Datasets Description

The dataset consists of small pathology images with binary labels (0 and 1). A positive label (1) are images of patients with cancer and at the center of these images, there is at least one pixel of tumor tissue in the region of 32x32px. The images with negative labels do not contain this tumor tissue. Our task here is to classify images by detecting this tumor tissue to either positive or negative class

Image Visualization

To understand the images better I will first display 5 images from each class to check if there is any visual difference between the images of both labels.

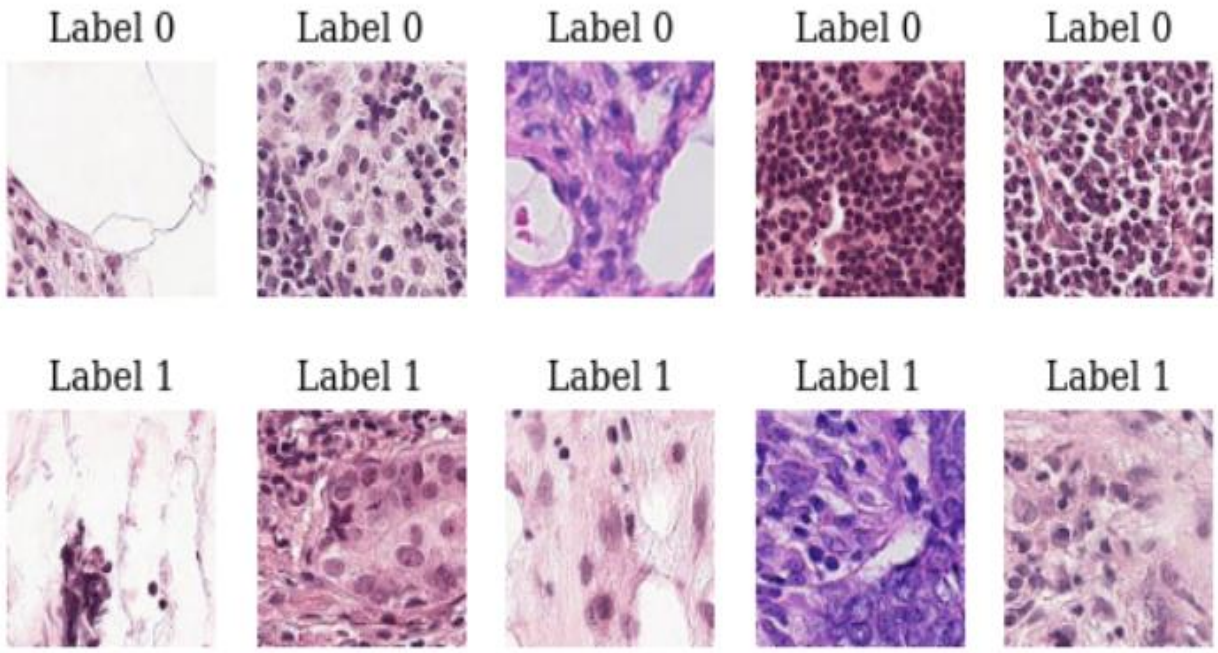


Figure 1 Images belonging to both class labels.

As we can see from the images above a person without domain knowledge can't detect the difference between both classes by looking at them.

EDA

The data is divided into both test and train datasets. There are a total of 220,025 images in training data and 57458 images in test data. The labels for training data are provided and the target is to predict the presence/absence of tumor in test data images.

EDA Issues

Since the dataset consists of images, we have limited options for performing data analysis here. I will apply the following techniques for the data analysis.

1. Analysis of the distribution of labels in test data
2. Analysis of outliers based on RGB values of images.

Distribution of Labels

I created a histogram to examine the distribution of label values in the training data. The above diagram reveals that there are nearly 130,000 images associated with class 0 and around 90,000 images associated with class 1. This indicates that while the data is not perfectly balanced, the amount of data for the minority class is sufficiently substantial for the classification task.

