Deep Learning

880008-M-6

Assignment

Using Deep Learning to Perform Multi-Class Classification on the Covid19 Chest X-ray Dataset

Report by:

Renee Pex - 2064669

Group Number: 29

Group Members:

Britt Geisler - 2014863

Thomas van der Woude - 2107919

Robbie Werner - 2038591

March 2023

1. Problem Definition

Image classification is a common machine learning task. Through image classification machine learning algorithms can be trained on a dataset containing X-ray chest photos and gain a higher accuracy score for correctly classifying the photos as showing signs of pneumonia or COVID-19. A baseline model has been given and the task is to improve on this baseline model by adding new layers, and experimenting with hyperparameters and algorithms that can improve the performance of the baseline model.

The labels consists of 4 different classes: Bacterial Pneumonia, COVID-19, No Pneumonia and Viral Pneumonia. The goal is to build a convolutional neural network that will classify the photos from the X-ray dataset, with the highest possible accuracy. It is important to create a good performing and reliable model to help identify people with Pneumonia or Covid-19, to provide them with adequate treatment.

Overview Classes					
	Bacterial Pneumonia (Class 0)	Covid - 19 (Class 1)	No Pneumonia (Class 2)	Viral Pneumonia (Class 3)	Total
Number of cases	2816	127	1606	1843	6392
%	44	2	25	29	100

Figure 1

2. Dataset Preprocessing

After loading in the data and using the provided code to resize all the images of the dataset to 156X156, an exploratory data analysis (EDA) has been done. Through the EDA the shape of the loaded data became clear and an overview of the unique class labels could be created. Figure 1 shows how the different classes are represented in the dataset.

Then the data was split into a training, validation and test set. Firstly, 20 percent of the data was split into a test set. Of the remaining 80 percent, also 20 percent was split for the validation set. Splitting the data is an essential part of the data preprocessing. By splitting the data into a training, validation and test set, the performance of the model and neural networks can be measured on unseen data. This is also a good way to minimize overfitting, which can happen when training the model. The stratified sampling technique was used for splitting the data, this ensures that the distribution of target variables among the different splits is the same. This is important since there is class imbalance in the dataset.

To improve interpretability and speed of model training the data has been normalized to floating point numbers between 0 and 1. Finally, the data has been one-hot-encoded and transformed from categorical classes to numerical data. The machine learning algorithms used for this problem require the data to be numerical.

3. Baseline Model

A baseline CNN algorithm was given to improve upon, it consisted of the following layers: two consecutive convolutional layers with 64 and 32 filters of size 3 ×3 with ReLU activations followed by a max pooling layer of size 2 ×2 and this block of convolutional layer followed by a max pooling layer is repeated two times. Finally, two dense layers of sizes 32 with Relu activation function and an output layer of size 4. For the final output layer, the Softmax activation function was chosen since there are multiple class labels (Basta, 2020).

Following the guidelines for the base model, the number of epochs have been set to 10 and the batch size to 32 with Adam as optimizer. The metric has been set to accuracy and the loss function to categorical cross entropy. The target labels have been one hot encoded during the preprocessing of the data, and since there are multiple classes, categorical cross entropy is considered to be the best loss function to use (Koech, 2020).

Figure 2 shows some performance measures of the base line model. Specificity measures the proportion of correctly classified negative instances. Sensitivity measures the proportion of correctly classified positive instances. Since the problem at hand is classifying medical data, a high specificity rate is crucial to ensure people who need medical treatment are not misclassified as being healthy. The F1 score measures how well the model predicts in general.
In figure 4 a confusion matrix is plotted to show how well the model predicts with regards to each of the 4 classes. Lastly, figure 3 shows the ROC curve with the AUC score, this plot gives a good visualization of the model's classification performance (Scikit-learn, 2023).
The baseline model resulted in an accuracy of 0.744 and a loss of 0.591, these values have been plotted and can be seen in figure 5.

Baseline Model				
	Bacterial Pneumonia (Class 0)	Covid - 19 (Class 1)	No Pneumonia (Class 2)	Viral Pneumonia (Class 3)
Specificity	0.871	0.995	0.945	0.814
Sensitivity	0.734	0.16	0.900	0.683
F1	0.774	0.229	0.872	0.638

