Duplicates Detection Project Review by Maulik Patel

Overview:-

Due to hardware limitation (Colab-12GB RAM), I was able run code on **half dataset only (Dataset.csv size = 5GB)**

O(N) :- ~25N

Subcategory:- Tunics

Biggest challenge:- Execute code having ~12GB RAM GPU limitation

Result:- Successfully able to detect duplicate/different items for following cases

1. **Duplicate** products having same Image
2. **Duplicate** Products having different color
3. **Duplicate** products having same texture, different color and/or different models(girls)
4. **Duplicate** products having different color and/or different pose
5. Different products almost same look like, but actually **different**

Fields used:- Images, Brand, Sleeve Type

Scope for Improvement:- we can use title, description & neck type field, which will help to improve model performance

About My Algorithm:-

When we download data from e-commerce site, mostly we don’t have labeled dataset. So I have used Pre-trained keras models (DenseNet & ResNet). These models basically used as feature extractor.

For any 2 images, feature vectors are calculated. These vectors used to calculate cosine similarity. If cosine similarity > similarity_threshold, we can say given 2 images are duplicates.

Que. Why two architectures DenseNet & ResNet are used?
Ans- Different model architectures are sensitive to different characteristics of images. Initially I was planning to use 3-4 models, but dropped plan because of 12GB RAM limitation

I have to do ~25N comparisons instead of N*N, thanks to unique_category logic. unique_category is classification of products based on brand & sleeve. So instead of comparing each image pairs, I have compared products/items fall under same "Unique Category"