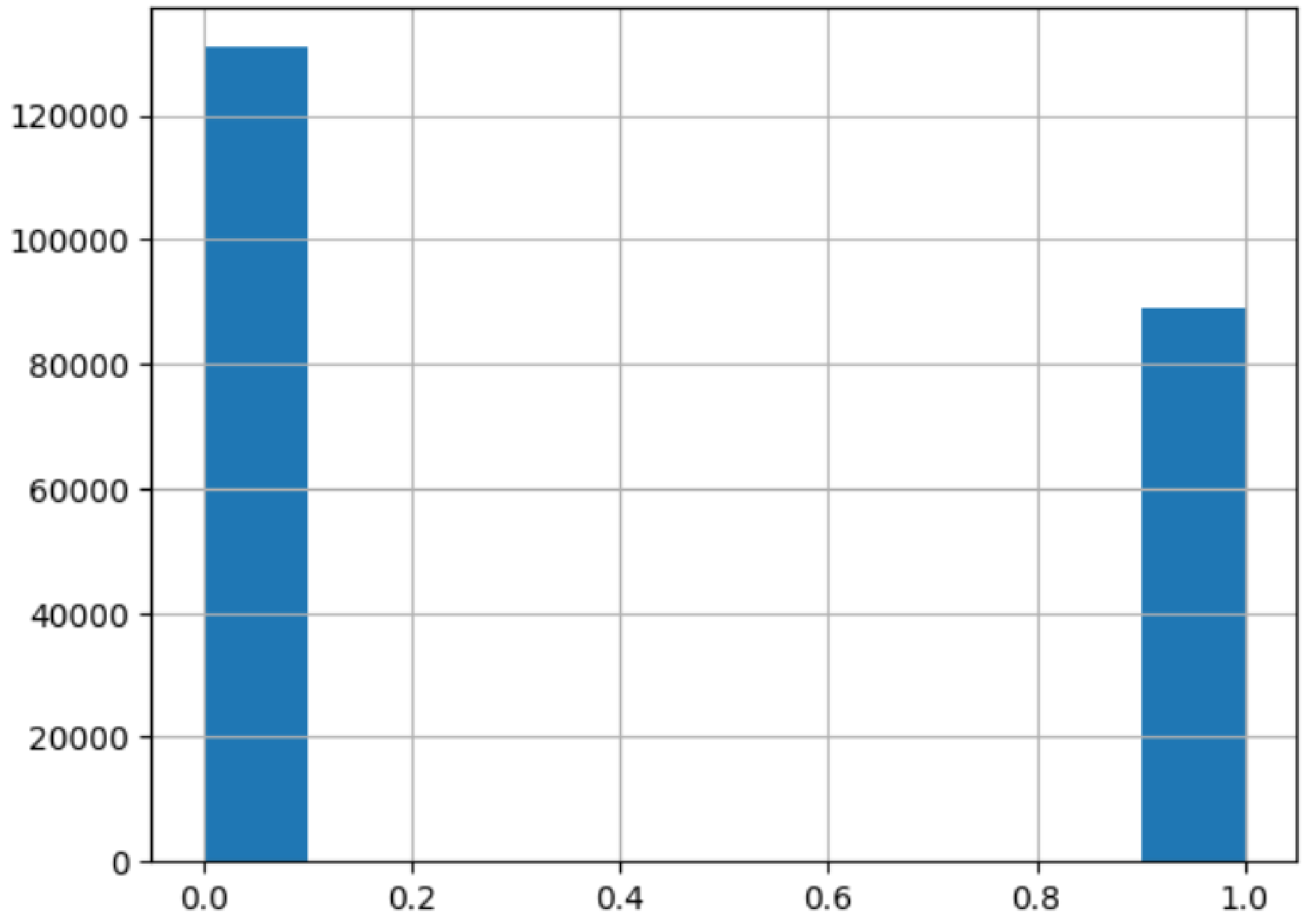


Figure 2 Class label distribution in training data

Analysis of outliers based on RGB values of images.

Since we cannot simply detect outliers from images with traditional approaches. I checked the average values of colors in the images. If the average values of the image are very low (<45) or very high (>245) then this image is an outlier. By using this technique, no outliers are detected in the data.

Model selection and evaluation.

To classify the image data, CNN is one of the most suitable options. I will build and train the CNN model and also use pre-train models to select the best model among them. Here are the models that I will apply for the classification.

1. CNN

2. VGG (Pre-Trained, Frozen)

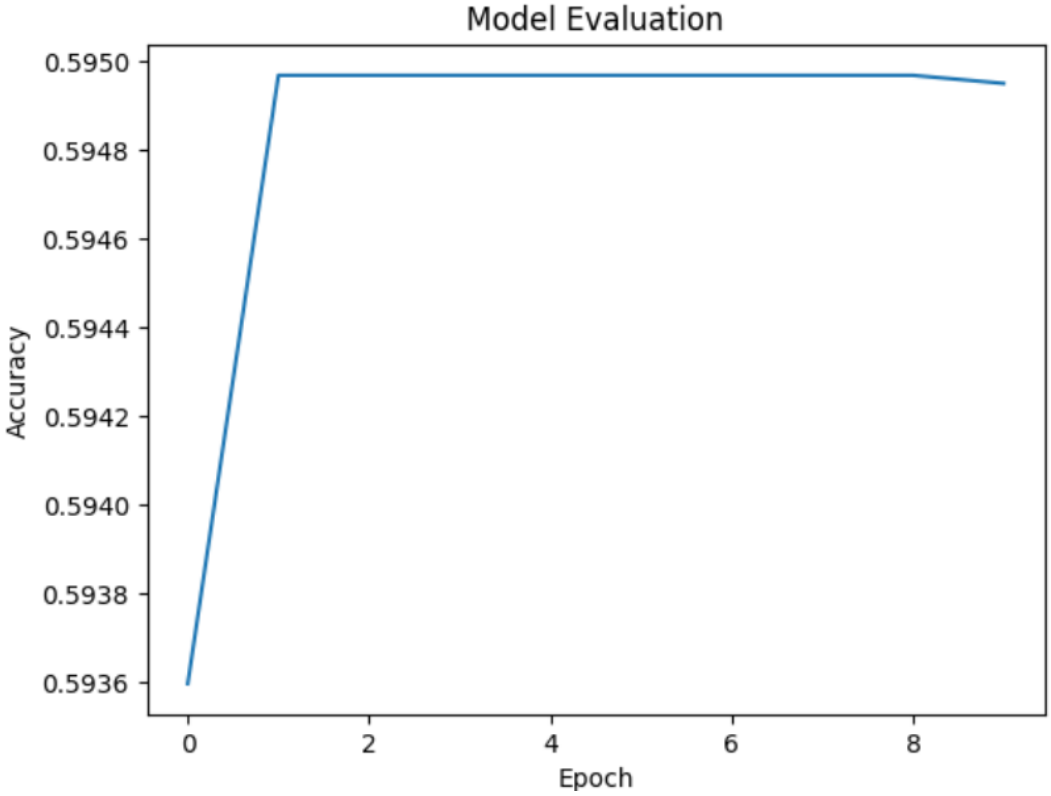
3. VGG (Pre-Trained, Trainable)

