Capstone Project 2: Chest X-Ray Image Classification Using Deep Learning

Jonathon Poage
Springboard Data Science Career Track
06/05/2019

Table of Contents

Introduction

Pneumonia is the leading infectious cause of death for children worldwide. According to the World Health Organization (WHO), pneumonia caused 920,136 deaths of children under the age of 5 in 2015.[1] Pneumonia affects children everywhere, but fatalities are most common in developing areas of the world. Fatality rates could potentially be reduced if patients are efficiently diagnosed and referred for proper treatment.

The two leading causes of pneumonia are bacterial and viral pathogens. Viral pneumonia is treated with supportive care, while bacterial pneumonia requires treatment with antibiotics. Low cost treatments for pneumonia are available. The issue is that fast and accurate diagnosis of pneumonia can be challenging in facilities that have limited resources.

Diagnosis of pediatric pneumonia may involve one or more medical tests. According to the National Heart, Lung, and Blood Institute (NHLBI), the best diagnostic test involves inspecting chest radiograph (X-ray) images for signs of inflammation in the lungs.[2] This project aims to develop methods to facilitate diagnosis of pneumonia from chest X-ray images.

Goals of Analysis

This project aims to develop a classification model to predict the presence or absence of pneumonia in patients, based on chest X-ray images. In addition, the model will distinguish between bacterial pneumonia and viral pneumonia. The ultimate goal of this project is to provide an automated, fast, and accurate method for interpreting chest X-ray images, in order to facilitate diagnosis of pediatric pneumonia.

Benefit to Clients

Clinicians and patients could benefit from this project's results, particularly in developing areas of the world. Clinicians could save time by using the results of this project to automatically interpret chest X-ray images when diagnosing patients. This could potentially greatly benefit facilities with limited resources, where clinicians may not currently have the means to quickly and accurately diagnose patients.

The results of this project could also potentially improve clinical outcomes in patients with pneumonia. Clinicians could use the results of this project to efficiently diagnose bacterial and viral pneumonia. This could ultimately facilitate rapid referral for appropriate treatment when required. As a consequence, clinical outcomes could potentially be improved worldwide.

Data

The raw dataset was obtained for this project by downloading the file ZhangLabData.zip from the following website: Mendeley.[3] The relevant dataset in the downloaded file was extracted for this project, and all other content was disregarded. The raw dataset was then processed using Python in preparation for the analysis.

Raw Data

The raw dataset contains thousands of anterior-posterior chest radiograph images from Guangzhou Women and Children's Medical Center. Kermany et al[4] describe the methods they used to collect and prepare the images in the raw dataset. The chest X-ray imaging was performed on pediatric patients ranging from one to five years old as part of routine clinical care. The images were labeled based on the absence or presence of pneumonia, with a distinction between viral and bacterial pneumonia.

The filenames provide the following information for each file:

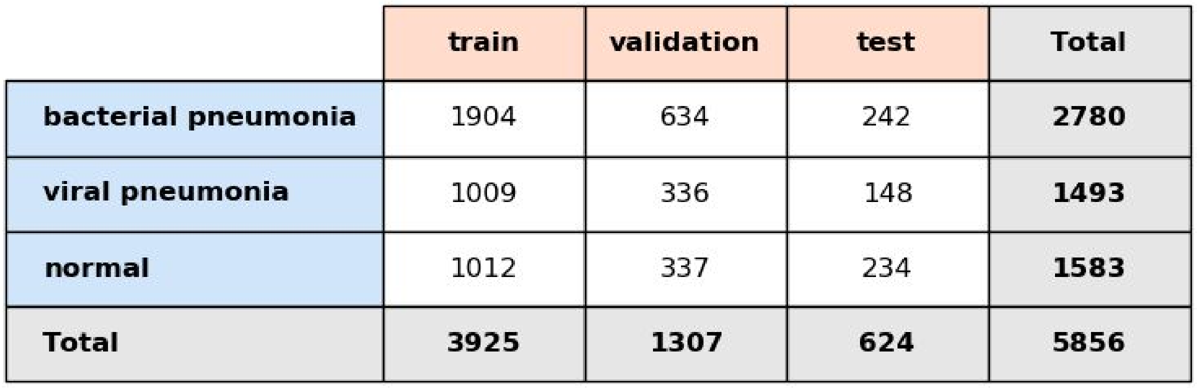
- Image Class (Normal, Bacterial Pneumonia, or Viral Pneumonia)
- Patient ID
- Image number by patient

Processed Data

The raw dataset was processed using Python and separated into train, validation, and test sets in preparation for the analysis. See Appendix A for the procedure used to process the raw data for this project.

Table 1 shows the number of files for each class in the train, validation, and test sets. The classes were not evenly distributed in any of the datasets, but the imbalances were not extreme. The classes had a roughly 2:1:1 ratio in the train and validation sets, and a roughly 5:3:5 ratio in the test set.

Table 1: File counts by image class in the train, validation, and test sets.



	train	validation	test	Total
bacterial pneumonia	1904	634	242	**2780**
viral pneumonia	1009	336	148	**1493**
normal	1012	337	234	**1583**
Total	**3925**	**1307**	**624**	**5856**

Feature and Target Arrays

Feature and target arrays were created from the processed data. These arrays were used as input for the classification models during the analysis. A separate pair of feature and target arrays was created for each of the train, validation, and test sets.

Each feature array was a NumPy array that contained the pixel intensity values in RGB color mode of every image in the designated dataset. Each feature array had dimensions N x W x H x 3, where N was the number of files in the dataset, and W and H were the pixel dimensions. W and H were set to 224 by default for this project.

Each target array was a NumPy array that indicated the class for every image in the designated dataset by using integer values 0 and 1. Each target array had dimensions N x 3, where N was the number of files in the designated dataset. The rows represented image files, and the columns represented classes. A cell value of 1 indicated that the row’s associated image file belonged to the column’s associated class. Conversely, a cell value of 0 indicated that the row’s associated image file did not belong to the column’s associated class.

Analysis

The analysis for this project involved four parts. The analysis framework is discussed below.

The first part of the analysis examined properties of the images in general. Sample images from each image class were displayed in order to visualize characteristics of typical chest X-rays for patients with bacterial pneumonia, viral pneumonia, or no pneumonia. In addition, a mean image was created for each image class by averaging over pixel intensity values. The properties of the mean images were examined to determine similarities and differences between the classes.

