ABSTRACT

A data wrangling and machine learning pipeline is implemented for a binary face recognition task. Using Inception ResNet v1 model pre-trained on the VGGFace 2 dataset and training it on manually labeled face image data, we were able to achieve 87% accuracy on the validation dataset when identifying the face of Bill Gates. The trained model is able to correctly identify most instances of Bill Gate’s face when tested on two separate video clips.

INTRODUCTION

In this project we implemented a pipeline for real time facial recognition in videos. We performed transfer learning using Inception ResNet v1, which has been pre-trained on the VGGFace 2 dataset. We train the model on a dataset collected from two separate public datasets, CelebA and the Flickr-Faces-HQ Dataset, and manually labeled with two classes "bill\_gates" and "not\_bill\_gates". We use stochastic gradient descent to train a logistical regression model with a sigmoid activation function.

The model achieves ~87% accuracy on validation set. We then use the model to perform facial recognition in videos. We use the DNN face detection model to output a bounding box around detected faces, and then feed the cropped face image to our facial recognition model to output either "bill\_gates" or "not\_bill\_gates". Using two video clips involving both Bill Gates and other people, we observe that our model is able to successfully detect Bill Gate’s face when it is not occluded. The model’s performance deteriorates for side views of faces and when the face is occluded or motion blurred.

iii. RELATED WORKS

iv. METHODS

We implement CNN network with transfer learning to achieve face recognition and classification. Transfer learning is a technique where we can use a model pre-trained for one task, and repurpose it on a new and related task to achieve higher performance than training a model from scratch.

In this project we choose Inception ResNet v1, which is a hybrid network inspired both by Inception and the performance of ResNet. We use a version of the model pre-trained on VGGFace 2, a large face database, as the input to our training. All layers of the model are kept unchanged except the last fully connected layer, which is trained on a dataset of approximately 1000 images labeled either "bill\_gates" or "not\_bill\_gates".

Two different loss functions, the standard PyTorch cross entropy loss for classification, and BCEWithLogitsLoss are used. BCEWithLogitsLoss is logistical loss layers on top of a binary cross entropy loss, enhanced with better numerical computational stability. The model trained with BCEWithLogitsLoss is chosen for video face recognition tasks due to its superior validation accuracy.

Then, we use the DNN face detection model to detect faces in a video. We extract detected faces as images, and input the images to our trained face classification model. Finally, we output the bounding boxes for faces and the prediction of our face classification model in a video.

v. EXPERIMENTS

We experimented with two different loss functions and different training dataset. We first tried using 600 images of 5 different celebrities from the CelebA dataset as training set, split between 20% “bill\_gates” images and 80% “not\_bill\_gates” images. Another 300 images of similar composition are used as the validation set. After many epochs of training, we were only able to achieve ~65% validation accuracy.

We then increased the number of images to about 1000 and achieved 75% validation accuracy. However, when tested on real world videos, the model almost always failed to recognize Bill Gates. We believe this is due to the fact that we used cropped face images from CelebA, and all five celebrities we chose are white males, which lead to biases in the model.

Therefore, we use a different dataset, Flickr-Faces-HQ, to expand our training data. We include an extra 500 images of faces from people of different ages, races, and genders for the “not\_bill\_gates” class. With the new training set, we achieved 79% best validation accuracy after 27 epochs. Despite relatively small improvement in accuracy, the model performed significantly better in videos, likely due to the better diversity of the Flickr-Faces-HQ dataset.

Finally, we experimented with a different loss function. Because we are only concerned about a binary classification task, we changed the cross entropy loss to PyTorch’s BCEWithLogitsLoss, which is optimized for binary classification. This helped us increase the validation accuracy to 89%.

vi. ANALYSIS AND POTENTIAL FOR FUTURE WORK

We implemented a binary face recognition model with relatively high accuracy that can be directly applied to videos. Our model has a validation accuracy of 89%, and is able to recognize most instances of the face of Bill Gates in our test video clips. We believe the data pipeline and model we implemented can aid the labeling of facial data in future projects. However, the current accuracy of our model makes it less capable in security related tasks that require a ultra-low false positive rate.

We note that our model has the following weaknesses:

1. When face is blurred or occluded, the recognition accuracy significantly decreases.

2. The model’s performance is worse among similar faces of white males – it often misclassified white male faces, while mostly correctly classifies female faces and faces of other races.

We believe we should continue this work in the following aspects:

1. Further increase the diversity of our training data to increase accuracy and decrease bias.

2. Use other metrics, such as FPR, FNR, to evaluate the performance of the model.

3. Use asymmetric loss functions to penalize false positives more than false negatives, due to the requirement of most face recognition tasks.

4. Experiment with multi-class classification to recognize the faces of multiple known individuals.