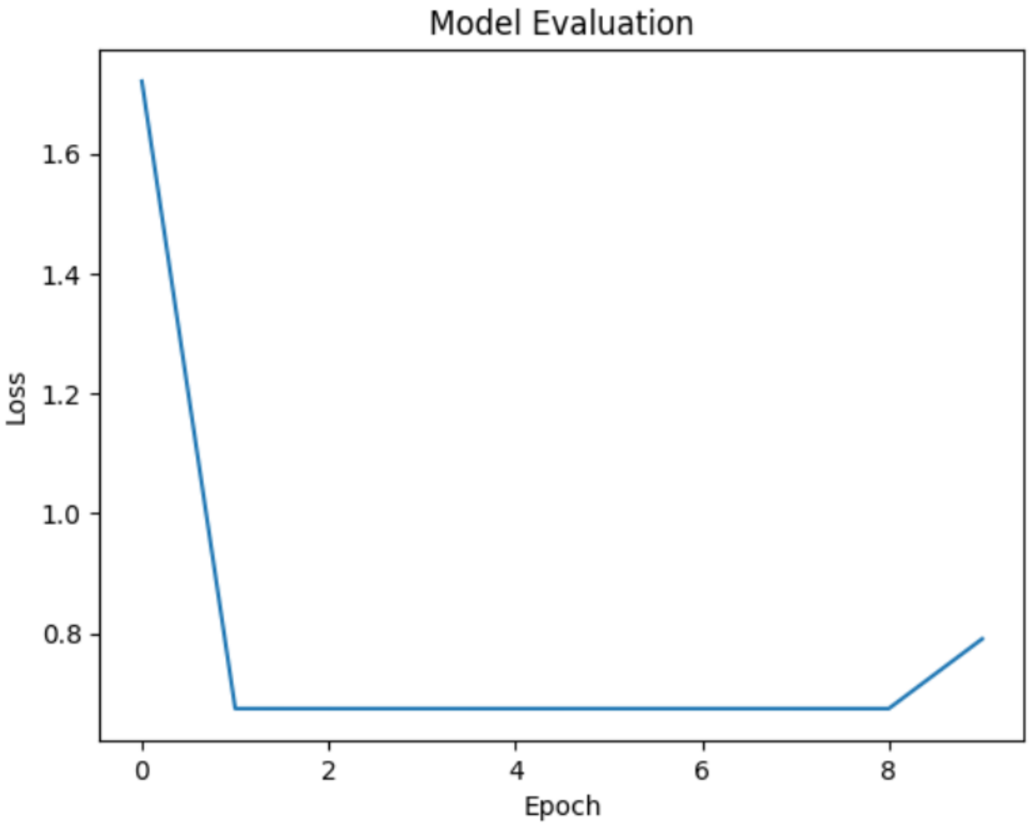
4. Restnet50 (Pre-Trained, Frozen)

5. Restnet50 (Pre-Trained, Trainable)

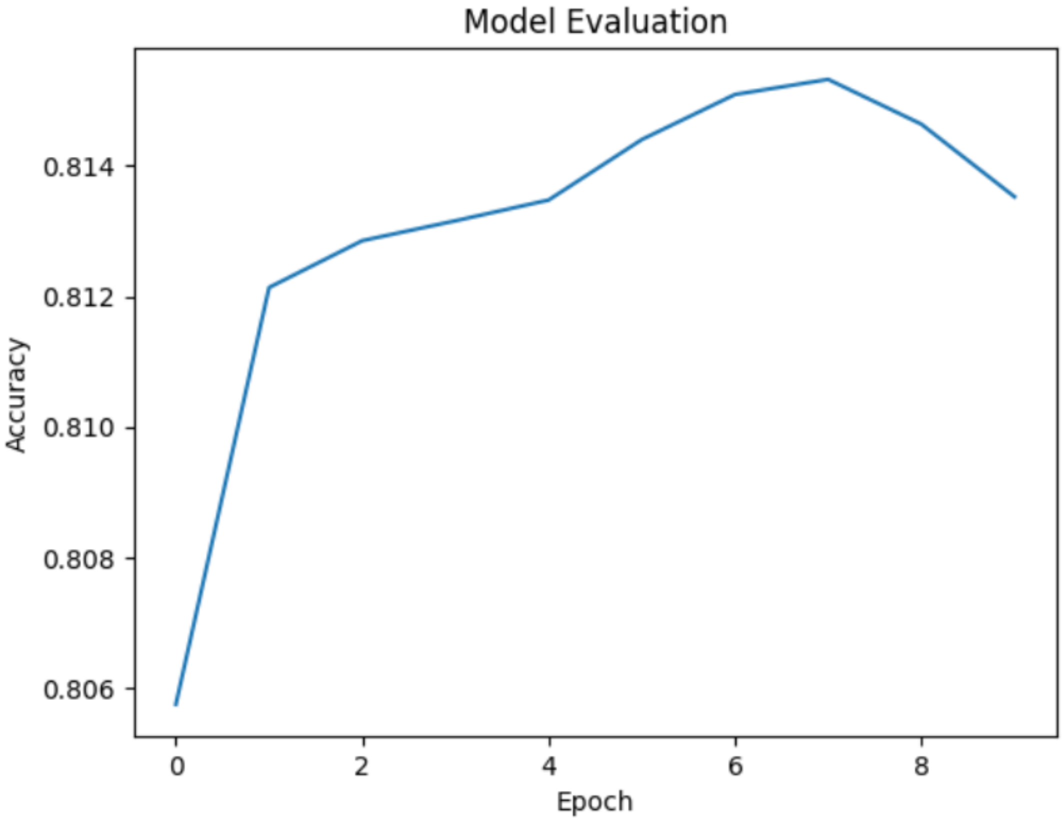
Accuracy and loss graphs of all the models are shown below:

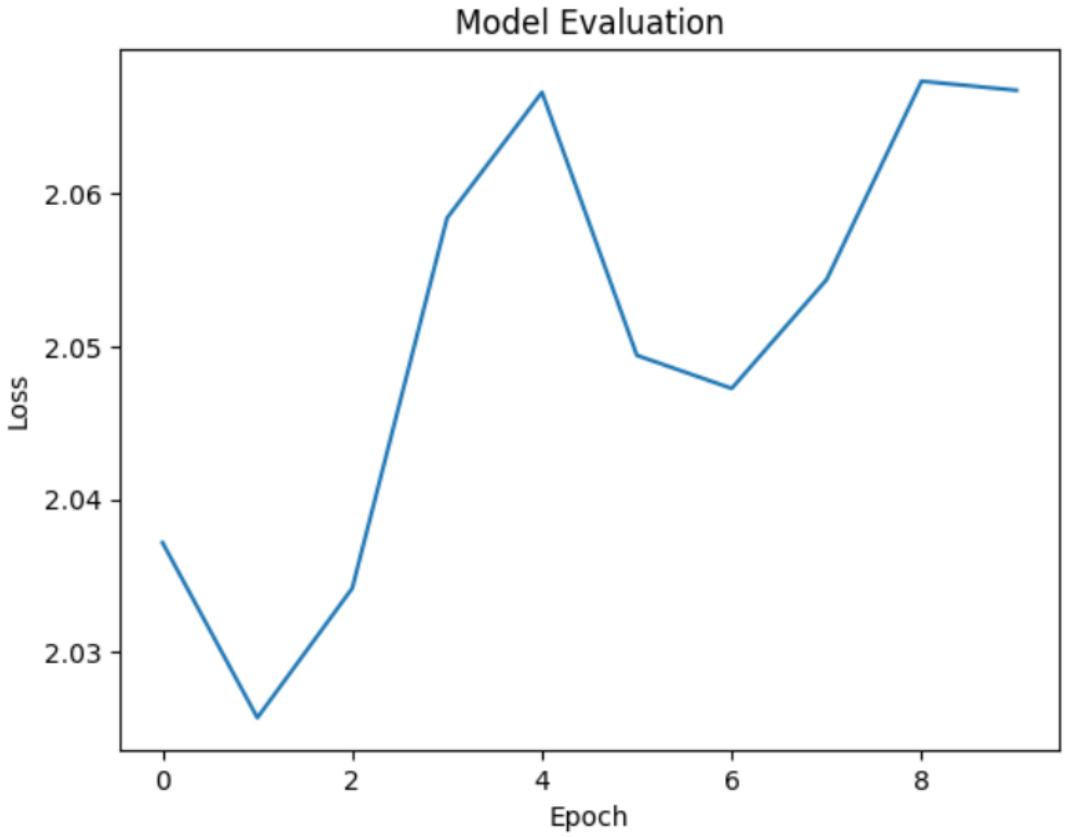
1. CNN



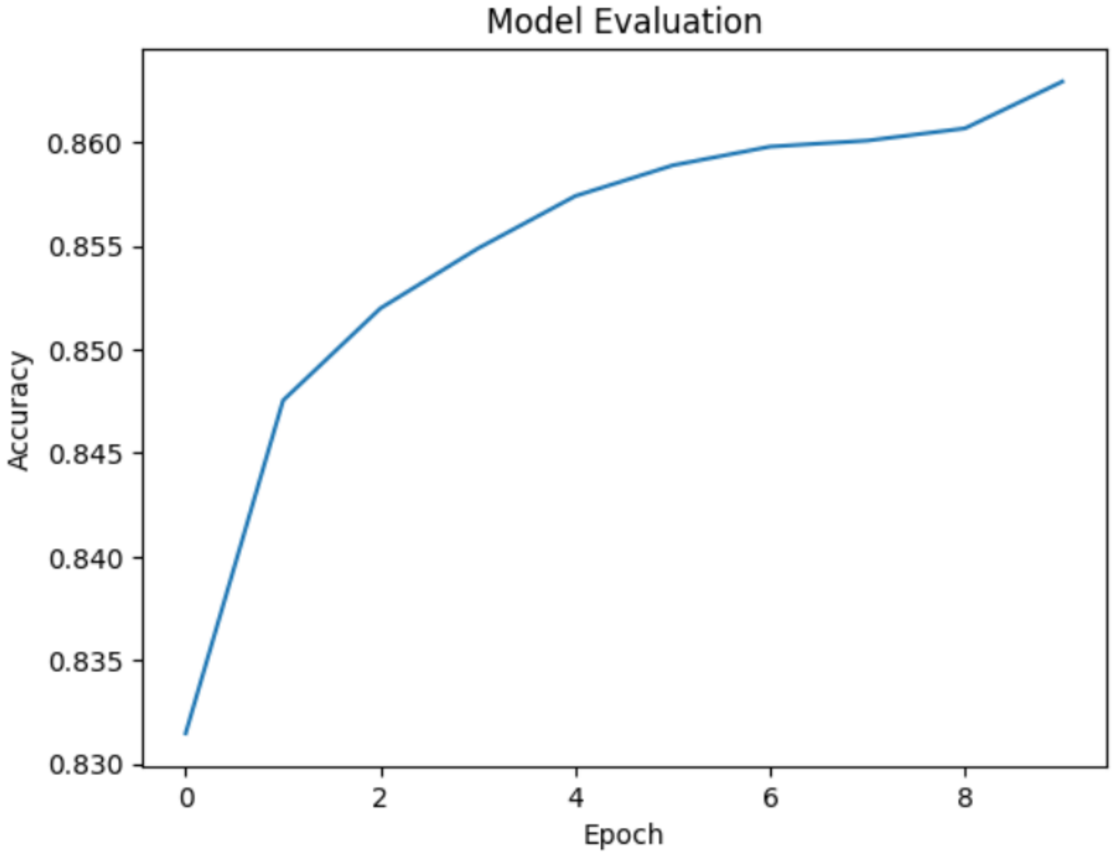


