FACEMASK DETECTOR USING MACHINE LEARNING

J Jahnavi (B19CSE109), Buruka Nikhitha (B19EE019))

Pattern Recognition and Machine Learning
Indian Institute of Technology, Jodhpur, Rajasthan, India

Abstract: This paper describes our comparison of multiple algorithms for detecting whether a person is wearing masks or not. These images are classified as either masked or unmasked. With the help of this project, a person who is intended to monitor the people can be seated in a remote area and still can monitor efficiently and give instructions accordingly. We show that machine learning algorithms (Support Vector Machine, K-Nearest Neighbors, Multi-layer Perceptron and Convolutional Neural Network) have accuracy above 99% when 50% of the given dataset was taken for testing data when trained with 50% of given dataset as testing data. This paper also describes the preprocessing steps needed in order to achieve high accuracy along with data visualization and performance evaluation of the pipeline implemented using cross validation and comparison between implemented concepts.

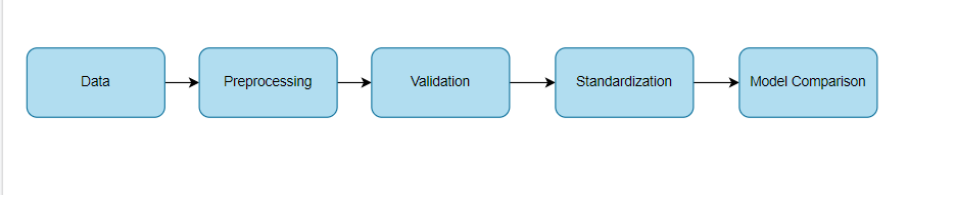
Index Terms: Visualization, Preprocessing, Binary classification, Classification Models

1. Introduction

The trend of sporting face mask publicly is rising because of the Covid-19 epidemic everywhere in the world. Because Covid-19 people wont to wear mask to shield their health from air pollution. Whereas other are self-conscious concerning their looks, they hide their emotions from the general public by activity their faces. Somebody treated the wearing face masks works on hindering Covid-19 transmission. The monitoring process involves the finding of anyone who is not sporting a face mask. Here we introduce a mask face detection model that's supported machine learning and image process techniques to determine whether a person is wearing masks or not.

2. Approach

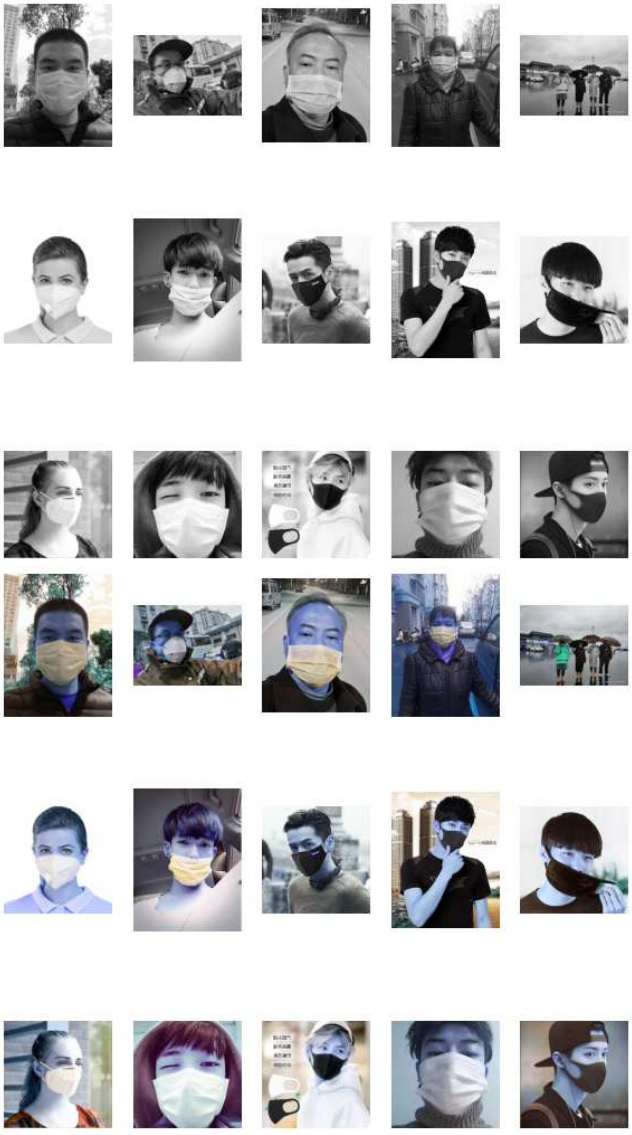
Our approach is to use different machine learning classifiers to classify the images as masked or unmasked. The machine learning classifiers are Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Multi-layer Perceptron (MLP) and Convolutional Neural Network (CNN). The architecture for the data visualization and preprocessing steps is shown in the figure below.