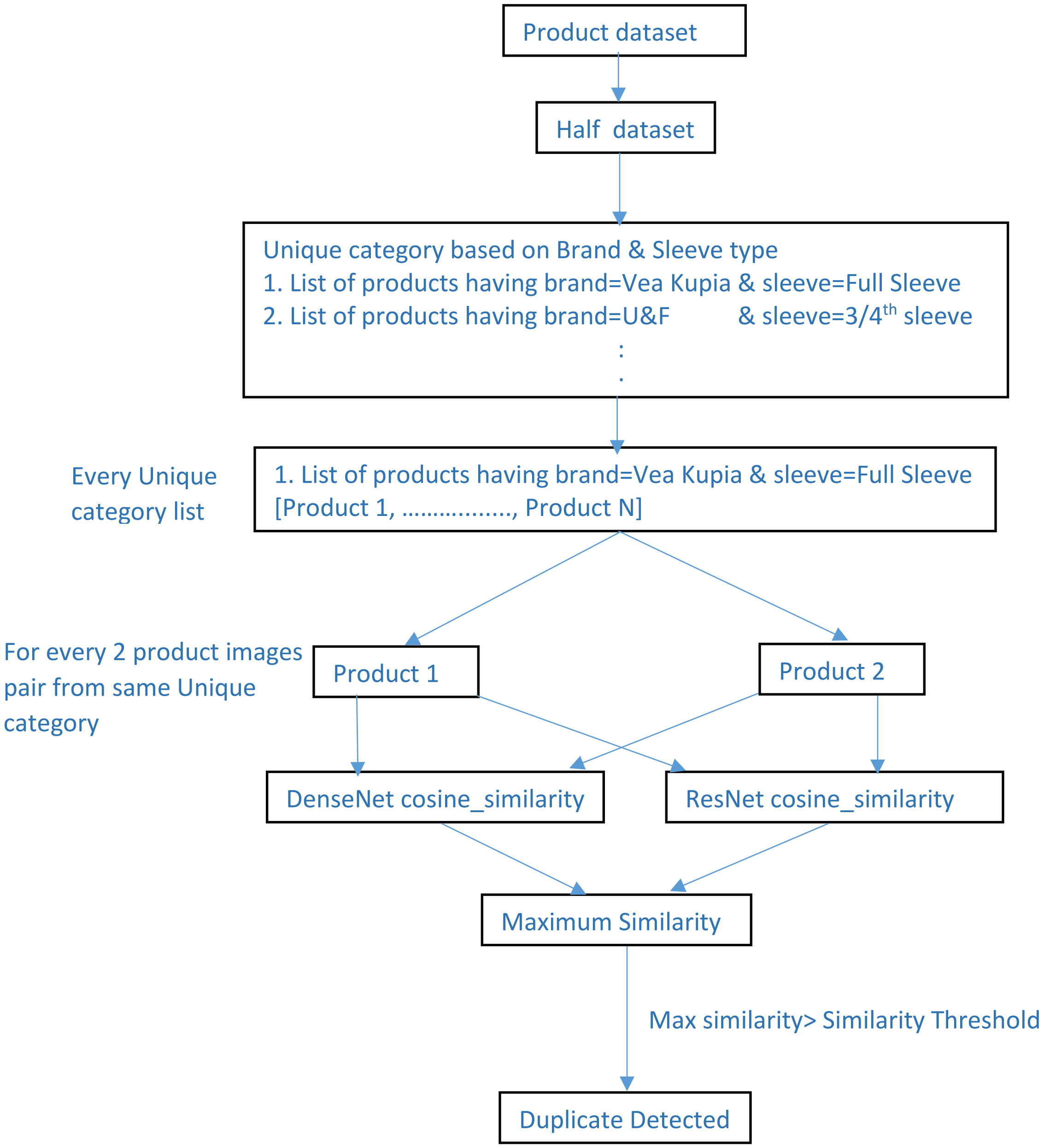

Product dataset
Half dataset
Unique category based on Brand & Sleeve type
1. List of products having brand=Vea Kupia & sleeve=Full Sleeve
2. List of products having brand=U&F & sleeve=3/4th sleeve
:
.
Every Unique category list
1. List of products having brand=Vea Kupia & sleeve=Full Sleeve
[Product 1, ……………., Product N]
For every 2 product images pair from same Unique category
Product 1
Product 2
DenseNet cosine_similarity
ResNet cosine_similarity
Maximum Similarity
Max similarity> Similarity Threshold
Duplicate Detected

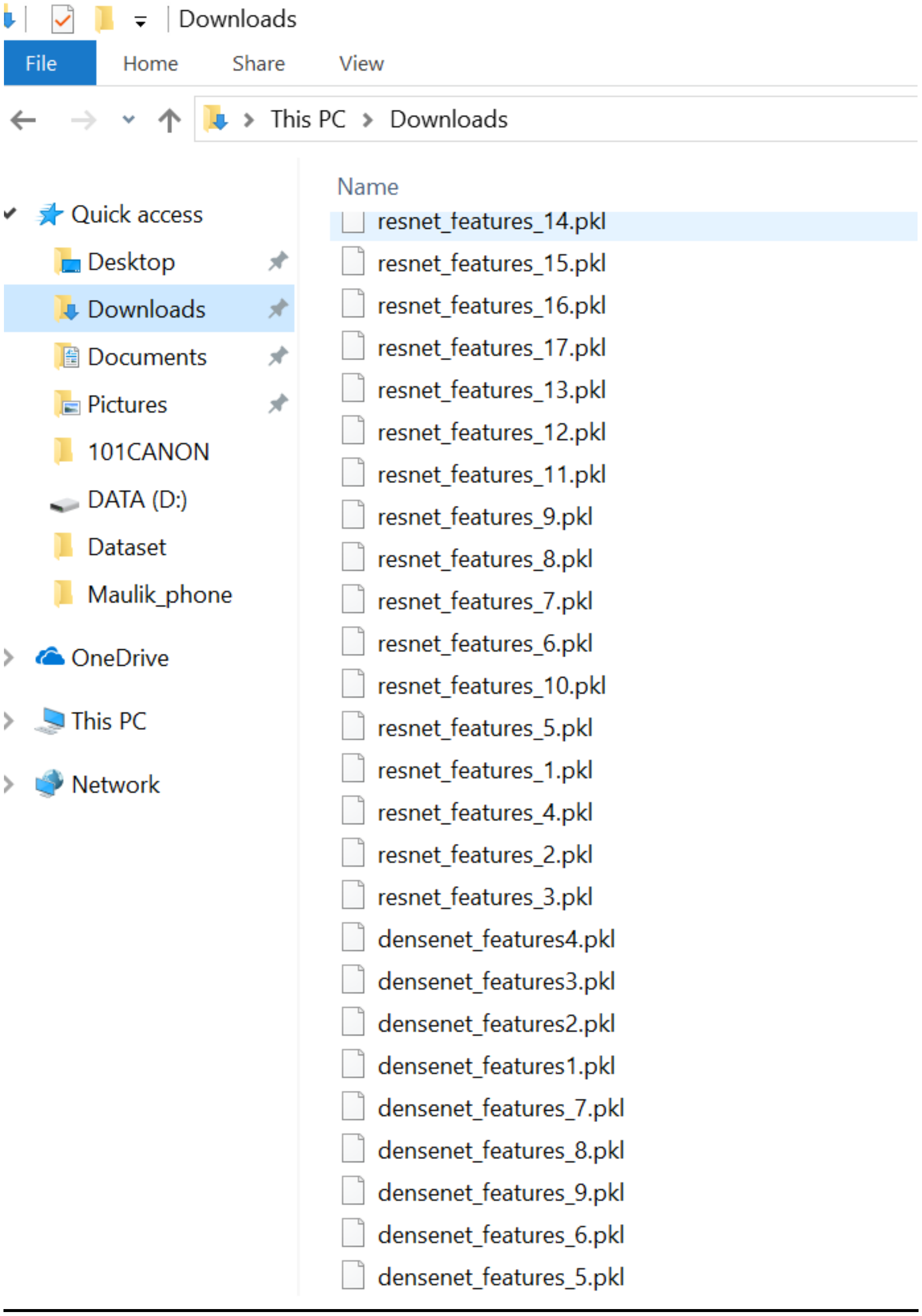
Figure 1. Duplicates Detection Block diagram