The second part of the analysis applied deep learning techniques to develop a set of multi-class classification models. The models predict the probabilities that a chest X-ray image indicates the presence of bacterial pneumonia, the presence of viral pneumonia, or the absence of pneumonia.

The third part of the analysis evaluated the predictive performance of the more promising models on a test set. The predictive performances of the models were assessed based on various scoring metrics, such as recall and F_1 score. The best performing model was identified according to a set of evaluation criteria.

The fourth part of the analysis assessed the capabilities and limitations of the best model in greater depth. The model was applied to a binary classification task that involved predicting the presence or absence of pneumonia, without specify the type of pneumonia. The model's performance was evaluated based on various scoring metrics.

Image Properties

The following sections describe properties of the chest X-ray images in general.

Sample Images

Sample images are displayed in Figure 1, in order to illustrate and compare typical characteristics of chest X-ray images for each class.

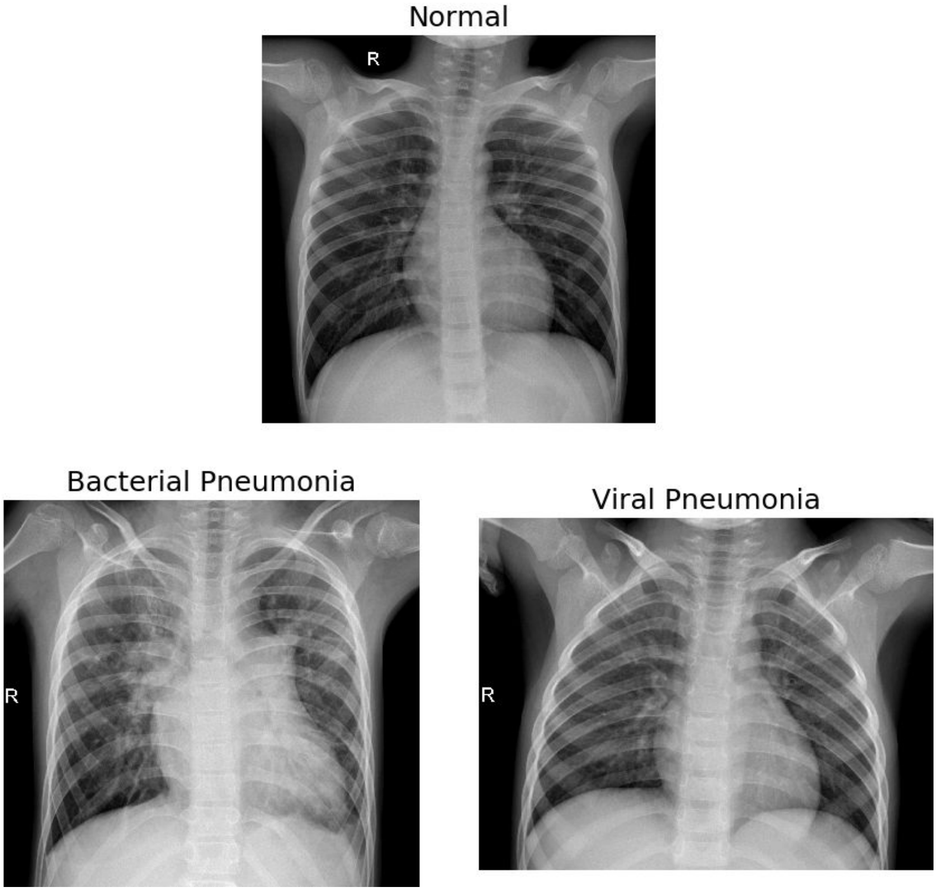


Figure 1: Sample chest X-ray images from the train set. The normal chest X-ray (top) depicts clear lungs. The bacterial pneumonia (bottom left) and viral pneumonia (bottom right) chest X-rays show areas of abnormal pulmonary opacification.

The normal image in Figure 1 depicts clear lungs as a baseline for comparison. The bacterial pneumonia and viral pneumonia images exhibit abnormal pulmonary opacification. Bacterial pneumonia chest X-rays typically contain concentrated opaque areas, which indicate lobar consolidations. Viral pneumonia chest X-rays typically display more diffuse patterns of opacity. Bacterial and viral pneumonia chest X-rays may display similar patterns, and in certain cases it may be difficult to visually determine from a chest X-ray whether an infection is bacterial or viral.

Mean Images

The raw image files in the train, validation, and test sets did not all have the same pixel dimensions. The images were all resized to have identical pixel dimensions in preparation for the analysis. A mean image was then created for each image class. Each mean image was obtained by averaging the pixel intensity values over all resized images in the designated class and then converting to grayscale. Figure 2 displays the mean images.

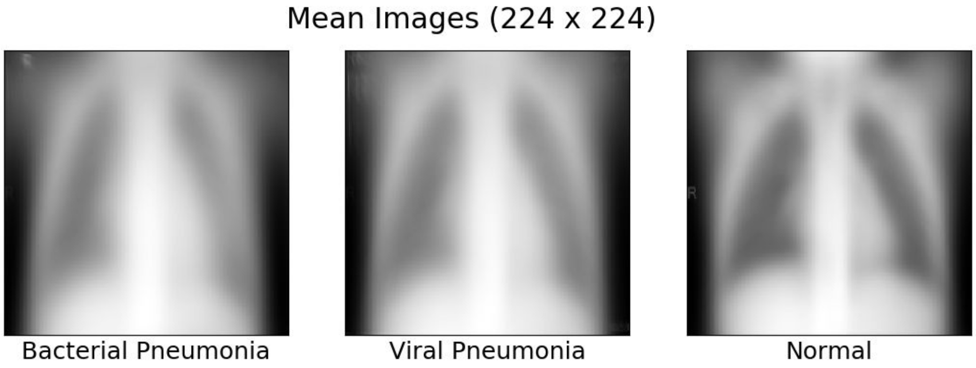


Figure 2: Mean image for each of the image classes. The image files were all resized to have pixel dimensions 224 x 224 before creating the mean images. The areas inside the lung regions for the normal (right) mean image seem to be darker than the corresponding areas in the bacterial pneumonia (left) and viral pneumonia (middle) mean images.

