Machine Learning Project Final Report

Jaskirath Singh (2018150)
Sarandeep Singh (2018364)
Simarjeet Singh (2018103)

Problem Statement - "Identify if a person is suffering from pneumonia (Binary Classification) using an image of his/her chest x-ray."

INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY **DELHI**

Related Work

The state of the art AI system uses a **convolutional neural network** pre-trained on medical images with **transfer learning techniques** to recognize the distinguishing features of X-ray images much faster and with significantly fewer training examples and less computational power.

The model had :

Metrics	Values
Accuracy	**92.8%**
Sensitivity(true +ve rate)	**93.2%**
Specificity(true -ve rate)	**90.1%**
AUC score	**96.8%**

Link
https://www.cell.com/cell/fulltext/S0092-8674(18)30154-5

Approaches used

Logistic Regression (Baseline)

Logistic Regression is our baseline model. For this, we took the pixel values of the images and passed them as input to model. We did hyperparameter tuning using GridSearchCV.

KNN

For KNN model, we extracted features from the images using VGG16's convolutional layers. Then used them for training and testing. Model selection and hyperparameter tuning was done by analyzing validation and training accuracies.

XGBClassifier

For XGBClassifier model, we extracted features from the images using vgg16's convolutional layers. Then used them for training and testing. Hyperparameter tuning was done using GridSearchCV.

Neural Networks (Best Results)

We used two methods in this approach. In the first method, we created a CNN with our own architecture and classifier layers. In the second method, we used the pretrained VGG16 model and replaced its lower classifier layers with our own fully connected layers. In this case, the training was done only on the classifier layers. In both methods, we used Adam optimizer.

Datasets used and Evaluation Metrics

Dataset

Dataset Description	Number of Images
Total Train Samples	5235 (1350: Normal, 3885: Pneumonia)
Total Test Samples	624(234: Normal, 390: Pneumonia)

Evaluation Metrics

- Accuracy
- Recall (True positive rate)
- Specificity (True negative rate)
- Area under the ROC Curve (AUC)
- F1 Score

Dataset Link

http://dx.doi.org/10.17632/rscbjbr9sj.2

Summary (Pneumonia: positive class)

Metrics\Model	Logistic Regression	KNN	XGBClassifier	Own defined CNN	VGG16 - Transfer Learning	State of the art AI system
Recall (test)	0.9897	0.9358	0.9564	0.9897	0.9333	0.932
Specificity (test)	0.3803	0.6239	0.7606	0.6623	0.8888	0.901
Test Accuracy	0.7612	0.8189	0.8830	0.8669	0.9166	0.928
Train Accuracy	0.9931	0.9249	0.9990	0.9195	0.9484	-
AUC (test)	0.6850	0.7799	0.8585	0.8260	0.9111	0.968
Normal F1 Score	0.54	0.72	0.83	0.79	0.89	-
Pneumonia F1 Score	0.84	0.87	0.91	0.90	0.93	-
Weighted F1 Score	0.73	0.81	0.88	0.86	0.92	-

