Name: Suraj Kumar Padhy

Email address: skp309skp@gmail.com

Contact number: 7978440416

Anydesk address:

Years of Work Experience: **1 year 8 month**

Date: **12th APR 2021**

Self Case Study -2: Buy Me That Look: An Approach for Recommending Similar Fashion Products.

"After you have completed the document, please submit it in the classroom in the pdf format."

Please check this video before you get started:
https://www.youtube.com/watch?time_continue=1&v=LBGU1_JO3kg

Overview

*** Write an overview of the case study that you are working on. ***(MINIMUM 200 words)*** ***

1. Nowadays E fashion influencers regularly post images, and users might be willing to mimic the looks of their influencers. Generally users might be interested in buying the entire look of the model. So in this problem we would like to solve, given a full pose image of a model ,the goal of our method is to recommend similar fashion products corresponding to the entire set of fashion articles worn by a model in the full-shot image.
2. However, fashion product recommendation is challenging. This is because fashion products have an enormous amount of variations present in the fashion items (eg., color, texture, shapes, viewpoint, illumination and styles). This problem is not only important to promote

cross-sells for boosting revenue, but also for improving customer experience and engagement.

3. First a fashion article images uploaded by users then, The main problem can be divided into following subproblems:
    a. Human keypoint detection
    b. Pose classification
    c. Article localisation and object detection
    d. Triplet network based image embedding model.
    e. retrieving and recommending similar products from our database using the embedding vector
4. FInally, we can say that this is challenging because, in contrast to performing recommendations for a single, primary article (for the query), we need to recommend similar products for the entire set of fashion articles worn by a model.

Research-Papers/Solutions/Architectures/Kernels

*** Mention the urls of existing research-papers/solutions/kernels on your problem statement and in your own words write a detailed summary for each one of them. If needed you can include images or explain with your own diagrams. ***

1. https://arxiv.org/pdf/2008.11638.pdf

   This is the main paper which describes how to solve the above problem

2. https://blog.tensorflow.org/2018/05/real-time-human-pose-estimation-in.html

   This is a blog from tensorflow which, describe how to use pose detection pretrained models i.e https://arxiv.org/pdf/1803.08225.pdf

3. https://github.com/switchablenorms/DeepFashion2 This link will redirect to the deepFashion2 dataset. We use this dataset to Human keypoint and pose detection and Article localisation and object detection
4. http://mmlab.ie.cuhk.edu.hk/projects/DeepFashion.html This link will redirect to the deepFashion dataset. To recommend similar fashion products we will use this dataset.
5. https://towardsdatascience.com/how-to-train-a-custom-object-detection-model-with-yolo-v5-917e9ce13208 This link will explain how to train custom object detection from a full shot image using yolo.
6. https://towardsdatascience.com/image-similarity-using-triplet-loss-3744c0f67973 this blog explains how to use the triplate loss train embedding model and get the embedding vector for each of the fashion items.

First Cut Approach

*** Explain in steps about how you want to approach this problem and the initial experiments that you want to do. ***(MINIMUM 200 words)*** ***

1. First, I will analyse the basic statics about both datasets i.e. Deepfashion2 , Deepfashion like size of the dataset,clothing categories and no of objects present in the full shot images.
2. Then, I will use the pretrain Human Key point estimation model Trained by tensorflow and using the model output we will filter out full shot images from the Deepfashion2 dataset.
3. After that I will train a custom Yolo based object detection model on the full shot image using IOU loss to detect different fashion articles in the full-shot image.

4. Then I will train a custom Triplet network based image embedding model using the triplet loss mentioned in the paper to get the embedding vector.
5. Lastly using the above embedding model, I will generate all embedding vectors for the fashion image present in the Deepfashion dataset.
6. Finally using the cosine similarity or inverse euclidean distance, I will recommended most similar products from the database for each of the detected fashion product by the yolo model in the full-shot image.

Notes when you build your final notebook:

1. You should not train any model either it can be a ML model or DL model or Countvectorizer or even simple StandardScalar
2. You should not read train data files
3. The function1 takes only one argument “X” (a single data points i.e 1*d feature) and the inside the function you will preprocess data point similar to the process you did while you featurize your train data
    a. Ex: consider you are doing taxi demand prediction case study (problem definition: given a time and location predict the number of pickups that can happen)
    b. so in your final notebook, you need to pass only those two values
    c. def final(X):

preprocess data i.e data cleaning, filling missing values etc

compute features based on this X

use pre trained model

return predicted outputs

final([time, location])

d. in the instructions, we have mentioned two functions one with original values and one without it

e. final([time, location]) # in this function you need to return the predictions, no need to compute the metric

f. final(set of [time, location] values, corresponding Y values) # when you pass the Y values, we can compute the error metric(Y, y_predict)

4. After you have preprocessed the data point you will featurize it, with the help of trained vectorizers or methods you have followed for your train data

5. Assume this function is like you are productionizing the best model you have built, you need to measure the time for predicting and report the time. Make sure you keep the time as low as possible

6. Check this live session: https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/4148/hands-on-live-session-deploy-an-ml-model-using-apis-on-aws/5/module-5-feature-engineering-productionization-and-deployment-of-ml-models