The differences between the normal mean image and either of the pneumonia mean images are visually apparent. The areas in the lung regions of the normal mean image are darker than the corresponding areas in the bacterial and viral pneumonia mean images. The differences between the bacterial pneumonia mean image and the viral pneumonia mean image are more difficult to detect visually.

The mean images are quantitatively compared by using their pixel intensity distributions. Figure 3 displays the intensity histograms for the mean images.

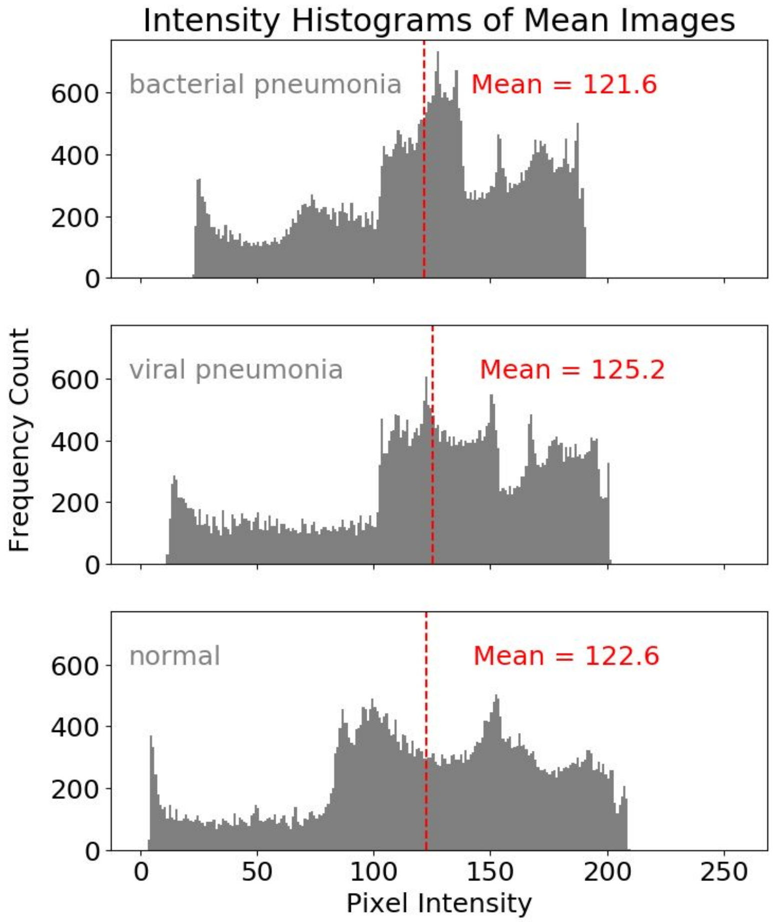


Figure 3: Intensity histograms for the mean images. Pixel intensity values indicate the brightness of the pixels, with 0 representing black and 255 representing white. The mean pixel intensity values of the mean images are indicated by dashed red lines and red text.

Figure 3 reveals similarities and differences between the pixel intensity distributions of the mean images. The distributions have peaks at slightly different locations, indicating that there are subtle differences in the predominant pixel intensity values for each of the mean images. The histograms indicate that the normal mean image pixel intensities are more dispersed, ranging in color from near black to light grey. The viral and bacterial pneumonia mean image pixel intensities are more concentrated towards moderate values, with a larger proportion of grey and light grey pixels compared to the normal mean image.

Image Summary

The results of the analysis of the general image properties indicate that the image classes have suitable differences for a comparison. The remainder of the analysis will apply deep learning in order to leverage the differences between the classes and build an image classification model.

Model Creation

Three Convolutional Neural Networks (CNNs) were developed for this analysis. Each CNN model consisted of a convolutional base followed by a classifier. Each model takes an array of image pixel values as input and produces a class probability distribution as output. The models were trained using Keras with a TensorFlow backend.

The first model was custom built for this project. The second and third models utilized transfer learning by adapting the pre-trained VGG16 model's convolutional base.[5] The second model used the convolutional base of the VGG16 model as a feature extractor by fixing the pre-trained weights during model training. The third model fine-tuned the convolutional base of the VGG16 model by fixing the pre-trained weights of the lower layers while allowing the weights in the upper layers to be modified during model training. See appendix B for a description of the convolutional bases.

A neural network (NN) classifier was custom built for this project. All three models used this NN classifier. See appendix C for a description of the classifier.

Table 2 summarizes the models that were used for this analysis.

Table 2: Summary of the models used in this analysis.

Model	Convolutional Base	Classifier
Model 1	CNN	NN Classifier
Model 2	VGG16 with Fixed Weights	NN Classifier
Model 3	VGG16 with Fine-Tuning	NN Classifier

Model Fitting

This analysis used Keras to fit the models. The fits were performed using the Adam optimization algorithm. Categorical Cross-Entropy was used as the loss metric, though accuracy scores were evaluated as well at the end of each epoch.

Data Preparation

The image data in the train and validation sets were transformed before fitting each model. For model 1, the image pixel values were rescaled from the range [0, 255] to the range [0, 1]. For models 2 and 3, the feature arrays were transformed using Keras's vgg16.preprocess_input function.

Training data were generated in batches from the transformed image data in the train set. The training data were generated with image augmentation in order to improve the model’s ability to generalize to unseen data. Validation data were generated in batches from transformed image data in the validation set. No image augmentation was performed on the validation data.

Anterior-posterior chest X-ray images show a view of the patient from a particular orientation, so care was taken to restrict the transformations performed during the image augmentation. For example, horizontal and vertical reflections were not permitted, while minor rotations were permitted.

Fit Results

The training histories for the three models are displayed below. The training history plots show the train and validation accuracy and loss scores that were evaluated at the end of each epoch.

Figure 4 shows the training history for model 1.

