HUMAN IMAGE CLASSIFICATION PROJECT (MAJOR)

NAME: TEETAS BHUIYA
BATCH: B18, FEB
DATE: 21/04/2021

A MODEL THAT WILL CLASSIFY INDIAN ORIGIN HUMAN FROM OTHER REGIONS

INTRODUCTION:

Machine learning, Big data and data analysis projects require data to work with. There are many ways in which we can acquire the data required to build the models.

1. Download manually the images from search engines like Google. (this takes long time for large data)
2. Download the dataset from Kaggle.com (doesn't always satisfy our requirements)
3. Use apps for web scraping (ex: Parsehub or Octaparse)
4. Use python libraries to scrape the images (Using Bing_image_downloader)
5. Building an Image crawler

In the Major Project (Indian human image classification),we will be using the approach of downloading images manually from Google as it has the highest ratio for accurate data set.

Our approach - Downloading images manually
1) Indian men images - 650 images
2) Indian women images - 400 images
3) Foreign men - 300 images
4) Foreign women - 300 images

And all these images are to be stored in folders accordingly in a file called "Image dataset". Other methods like bing_image_downlaoder and apps for web scraping are almost similar when it comes to collecting dataset (the images collected have some duplicates, and mostly poor standard).

The reference links which we planned to use for data collection are listed below :
1) https://www.pexels.com/search/indian%20man/
2) https://www.pexels.com/search/indian%20woman/
We used these links for getting the bulk of the data, downloaded using bing downloader and fatkun image downloader.

FACE DETECTION WITH HAAR CASCADE:

It is an Object Detection Algorithm used to identify faces in an image or a real time video. The model created from this training is available at the OpenCV GitHub repository https://github.com/opencv/opencv/tree/master/data/haarcascades.

Face Detection, a widely popular subject with a huge range of applications. Modern day Smartphones and Laptops come with in-built face detection softwares, which can authenticate the identity of the user. There are numerous apps that can capture, detect and process a face in real time, can identify the age and the gender of the user, and also can apply some really cool filters. The list is not limited to these mobile apps, as Face Detection also has a wide range of applications in Surveillance, Security and Biometrics as well.

SVM ALGORITHM:

“Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well (look at the below snapshot).

Pros and Cons associated with SVM:

Pros:

- It works really well with a clear margin of separation
- It is effective in high dimensional spaces.
- It is effective in cases where the number of dimensions is greater than the number of samples.
- It uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.

Cons:

- It doesn't perform well when we have large data set because the required training time is higher
- It also doesn't perform very well, when the data set has more noise i.e. target classes are overlapping
- SVM doesn't directly provide probability estimates, these are calculated using an expensive five-fold cross-validation. It is included in the related SVC method of Python scikit-learn library.

POLYNOMIAL KERNEL:

In machine learning, the polynomial kernel is a kernel function commonly used with support vector machines (SVMs) and other kernelized models, that represents the similarity of vectors (training samples) in a feature space over polynomials of the original variables, allowing learning of non-linear models.

Although the RBF kernel is more popular in SVM classification than the polynomial kernel, the latter is quite popular in natural language processing (NLP). The most common degree is d = 2 (quadratic), since larger degrees tend to overfit on NLP problems.

DATASET CLEANING AND BUILDING THE MODEL:

Our target is to remove unwanted pictures(not human pictures),which can be done by cleansing the dataset using a python code. For this we used OpenCV, which contains haarcascades package from which we used face cascade and eye cascade, which are used to detect the face and eyes.
We have written the code which basically detects whether the image has a face and 2 eyes, if yes crop it and save it in the new folder called Cropped, else discard it. So this code discards unwanted images and we get a dataset which can be used to train our model.

Now we have to build a model using the dataset which is already split into Indian and non-Indian (foreign) folders. We also have to preprocess the dataset by resizing and flatten the images(from 3D to 1D array). Then use those flattened preprocessed data as input for our model.So all we have to do is take those pictures as input X,Y. Using
train_test_split data we split the data into train and test data. So 80% data to train our model, and 20% to test our model.

Now for our model the best algorithm was SVM (kernel used was 'poly') and gave us the most accuracy. So now we had to feed the model with dataset and

then use it to test our outputs based on the inputs test dataset. So we got a accuracy of 75-80%.

Now we also saved the file in pickle, so that we don't have to run it every time we open it.

Now we we want to test our model, all we have to do is give the input:
1) An URL of the image you are giving as input.(mostly jpg format)
2) The datapath of the image you have stored in your
pc. (ex:C//users//XYZ//Downloads//image1.jpg)

Now the output will be the image that we chose presented, and also stating whether the person is an Indian or foreign. And also store the image into the respective folder (if Indian face store it in Indian folder containing Indian faces, and vice versa).

This model which we created has an accuracy of 75-80%.