Biggest Challege:-

Colab RAM limitation: - 12GB

Dataset size:- 5GB
Feature vectors output size:- >4.5GB (For each model , DenseNet & ResNet)

To able run complete code, I had to process split many variables -> save/download splitted variable -> combine on another colab machine -> restore/upload complete variable to main colab machine



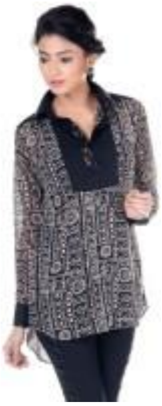
Name
resnet_features_14.pkl
resnet_features_15.pkl
resnet_features_16.pkl
resnet_features_17.pkl
resnet_features_13.pkl
resnet_features_12.pkl
resnet_features_11.pkl
resnet_features_9.pkl
resnet_features_8.pkl
resnet_features_7.pkl
resnet_features_6.pkl
resnet_features_10.pkl
resnet_features_5.pkl
resnet_features_1.pkl
resnet_features_4.pkl
resnet_features_2.pkl
resnet_features_3.pkl
densenet_features4.pkl
densenet_features3.pkl
densenet_features2.pkl
densenet_features1.pkl
densenet_features_7.pkl
densenet_features_8.pkl
densenet_features_9.pkl
densenet_features_6.pkl
densenet_features_5.pkl

Result

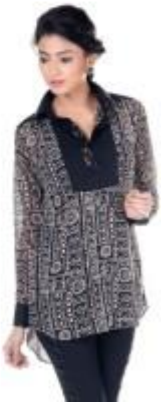
1. Duplicate products having same Image

- Algorithm easily detected **duplicate** products

Product ID = 18237



Product ID =18238

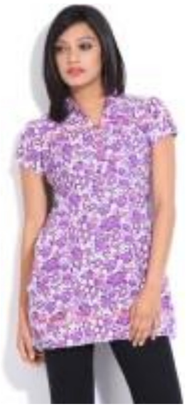


2. Duplicate Products having different color

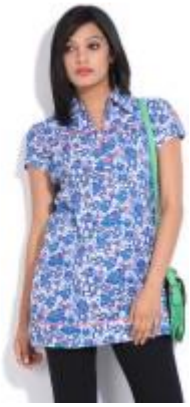
- Both products have similar texture & other attribute. Just color is different