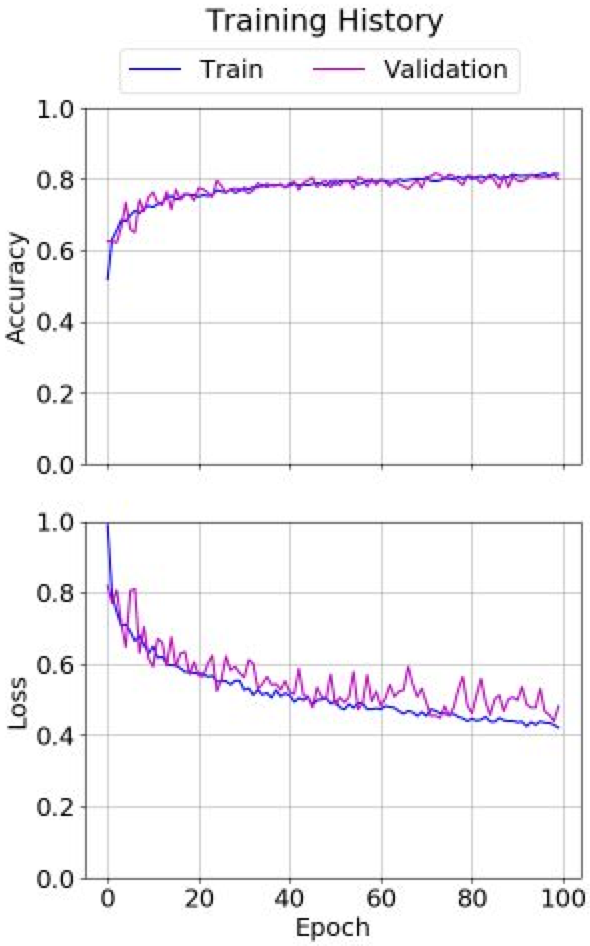


Figure 4: Training history for model 1. The train and validation accuracy scores are comparable, as are the loss scores.

The training history for model 1 indicates that the model generalizes well to unseen data. The train and validation accuracy scores reached values close to 0.8, and the loss scores reached values around 0.4 to 0.5.

Figure 5 shows the training history for model 2.

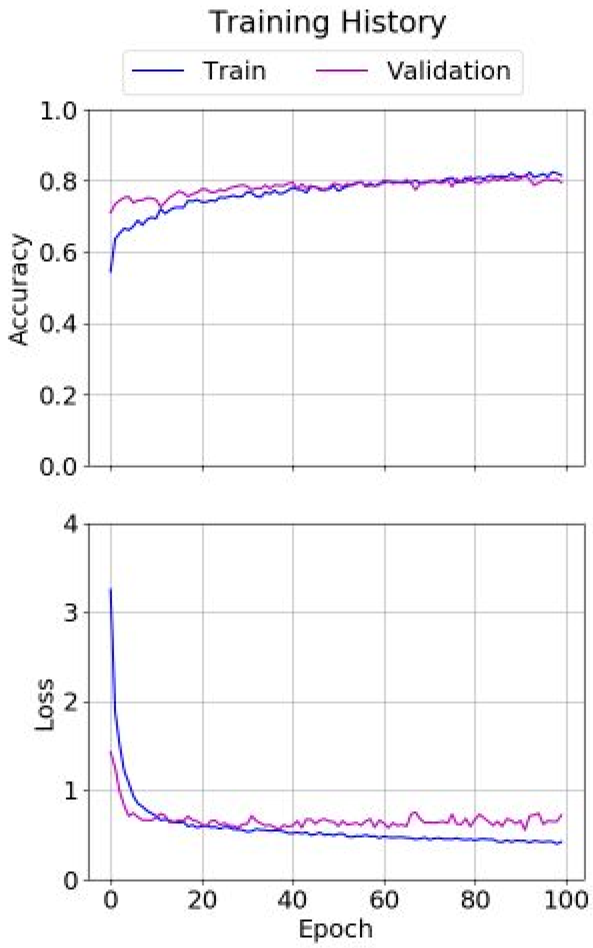


Figure 5: Training history for model 2. The train and validation accuracy scores are comparable at later epochs. The train and validation loss scores are comparable as well, but the train loss scores are slightly lower than the validation loss scores at later epochs.

The training history for model 2 indicates that the model generalizes fairly well to unseen data. The train and validation accuracy scores reached values close to 0.8. The train loss scores reached values around 0.4, while the validation loss scores reached values around 0.6.

Figure 6 shows the training history for model 3.

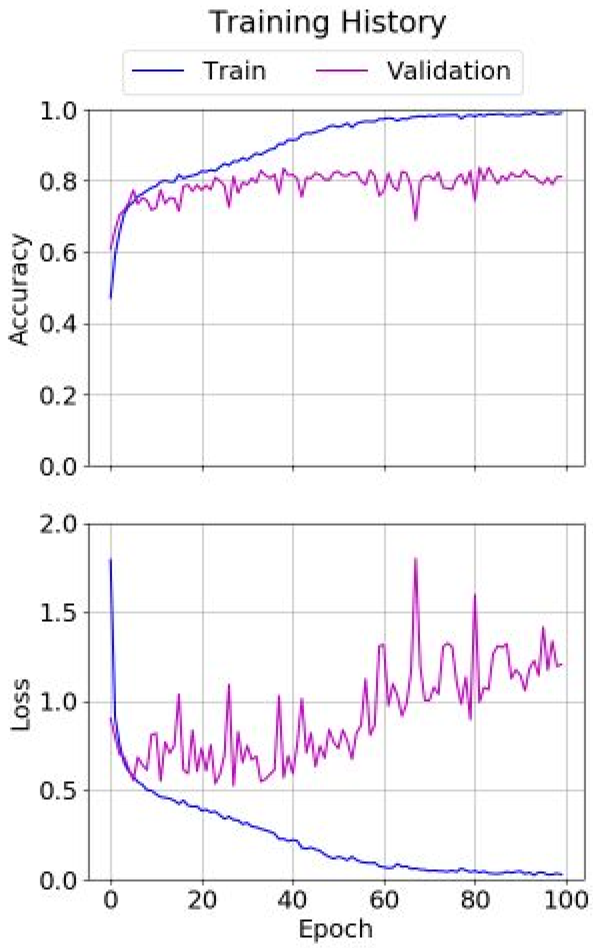


Figure 6: Training history for model 3. The training history indicates that model 3 has been overfit to the training data.

The training history for model 3 indicates that the model does not generalize very well to unseen data. At later epochs, there are large discrepancies between the train and validation accuracy scores as well as between the train and validation loss scores.

Fit Summary

Models 1 and 2 show similar performance in terms of validation and loss scores. Model 3 shows similar performance in terms of validation accuracy, but worse performance in terms of validation loss scores.

