|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **MODEL** | **EXPERIMENT NUMBER** | **HYPERPARAMETERS & LAYERS** | **PARAMETERS** | **RESULT** | **DECISION + EXPLANATION** |
| Conv3D | 1 | Two Conv3D layers of 32 and 64 neurons each. One dense layer of 128 neuron  No of epochs = 10  Optimizer = adam | 6,481,285 | Accuracy: 0.73  Loss: 1.62 | Very heavy model with higher loss. Also tending to overfit. Let’s try adding dropouts |
| 2 | Two Conv3D layers of 32 and 64 neurons each including dropouts. One dense layer of 128 neuron  No of epochs = 20  Optimizer = adam | 6,481,285 | Accuracy: 0.62  Loss: 2.05 | Model is not able to learn due to high number of parameters |
| 3 | Four Conv3D layers two of 32 and 64 neurons each. One dense layer of 128 neuron  No of epochs = 20  Optimizer = adam | 18,112,997 | Accuracy: 0.61  Loss: 2.84 | Heavy model with low accuracy and higher loss. Model is not able to learn after some accuracy. Let’s add normalization and dropouts |
| 4 | Four Conv3D layers two of 32 and 64 neurons each including batchnormalization and dropouts. One dense layer of 128 neuron  No of epochs = 10  Optimizer = adam | 18,112,997 | Accuracy: 0.38  Loss: 10.55 | Looks like normalization had negative effect on the model. Also lets reduce the parameter |
| 5 | Four Conv3D layers of 32, 64, 128 and 265 neurons each, including dropouts. One dense layer of 128 neuron  No of epochs = 10  Optimizer = adam | 1,246,981 | Accuracy: 0.62  Loss: 0.95 | Looks like accuracy is increasing and loss is reducing. Let’s increase the number of epochs |
| Same model with 30 epochs | 1,246,981 | Accuracy: 0.86  Loss: 0.59 | This is the best model so far with decent parameters |
| Conv2D + GRU | 6 | Two Conv2D layers of 32 and 64 neurons each, with 128 dense layer and 128 GRU layer  No of epoch= 10  Optimizer = adam | 6,541,765 | Accuracy: 0.64  Loss=1.45 | Model is not learning after 0.6 accuracy and loss is increasing. Let’s increase the layers |
| 7 | Three Conv2D layers of 32, 64 and 128 neurons each, with 128 dense layer and 128 GRU layer. Along with dropouts  No of epoch= 30  Optimizer = adam | 2,961,989 | Accuracy: 0.69  Loss: 1.56 | Accuracy seems to get flatten at around 0.65 while loss gradually increased |
| Conv2D + LSTM | 8 | Two Conv2D layers of 8 and 16 neurons each. ConvLSTM2D layer of 128 neurons and a dense layer  No of epochs = 30  Optimizer=adam | 1,441,653 | Accuracy: 0.61  Loss: 1.91 | In this model as well, accuracy seems to get flatten at around 0.60 while loss gradually increased. Let’s try making a dense layer with dropouts |
| 9 | Three Conv2D layers of 32, 64, 128 neurons each. ConvLSTM2D layer of 32 neurons and a dense layer of 64  No of epochs = 30  Optimizer=adam | 525,893 | Accuracy: 0.62  Loss: 1.49 | In this model as well, accuracy seems to get flatten at around 0.60. Let’s increase the cells of LSTM and try |
| 10 | Three Conv2D layers of 32, 64, 128 neurons each. ConvLSTM2D layer of 18 neurons and a dense layer of 128  No of epochs = 30  Optimizer=adam | 3,256,645 | Accuracy: 0.72  Loss: 1.57 | Accuracy seems to get flatten at 0.65 and loss also get flatten at 1.8 |

Beside the above experiments lot of other experiments were performed changing the optimizers and kernel sizes, but results were more or less similar

Final Model