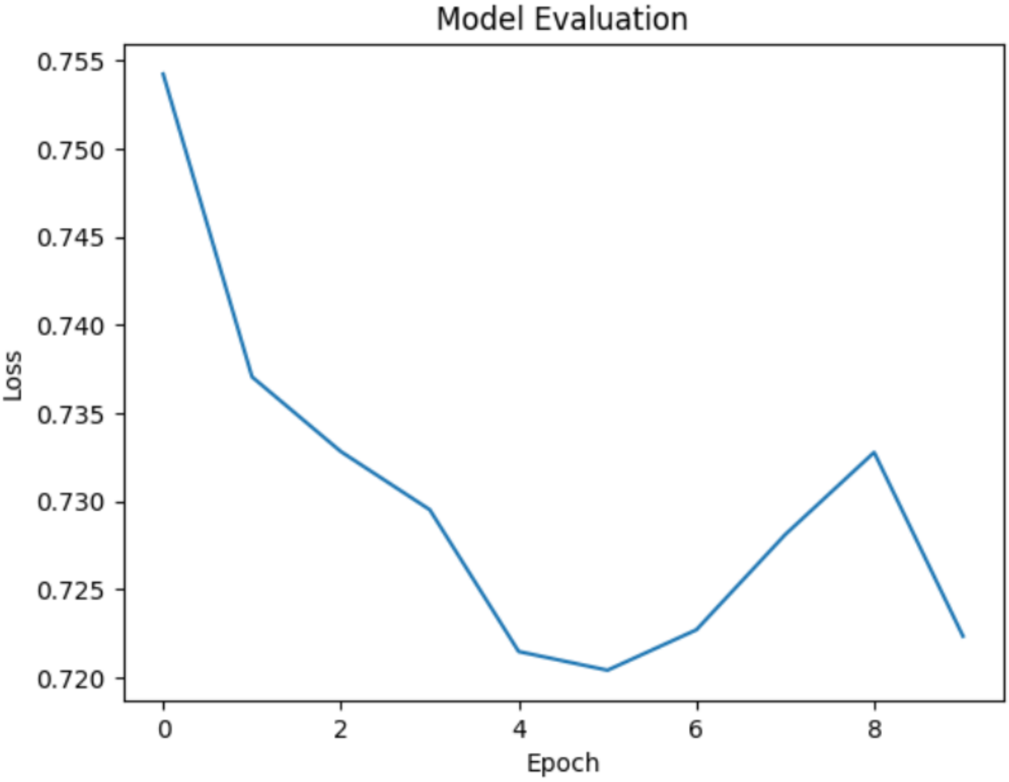
2. VGG (Pre-Trained, Frozen)