Accuracy: 0.744
Loss: 0.591

Figure 2

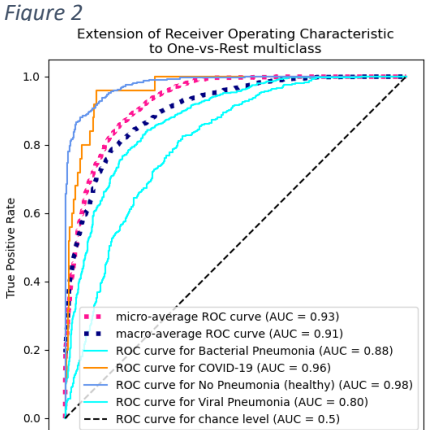

Figure 3

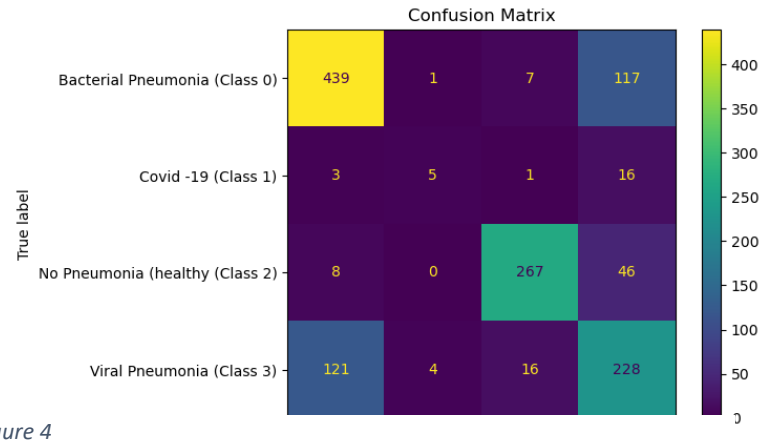

Figure 4

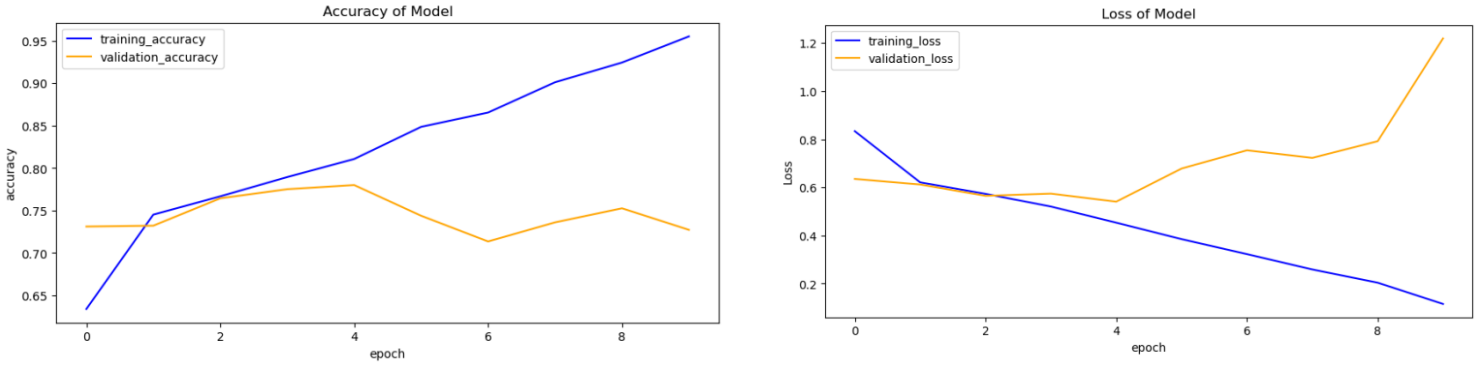

Figure 5

4. Improved (Fine-tuned) Model

Multiple different models were build and tested to improve upon the baseline model. To create a clear overview of the performance of the tested models, a table has been created that shows the accuracy and loss scores for the different models. This table can be seen in figure 6.

Models							
	Early Stopping	Class Weight	Augmented Data	Dropout-Model	L2 Regularization	Regularization and dropout	Full Model
Accuracy	0.775	0.733	0.447	0.784	0.765	0.772	0.497
Loss	0.556	0.592	1.213	0.559	0.590	0.562	1.083

Figure 6

The model with the best performance, is the dropout-model. It has an accuracy of 0.784 and a loss of 0.559. The dropout-model reduces the overfitting of the neural networks by randomly dropping a set of neurons during each iteration. This causes the model to be more generalized and prevents the model from learning the statistical noise (Yadav, 2022).

Besides the dropout-model there were multiple other models that were tested. Like the early stopping model to avoid overfitting by stopping the model training after a certain threshold of unimproved scores of the chosen metric. A model with different class weights was also tried to address the class imbalance in the data. An Augmented data model was tried were mirrored images of the original images were also added to the data input, in combination with contrast stretching of the images. This model did not perform very well as can be seen in figure 6.