- Algorithm successfully detected this **duplicates**

Product ID = 1426983



Product ID = 1944551

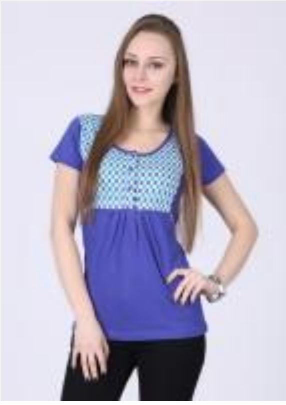


3. Duplicate products having same texture, different color and/or different models(girls)

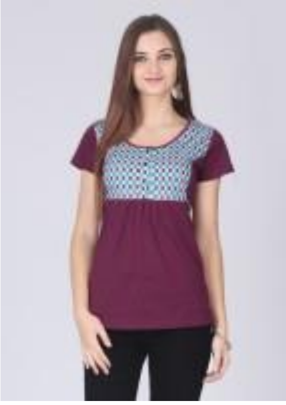
- Both girls are wearing similar dress , only color is different

- Algorithm successfully detected this **duplicates**

Product ID = 731912



Product ID = 1382730

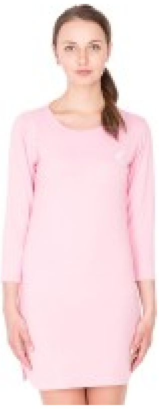


4. Duplicate products having different color and/or different pose

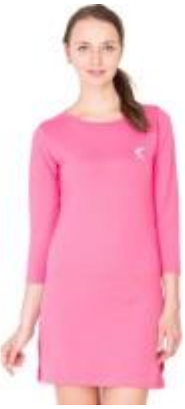
- Same girl wearing same dress of different color , little different pose

- Algorithm successfully detected this **duplicates**

Product ID = 1534916



Product ID = 1534917

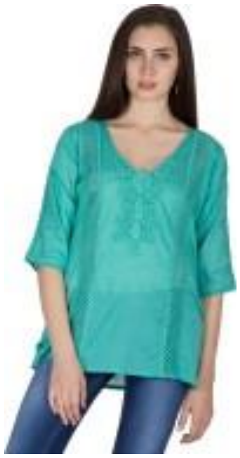


5. Different products almost same look like, but actually different

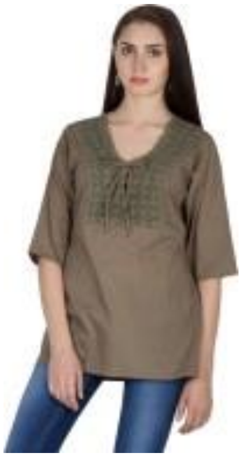
- Algorithm/model successfully identify below dresses are different, **not duplicate** (Even it looks similar)

Similar looks, but different products

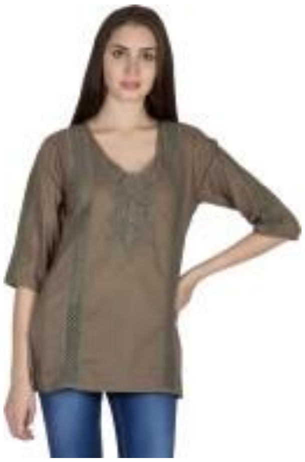
Product ID = 18239



Product ID = 18295

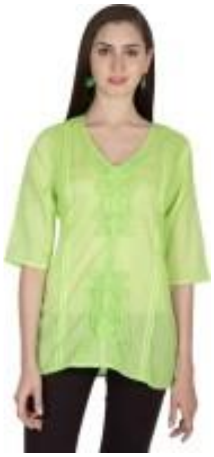


Product ID = 18305



Texture is different, different product

Product ID = 1267053



Product ID = 1646297