Leveraging Deep Learning for Product Similarity Analysis

A case study for PhD job position assessment
at Technical University of Denmark

Presented by:
Nastaran Moradzadeh

TABLE OF CONTENTS

01 Introduction & Problem Definition

02 Data Analysis

03 Data Processing

04 Deep Learning Models

05 Implementation & Results

06 CONCLUSION & More Ideas

Introduction

Recommender systems

A software tool
Suggesting personalized recommendations based on user behavior or input

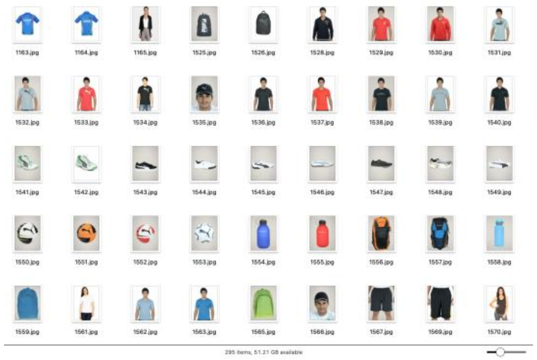
Our Problem

Outfit recommender system
Traditional systems: based on user history (content-based) and users similarities (collaborative-based)
Here: based on image uploaded by user

Data Resources

Product Images

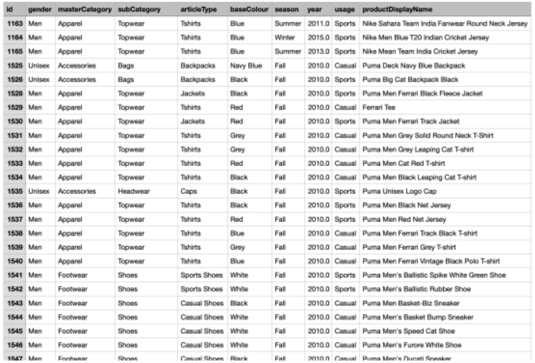
Includes 295 images



295 items, 51.21 GB available

Product Labels

Includes 9 categorical columns



id	gender	masterCategory	subCategory	articleType	baseColour	season	year	usage	productDisplayName
1163	Men	Apparel	Topwear	Tshirts	Blue	Summer	2011.0	Sports	Nike Sahara Team India Fanwear Round Neck Jersey
1164	Men	Apparel	Topwear	Tshirts	Blue	Winter	2015.0	Sports	Nike Men Blue T20 Indian Cricket Jersey
1165	Men	Apparel	Topwear	Tshirts	Blue	Summer	2013.0	Sports	Nike Mean Team India Cricket Jersey
1525	Unisex	Accessories	Bags	Backpacks	Navy Blue	Fall	2010.0	Casual	Puma Deck Navy Blue Backpack
1526	Unisex	Accessories	Bags	Backpacks	Black	Fall	2010.0	Sports	Puma Big Cat Backpack Black
1528	Men	Apparel	Topwear	Jackets	Black	Fall	2010.0	Sports	Puma Men Ferrari Black Fleece Jacket
1529	Men	Apparel	Topwear	Tshirts	Red	Fall	2010.0	Casual	Ferrari Tee