Model 3 outperformed models 1 and 2 in terms of train accuracy and loss scores, though it did seem to be overfit to the training data. It is possible that a model that utilizes VGG16’s base with fine-tuning could outperform models 1 and 2, if the appropriate adjustments are made to the convolutional base and the model fitting procedure.

Model Evaluation

The predictive performances of models 1 and 2 were evaluated using the image data in the test set. The predictive performance of model 3 was not evaluated, because the model seemed to overfit to the training data. The results of the model evaluations are discussed in the following sections.

Evaluation Criteria

The predictive performances of models 1 and 2 were evaluated by considering all of the following metrics:

1. Bacterial pneumonia recall score.
2. Proportion of viral pneumonia images misclassified as normal.
3. Weighted average F_1 score.

The following discussion describes the motivation for using the metrics listed above to evaluate the predictive performance of each model.

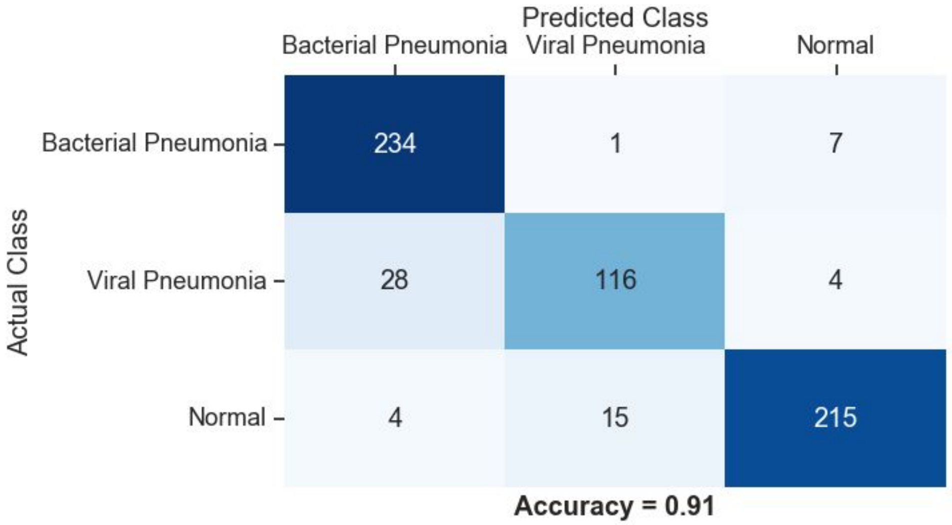
The test dataset was imbalanced. The model evaluation criteria must be suitable for handling class imbalances.

Certain types of misclassification would have more severe impacts on patients than other types of misclassification. Misclassification of bacterial pneumonia images would have the most severe impact on patients, because bacterial pneumonia requires the most intensive treatment. Misclassification of viral pneumonia images as normal would also have a severe impact on patients. The model evaluation criteria must prioritize these severe types of misclassifications.

The less severe types of misclassification are also of practical significance. Misclassification of viral pneumonia images as bacterial pneumonia would inconvenience patients by subjecting them to unnecessary treatment, as would misclassification of normal images. The model evaluation criteria must assess each model’s overall ability to correctly classify images.

Model 1 Evaluation

Figure 7 shows the confusion matrix for model 1.



Actual Class \ Predicted Class	Bacterial Pneumonia	Viral Pneumonia	Normal
Bacterial Pneumonia	234	1	7
Viral Pneumonia	28	116	4
Normal	4	15	215

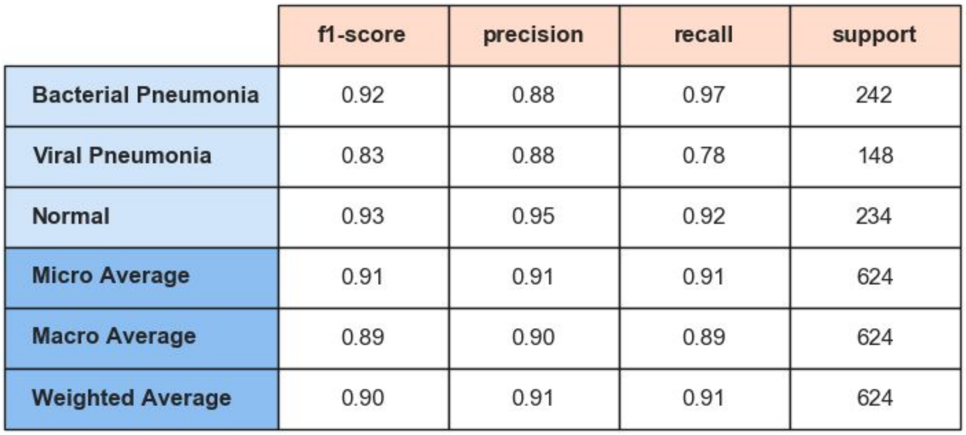
Accuracy = 0.91

Figure 7: Confusion matrix for model 1. The accuracy score indicates that the model correctly classified 91% of the images in the test set.

The upper right corner of the confusion matrix displays the three misclassification types that most severely impact patients. This region of the confusion matrix for model 1 contains 12 instances of misclassified images. In particular, 4 viral pneumonia images were misclassified as normal. The confusion matrix also shows that 18.9% of the viral pneumonia images were misclassified as bacterial pneumonia. This indicates that model 1 had relative difficulty classifying viral pneumonia images.

Table 3 shows a classification report for model 1.

Table 3: Classification report for model 1.