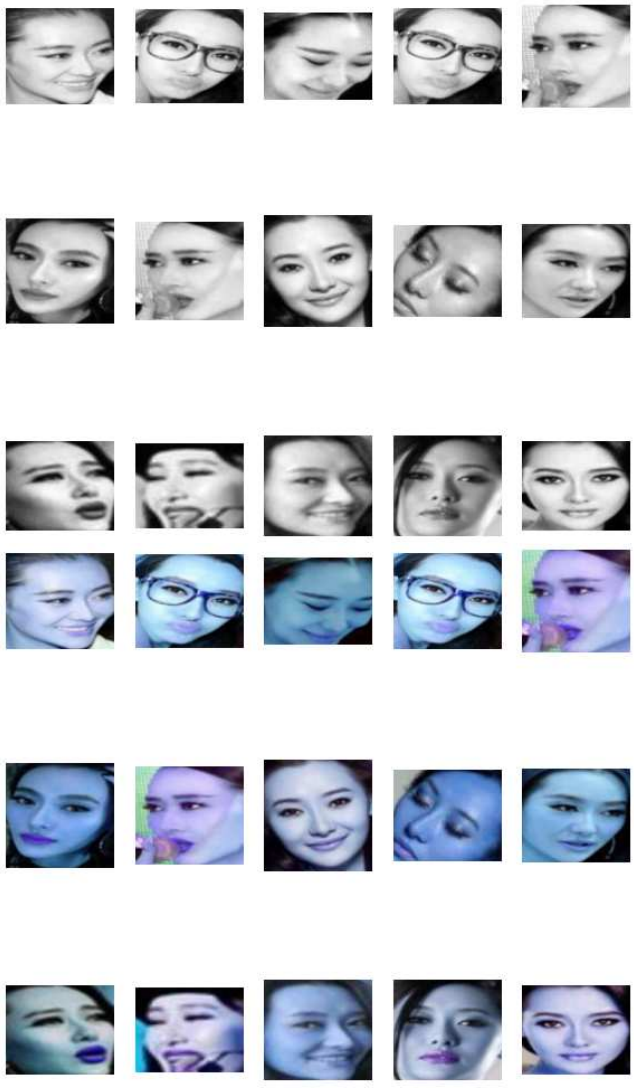
3. Display and Visualization

Data visualization is the graphical representation of information and data. Following are few images from the two classes of the dataset i.e. masked and unmasked. Data visualization is one of the steps of the data science process, which states that after data has been collected, it must be visualized for data information before model training.

3.1. Masked Images



3.2. Unmasked Images



4. Preprocessing

The dataset we are using consists of images with different colors, different sizes, and different orientations. Therefore, we need to convert all the images into grayscale because we need to be sure that color should not be a critical point for detecting mask. We made dataframes out of the given dataset for its ease of computation.

4.1. Performing Data Standardization

Standardization refers to shifting the distribution of each attribute to have a mean of zero and a standard deviation of one (unit variance). It is useful to standardize attributes for a model that relies on the distribution of attributes such as Gaussian processes.

In summary, the data was ideal to be used for training after following the preprocessing steps.