1530	Men	Apparel	Topwear	Jackets	Red	Fall	2010.0	Sports	Puma Men Ferrari Track Jacket
1531	Men	Apparel	Topwear	Tshirts	Grey	Fall	2010.0	Casual	Puma Men Grey Solid Round Neck T-Shirt
1532	Men	Apparel	Topwear	Tshirts	Grey	Fall	2010.0	Casual	Puma Men Grey Leaping Cat T-shirt
1533	Men	Apparel	Topwear	Tshirts	Red	Fall	2010.0	Casual	Puma Men Cat Red T-shirt
1534	Men	Apparel	Topwear	Tshirts	Black	Fall	2010.0	Casual	Puma Men Black Leaping Cat T-shirt
1535	Unisex	Accessories	Headwear	Caps	Black	Fall	2010.0	Sports	Puma Unisex Logo Cap
1536	Men	Apparel	Topwear	Tshirts	Black	Fall	2010.0	Sports	Puma Men Black Net Jersey
1537	Men	Apparel	Topwear	Tshirts	Red	Fall	2010.0	Sports	Puma Men Red Net Jersey
1538	Men	Apparel	Topwear	Tshirts	Blue	Fall	2010.0	Casual	Puma Men Ferrari Track Black T-shirt
1539	Men	Apparel	Topwear	Tshirts	Grey	Fall	2010.0	Casual	Puma Men Ferrari Grey T-shirt
1540	Men	Apparel	Topwear	Tshirts	Blue	Fall	2010.0	Casual	Puma Men Ferrari Vintage Black Polo T-shirt
1541	Men	Footwear	Shoes	Sports Shoes	White	Fall	2010.0	Sports	Puma Men's Ballistic Spike White Green Shoe
1542	Men	Footwear	Shoes	Sports Shoes	White	Fall	2010.0	Sports	Puma Men's Ballistic Rubber Shoe
1543	Men	Footwear	Shoes	Casual Shoes	Black	Fall	2010.0	Casual	Puma Men Basket-Biz Sneaker
1544	Men	Footwear	Shoes	Casual Shoes	White	Fall	2010.0	Casual	Puma Men's Basket Bump Sneaker
1545	Men	Footwear	Shoes	Casual Shoes	White	Fall	2010.0	Casual	Puma Men's Speed Cat Shoe
1546	Men	Footwear	Shoes	Casual Shoes	White	Fall	2010.0	Casual	Puma Men's Furore White Shoe
1547	Men	Footwear	Shoes	Casual Shoes	Black	Fall	2010.0	Casual	Puma Men's Ducati Sneaker