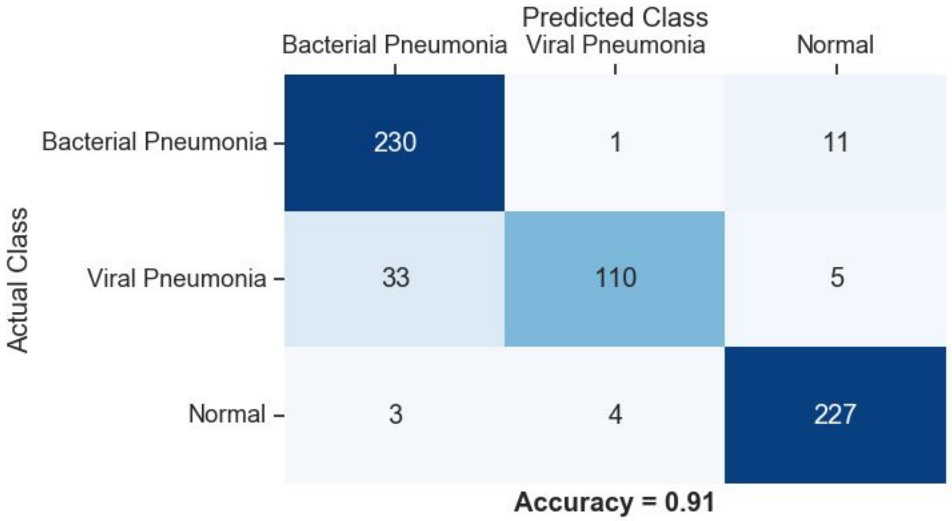


	f1-score	precision	recall	support
Bacterial Pneumonia	0.92	0.88	0.97	242
Viral Pneumonia	0.83	0.88	0.78	148
Normal	0.93	0.95	0.92	234
Micro Average	0.91	0.91	0.91	624
Macro Average	0.89	0.90	0.89	624
Weighted Average	0.90	0.91	0.91	624

The bacterial pneumonia recall score indicates that model 1 correctly classified 97% of the bacterial pneumonia images. The weighted average F_1 score of 0.90 indicates fairly strong overall predictive performance. The relatively low bacterial pneumonia precision score and viral pneumonia recall score reflect the model’s tendency to misclassify viral pneumonia images as bacterial pneumonia.

Model 2 Evaluation

Figure 8 shows the confusion matrix for model 2.



		Predicted Class	
Actual Class	Bacterial Pneumonia	Viral Pneumonia	Normal
Bacterial Pneumonia	230	1	11
Viral Pneumonia	33	110	5
Normal	3	4	227

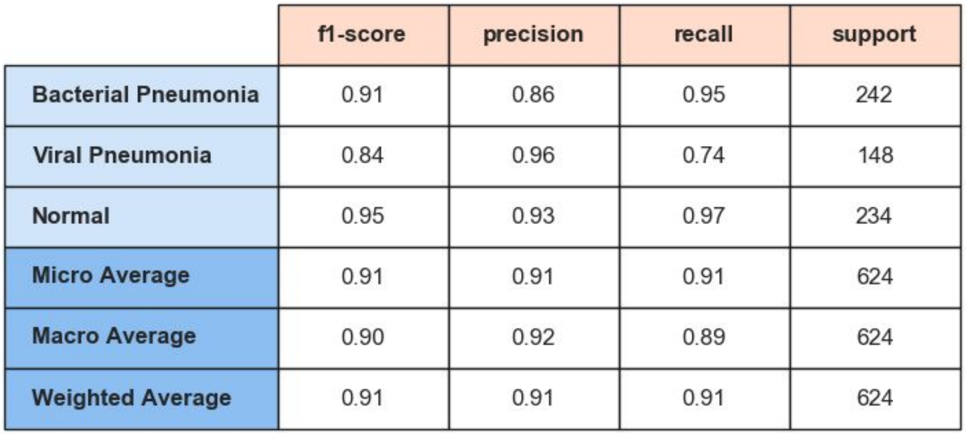
Accuracy = 0.91

Figure 8: Confusion matrix for model 2. The accuracy score indicates that the model correctly classified 91% of the images in the test set.

The upper right corner of the confusion matrix displays the three misclassification types that most severely impact patients. This region of the confusion matrix for model 2 displays 17 instances of misclassified images. In particular, 5 viral pneumonia images were misclassified as normal. The confusion matrix also shows that 22.3% of the viral pneumonia images were misclassified as bacterial pneumonia. This indicates that model 2 had relative difficulty classifying viral pneumonia images.

Table 4 shows a classification report for model 2.

Table 4: Classification report for model 2.



	f1-score	precision	recall	support
Bacterial Pneumonia	0.91	0.86	0.95	242
Viral Pneumonia	0.84	0.96	0.74	148
Normal	0.95	0.93	0.97	234
Micro Average	0.91	0.91	0.91	624
Macro Average	0.90	0.92	0.89	624
Weighted Average	0.91	0.91	0.91	624

The bacterial pneumonia recall score indicates that model 2 correctly classified 95% of the bacterial pneumonia images. The weighted average F_1 score of 0.91 indicates fairly strong overall predictive performance. The relatively low bacterial pneumonia precision score and viral pneumonia recall score reflect the model's tendency to misclassify viral pneumonia images as bacterial pneumonia.

Model Comparison

Overall, models 1 and 2 showed relatively similar predictive performances. Model 1 had slightly better performance regarding the first two evaluation criteria, while model 2 had slightly better performance regarding the third criterion. The choice of best model depends on the order in which the evaluation criteria are prioritized. The ability to identify bacterial pneumonia images was prioritized for this analysis, and so model 1 was selected as the best model.

Binary Classification

The results of the multi-class classification analysis indicated that the best model had relative difficulty classifying viral pneumonia images. Follow-up analysis was performed by applying the best model to a binary classification task, in order to address this limitation and evaluate the model's predictive performance in more detail. The task involved predicting the presence or absence of pneumonia, without specifying the type of pneumonia.

The binary classification analysis utilized the predictions from the multi-class classification analysis. The actual and predicted class labels for viral and bacterial pneumonia were changed to the class label of pneumonia. The predicted probabilities for the viral and bacterial pneumonia classes were summed to obtain the predicted probability for the pneumonia class.

Binary Classification Results

The results of the binary classification analysis are presented in this section. Figure 9 shows the confusion matrix for the binary classification analysis.