Summary : Confusion Matrices

Logistic Regression

		Predicted 0	Predicted 1
Actual	0	89	145
	1	4	386

KNN

		Predicted 0	Predicted 1
Actual	0	146	88
	1	25	365

XGBClassifier

		Predicted 0	Predicted 1
Actual	0	178	56
	1	17	373

CNN

		Predicted 0	Predicted 1
Actual	0	155	79
	1	4	386

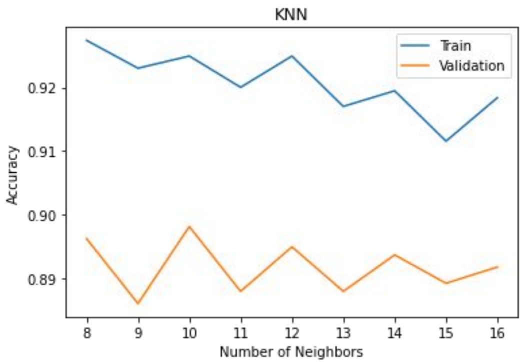
VGG16

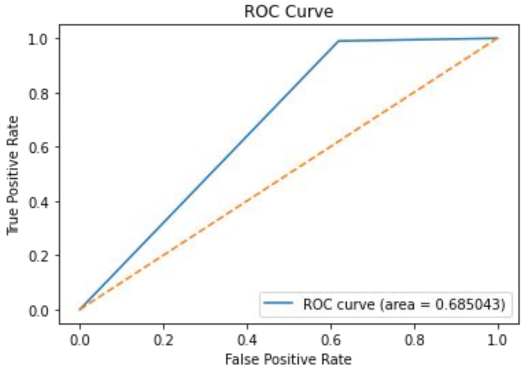
		Predicted 0	Predicted 1
Actual	0	208	26
	1	26	364

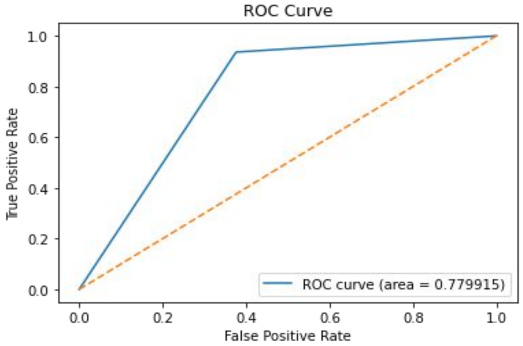
Analyses:

Logistic Regression

KNN

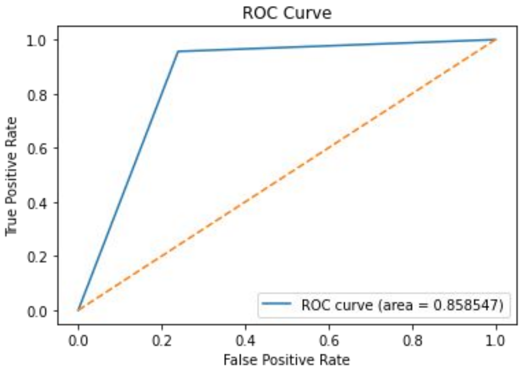




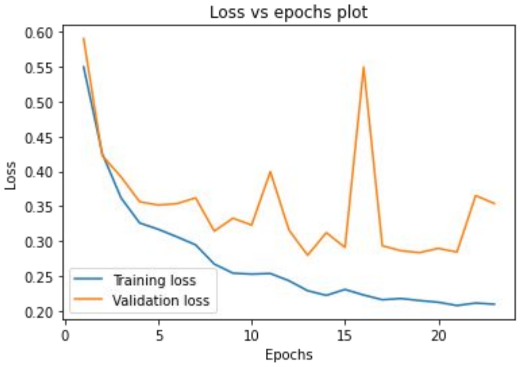


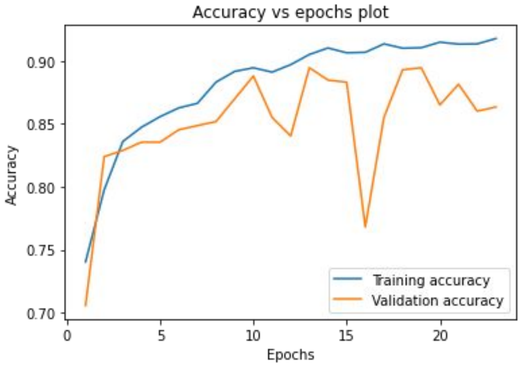
Analyses:

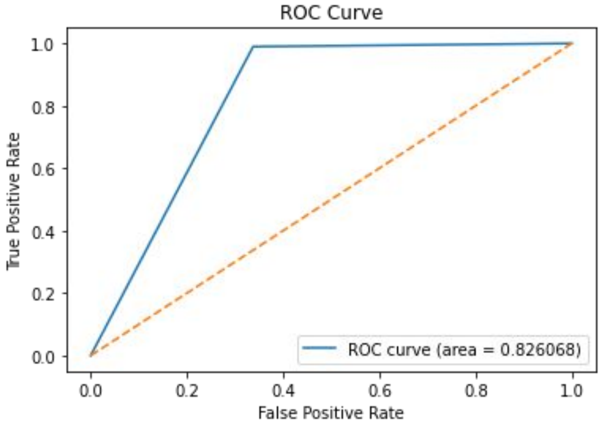
XGBClassifier



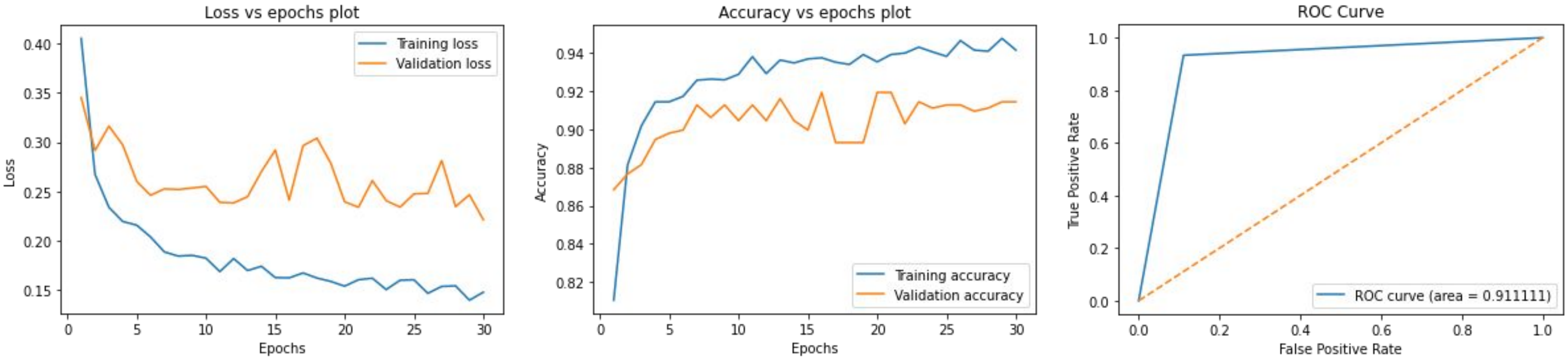
CNN







Analyses: VGG16-Transfer Learning



Note: Some extent of overfitting can be seen in Logistic regression, KNN and XGBClassifier. Overfitting was finally eliminated using transfer learning (VGG16) and CNN models.

Error Analyses: Improving over baseline models

- Earlier, features were taken as pixel values of the images but we improved our accuracy by generating features from VGG16 model's convolutional layers, which is pretrained on thousand of images. New features count - 8192.
- Earlier, our features were linearly separable (tested by linear SVM). This caused high overfitting, leading to low specificity. After passing through VGG16's convolutional layers, we improved the overfitting problem to some extent.

Old KNN results

Recall of test set: 0.9717948717948718
Specificity of test set: 0.5042735042735043
Accuracy on test set: 0.7964743589743589
ROC test set: 0.7380341880341881
Accuracy of train set: 0.9622477760902582

Improved KNN results

Recall of test set:0.9358974358974359
Specificity of test set: 0.6239316239316239
Accuracy on test set: 0.8189102564102564
ROC test set: 0.7799145299145299
Accuracy of train set: 0.9249454148471615

- Used GridSearchCV for improving error in XGBClassifier.
- Used validation error analysis to get best results from KNN.
- The first CNN model that we created didn't had the required complexity for this image classification task. So, we improved on this by applying transfer learning on a pre-trained image classification model of significant complexity and training size. This led to significant improvements in reducing errors.

Conclusion

We were able to create the models which were significantly close in performance to the actual state of the art AI system mentioned. But the state of the art AI system has a bit of advantage because it is already pre-trained on thousands of medical images. We achieved accuracy of 91.66% where as state of the art AI system has an accuracy of 92.8%.

Learning

- We learned how to work with image data sets, various types of preprocessing techniques used to handle images.
- We learned how to tune pre trained model for our classification task.
- We as a team learned how to break complex tasks into small steps and parts, how to manage time and refine understanding by regular discussions and explanation.
- We also got hands on experience on how to do machine learning tasks, how to handle them on industrial scale. We learned the basic methodologies and cycles of development of big machine learning projects.