User Input Image

New clothing photo uploaded by the user

Product Labels Details

gender

Men 201
Unisex 51
Women 43

masterCategory

Apparel 182
Footwear 61
Accessories 45
Sporting Goods 7

subCategory

Topwear 154
Shoes 59
Bags 34
Bottomwear 28
Sports Equipment 7
Headwear 5
Water Bottle 5
Sandal 2
Accessories 1

articleType

Tshirts 139
Sports Shoes 43
Backpacks 28
Casual Shoes 16
Shorts 13
Track Pants 11
Jackets 10
Footballs 6
Water Bottle 6
Caps 5
Handbags 5
Swimwear 4
Sweatshirts 4
Sandals 2
Basketballs 1
Duffel Bag 1
Tops 1

baseColour

Black 64
White 63
Blue 49
Grey 29
Red 26
Green 15
Navy Blue 11
Brown 10
Pink 7
Purple 6
Yellow 6
Orange 3
Silver 2
Beige 2
Maroon 1
Cream 1

season

Fall 187
Summer 69
Winter 31
Spring 8

year

2010.0 216
2011.0 70
2012.0 5
2015.0 2
2013.0 1
2016.0 1

usage

Sports 170
Casual 117
Travel 8

Data Analysis

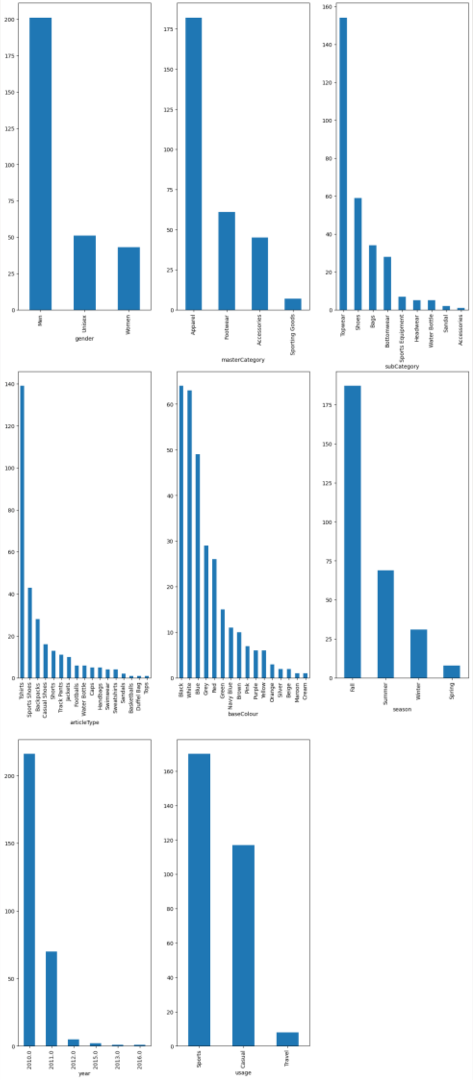
Histograms

Labels histograms are drawn
Data is not balanced across almost all labels

Clustering

Product images are clustered
Based on various feature embedding extractors

This method is used to evaluate feature embeddings



Data Processing (Product images)

Step 1

Resize input
to 256*256

Step 2

Crop center of resized image in 224 * 224

Step 3

Converion to tensor

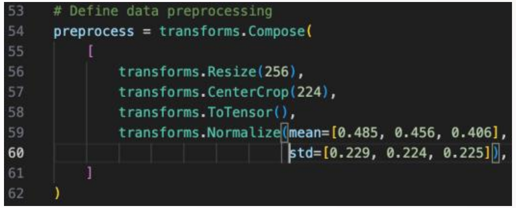
Step 4

Normalization

Data Processing (User Input Image)

Step 0

Preprocess such as:
clothes detection
alignment
background removal



```
# Define data preprocessing
preprocess = transforms.Compose(
    [
        transforms.Resize(256),
        transforms.CenterCrop(224),
        transforms.ToTensor(),
        transforms.Normalize(mean=[0.485, 0.456, 0.406],
                             std=[0.229, 0.224, 0.225]),
    ]
)
```

Step 1-4

Same as product images processing

Proposed Solution

Step 1

Extract feature embeddings from product images using deep learning.

Save the extracted features for subsequent utilization.

Step 2

A web application is implemented to facilitate the upload of new images and display similar products.

Step 3

The uploaded image is processed, and its feature embedding is extracted.

Using a similarity metric, similar products are retrieved.

Step 4

To enhance the results, the labels of products are utilized to identify less probable items.

Label voting is conducted among selected items,
those with a label different from the computed label are marked accordingly.

Results are presented to the user

Main Challenges

1 User input photo preprocessing such as clothes detection, alignment and background removal

2 Feature extractor model -> dedicated model in outfit area will have better performance

Simplifying assumptions!

The system input photos are well-formed, featuring a single outfit and devoid of complex backgrounds.

Additionally, pretrained neural networks trained on the ImageNet dataset are employed as feature extractors.

Deep Learning Models

ResNet50 & ResNet18

Both resnet50 and resnet18 models performed well as feature extractors

Evaluation method

Subjective tests based on personal opinion.
Clustering results

VGG16

VGG16 model did not perform well as a feature extractors *

so is not utilized

Final Model

To further improve the results both ResNet50 and ResNet18 feature vectors are concatenated *

* In the notebook file, the results are presented.

Implementation Details

Utilized Tools

PyTorch: Deep learning models

Dash – Plotly: Web app implementation

Docker: Containerization for deployment and scalability

Uploaded files

Are saved!

Feature Embeddings

Have been saved!

To prevent re-computation upon each restart.