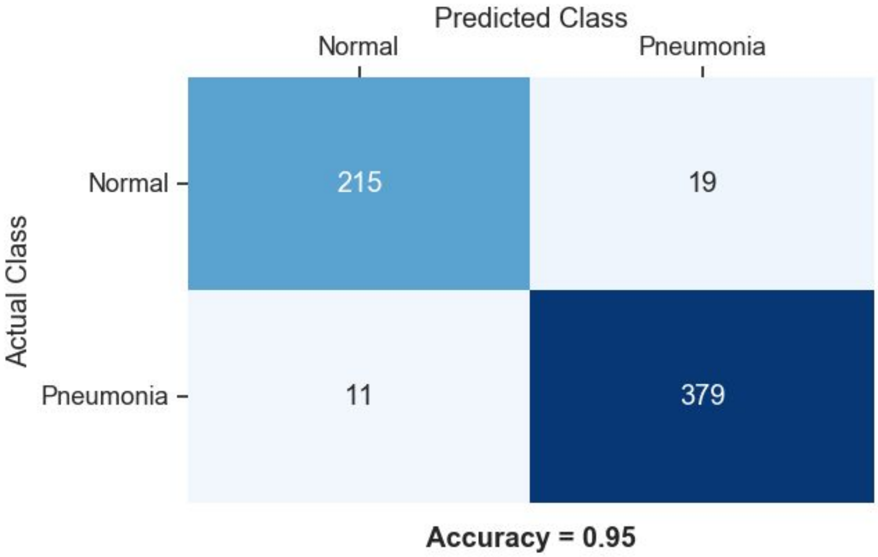
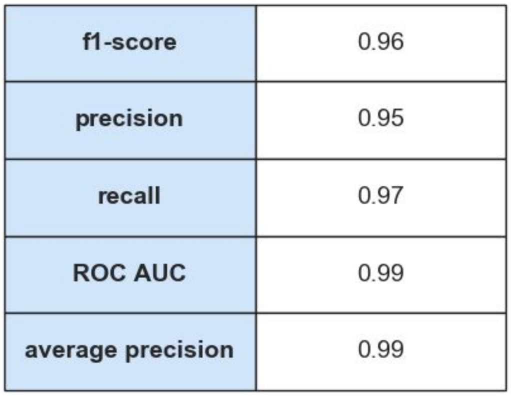


Figure 9: Confusion matrix for the binary classification analysis. The accuracy score indicates that the model correctly classified 95% of the images.

False negatives would have a more severe impact on patients than false positives. The confusion matrix indicates a false negative rate of 2.8% and a false positive rate of 8.1%.

Table 5 shows various metric scores for the binary classification analysis.

Table 5: Metric scores for the binary classification analysis. The recall score indicates that 97% of the pneumonia images were correctly classified.



f1-score	0.96
precision	0.95
recall	0.97
ROC AUC	0.99
average precision	0.99

The metric scores indicate fairly strong overall performance for the binary classification task. In particular, the area under the receiver operating characteristic curve (ROC AUC) and the average precision score indicate that the model's predicted class probabilities distinguish between the classes fairly well. Figure 10 shows the ROC curve and the precision-recall curve for the binary classification analysis.

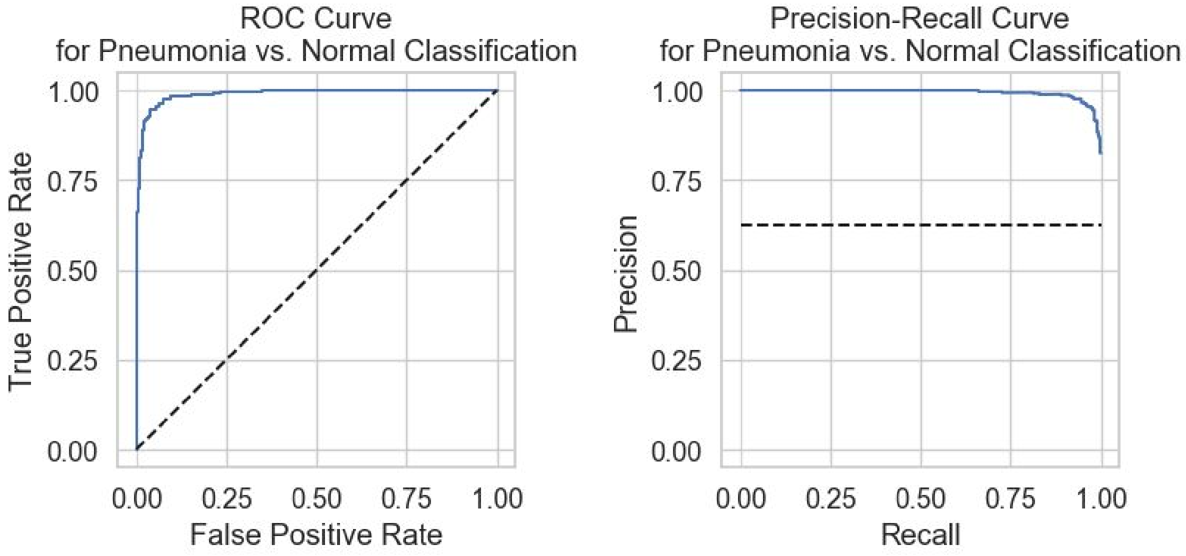


Figure 10: ROC curve (left) and precision-recall curve (right) for the binary classification analysis. The curves are indicated by blue lines. The black dashed lines show the baselines curves that would be obtained by classifying the images at random.

Both curves indicate that the model performs well for a variety of decision thresholds.

Binary Classification Summary

The results of the binary classification analysis indicate that the best model performs well at predicting the presence or absence of pneumonia from a chest X-ray image. The model's tendency to misclassify viral pneumonia as bacterial pneumonia when performing multi-class classification does not affect its performance for the task of binary classification.

Summary

This project succeeded in developing a classification model to predict the presence or absence of bacterial pneumonia or viral pneumonia in pediatric patients. The model could provide an automated, fast, and accurate method for clinicians to interpret chest X-ray images. The results of this analysis could facilitate diagnosis of pediatric pneumonia, particularly in facilities with limited resources.

Models 1 and 2 showed comparable predictive performance. Model 1 used a custom built CNN, and model 2 utilized transfer learning by adapting VGG16's base with fixed pre-trained weights. Each model demonstrated slight advantages and disadvantages with respect to the different evaluation criteria. This project's analysis prioritized the ability to correctly classify bacterial pneumonia images over the other criteria. Model 1 was selected as the best performing model for this project because it correctly classified a higher proportion of bacterial pneumonia images than model 2.

Model 1 has a limitation that must be considered when performing multi-class classification. The model showed a tendency to misclassify viral pneumonia images as bacterial pneumonia. Although the model had relative difficulty distinguishing between the two types of pneumonia, the binary classification analysis demonstrated the model's ability to accurately identify the presence or absence of pneumonia. Model 1's limitation can be circumvented by taking appropriate measures.

