**RESEARCH PROJECT ON COVID-19 CLASSIFICATION**

**Dataset Description**

The dataset we consider for our study is obtained from Kaggle’s dataset.

This dataset contains two folders for COVID and non-COVID images. Some of the augmentations include geometric image transformations such as flipping, rotation, translation and scaling. The dataset does not contain proper image distribution for training purposes. So, in order to avoid the problem of class imbalance, we manually split the data to have equal images in both classes, resulting in 4,044 images for each class (both COVID and non-COVID). The non-COVID dataset also contains the X-ray images of other respiratory diseases such as viral pneumonia, SARS, streptococcus and pneumocystis. This dataset includes anteroposterior, posteroanterior and lateral X-ray images. All the images are converted into 227×227×3 dimensional images before passing it on to the models.

**Experiment 1 (CNN as a classifier)**

As our first experiment, we use various CNN models (including three pre-trained models) to classify the images as COVID positives and negatives. The dataset containing 8,088 images is split in such a way that 80% of the input raw images are used for training the model while the remaining 20% are used up for testing. We use customised CNN models as well as pretrained ResNet-152, AlexNet and VGG-16 as the models for performance evaluation.

**Experiment 2 (Flattened Layer Output)**

Since CNNs have the exceptional capability of converting lower-dimensional features to higher dimensional features, we use it as a feature extractor. We obtain the higher dimensional features from the flattened layer of the model. Since the flattened output is very large, PCA is done on the features to reduce the dimensions. These features are then passed on to five different ML classifiers for the task of classification.

**Experiment 3 (Fully Connected Layer output)**

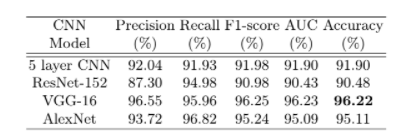
Similar to Experiment 2, we now add a fully connected layer of 512 neurons to the flattened layer and obtain the outputs from respective CNN models considered for our study. The output with a dimension of 512 is then used as the input feature vectors for ML classifiers.

**Performance Metrics**

The evaluation of the classifier is based on how accurately it distinguishes between positive and negative COVID-19 samples. Based on this, four outcomes are considered: true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN).

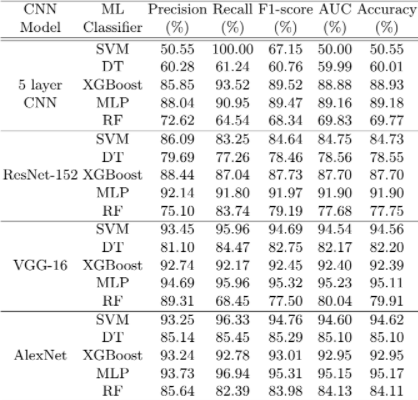
**Performance of CNN Classifier**

The performance of the trained CNN models is evaluated by testing the models’ ability to accurately predict the target classes of the test dataset. The test set, which contains 20% of the dataset, are provided as the inputs for the CNN models. Based on the learning that the models have obtained from the train set, the CNN models achieve state-of-the-art results as shown. VGG-16 outperforms all the other models with a test accuracy score of 96.22%.



**Performance of features extracted from the Flattened Layer**

From the flattened layer of the CNN models, the corresponding higher dimensional features are extracted. These features that have the ability to distinguish between different target classes, are then passed on to ML classifiers for training and testing purposes. Different classifiers are trained with features of the train set. The performance of classifiers tests the CNN’s ability to capture distinguishable features. The below mentioned table shows the comparison of performances that are exhibited by the five ML classifiers for every model by means of test accuracy.



**Performance of features extracted from the Fully Connected Layer:**

Features are extracted from the fully connected layer of the CNN models for the respective input images. These features are then handed over to the ML classifiers for classification purposes. The ML classifiers are trained with the train set that were extracted by the CNN models and are then evaluated using the test set. The results in terms of test accuracy obtained by the ML classifiers for the four CNN models are tabulated in Table below.

