PRML PROJECT

IIT JODHPUR

FACE MASK DETECTION

Nitin Singh Kanyal(B19EE058),Ram Khandelwal(B19CSE116)

I. Introduction

The problem statement expects us to perform detection analysis of whether person is wearing a mask or not.For this we will use various classifiers and calculate their accuracy scores and confusion matrix and compare them.

II. Observing the Dataset

The dataset was given contained various folders containing images of people with and without masks.So first we created only two folders ‘masked’ and ‘unmasked’.The folder masked includes images of people wearing mask and unmasked folder includes images of people not wearing mask.

III.Data Preprocessing

First we have imported important libraries such as numpy,os,pil,cv2,pickle.Mounted the google drive.

IV.Feature Extraction

A. Converting image data into array data

We have converted the image data of from two folders to array data as working and applying classifiers on array data is easier.

B. Converting to numpy and plot some data set images

.We have converted the array into numpy array so that we can apply functionalities that run on the numpy arrays.We have reshaped the features array.We have also plotted some of the images from the dataset.

C. Save dataset as pickles and load the pickles

We have used pickles serialization and deserialization of dataset.This process includes saving the data into pickle format and using it for comparing the various models and calculating the results.Later pickles are restored so that training data is not lost and then this data could be used in other projects as well.

D. Total Classification counts and split data into train and test

We have got counts of masked and unmasked data.We denote 0 for person without a mask and 1 for person wearing the mask.

IV. CLASSIFIERS USED

A. K NEAREST NEIGHBOURS

This algorithm assumes that all the data points belonging to same class lie close to each other and form a cluster.So in this way it separates the data into k clusters and classifies it accordingly.

B.SUPPORT VECTOR MACHINES

Support vector machine is a supervised learning technique used to classify data with the help of kernel functions which include linear,polynomial and gaussian functions.It is effective in use with high dimensional data.Various hyperparameters such as C value,gamma value,tolerance,coeff_, etc affect the accuracy produced.

C.CONVOLUTIONAL NEURAL NETWORKS

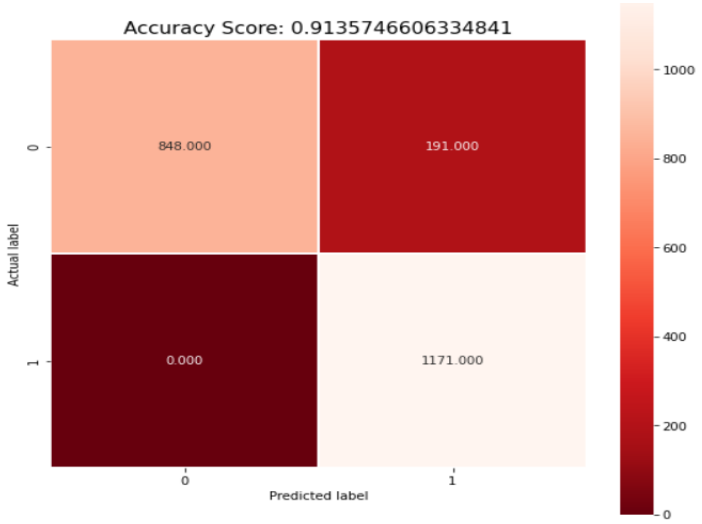
It is deep learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms.

