
# COMP 6721

## Applied Artificial Intelligence

Project-1 Report

Submitted To: Prof. Dr. René Witte

Team Id: RN_08

Raj Mistry (40119206) - Data Specialist

Yash Jayeshkumar Pandya (40119272) - Training Specialist

Piyush Thummar (40125029) - Evaluation Specialist

## Dataset:

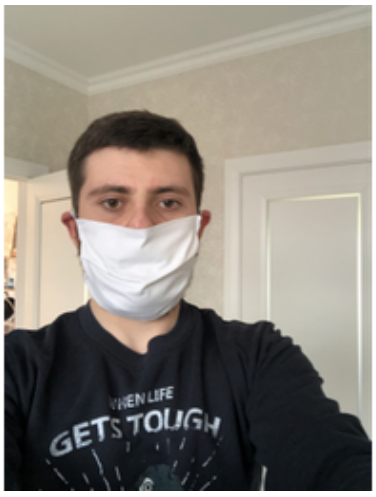
AI is the field that its projects are mostly dependent on datasets. By that, it can be deduced that to get good results from AI projects, we need a very well balanced and good dataset, because then only, the model can be trained for almost every probable outcome/result/class. Dataset performs the key role in the training and testing phase.

Here, we need to divide our dataset in the balanced manner for these three phases for precision and accuracy. In the ideal case, the whole dataset (3000 images) can be divided in 80-20 ratio, that is 80% data is for the training phase, and rest 20% is for the testing phase.

- **Training phase:** Here, the model is trained using a dataset and algorithm or pattern is built by the AI to distinguish objects.
- **Testing phase:** In this step, the trained model gets tested for the unknown data, that is isolated from the dataset to measure the performance of the model or algorithm using several performance matrices.

In this project, we have used 1000s of samples for each three classes that are person with mask, person without mask and not a person. We have used the balanced dataset with different varieties such as age, color, gender, etc. Some dataset has been taken from the data-repositories, for example, kaggle. And for some, we cannot find such repositories, where we have to come up with plenty of images from different sources from google or the internet.

For example the training dataset contains following types of images for each category:




***Person with Mask***

***Person Without Mask***

***Not a Person***

## CNN Architecture:

Here, we have implemented the Convolutional Neural Network to train the model for given classes. First of all, the given dataset must be preprocessed to make them balanced and of equal size(100*100). Here, 3 different datasets for each class will be processed, and their paths will be set. Those classes are labeled as below: ‘0’: Person with Face Mask, ‘1’: Person without Face Mask, and ‘2’: Not a Person.

Now, we divide our dataset as training and testing dataset, where 80% data is used for training, and rest is for testing phase. Also, here we have to set the weights of images to normalize the dataset, as the dataset size is different for each class, which may end up in unexpected results from the model. So, to overcome this problem, this weight information is passed to the loss function, which can be done via assigning the weight to each class according to its representability in the dataset. Here, we assign more weights to that class which has less images. So, when the model predicts incorrect labels, it can penalize those classes’ weights. However, smaller weights will be assigned to the classes with more images.

After this step, we load our training and testing dataset, and convert it to tensor with a label, and then load it using a data loader with batch size. Here, we load data with training and testing data loaders.

As per the CNN diagram, it can be depicted that the firstly, input image with input channel of 100 is supplied to our 1st Convolutional 2D layer with RGB channel and its output channel of 64, and padding 1. Now, this is normalized and provided as input to ReLU activation function. Then, its output is provided for MaxPooling with stride 2, where its size will be minimized and then it will be passed to the 2nd Convolutional 2D Layer. Here, its output channel will be 128 with padding 1, and then next steps are as above, which are ReLU activation function and MaxPooling. Then, it is passed to our 3rd Convolutional Layer, which produces the output with 256 channels and padding 1. And again, ReLU activation function and MaxPooling takes place. Then, at last we get the output (0/1/2) label, which represents “Masked Person”, “Person without Mask”, and “Not a Person”.

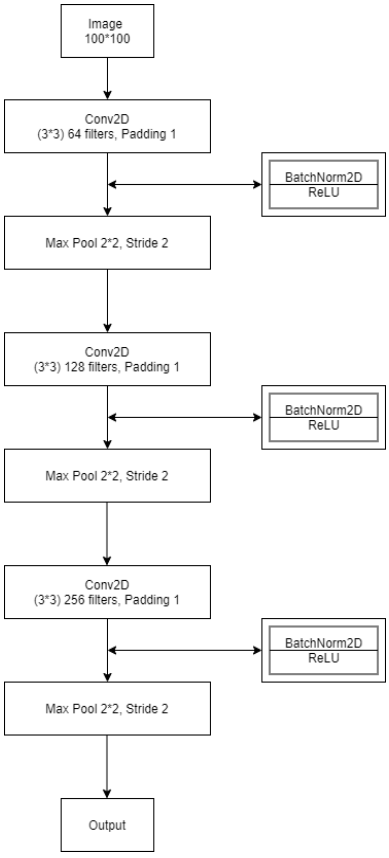



Image 2: CNN Architecture

## Evaluation:

After these three phases, we can evaluate the algorithm or the model by using confusion matrix, and several other matrices for its performance. In this project, the model is trained for 10 epochs. In every epoch, there is learning at the rate of 0.0001. Here, we evaluated the performance using recall, precision, accuracy and f1-measure for all phases. By using a confusion matrix, we can say some of the images have been misclassified due to imbalance data, that means we have 3rd class (not a person) as a more generic one, where we may not have enough data. Which can lead the model for this confusion and misclassification. Here, we can see that our model was trained with the accuracy of 98%; whereas, for testing, we have the accuracy of 91%.

