Assignment 2 (CS 747): Building CNN classifiers for images

Pankaj Kumar Jatav – G10338769

Goal:

This assignment consists of building, validating, and testing CNN binary classifiers for the same repository of images you used in Assignment 0.

Approach:

I followed the below diagram step to solve this assignment.

Pre-Processing of Data:

Below are the steps taken to pre-process that data.

1. Move the data into two folder names food and no-food so I am can use flow_from_directory method from ImageDataGenerator class of TensorFlow lib for loading images.
2. Divide the each image pixels by 255 ie ImageDataGenerator(rescale=1./255).

Model Training:

I have used the same archeture as Alexnet from activity 2.

Model: "sequential"

Layer (type)	**Output Shape**	**Param #**
conv2d (Conv2D)	**(None, 55, 55, 96)**	**34944**
batch_normalization (BatchNormalization)	**(None, 55, 55, 96)**	**384**
max_pooling2d (MaxPooling2D)	**(None, 27, 27, 96)**	**0**
conv2d_1 (Conv2D)	**(None, 27, 27, 256)**	**2973952**

batch_normalization_1 (Batc (None, 27, 27, 256) 1024
hNormalization)

conv2d_2 (Conv2D) (None, 27, 27, 384) 885120

batch_normalization_2 (Batc (None, 27, 27, 384) 1536
hNormalization)

conv2d_3 (Conv2D) (None, 27, 27, 384) 1327488

batch_normalization_3 (Batc (None, 27, 27, 384) 1536
hNormalization)

conv2d_4 (Conv2D) (None, 27, 27, 256) 884992

batch_normalization_4 (Batc (None, 27, 27, 256) 1024
hNormalization)

max_pooling2d_1 (MaxPooling (None, 13, 13, 256) 0
2D)

flatten (Flatten) (None, 43264) 0

dense (Dense) (None, 4096) 177213440

dropout (Dropout) (None, 4096) 0

dense_1 (Dense) (None, 4096) 16781312

dropout_1 (Dropout) (None, 4096) 0

dense_2 (Dense) (None, 2) 8194

===

For models trining I have used the tensorflow library and below are the parameter for used in model training.

```
dropout = [0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]
early_stopping = [1, 3, 5, 10, 50]
regularization = [0, 1, 0.5, 0.1, 0.01, 0.001, 0.0001, 0.00001, 0.000001]
```

Model with vgg16:

Model: "sequential"

_____

Layer (type) Output Shape Param #

===

vgg16 (Functional) (None, 7, 7, 512) 14714688

global_average_pooling2d (G (None, 512) 0

lobalAveragePooling2D)

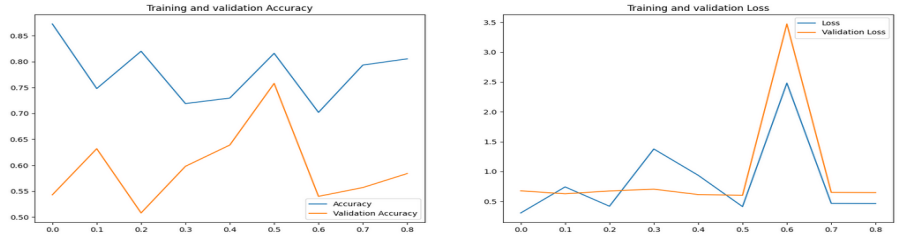
dense (Dense) (None, 2) 1026

==========

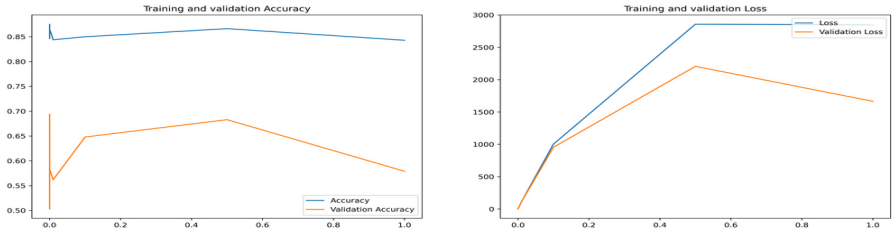
Model Selection(Results):