In the L2 regularization (or Ridge Regression) model, a penalty was added to the loss function to try and battle overfitting. Finally 2 combined models were tried, one with the regularization and dropout model, which performed relatively well, and a full model, a combination of all previously tested models. The full model performed poorly.

The dropout model was tested with eight different optimizers to see which combination works best (Courville, n.d.). The performance of the model in combination with the different optimizers can be seen in figure 7.

Optimizers with Dropout - Model								
	SGD	RMSprop	Adam	Adadelta	Adagrad	Adamax	Nadam	Ftrl
Accuracy	0.753	0.773	0.770	0.735	0.748	0.782	0.765	1.241
Loss	0.581	0.555	0.582	0.735	0.683	0.546	0.595	0.441

Figure 7

The Adamax optimizer returned the best result, so this optimizer was used in combination with the dropout model to experiment with the batch size. These results can be seen in figure 8. A batch size of 16 returns the highest accuracy of 0.794 and a loss of 0.564. The Accuracy and loss of the fine tuned model has been plotted in figure 9.

Batch Sizes with Dropout - Model with Adamax Optimizer							
	1	2	4	8	16	32	64
Accuracy	0.776	0.784	0.780	0.771	0.794	0.769	0.778
Loss	0.575	0.557	0.549	0.546	0.564	0.566	0.575

Figure 8

So the final fine-tuned model was the dropout model, with the Adamax optimizer and a batch size of 16. Figure 10 shows the Specificity, Sensitivity and F1 score for this fine tuned model. The ROC curve has been plotted and is displayed in figure 11. And lastly, figure 12 shows the confusion matrix.

Improved Model				
	Bacterial Pneumonia (Class 0)	Covid - 19 (Class 1)	No Pneumonia (Class 2)	Viral Pneumonia (Class 3)
Specificity	0.806	0.998	0.972	0.880
Sensitivity	0.858	0.04	0.900	0.615
F1	0.816	0.069	0.907	0.644

