results the matches the would get retrieved.

💡 How to run:

Make sure to pandas, fuzzywuzzy, elasticsearch flask modules. Simply run the 'demoserver.py' would start the web app, and user may start to make any

query. The retrieved results will show after click 'search'.

Some examples:

◆ Example 1

Make a query

```
lenovo 1920x1080 512gb  
Located attributes: Company ScreenResolution Memory
```

```
{
    "IdeaPad 320-15ABR": {
        "Company": "Lenovo",
        "Cpu": "AMD A12-Series 9720P 3.6GHz",
        "Gpu": "AMD Radeon 530",
        "Inches": "15.6",
        "Memory": "512GB SSD",
        "OpSys": "Windows 10",
        "Price_euros": "949.00",
        "Product": "IdeaPad 320-15ABR",
        "Ram": "12GB",
        "ScreenResolution": "Full HD 1920x1080",
        "TypeName": "Notebook",
        "Weight": "2.2kg",
        "laptop_ID": "598"
    },
    "Legion Y520-15IKBN": {
        "Company": "Lenovo",
        "Cpu": "Intel Core i7 7700HQ 2.8GHz",
        "Gpu": "Nvidia GeForce GTX 1060",
        "Inches": "15.6",
        "Memory": "512GB SSD",
        "OpSys": "No OS",
        "Price_euros": "1398.00",
        "Product": "Legion Y520-15IKBN",
        "Ram": "16GB",
        "ScreenResolution": "Full HD 1920x1080",
        "TypeName": "Gaming",
        "Weight": "2.4kg",
        "laptop_ID": "209"
    },
    "ThinkPad P51": {
        "Company": "Lenovo",
        "Cpu": "Intel Core i7 7700HQ 2.8GHz",
        "Gpu": "Nvidia Quadro M1200",
        "Inches": "15.6",
        "Memory": "512GB SSD",
        "OpSys": "Windows 10",
        "Price_euros": "1925.00",
        "Product": "ThinkPad P51",
        "Ram": "8GB",
        "ScreenResolution": "Full HD 1920x1080",
        "TypeName": "Workstation",
        "Weight": "2.67kg",
        "laptop_ID": "417"
    },
    "ThinkPad P51s": {
        "Company": "Lenovo",
        "Cpu": "Intel Core i7 6500U 2.5GHz",
        "Gpu": "Nvidia Quadro M520M",
        "Inches": "15.6",
        "Memory": "512GB SSD",
        "OpSys": "Windows 7",
        "Price_euros": "1855.00",
        "Product": "ThinkPad P51s",
        "Ram": "16GB",
        "ScreenResolution": "Full HD 1920x1080",
        "TypeName": "Workstation",
        "Weight": "2.18kg",
        "laptop_ID": "715"
    }
}
```

Top 4 results showed, retrieved correct results for query 'lenovo 1920×1080 512gb'

◆ Example 2

Make a query

```
core i7 512gb 12gb hp 1920x1080
```

Search

```
core i7 512gb 12gb hp 1920x1080
Located attributes: Cpu Memory Ram Company ScreenResolution
```

```
{
  "ThinkPad T460s": {
    "Company": "Lenovo",
    "Cpu": "Intel Core i7 6600U 2.6GHz",
    "Gpu": "Intel HD Graphics 520",
    "Inches": "14",
    "Memory": "512GB SSD",
    "OpSys": "Windows 7",
    "Price_euros": "2299.00",
    "Product": "ThinkPad T460s",
    "Ram": "12GB",
    "ScreenResolution": "Full HD 1920x1080",
    "TypeName": "Ultrabook",
    "Weight": "1.4kg",
    "laptop_ID": "1175"
  }
}
```

The query attribute detection is right, but there's no such matching results for query 'core i7 512gb 12gb hp 1920×1080'.