5. Performing Linear Discriminant Analysis

Linear Discriminant Analysis is a dimensionality reduction technique which is commonly used for the supervised classification problems. It is used for modeling differences in groups i.e. separating two or more classes. It is used to project the features in higher dimension space into a lower dimension space.

6. Performing Cross Validation

The process of deciding whether the numerical results quantifying hypothesized relationships between variables, are acceptable as descriptions of the data, is known as validation. Generally, an error estimation for the model is made after training, better known as evaluation of residuals. In this process, a numerical estimate of the difference in predicted and original responses is done, also called the training error. However, this only gives us an idea about how well our model does on data used to train it. Now its possible that the model is underfitting or overfitting the data. So, the problem with this evaluation technique is that it does not give an indication of how well the learner will generalize to an independent/ unseen data set. Getting this idea about our model is known as Cross Validation.

7. Comparing Algorithm Performances

We test different classifiers using: Support Vector Machine, K-Nearest Neighbors, Multi-layer Perceptron and Convolutional Neural Network

7.1. Support Vector Machine

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane. SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane
The Accuracy obtained using SVM is **99.9**%.

7.2. K-Nearest Neighbors

K-nearest neighbors (KNN) algorithm is a type of supervised ML algorithm which can be used for both classification as well as regression predictive problems. However, it is mainly used for classification predictive problems in industry. The following two properties would define KNN well:
- Lazy learning algorithm KNN is a lazy learning algorithm because it does not have a specialized training phase and uses all the data for training while classification.
- Non-parametric learning algorithm KNN is also a non-parametric learning algorithm because it doesn't assume anything about the underlying data. .

The Accuracy obtained using K-Nearest Neighbors model is **99.9**%.

7.3. Multi Layer Perceptron

Multi layer Perceptron (MLP) is a supplement of feed forward neural network. It consists of three types of layers—the input layer, output layer and hidden layer. The input layer receives the input signal to be processed. The required task of classification is performed by the output layer. So basically, MLPClassifier relies on an underlying Neural Network to perform the task of classification. An arbitrary number of hidden layers that are placed in between the input and

output layer are the true computational engine of the MLP.The neurons in the MLP are trained with the back propagation learning algorithm. The major use cases of MLP are pattern classification, recognition, prediction and approximation.

The Accuracy obtained using Multi layer Perceptron Classifier model is **99.9**%.

7.4. Convolutional Neural Network

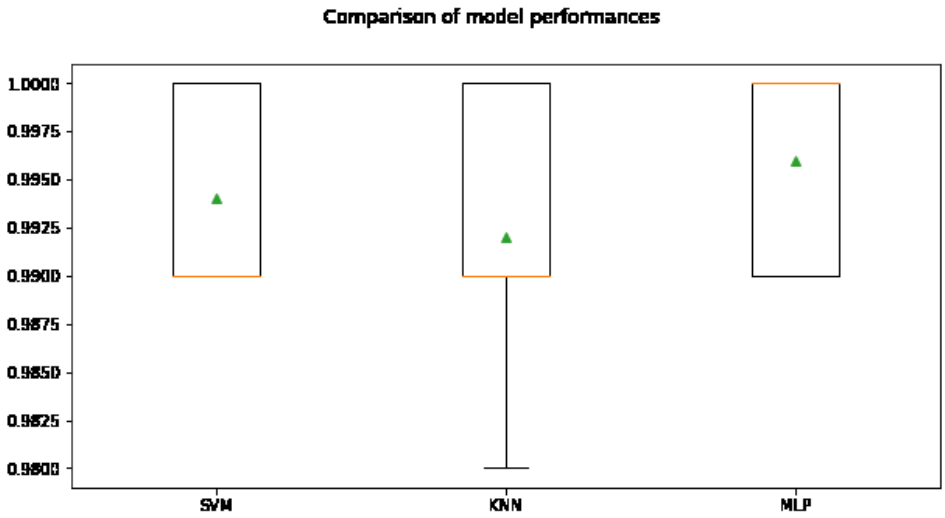
A Convolutional Neural Network (CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learn-able weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a CNN is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, CNNs have the ability to learn these filters/characteristics

