

Context: Due to limited resources, I've trained the model 1-5 for 5 epochs and remaining for 8 epochs.

Experiment Number	Model	Result	Decision + Explanation
1	Model with 3 Conv3d + Batch Normalization + Pooling layers	<ul style="list-style-type: none"> categorical_accuracy: 0.3092 val_categorical_accuracy: 0.2000 Overfitting, very low and fluctuating validation accuracy. 	Increase layers to extract more features.
2	Increasing Layer Set on Experiment 1	<ul style="list-style-type: none"> categorical_accuracy: 0.2443 val_categorical_accuracy: 0.2400 Number of parameters dropped sharply. 	Use more Kernels in dense layer of this model
3	Increasing Dense Layer Kernels on Experiment 2	<ul style="list-style-type: none"> categorical_accuracy: 0.2941 val_categorical_accuracy: 0.4400 Both validation and training accuracies are increasing steadily and the losses are also decreasing steadily. Also unlike Experiment 1, the number of parameters are much lower as well. 	Model is not able to converge in 5 (low) epochs so we shall try increasing speed with increasing batch size.
4	Increasing batch size on Experiment 3	<ul style="list-style-type: none"> categorical_accuracy: 0.4344 val_categorical_accuracy: 0.5200 The training and validation accuracies both struggle in the beginning but start to increase very slowly later. 	We shall try increasing the learning rate as well and check if the model starts performing better
5	Increasing learning rate on Experiment 4	<ul style="list-style-type: none"> categorical_accuracy: 0.5143 val_categorical_accuracy: 0.5600 Increasing learning rate did not help achieve the desired output. The model now overfits a lot. 	We shall try to increase the number of epochs of model and bring down the batch size to 32 and see if it is a better one
6	Increasing epochs to 8 on Experiment 3	<ul style="list-style-type: none"> categorical_accuracy: 0.3756 val_categorical_accuracy: 0.4000 Both training and validation accuracy has reduced 	Still not getting desired output we will change the architecture for experimentation
7	CNN+RNN with MobileNetV2 Transfer Learning + GRU layers	<ul style="list-style-type: none"> categorical_accuracy: 0.8462 val_categorical_accuracy: 0.6700 Model is overfitting 	So we shall try adding dropout layer to mobilenetv2.
8	Decreasing the number of output neurons from MobileNet on Experiment 7	<ul style="list-style-type: none"> categorical_accuracy: 0.7783 val_categorical_accuracy: 0.5200 The model is still overfitting and there has been no improvement. 	We will now try combining multiple changes with some increase in parameters and some decrease. Also we shall tweak optimiser as well
9	Decreasing batch size and learning rate, increasing GRU Kernels	<ul style="list-style-type: none"> categorical_accuracy: 0.7677 val_categorical_accuracy: 0.6200 The model has a high tendency of overfitting. Adding dropouts and the learning rate reduction didnt help much 	There is not much improvement.

Conclusion: Increase the number of epochs and till the model converges. As of now the model doesn't converge and not in a shape to be used in production.