Model 1 : (OWN CNN)

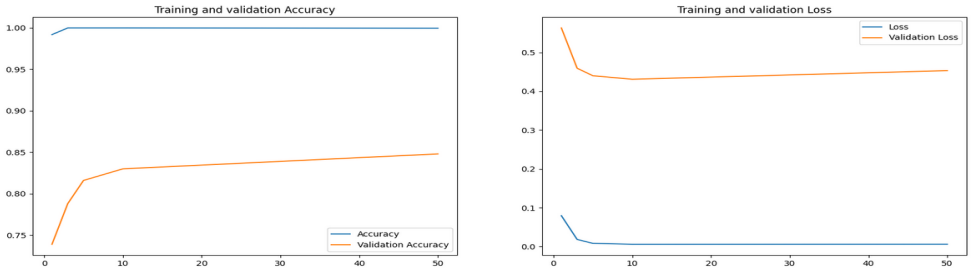
The model selection is done by selecting the best performing model on different hyperparameter(dropout, early stopping and regularization value) on validation data. The maximum accuracy was 0.8680 and loss 0.5183 on evaluation data.
Below is the plots for the same. Also same values can be find on miscellaneous folder.



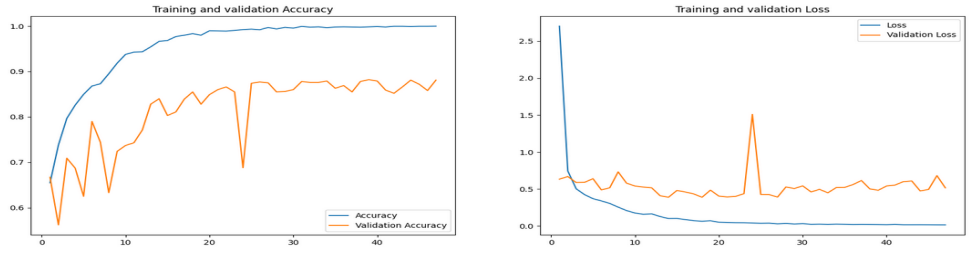
Plot a) Model Tunning for dropout



Plot b) Model Tunning for regularization



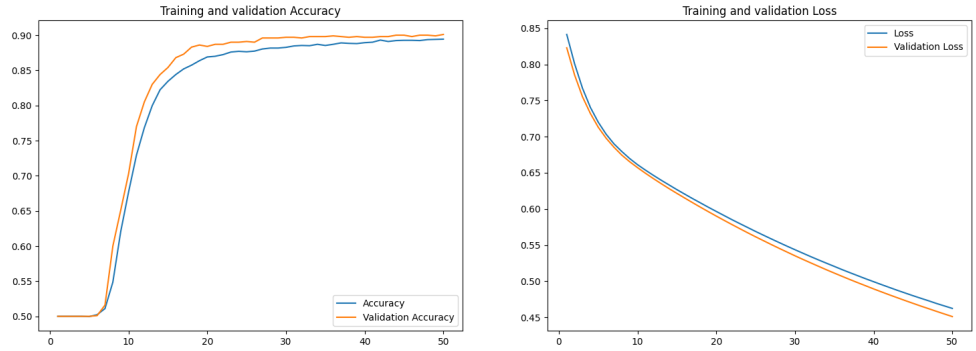
Plot c) Model Tunning for Early Stop



Plot d) Final Model Traing Using Dropout

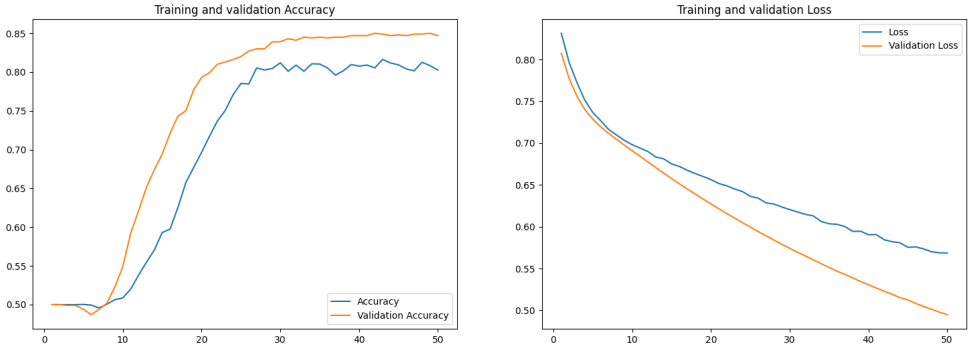
Model 2 (Transfer Learning without augmentation)

The accuracy was 0.8830 and loss: 0.4606 on evaluation data. Below is plot for model training.



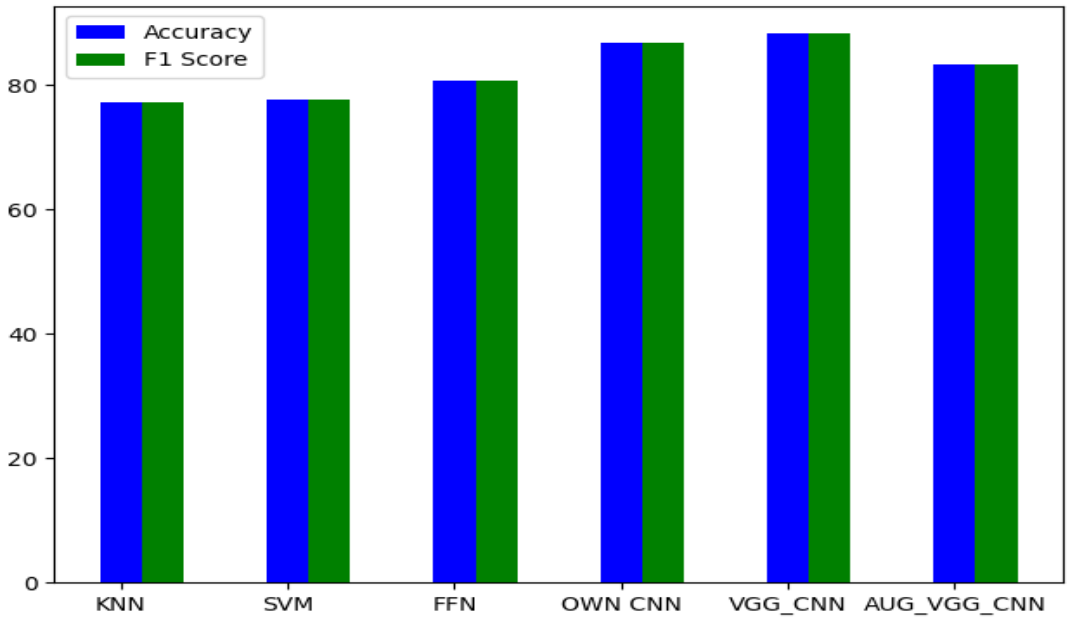
Model 2 (Transfer Learning with augmentation)

The accuracy was 0.8340 and loss: 0.5050 on evaluation data. Below is plot for model training.



Resuls Comparison:

Model without augmentation outperform the others and give the best accurary on evaluation data which is 80.7 percent.



Conclusion:

1. Custom cnn model has few noticeable points.
    1.1. The model does not perform well with Early Step(plot c), and model become highly overfitted with traning data.
    1.2. Too low and Too high value of Regularization value does not perform well on validation data set.
    1.3 The dropout perform well on traning and validation data set. At value 0.5 model give high accuracy and low loss for traning and validation data set.

2. Model without augmentation perform quite well as compare other and give high accuracy. But as we can notice from the plot adding too much epochs does not really help with traning and validation accuracy and loss.

3. Model with augmentation gave lowest accuray compare to other 2 CNN models. The reason for that may be extra noise or missing extra features from the new images.