Figure 10

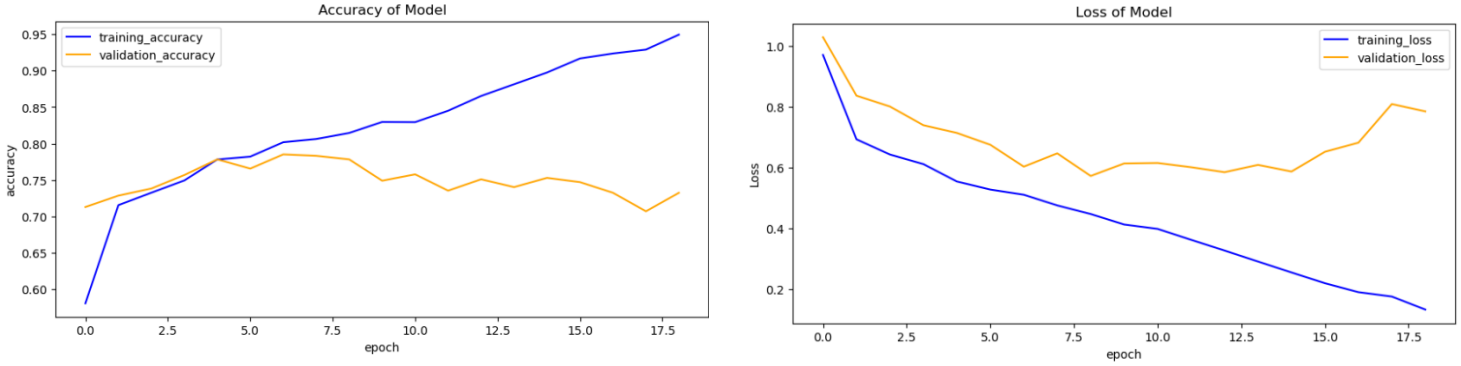

Figure 9

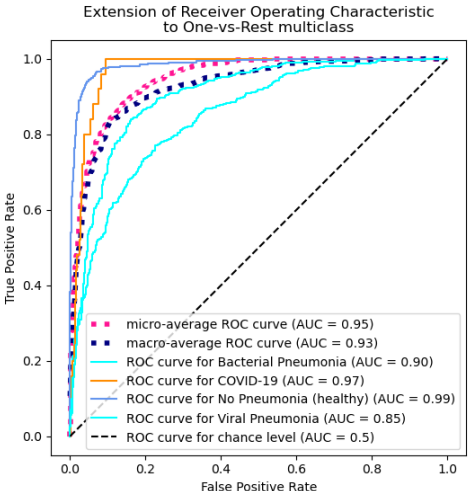

Figure 11

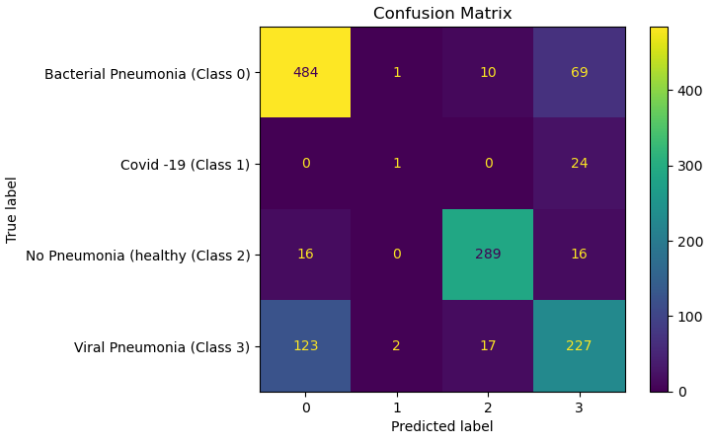

Figure 12

5. Transfer Learning Model

Transfer learning can be defined as: "utilizing the feature learning layers of a trained CNN to classify a different problem than the one it was created for (Cassimiro, 2021)." Two neural networks were tested for the classifying problem: VGG16 and ResNet50. Both of these neural networks were not designed and trained for classifying Pneumonia and COVID-19 in X-ray chest photos.

Both VGG16 and ResNet50 improved the model's accuracy, however the difference in improvement between the two neural networks was very small. In the end, VGG16 was used since it gave the model a slightly better accuracy score compared to ResNet50. VGG16 was trained on the ImageNet dataset which consists of over 14 million high resolution images belonging to 1000 different labels. The patterns that the VGG16 neural network found useful to classify images on this dataset are being "transferred" to help classify the X-ray chest photos.

Figure 13 gives an overview of the specificity, sensitivity and F1 score for the 4 different classes of the Transfer Learning model, with the VGG16 algorithm. The accuracy of the transfer learning model is 0.797 and the loss is 0.488.

Transfer Learning Model				
	Bacterial Pneumonia (Class 0)	Covid - 19 (Class 1)	No Pneumonia (Class 2)	Viral Pneumonia (Class 3)
Specificity	0.822	0.998	0.969	0.889
Sensitivity	0.855	0.12	0.956	0.615
F1	0.823	0.200	0.933	0.651

Accuracy: 0.797
Loss: 0.488

Figure 13

Figures 14, 15 and 16 show the accuracy and loss plots, ROC-curve and confusion matrix of the transfer learning model.

