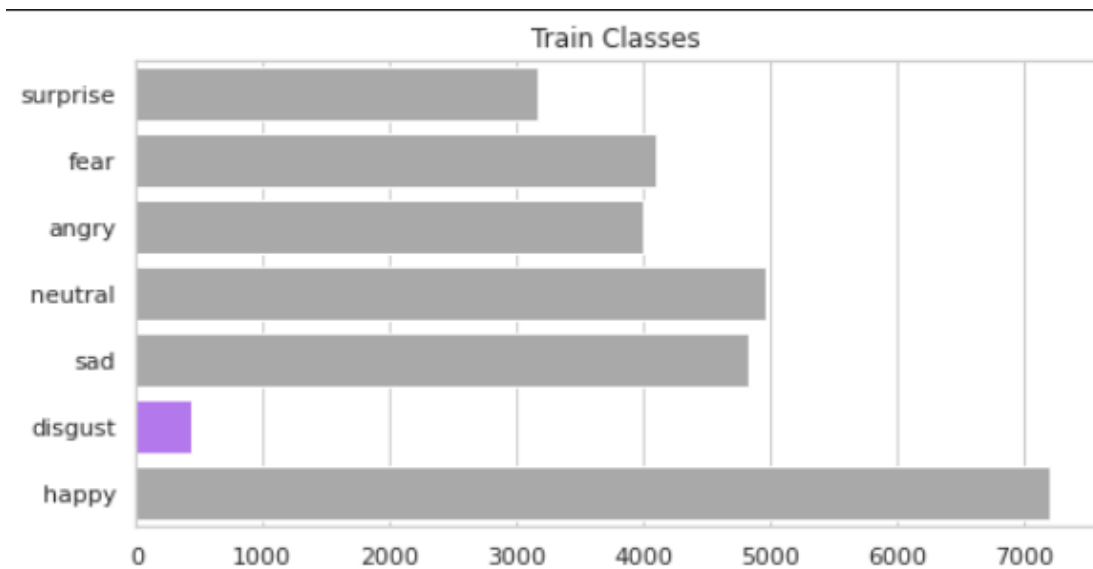# SOI : Hardly Human

Team - Intrinsic Nubs

To classify emotions using **fer1013**, we used a convolutional neural network. But before that let us take a close look at the dataset.

The below figure shows the distribution of no. of training images present under each folder.


Augmentation

Oh! There seems to be a problem, no. of training images for class *disgust* is very less, while for that of *happy* is around 7000! To fix this issue we used techniques of ***Oversampling and Undersampling.***

- **Oversampling:** We randomly picked images from each class except *happy* and augment them using the following transforms, until 5000 images were there in total.
    a. Random Horizontal Flips
    b. Random 20 degree Rotations.
- **Undersampling:** We randomly deleted images from class *happy* until we had around 5000 images left.

After data augmentation we find the following file distribution!

Train Classes
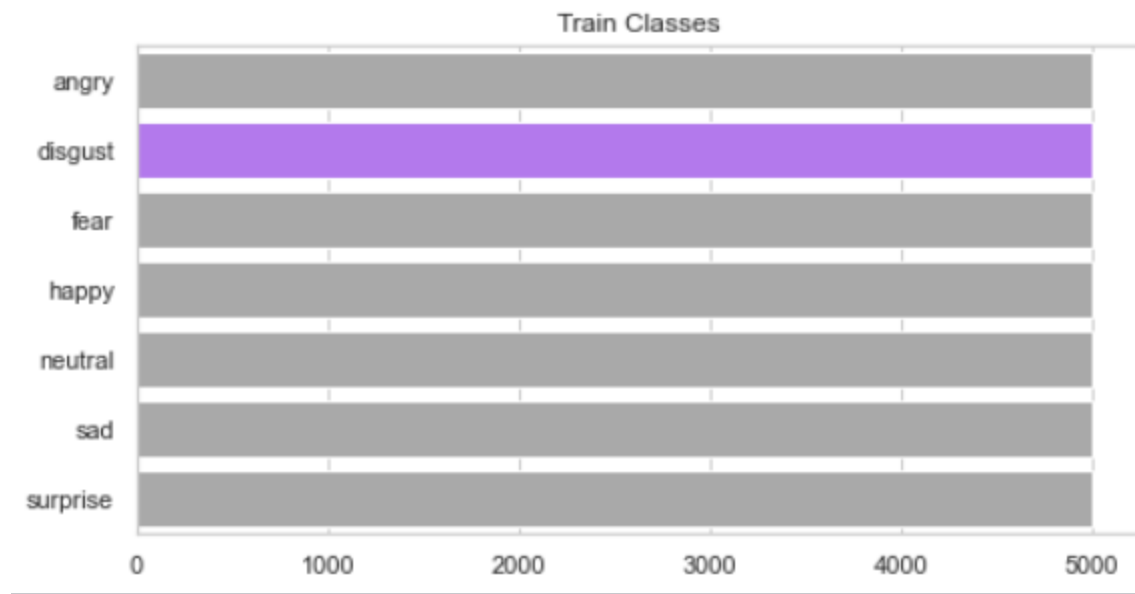
Then we normalized the training data by dividing each element by 255.

