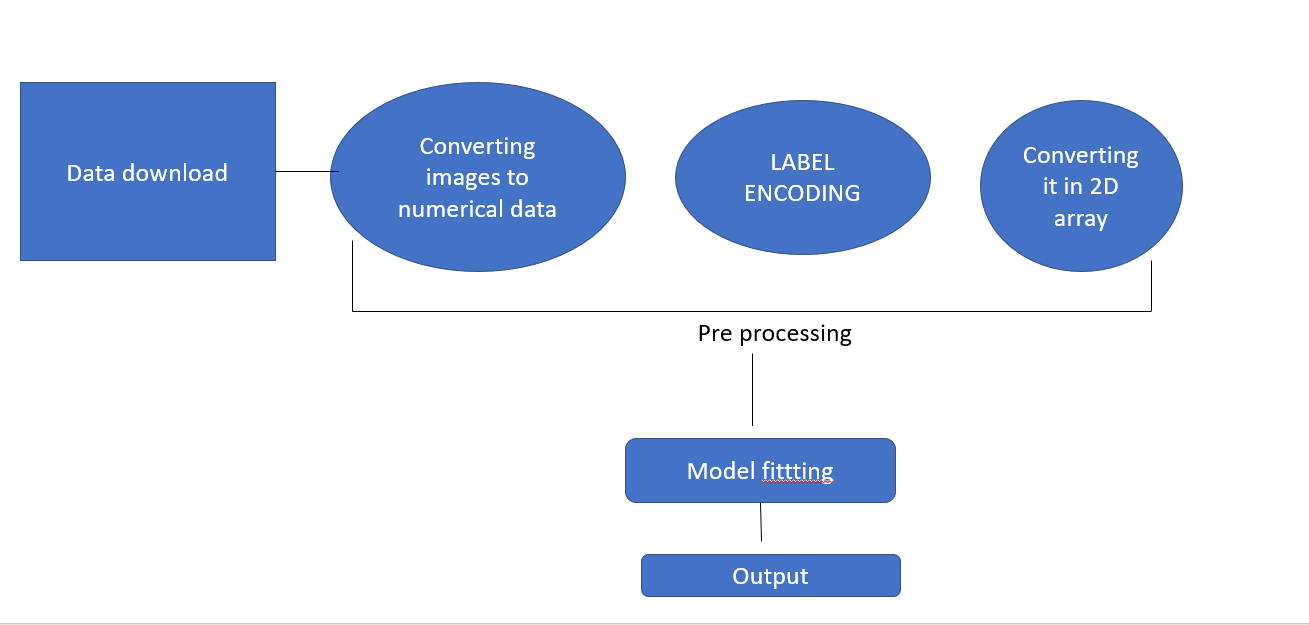
CSL2050 Face Mask Detection

Abhishek Jamhoriya(B19CSE003),Ajay Meena(B19EE005)

**Introduction**

In current pandemic facial masks are a crucial tool in avoiding covid, because of this importance of facial mask detection algorithm has increased tremendously. We have implemented an end to end pipeline to achieve the same.

**Pipeline**



**Preprocessing**

We have been provided with a dataset containing two folders which separately contain images of people wearing masks and without wearing masks. Most of the preprocessing is converting the images to a ML readable format and giving these obtained data points labels.

Once we have completed this we changed the labels from categorical format to numerical format using LabelEncoder from sklearn.model\_selection library. Now we have two lists so we convert them to numpy arrays as machine learning models only take them as inputs. Further we reshape the multidimensional array to a two dimensional array.

And finally we split the dataset into two equal half parts one being the train set and other test set, with this our preprocessing is complete.

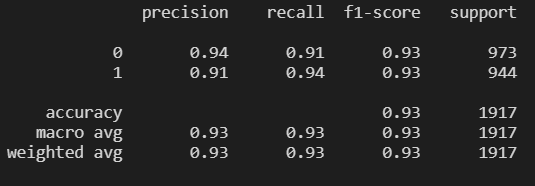
**Model implementation**

We have used three classification models namely Multi-Layer perceptron, Random Forest Classifier and Gaussian Naive Bayes.

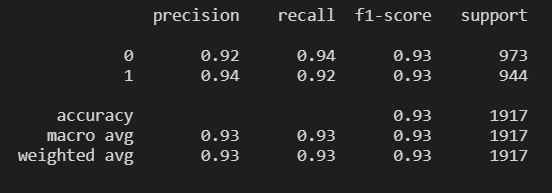
MLP is a Multi-Layer perceptron since it works well to recognize faces. We have kept all the parameters as default for this classifier. And the last classifier which we used is GaussianNB, it gave the worst performance out of all three classifiers. GaussianNB is implemented by Abhishek(B19CSE003) and the rest MLPClassifier and Random Forest are implemented by Ajay(B19EE005).

**Classification reports**

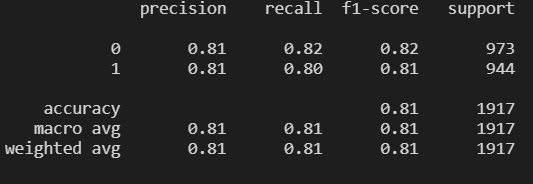
First we trained MLPClassifier, it took 5m 30sec to train which was the highest out of all three but it also gave the best classification out of these three with accuracy of 0.93. Here we attach the classification report for it-



Second is the Random Forest classifier, it also gave close to 0.93 accuracy. These are the classification report -



And finally gaussianNB, it took 5 sec but gave much worse accuracy. Upper limit being 0.86 in k fold cross validation.



**Conclusion and extra findings**

We also tried SVM but it could not complete the fitting process as we could not preprocess the data properly and because of it the server could not allocate high memory for 2 dimensional arrays. So we tried LDA but it also had the same problem. GaussianNB is implemented by Abhishek(B19CSE003) and the other 2 are implemented by Ajay(B19EE005).