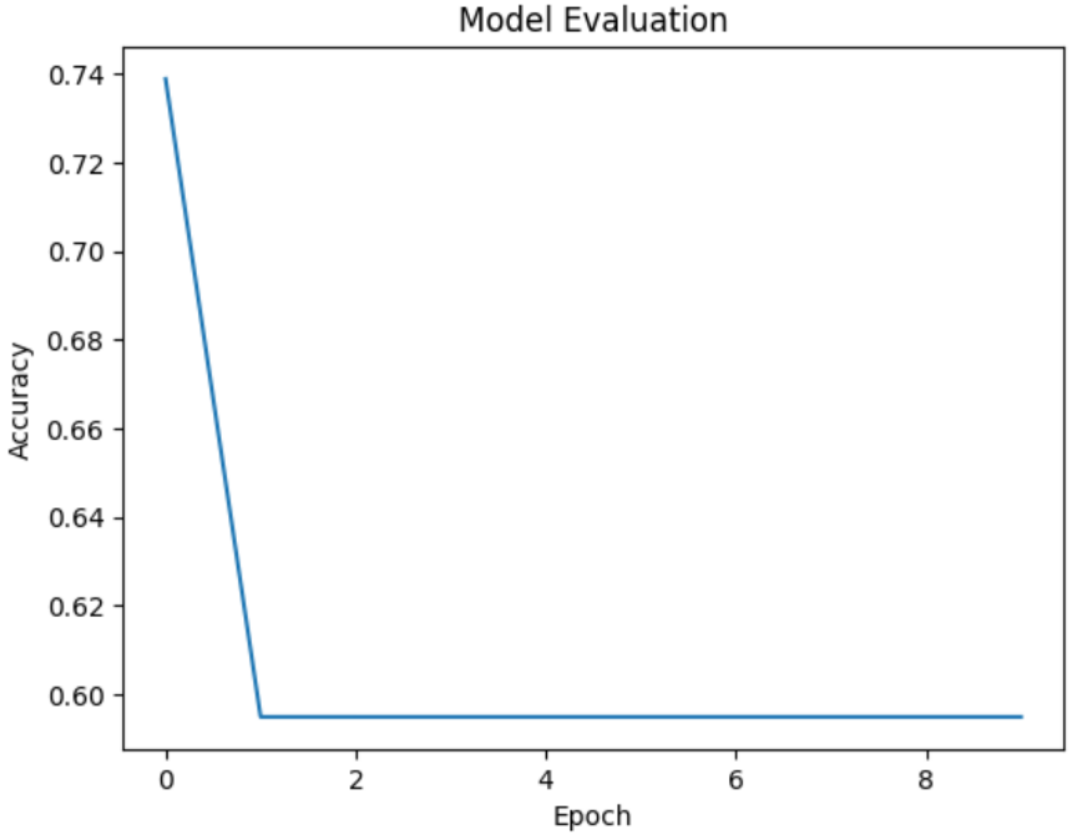

Model Evaluation
Loss
2.06
2.05
2.04
2.03
Epoch
0
2
4
6
8

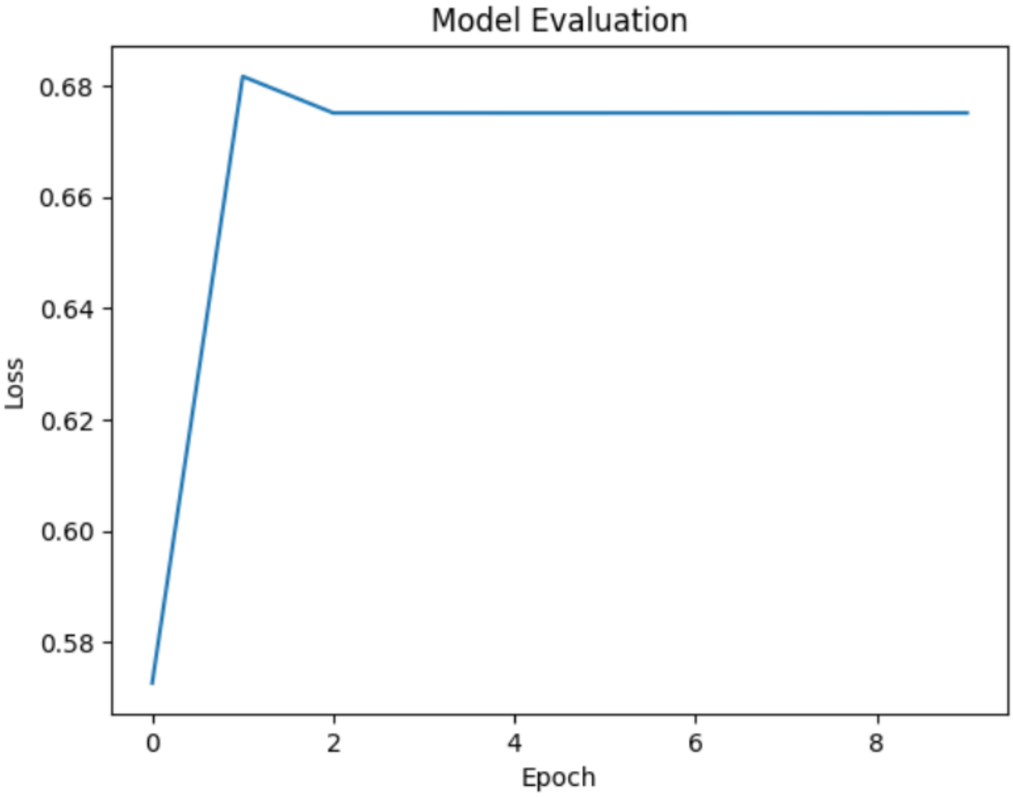
3. Restnet50 (Pre-Trained, Frozen)



