This is a sample write-up. The write-up need not be in tabular form.

It doesn’t state that ConvLSTM will give you better results than Conv3D. The explanation should be as detailed as possible so that the logic behind the decision is conveyed. Also, there are a lot of things you can experiment with in the generator function and elsewhere. Please do not forget to specify the exact metric values, here Accuracy which drives your decision.

You can draw inspiration from the concepts taught in the Industry demo in CNNs to experiment with the data and different architectures.

|  |  |  |  |
| --- | --- | --- | --- |
| **Experiment Number** | **Model** | **Result** | **Decision + Explanation** |
| **1** | **Model-1**  **Conv3D** | **Total parameters: 42,224,773**  **Trainable parameters: 42,224,773**  **Non- Trainable parameters: 448**  **ModelAccuracy: 32.238**  **Val accuracy : 20.0** | **Crop the images correctly, try to overfit on less amount of data** **Tried to hyper tune the network and apply layers and other possible ways to get model performance and train and validation accuracy.** |
| **2** | **Model -2**  **Here we change optimizer “Adam” to "SGD" & Added L2 Regularization**  **Conv3D** | **Total parameters: 1,344,453**  **Trainable parameters: 1,343,973**  **Non- Trainable parameters: 480**  **ModelAccuracy: 67.16**  **Val accuracy: 60.02** | **Reduce the size of the image/Reduce the number of layers Increased the amount of trainable data/ reduce the filter size. Here we will increase the accuracy hence we are trying different CNN layers and trying to reduce over fitting while increasing the accuracy decreasing loss.** |
| **3** | **Model -3**  **Adding More CNN Layers**  **Conv3D** | **Total parameters: 1,400,357**  **Trainable parameters: 1,399,429**  **Non- Trainable parameters: 928**  **ModelAccuracy: 68.80**  **Val accuracy : 58.99** | **Increase the amount of trainable data/ reduce the filter size We Added more hidden layers for control the over fitting of training data and increase the val accuracy.** |
| **4** | **Model-4**  **Reducing Filter (2,2,2) & Added more Hyper Tuning Parameters**  **Conv3D** | **Total parameters: 1,139 ,253**  **Trainable parameters: 1,338,773**  **Non- Trainable parameters: 480**  **ModelAccuracy: 61.79**  **Val accuracy: 44.00** | **Reduced the size of the image and. we have also tried to hyper tuning of parameters through dropout and Batch normalization for batter accuracy.** |
| **5** | **Model -5 Adding one more dense layers** | **Total parameters: 746,293**  **Trainable parameters: 745,213**  **Non- Trainable parameters: 480**  **ModelAccuracy: 39.85**  **Val accuracy : 37.00** | **We Added one more dance layers with reduced filter size of (2,2,2) so it will increase the accuracy.** |
| **6** | **Model – 6**  **CNN or SIMPLE RNN** | **Total parameters: 755,557**  **Trainable parameters: 755,077**  **Non- Trainable parameters: 480**  **ModelAccuracy: 92.68**  **Val accuracy : 60.00** | **Here we used CNN and RNN for sequential feature analysis and producing the good accuracy but it might get overfit.** |
| **7** | **Model – 7**  **CNN or LSTM** | **Total parameters: 1,230,181**  **Trainable parameters: 1,229,701**  **Non- Trainable parameters: 480**  **ModelAccuracy: 72.98**  **Val accuracy : 56.99** | **We observed that vanilla architecture for CNN+RNN requires more hyper tuning due to over fitting hence we tried to add dropout and increase neurons to gather as much as input from image processing.** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| **Final Model** | **Model – 8**  **CNN or GRU** | **Total parameters: 1,936 ,485**  **Trainable parameters: 1,936 ,005**  **Non- Trainable parameters: 480**  **ModelAccuracy: 84.02**  **Val accuracy : 62.00** | **Here we combined CNN or LSTM in order to check the accuracy but we observed major gap between accuracy and validation accuracy. It is all over good to compare of other models.** |