Sample input images

Are prepared and uploaded in the github

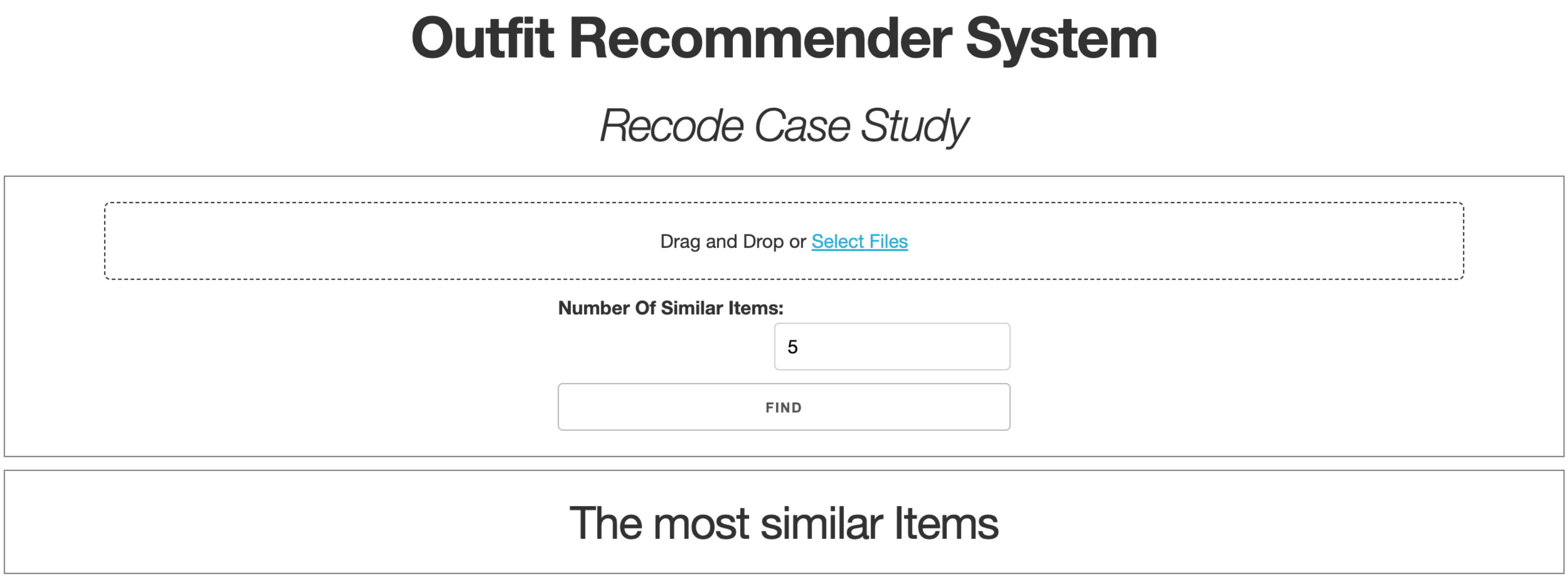
Similarity metric

Cosine similarity

Further results enhancement

The 'articleType' label is utilized.