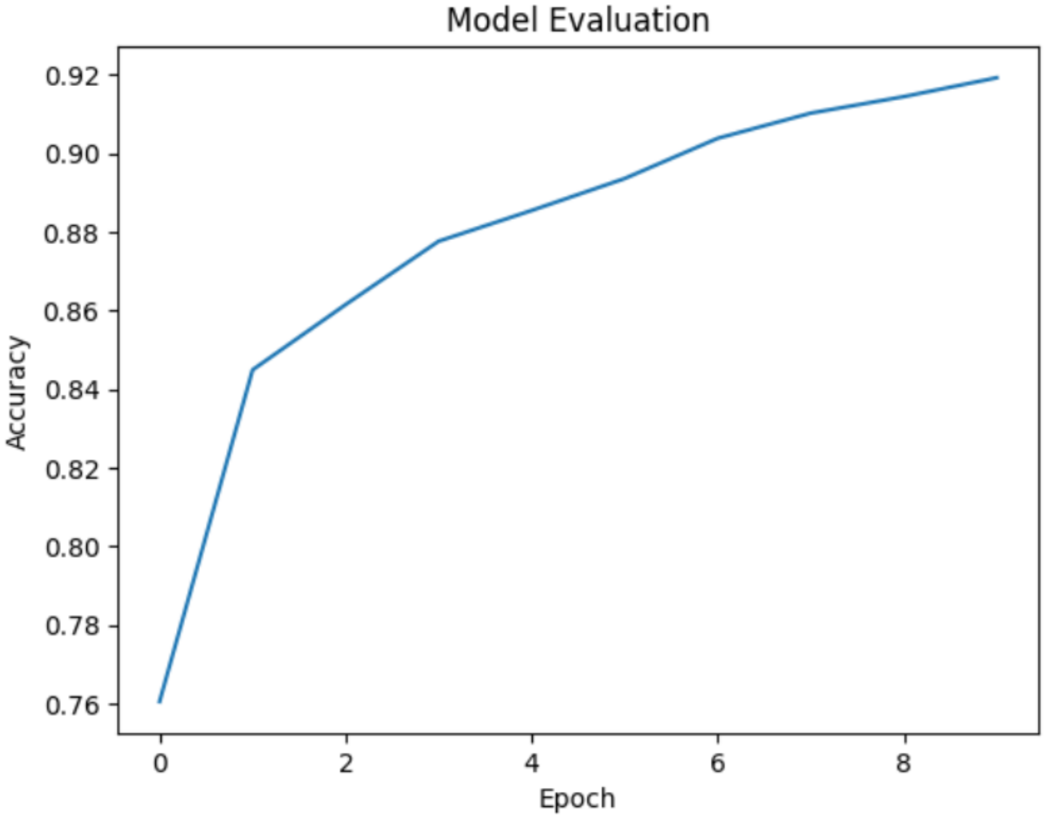


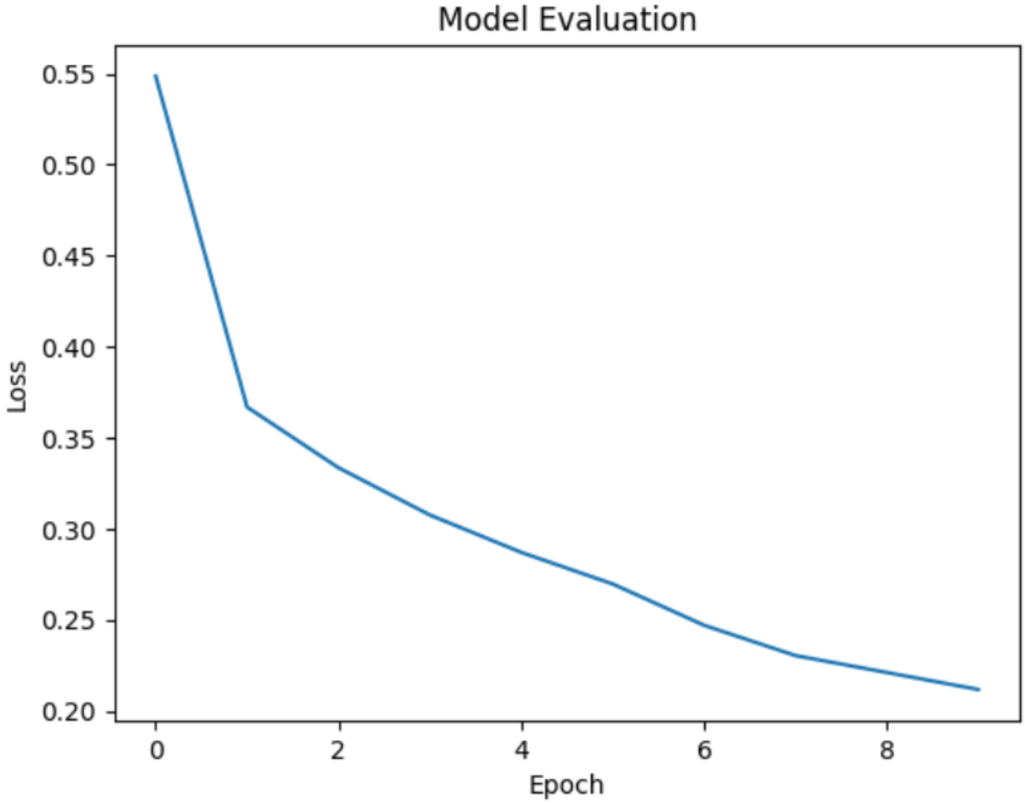
4. VGG (Pre-Trained, Trainable)