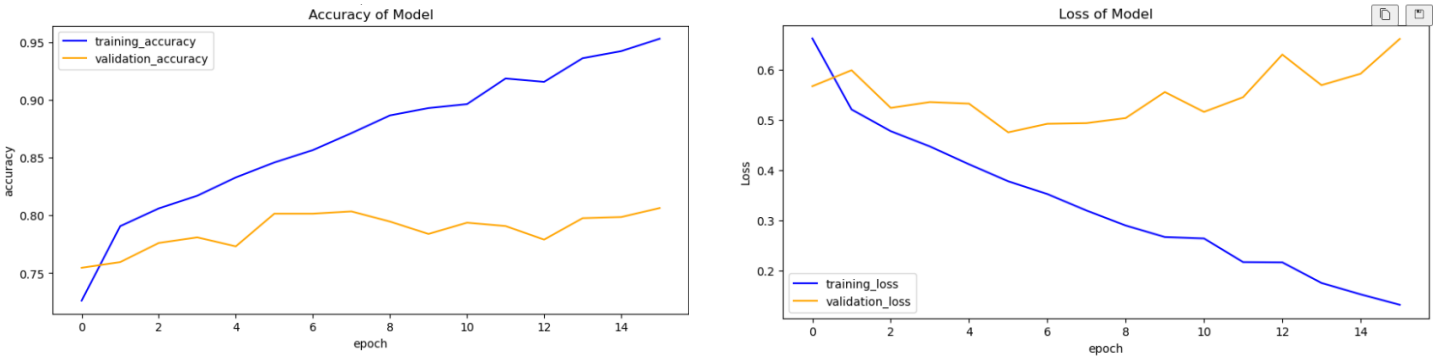

Figure 16

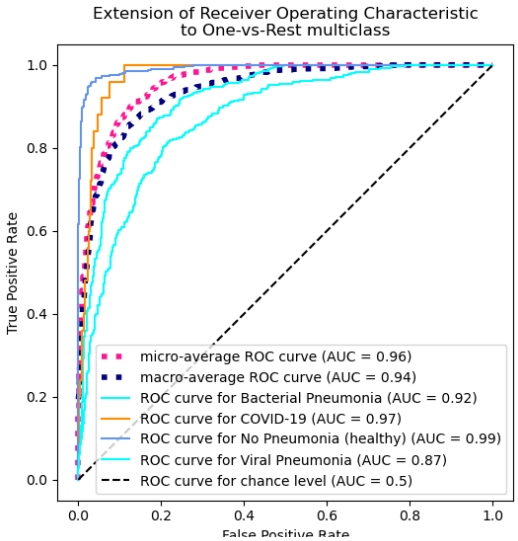

Figure 15

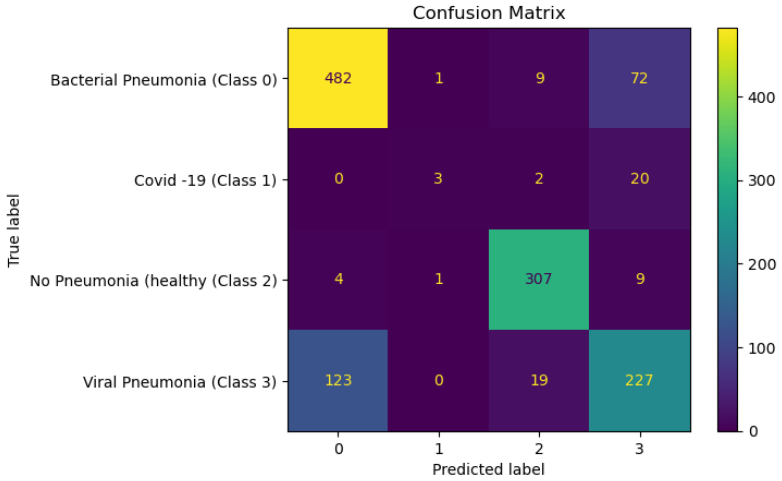

Figure 14

6. Discussion

The baseline model has an accuracy of 0.744 and a loss of 0.591. After trying multiple models to improve the accuracy and decrease the loss initially, the dropout model, with the Adamax optimizer and a batch size of 16 returned the best performance results. The accuracy increased from 0.744 → 0.794. Besides an improvement in accuracy the model also showed a decrease in loss, from 0.591 → 0.564.

The transfer learning model returned an even better accuracy score, the score accuracy improved from 0.794 → 0.797. A very small improvement, the loss however showed a bigger improvement and decreased from 0.564 → 0.488. The early stopping monitor parameter was set to the validation loss, this has contributed to the decrease in validation loss.

The improved models were able to correctly classify more true positives and true negatives, especially for the “bacterial pneumonia” and “no pneumonia” class. With a smaller improvement for the “viral pneumonia” class. Correctly predicting the “COVID-19” class proved to be more difficult, even for the improved models. This is likely caused by the fact that there’s a big class imbalance regarding the “COVID-19” class, as it only accounts for 2% of the total dataset, which can be seen in figure 1.

In order for the models to be able to correctly classify COVID-19 with more accuracy, more data is needed with regard to the COVID-19 class.

7. References

(2023). Retrieved from Scikit-learn: https://scikit-learn.org/stable/auto_examples/model_selection/plot_roc.html

Basta, N. (2020, 04 01). Retrieved from medium: https://medium.com/arteos-ai/the-differences-between-sigmoid-and-softmax-activation-function-12adee8cf322#:~:text=Softmax%20is%20used%20for%20multi,in%20the%20Logistic%20Regression%20model.

Cassimiro, G. (2021, 06 16). Retrieved from towardsdatascience.com: https://towardsdatascience.com/transfer-learning-with-vgg16-and-keras-50ea161580b4

Courville, I. G. (n.d.). Retrieved from Deep learning: http://www.deeplearningbook.org

Igareta, A. (2021, 07 21). Retrieved from towardsdatascience: https://towardsdatascience.com/stratified-sampling-you-may-have-been-splitting-your-dataset-all-wrong-8cfdd0d32502

Koech, K. E. (2020, 10 02). Retrieved from towardsdatascience: https://towardsdatascience.com/cross-entropy-loss-function-f38c4ec8643e

Yadav, H. (2022, 07 05). Retrieved from towardsdatascience: https://towardsdatascience.com/dropout-in-neural-networks-47a162d621d9

COVID-19 Chest X-Ray Dataset. Darwin. (2020). Retrieved February 16, 2023, from https://darwin.v7labs.com/v7-labs/covid-19-chest-x-ray-dataset