Overview of the Web App



Outfit Recommender System

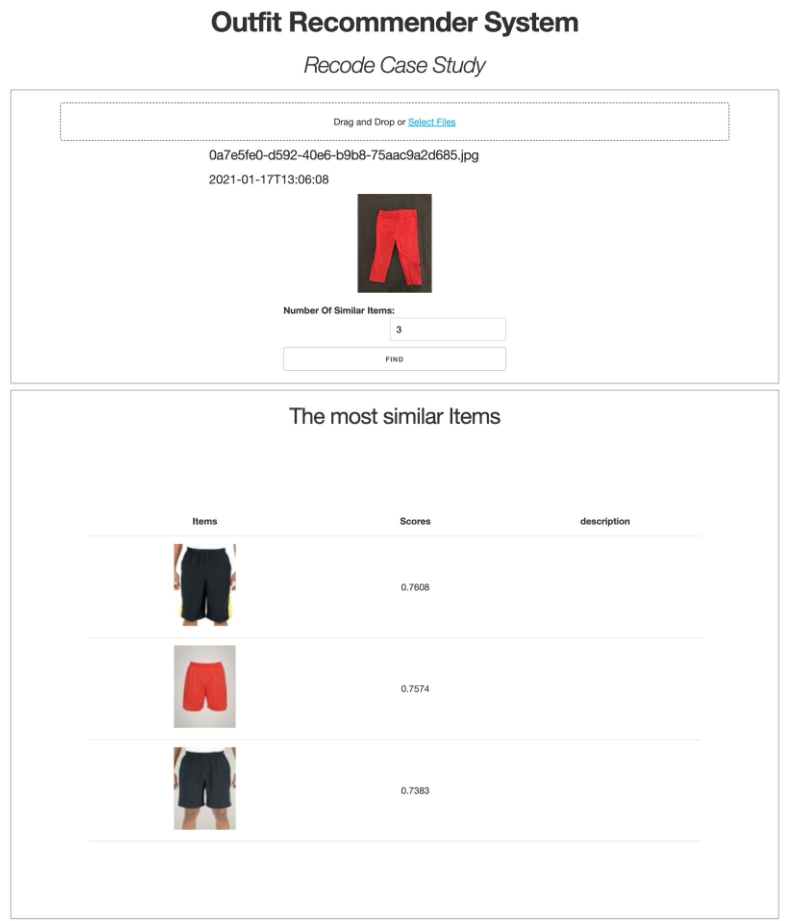
Recode Case Study

Drag and Drop or Select Files

Number Of Similar Items: 5

The most similar Items

Results



Outfit Recommender System

Recode Case Study

Drag and Drop or Select Files

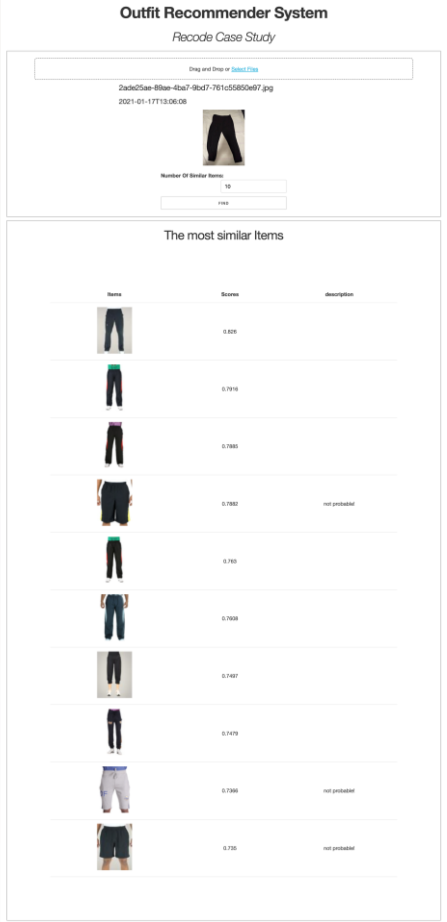
0a7e5fe0-d592-40e6-b9b8-75aac9a2d685.jpg

2021-01-17T13:06:08

Number Of Similar Items: 3

The most similar Items

Items	Scores	description
	0.7608	
	0.7574	
	0.7383	



Outfit Recommender System

Recode Case Study

Drag and Drop or Select Files

2ade25ae-89ae-4ba7-9bd7-761c55850e97.jpg

2021-01-17T13:06:08

Number Of Similar Items: 10

The most similar Items

Items	Scores	description
	0.826	
	0.7916	
	0.7885	
	0.7882	not probable!
	0.763	
	0.7608	
	0.7497	
	0.7479	
	0.7366	not probable!
	0.735	not probable!

Conclusion & more Ideas

Conclusion

If the input is sufficiently well-formed and similar products are present in the dataset, the final recommended items are desirable.

More ideas

1- Capturing user's interested labels and categories can enhance performance.

2- Recommending items from related categories can be beneficial. For instance, if a user uploads a blue T-shirt, recommending a blue cap or navy jeans can complement their outfit.

3- Suggest complementary accessories, such as belts, bags, or sunglasses, based on the uploaded item.

4- Offer variations of the uploaded item, such as different colors, patterns, or styles, to provide options for the user.

5- Include options for user feedback or customization preferences to further refine recommendations and improve user experience.

RESOURCES

[1] https://pytorch.org/vision/0.8/models.html

[2] https://github.com/alexeygrigorev/clothing-dataset

THANKS!

any questions?