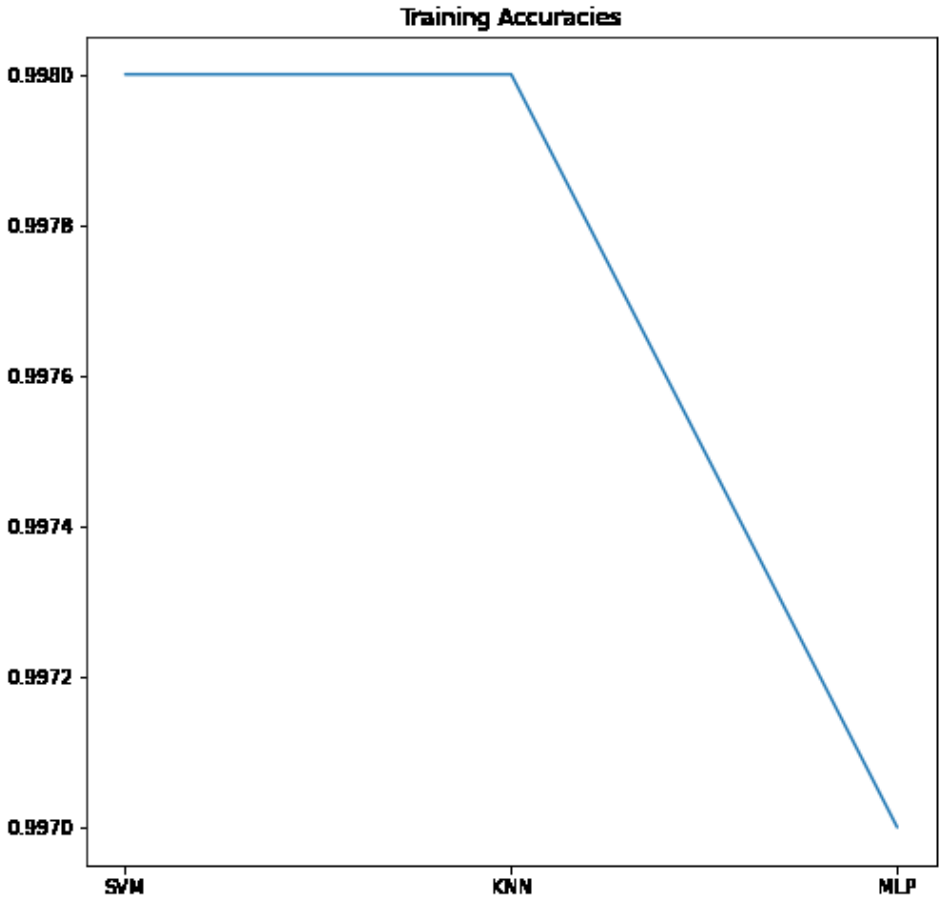
The Accuracy obtained using Multi layer Perceptron Classifier model is **99.69**%.