Architecture

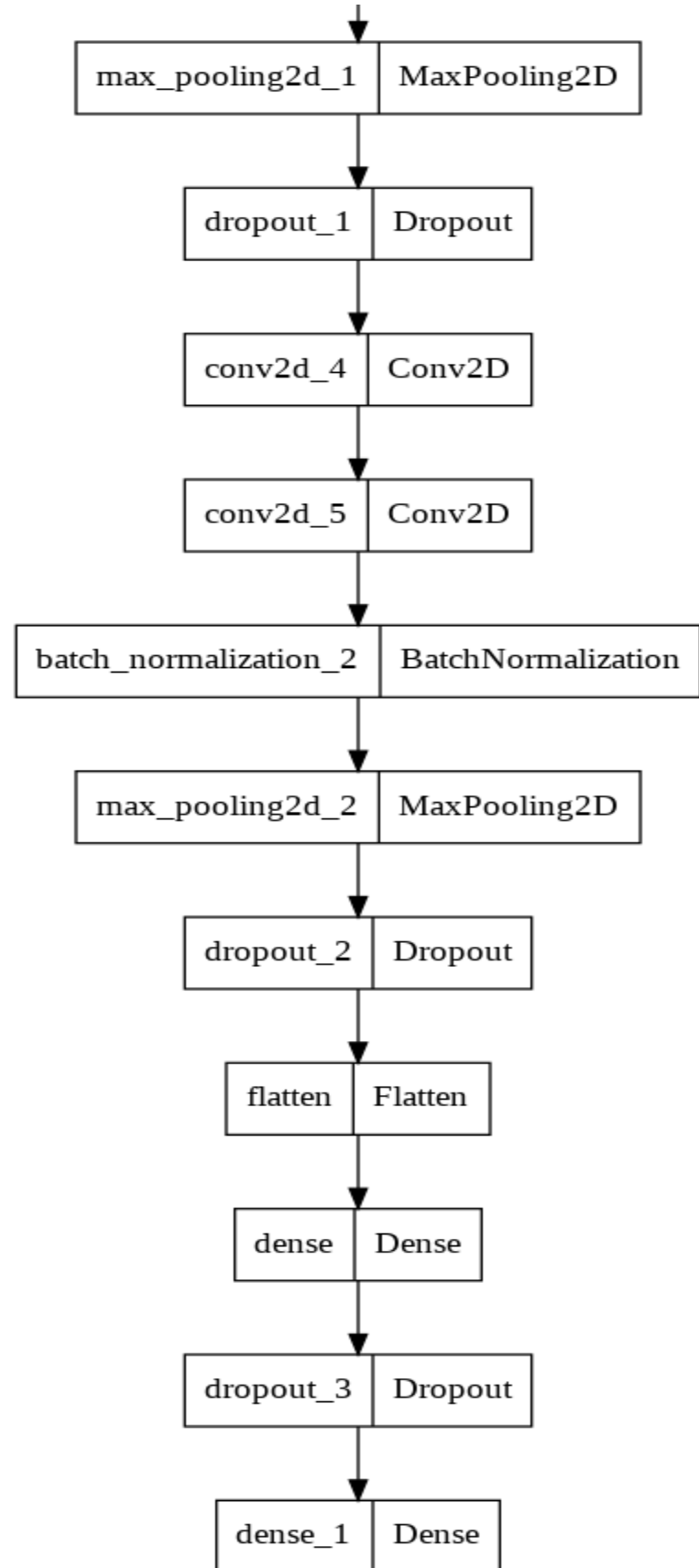Next let us look at the architecture of our CNN.

- We used 6 convolutional layers.
- After every 2 layers we added MaxPooling, Batch Normalization and Dropout.
- We kept the probability of dropout to be 0.25.
- In the end we added 2 dense layers with dropout set to 0.5.

The exact parameters used are visible in *SOI_HH_Training.ipynb* file. An image depicting our model architecture is shown in the next page.

Trainable Parameters	8,815,559
Non Trainable Parameters	840
Total Parameters	8,816,399

The model was trained using Adam optimizer with the following add-ons.