V. RESULTS AND COMPARISON

For class 0 - with mask samples

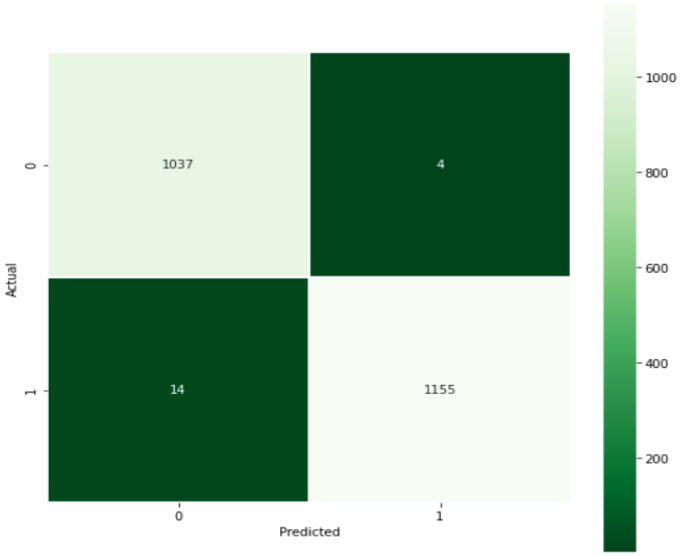
For class 1- without mask samples

Confusion matrix and accuracy report of KNN



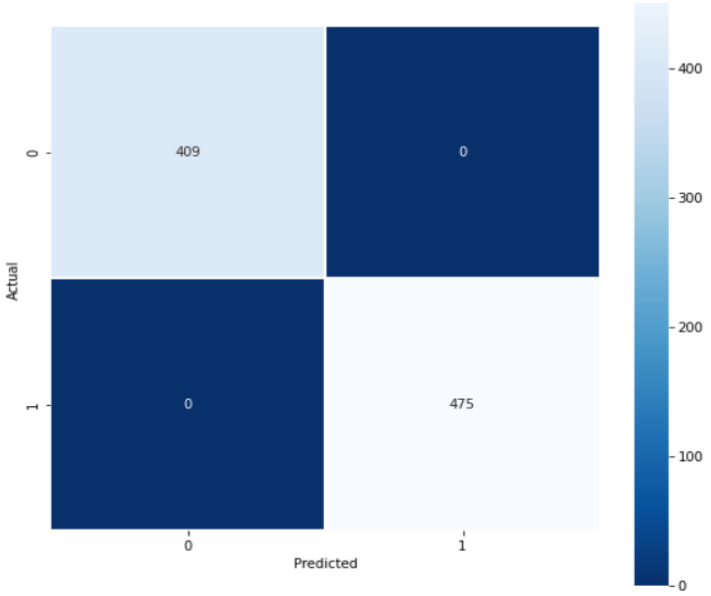
	precision	recall	f1-score	support
0	1.00	0.82	0.90	1039
1	0.86	1.00	0.92	1171
accuracy			0.91	2210
macro avg	0.93	0.91	0.91	2210
weighted avg	0.93	0.91	0.91	2210

Confusion matrix and accuracy report of SVM



	precision	recall	f1-score	support
0	0.99	1.00	0.99	1041
1	1.00	0.99	0.99	1169
accuracy			0.99	2210
macro avg	0.99	0.99	0.99	2210
weighted avg	0.99	0.99	0.99	2210

Confusion matrix and accuracy report of CNN



	precision	recall	f1-score	support
0	1.00	1.00	1.00	409
1	1.00	1.00	1.00	475
accuracy			1.00	884
macro avg	1.00	1.00	1.00	884
weighted avg	1.00	1.00	1.00	884

On comparing the above performance we see that

For class 0

Comparison	Classifier
Precision	CNN=KNN>SVM
Recall	CNN=SVM>KNN
F1 score	CNN>SVM>KNN

For class 1

Comparison	Classifier
Precision	CNN=SVM>KNN
Recall	CNN=KNN>SVM
F1 score	CNN>SVM>KNN

Below is the results of the accuracy results of all the three classifiers used above to predict the model.

Classifier	Accuracy
KNN	0.914
SVM	0.991
CNN	1.000

Thus we could clearly observe that CNN performed the best than SVM and than KNN.

VI.CONTRIBUTIONS

- **Ram** - Feature extraction,KNN
- **Nitin-** Data preprocessing,SVM,CNN
  And equal contributions in report and readme file.

CONCLUSIONS

We conclude that CNN is the best classifier to be used for face mask detection.It gives us 100% accuracy as shown above.We followed machine learning pipeline step by step which helps us to systematize our approach towards solving problems for machine learning.

Acknowledgment

.We have successfully completed the face mask detection project by using various classifiers and comparing their accuracies.We also learnt how to use machine learning pipeline in a project from preprocessing,feature selection,learning a model to classification and finding accuracy scores.We got to learn a lot from this project.We express our gratitude to Dr Richa for giving us this opportunity to work on this project.

References

[1] https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html
[2] https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html
[3] https://scikit-learn.org/stable/modules/neural_networks_supervised.html
[4] https://scikit-learn.org/stable/modules/feature_extraction.html