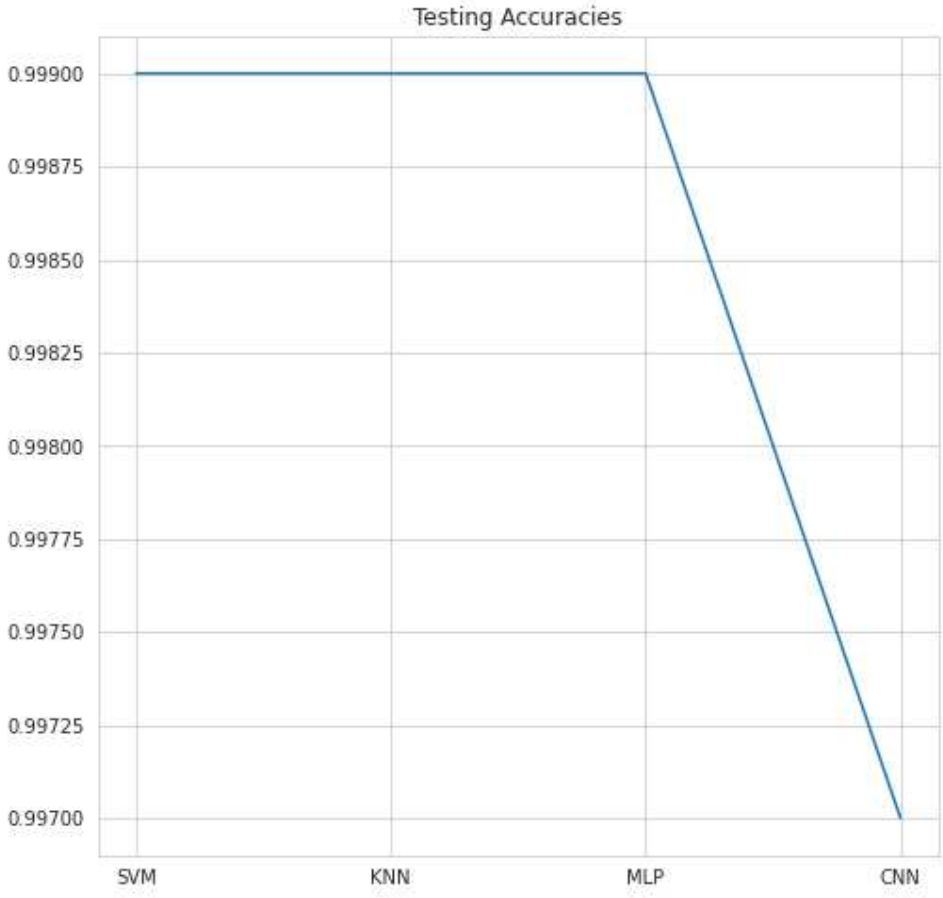
7.5. Model Performance

Evaluating the performance of a model indicates how successful the scoring (predictions) of a data set has been by a trained model. Model evaluation metrics are required to quantify model performance.

Model	Accuracies
Support Vector Machine	99.9%
K-Nearest Neighbors	99.9%
Multi Layer Perceptron	99.9%
Convolutional Neural Network	99.69%




Training Accuracies
0.9980
0.9978
0.9976
0.9974
0.9972
0.9970
SVM
KNN
MLP



Observation

Observing the accuracy values of the models, the first three models perform with an equivalent accuracy. Although MLP performs well, it is high in computational cost.

Results and Discussion

The accuracies obtained using the Support Vector Machine, K-Nearest Neighbors, Multi-layer Perceptron and Convolutional Neural Network are 99.9, 99.9, 99.9, 99.69 percentage respectively. Looking at the accuracies obtained, Logistic regression is preferred if a tweet is a positive or a negative one. If computational time is taken as a deciding factor, Support Vector Machine classifier is preferred as it is computationally light with comparable accuracy with the other classifiers.

Conclusions

We presented multiple algorithms for detecting whether a person is wearing masks or not. We used multiple libraries and models to understand the pros and cons of each of the models for this particular data set. We show that preprocessing of the data set is an effective way to increase the accuracy. Machine learning algorithms (Support Vector Machine, K-Nearest Neighbors, Multi-layer Perceptron and Convolutional Neural Network) can achieve accuracy greater than 99% for classifying images that are either masked or unmasked when using this method. Machine learning algorithms are shown to classify the images with similar performance as demonstrated in this report.

Contributions

This project was created for **CSL2050 Pattern Recognition and Machine Learning**.
- **J Jahnavi (B19CSE109)** made the Observations and Conclusions, performed Data Mining, Data Preprocessing, trained the Support Vector Machine and the Convolutional Neural Network Model.
- **Buruka Nikhitha (B19EE019)** made the Introductory section for the report, performed Data Visualisation, Data Standardisation, trained the K-Nearest Neighbors and the Multilayer Perceptron Model.