5. Restnet50 (Pre-Trained, Trainable)

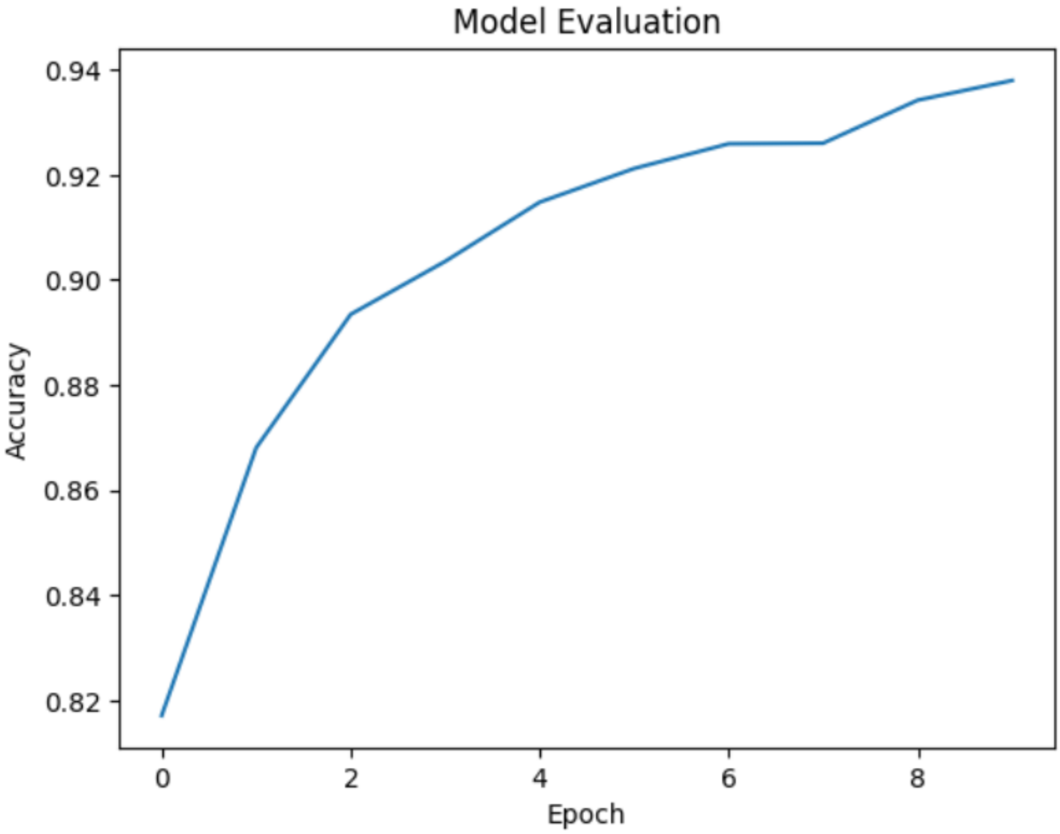


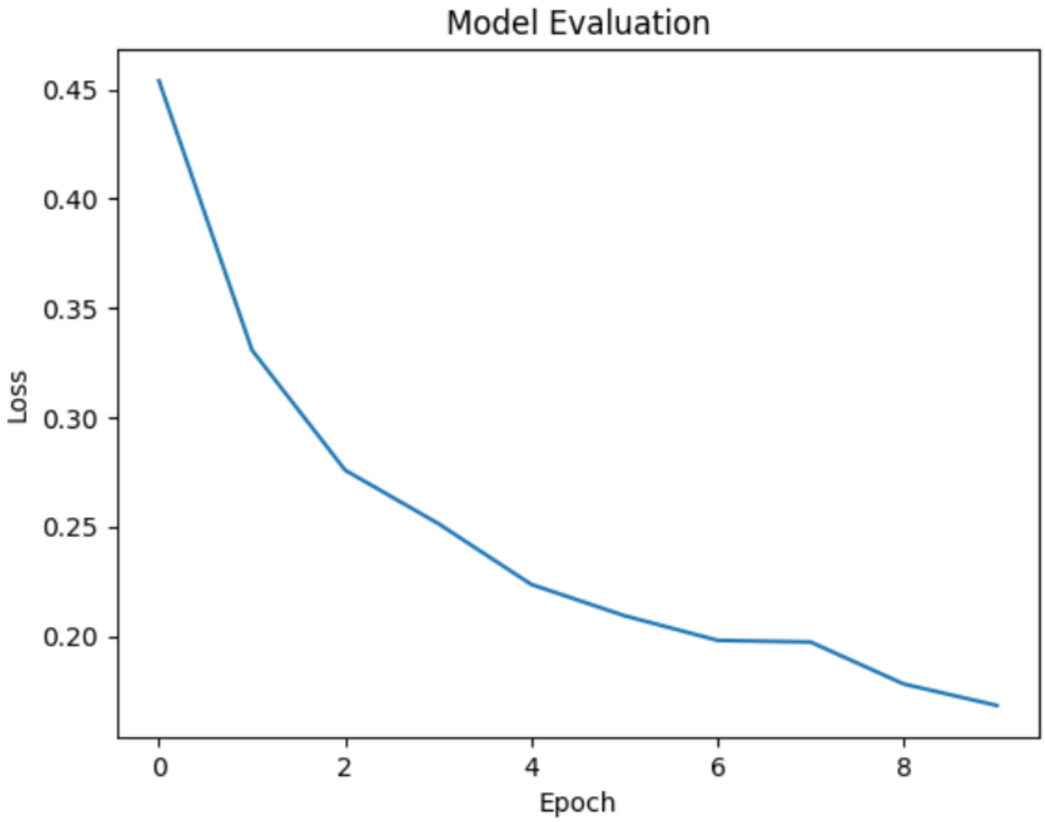


By analyzing all the figures above, it is clear that the retsnet 50 trainable model performs best for this problem.

Hyperparameter Tuning

To tune the parameters of restnet50 trainable model, we used different learning rate and the value 10^{-3} gave the best results. Here are plots for the loss and accuracy of our best model.





Conclusion

The dataset consists of images and the target problem was classification. We trained models and found that transfer learning with trainable models (restnet50) is best for this problem. Not all transfer learning models work well as we can see that the performance of the VGG model is worse. In the future, we can further work with parameter tuning and modifying the model by adding some extra layers to check the performance.