

Dataset information:

CelebA Dataset is a large-scale face attributes dataset with more than 200K celebrity images, each with 40 attribute annotations. The images in this dataset cover large pose variations and background clutter. CelebA has large diversities, large quantities, and rich annotations, including

- 10,177 number of identities,
- 202,599 number of face images, and
- 5 landmark locations, 40 binary attributes annotations per image.

The dataset can be employed as the training and test sets for the following computer vision tasks: face attribute recognition, face detection, landmark (or facial part) localization, and face editing & synthesis.

Sample Images



Data Source : <https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html>

Business Problem: Given a image, the algorithm should predict the 5 similar images

Solution:

This problem has been solved using four approaches, However, I am keeping the codes only for three approaches.

It is as follows:

- Using Classical Machine learning
- Using Auto Encoders and machine learning
- Using VGG Face pre-trained weights Machine learning
- Using Siamese Neural Network

I personally feel that there can be many solutions or approaches for the given problem statement. However, I feel that the four approaches, I used it convenient.

Please Note that I have not much worked on accuracy adjustment and I have written the code using Tensorflow. I am convenient also in Pytorch. For simplicity, I have taken only 1000 samples out of 200k as I was having issue related to Ram and Computational time

Using Classical Machine learning:

Images are nothing but an array or a matrix, which is having values ranging from 0 to 255. This problem statement can be solved by using KNN-classifier.

Please find the below point for approach:

- First, I read all images and flatten all the images. I have even normalize the images.
- Second, I can even use PCA or any other dimension reduction technique to decrease the vector space. However, I have not used PCA technique. I directly fit the model using KNN-classifier with Euclidian distance as distance parameter.
- Third, after the model is fit, I am using KNN.neighbours for predicting the similar 5 images. I have written short code to display the images.
- Out of 5, Images predicted, the algorithm is able to predict 3 similar images successfully. If I finetune the model, I might increase the accuracy of the algorithm.

In the current problem given, We have already the identity CSV files, which talks about, which all images are having same identity. If it was not given also, we can do K-Means clustering first to get the labels. Once , I have the labels, I can fit on KNN-classifier. So, Both can be done.

Using Auto Encoders and machine learning:

Auto Encoders is artificial Neural Network, which basically try to produce, the approx. image as input images. Here Please find the below point for approach:

- First, I created a simple Auto Encoder model and I trained it for 30 epochs.
- Once, the network is trained, I took the Encoding part of the Network and created **Embeddings** for the images in Latent Space. Latent space is nothing but, Space which is having similar data points closer to each other and it is the space, where all information is lied in compression form. It also helps in dimension reduction.

- Once, the Embeddings are created, I have used KNN classifier and fit the embeddings on it and with the help of KNN.neighbours, I was able to predict 5 Similar Images.

Using VGG Face Net and machine learning:

VGG-Net is a powerful network for performing image classification task and still it is widely used.

Please find the below approach:

- Here, I have used Pre-trained VGG-face weights and downloaded from <https://www.kaggle.com/acharyarupak391/vggfaceweights>
- I defined the VGG face net in the code and then for the last layer of 2622 size, I have taken out the Embeddings for all the images.
- Once, the Embeddings are created, I have used KNN classifier and fit the embeddings on it and with the help of KNN.neighbours, I was able to predict 5 Similar Images.

Using Siamese Neural Network

Siamese Neural Network is very effective for low amount of data and it is still more powerful Network for predicting Face recognition, Signature verification and others.

I have tried to fit the data using this network, but, unfortunately, My GPUs quota was over.

However, I am writing my approach to solve the problem using Siamese Neural Network:

Please find the below approach:

- In the first step, I have created the dataset of Anchor, positive, Negative images.
- As we know that Anchor, Positive and negative images share the same weights of the network across each other.
- I have used here Transfer learning using Mobile Net network which pretrained weights of 'ImageNet' and I kept top=False so that it doesn't give the last layers.
- I added flatten layer and 2 dense layers of 512 and 256 in the network, obviously, the last layer will create Embeddings for Anchor, Positive and Negative Images. I am ever training last few 20 layers in the network.
- Loss to be used here is triplet loss with margin of 0.5, so, Network will create latent space, where similar images will be more near and dissimilar images will be more far from each other.
- Once, the Network will learn, I was planning to find the images, which is closer to each other using a function, where I am going to define distance parameter. I mean to say that; the code will try to find images whose distance is less than the distance parameter given.

