After reading the assignment, I opened some of the URLs to get a better idea of how they look. As suggested in the assignment, some of them contained furniture products, some of them were unavailable and some of the site didn't even work at all.

Before I start thinking about a solution, I like to deeply analyze my data set until I feel comfortable that I know what I am dealing with.

List of URLs that work and display a product:

https://4-chairs.com/products/mason-chair

https://pinchdesign.com/products/tables-and-desks/yves-desk

https://24estyle.com/products/eliza-fan

https://www.goodwoodfurniture.com.au/products/wardrobes/colonial-robe/

https://www.comfortfurniture.com.sg/sale/products/office

https://www.fads.co.uk/products/living/sofas/sofa-beds/

https://warmnordic.com/collections/news

https://mulamu.com/products/membership

https://big-sale-furniture.com/products/amsterdam-bench-150-x-35-be-150-35-ta

https://www.madebyhame.co.uk/products/ladder

List of URLs that work but do not display an available product:

https://www.scandesign.com/products/pavia-sectional-ivory

https://shopspencerfurnituresiouxfalls.com/products/glass-lamp

https://vincentdesign.com.au/products/spaltekanden-big-white

https://magnolialane.biz/products/anika-cushion

https://kokocollective.com.au/products/esme-natural-rattan-day-bed

https://totalpatioaccessories.com/products/product07

https://urbanfurnishing.net/products/5a-rio

https://premiumpatio.com.au/products/outdoor-tables/page/3/

List of URLs that don't work at all:

https://www.homekoncepts.com/products/furniture/tables/end-tables/

https://capsulehome.com/products/frey-sofa'%3EFrey%20Sofa%3C/a%3E%20and%20I%20wanted%20to%20share%20it%20with%20you

https://www.rockymountain.furniture/products/jofran/15003.html

Bibliography:

Since I am not very familiar with NER, I started watching a basic video to get a better understanding of how it works:

https://www.youtube.com/watch?v=2XUhKpH0p4M

https://www.youtube.com/watch?v=h4rI8v6UjV0

https://www.youtube.com/watch?v=fEU37G70SFc

1st method: Simple Lookup (add words in the vocabulary and just check if the words appear in a new search). I do not like this idea since it implies hard coding.

2nd method: Rule based NER: Find a rule for the entities that you are searching for

3rd method: Machine Learning: Use Conditional Random Fields (BERT)

First thinking:

Extract the HTML of 100 sites and store it in a dataset.

Find a way to locate the Products in these examples

Train the model on these sites.

Use the model to test new sites.

Evaluate the results.

Create a dictionary of products that contain name and counter. For each new product found, check whether it already exists in the list (thus increase the counter) or if it doesn't exist, create a new element with that name and counter 1.

When searching in a site, do not count the name of 1 product multiple times (for example here https://4-chairs.com/products/mason-chair the product 'Mason Chair' appears in the title and in the description of the item). Use a list of items every time you search an URL and add every item only once.

Ideea 1: cauta dupa titlu si headere

Sa fie <title> PRODUS </title>

<h1>

<h2>

Sa fie cu majuscula?

Ideea 2: Introdu 100 de exemple si doar da tag la produse fara a specifica vreo regula, lasa modelul sa gandeasca de acolo

Trying to find a rule for entity recognition:

https://www.youtube.com/watch?v=uj_bbI3Ao-s

Basic operations (annotation, tokens and pipelines):

https://www.youtube.com/watch?v=fsS057SNFtg

BIO training:
https://www.youtube.com/watch?v=7CRyqwCZFY0

https://towardsdatascience.com/named-entity-recognition-ner-with-bert-in-spark-nlp-874df20d1d77

I looked up free annotation tools to tag the product entities:

https://medium.com/dida-machine-learning/the-best-free-labeling-tools-for-text-annotation-in-nlp-844525c5c65b

For my case it seems that doccano would be the best choice.

Web scraping with Python BeautifulSoup

https://www.youtube.com/watch?v=QhD015WUMxE

Tried to scrape all the content from the pages, too much information to tag entities.

SEARCH FOR PRODUCT TITLE CLASS!!

If I search only for title, sometimes it doesn't count the other products displayed under the main product.

Solution: search for either product or title, then I filter it manually in doccano

Create 2 labels: Product full name and Product name (only the product name will be saved in the list)

For each site, look how the main product is displayed (until which class). Click all the links and see if the link that u clicked contains that class. If not, stop. If yes, add the link to a list and search for the title of the same product. If all the sites have already been searched, stop.

For test data: Only use the main page and tag the title entity. Add the code of recursion searching after you implement the model.

Take in consideration only the links that contain a relative path. Therefore, search for

```
href="/
```

This idea limits the effort of the model, because it does not have to search for multiple products on the same page. It limits it to finding only 1 product per page. It does not matter if the main page does or does not contain a main product (if it does not contain a main product, the model will look for recommendations and will take a product from there)

This also limits the work of entity tagging, as you only have to tag 1 entity per website, thus making the process quicker.

Another idea to be sure in the case the page does not contain a main product: let the model search 2 times for the name of the product and take the second guess as good.

All the URLs contain /product . Only look for URLs containing /product .

Bad idea. The sites that does not display any product will display products on other URLs that do not contain /product (for example https://kokocollective.com.au/products/esme-natural-rattan-day-bed

And https://vincentdesign.com.au/products/spaltekanden-big-white)

https://www.youtube.com/watch?v=teHDlOzfN-A

I need a function that looks for all the links in a URL.

A function that converts the text from html to readable text.

Forgot to do doccano task in another terminal in order to import data set. Data was infinitely importing and I thought it was something wrong with my data. I tried creating a new empty data and same results. Googled up the problem and found the solution that I forgot to que doccano task.

Debugged sites that work (connection successful) but there is no text displayed.

At first I will see how the model does with only tagging the Full Product Name and Product. If it doesn't perform well, I will also tag the description and the price to give an indicator of where the product should be located and also to differentiate the products from simple title sites.

Installing sparknlp:
https://github.com/JohnSnowLabs/spark-nlp/discussions/1022%20GitHubGitHub%20How%20to%20correctly%20install%20Spark%20NLP%20on%20Windows%208%20and%2010%20%C2%B7%20Discussion%20#1022%20%C2%B7%20JohnSnowLabs/spark-nlp%20State%20of%20the%20Art%20Natural%20Language%20Processing.%20Contribute%20to%20JohnSnowLabs/spark-nlp%20development%20by%20creating%20an%20account%20on%20GitHub.%20Veysel%20%2010%20minutes%20ago%20And%20this%20is%20how%20it%20is%20being%20done%20in%20Docker%20https://github.com/JohnSnowLabs/spark-nlp/discussions/1714

Found this tutorial and started the sparknlp session in colab

https://www.youtube.com/watch?v=F2ph02HWWAo