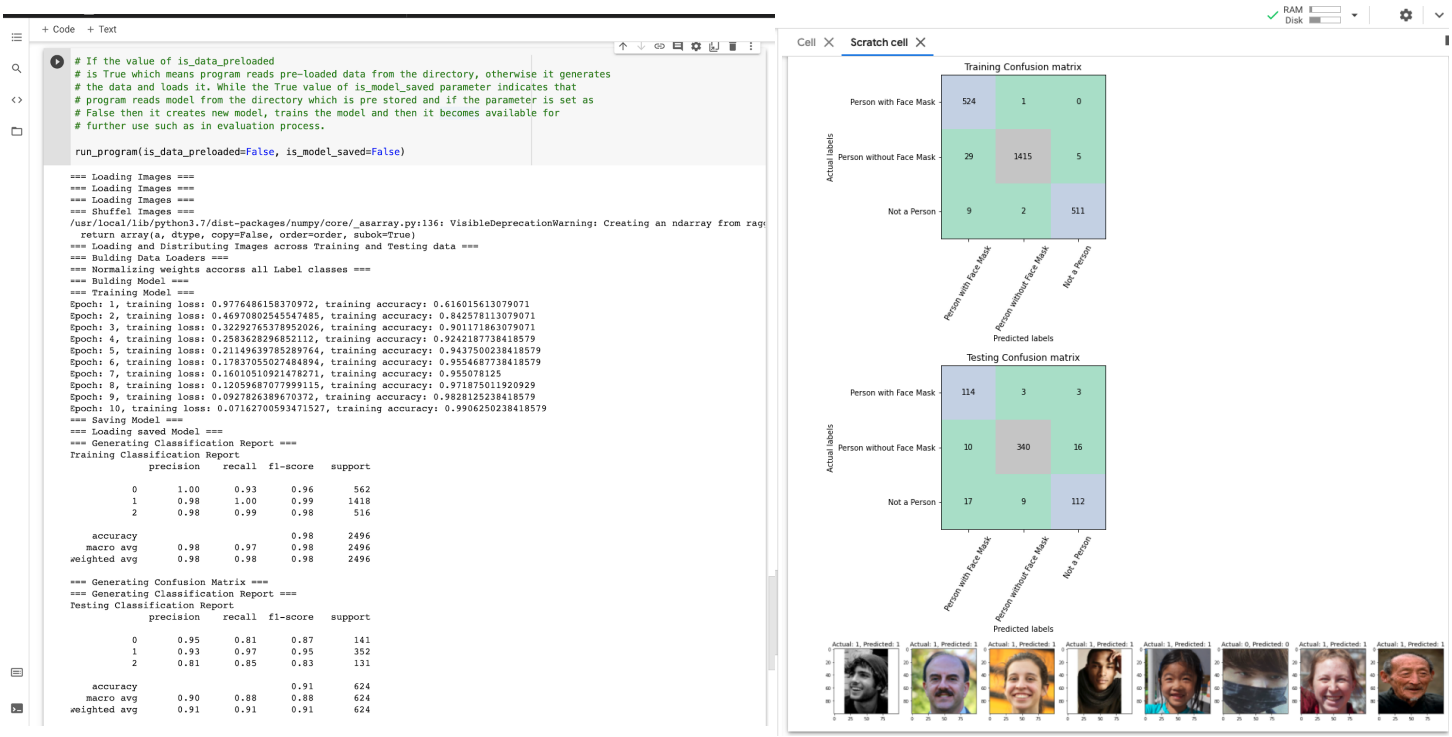



Image 2: Classification report and Confusion Matrix of our model

As mentioned above, some data are misclassified for the reason being is imbalanced data for each three classes, where human with mask and without mask can be categorized with several parameters, but the non-human class is very broad, where it can contain animals, non-living objects, statues, plants, etc.

In the next phase of the project, we are going to increase the data for the masked person and not a person categories and also add a validation layer of K-fold cross to remove bias and improve project performance.

## References:


- https://medium.com/swlh/introduction-to-cnn-image-classification-using-cnn-in-pytorch-11eefae6d83c
- https://towardsdatascience.com/how-i-built-a-face-mask-detector-for-covid-19-using-pytorch-lightning-67eb3752fd61
- https://towardsdatascience.com/covolutional-neural-network-cb0883dd6529
- https://core.ac.uk/reader/328808130