- Saving the best model based on validation accuracy after each epoch.
- Initial learning rate as 0.0001.
- Decay of 1e-6.

```
conv2d_input | InputLayer
        |
        v
conv2d | Conv2D
        |
        v
conv2d_1 | Conv2D
        |
        v
batch_normalization | BatchNormalization
        |
        v
max_pooling2d | MaxPooling2D
        |
        v
dropout | Dropout
        |
        v
conv2d_2 | Conv2D
        |
        v
conv2d_3 | Conv2D
        |
        v
batch_normalization_1 | BatchNormalization
        |
        v
max_pooling2d_1 | MaxPooling2D


max_pooling2d_1 | MaxPooling2D
        |
        v
dropout_1 | Dropout
        |
        v
conv2d_4 | Conv2D
        |
        v
conv2d_5 | Conv2D
        |
        v
batch_normalization_2 | BatchNormalization
        |
        v
max_pooling2d_2 | MaxPooling2D
        |
        v
dropout_2 | Dropout
        |
        v
flatten | Flatten
        |
        v
dense | Dense
        |
        v
dropout_3 | Dropout
        |
        v
dense_1 | Dense
```

Metrics

After training the following metrics were obtained for the training and test sets.

Training Accuracy	83.49 %
Testing Accuracy	66.95 %

The final F1-score was obtained to be **0.647**

The following images show the complete classification report

.

- **Classification report.**

Training					Testing				
	precision	recall	f1-score	support	Classificationm Report				
						precision	recall	f1-score	support
0	0.78	0.78	0.78	5000					
1	0.98	1.00	0.99	5000	0	0.56	0.62	0.59	958
2	0.79	0.67	0.73	5000	1	0.73	0.65	0.69	111
3	0.92	0.92	0.92	5000	2	0.56	0.44	0.49	1024
4	0.72	0.83	0.77	5000	3	0.89	0.84	0.87	1774
5	0.76	0.70	0.73	5000	4	0.57	0.70	0.63	1233
6	0.88	0.93	0.91	5000	5	0.58	0.53	0.55	1247
					6	0.75	0.81	0.78	831
accuracy			0.83	35000					
macro avg	0.83	0.83	0.83	35000	accuracy			0.67	7178
weighted avg	0.83	0.83	0.83	35000	macro avg	0.66	0.66	0.66	7178
					weighted avg	0.67	0.67	0.67	7178

- **Labels**

Index	Label
0	angry
1	disgust
2	fear
3	happy
4	neutral
5	sad
6	surprise

- **Confusion matrix.**

Training	Testing
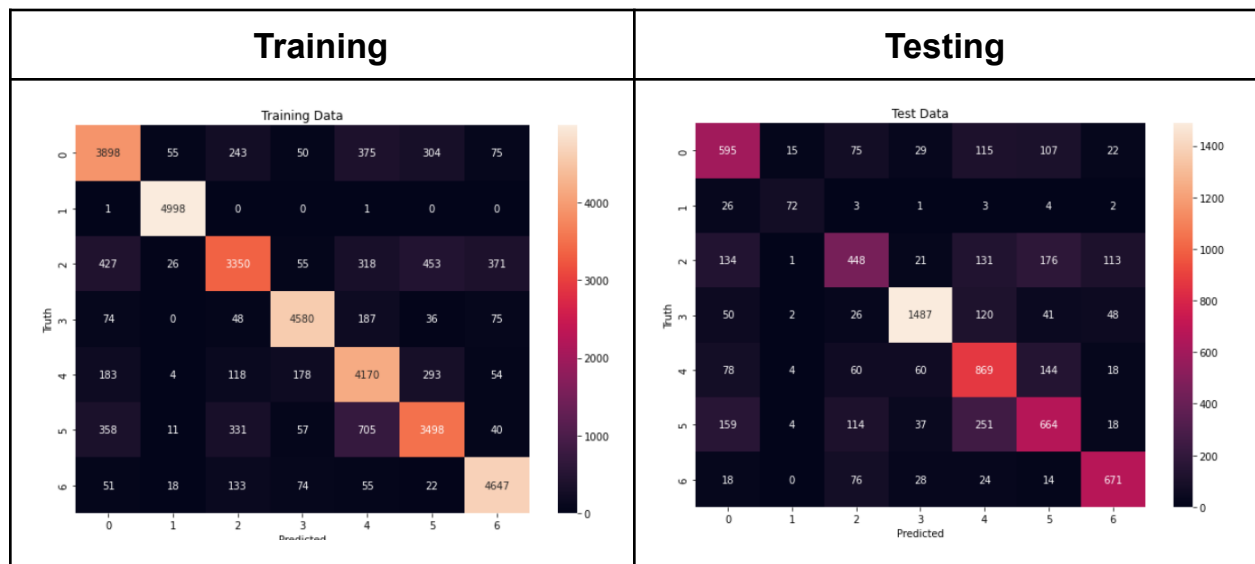
- *These reports can be found in 'SOI_HH_Metrics.ipynb' file.*

Team: **Intrinsic Nubs.**

- Shashank P - 200010048

- Tabish Khalid Halim - 200020049