Group – Ambareesh and Parimal

As per Jeff Heaton, author of “Introduction to Neural Networks in Java”, the optimal size of the hidden layer is usually between the size of the input and size of the output layers. Since the output layer size is 5, we want to start with maximum layers to overfit and hence started with 5 layers.

To make the model more adaptive and the subsequent hidden layers has to learn smaller details, we went by doubling the neuron size in each layer.

**H5 file google drive link** - <https://drive.google.com/drive/folders/1dImij88VHJXe6srkGtmao_HfR-CJ68u7?usp=sharing>

|  |  |  |  |
| --- | --- | --- | --- |
| **Experiment Number** | **Model** | **Result** | **Decision + Explanation** |
| **Model 0** | **Conv3D**  batch\_size = 16  img\_height = 120  img\_width = 120  num\_epochs = 5  - **without** augmentation  - **without** Batch normalization | Training accuracy: 16.74%  Validation accuracy: 21.00% | Model has very very low training accuracy. Now next step is to do Batch normalization to see the effect. |
| **Model 0\_1** | **Conv3D**  batch\_size = 16  img\_height = 120  img\_width = 120  num\_epochs = 5  - **without** augmentation  - **with** Batch normalization | Training accuracy: 69.38%  Validation accuracy: 37.00% | Model still has low training accuracy but improved alot after batch normalization was added. Next step is to add augmentation to see the effect. |
| **Model 1** | **Conv3D**  batch\_size = 16  img\_height = 120  img\_width = 120  num\_epochs = 5  - **with** augmentation  - **with** Batch normalization | Training accuracy: 72.25%  Validation accuracy: 45.00% | After adding batch normalization and augmentation, Model improved a lot from the base model. But still training and validation accuracy is low. |
| **Model 2** | **Conv3D**  batch\_size = 16  img\_height = 80  img\_width = 80  num\_epochs = 5 | Trainingaccuracy: 71.19%  Validation accuracy: 35.00% | Training accuracy went down as well as validation accuracy also went down |
| **Model 3** | **Conv3D**  batch\_size = 16  img\_height = 80  img\_width = 80  num\_epochs = 5  -> changing the optimizer | Trainingaccuracy:  71.04%  Validation accuracy: 69.00% | After changing the optimizer, it has impact on validation accuracy. It almost double. |
| **Model 4** | **Conv3D**  batch\_size = 32  img\_height = 120  img\_width = 120  num\_epochs = 10 | Trainingaccuracy:  87.03%  Validation accuracy:  25.00% | Changing the batch size increased the training accuracy but it reduced the validation accuracy. Model is overfitting. |
| **Model 5** | **Conv3D**  batch\_size = 20  img\_height = 120  img\_width = 120  num\_epochs = 10  -> adding drop outs after convolution layer | Trainingaccuracy:  82.20%  Validation accuracy:  62.00% | Adding dropout and reduced the training accuracy. But in this case Batch size was also reduced. Next step is to keep Batch size same as model 4 to see the impact. |
| **Model 6** | **Conv3D**  batch\_size = 32  img\_height = 120  img\_width = 120  num\_epochs = 10  -> adding dropouts after convolution layer | Trainingaccuracy:  58.82%  Validation accuracy:  23.00% | After increasing batch size to 32, and adding drop outs after convolution layer. training as well ass validation accuracy dropped drastically |
| **Model 7** | **Conv3D**  batch\_size = 32  img\_height = 160  img\_width = 160  num\_epochs = 10  -> back to initial model(removed drop outs) | Trainingaccuracy:  93.06%  Validation accuracy:  47.00% | Comparing with Model 4, we can see validation accuracy increased after increasing the image size. But it is still very low.  Next we will reduce the batch size to 20 and will increase epoch to see the impact. |
| **Model 8** | **Conv3D**  batch\_size = 20  img\_height = 160  img\_width = 160  num\_epochs = 15 | Trainingaccuracy:  Validation accuracy: 72.00% | Now after 15 epoch, training accuracy is very high and validation accuracy is 72%. We need to see how to increase the validation accuracy. Next step to increase the epoch. |
| **Model 9** | **Conv3D**  batch\_size = 20  img\_height = 120  img\_width = 120  num\_epochs = 20 | Trainingaccuracy:  93.21 %  Validation accuracy:  84.00% | This model gave the best training and validation accuracy so far. |
| **Model 10** | **Conv3D**  batch\_size = 20  img\_height = 120  img\_width = 120  num\_epochs = 20  filter size --> (2,2,2) | **Training accuracy: 96.68%**  **Validation accuracy:**  **85.00%** | After putting a filter of 2,2, accuracy figure reached 96% and validation is 85%. **The best model.** |
| **Model 11** | **Conv3D + LSTM**  batch\_size = 20  img\_height = 120  img\_width = 120  num\_epochs = 20 | Trainingaccuracy: 60.63%  Validation accuracy:  58.00% | CNN + LSTM model does not give high accuracy with lstm\_cells=64 |