The training history for model 3 indicated that it was overfit to the training data. Model 3 utilized VGG16's base by fine-tuning the pre-trained weights in the upper convolutional blocks. Although model 3 did not generalize well to validation data during training, it showed desirable train accuracy and loss scores. It is possible that further analysis could utilize transfer learning to create a new model that outperforms models 1 and 2.

Recommendations

The following actions are recommended, based on the results of this project:

1. It is recommended that follow up analysis is performed to evaluate the methods currently used by medical facilities to diagnose pediatric pneumonia. The performance of these methods could be used as a baseline for comparison.

2. If the best model from this analysis outperforms the current methods used by a medical facility, then it is recommended that clinicians at that facility implement this model to facilitate diagnosis of pediatric pneumonia.

3. It is recommended that supplemental follow-up testing is performed if the model predicts the presence of bacterial pneumonia in a patient. This supplemental testing would circumvent the model's limitations, and confirm the diagnosis before treatment is administered.

In addition, further analysis could expand upon the results of this project, with the goal of developing a new model that outperforms the best model from this analysis.

Reference List

1. Pneumonia. World Health Organization Web site. https://www.who.int/news-room/fact-sheets/detail/pneumonia. Published November 7, 2016. Accessed April 30, 2019.

2. Pneumonia. National Heart, Lung, and Blood Institute Website. https://www.nhlbi.nih.gov/health-topics/pneumonia. Accessed April 30, 2019.

3. Kermany D, Zhang K, Goldbaum M. Large dataset of labeled optical coherence tomography (OCT) and chest X-ray images. Mendeley Web site. https://data.mendeley.com/datasets/rscbjbr9sj/3. Published June 1, 2018. Accessed April 13, 2019.

4. Kermany DS, Goldbaum M, Cai W, et al. Identifying medical diagnoses and treatable diseases by image-based deep learning. *Cell*. 2018;172(5):1122-1131. doi:10.1016/j.cell.2018.02.010.

5. Karen S, Andrew Z. Very Deep Convolutional Networks for Large-Scale Image Recognition. Paper presented at: 3rd International Conference on Learning Representations; May 7-9, 2015; San Diego, CA. https://arxiv.org/pdf/1409.1556.pdf. Accessed May 30, 2019.

6. Applications. keras.io. https://keras.io/applications/. Accessed May 1, 2019.

7. A Brief Report of the Heuritech Deep Learning Meetup #5. heuritech.com. https://blog.heuritech.com/2016/02/29/a-brief-report-of-the-heuritech-deep-learning-meetup-5/. Published February 29, 2016. Accessed May 28, 2019.

Appendix A - Data Processing Procedure

The raw dataset was processed for this project according to the following procedure:

1. The raw data image files had already been split into initial train and test sets. The image files in the initial train and test sets were organized into final train, validation, and test sets.
    - The image files in the initial train set were separated into the final train and validation sets by using stratified random sampling. For each image class, approximately 25% of the files in the initial train set were assigned to the final validation set. The remaining files were assigned to the final train set.
    - The final test set was identical to the initial test set.

2. A Pandas DataFrame was created for the final train set. Each row in the DataFrame represented an image.
    The columns represented the following fields:
    - image_file_base_path - The basename of the filepath.
    - Image_class - The class label of the image (bacterial_pneumonia, viral_pneumonia, or normal).
    - Pixel_array_custom_image_size - A NumPy array of dimensions W x H x 3 with pixel intensity values in RGB color mode. The pixel dimensions W and H are each set to 224 by default for this project.

3. Step 2 was repeated for the final validation set.

4. Step 2 was repeated for the final test set.

5. The DataFrames from steps 2 - 4 were saved to pickle files in order to preserve the format of the NumPy arrays.

Appendix B - Convolutional Bases

Base 1: Convolutional Neural Network

Convolutional base 1 was built from scratch for this analysis. The base consists of four convolutional layers, each paired with a max pooling layer, followed by a flatten layer. The convolutional layers all use a 3 x 3 kernel and ReLU activation.The pooling layers all use a 2 x 2 pool size. Figure 11 shows the architecture of convolutional base 1.

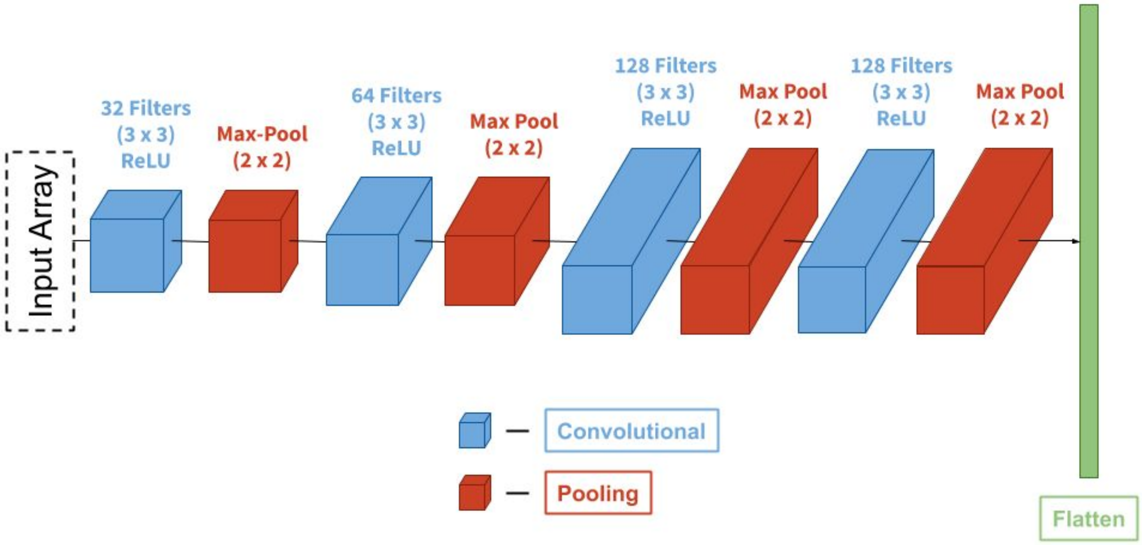


Figure 11: Architecture of the custom CNN convolutional base.

Base 2: VGG16 with Fixed Weights

Convolutional base 2 utilized the VGG16 model's base with weights pre-trained on ImageNet. The VGG16 model was obtained using Keras.[6] The top layers that serve as VGG16's classifier were excluded, and a flatten layer was added to the remaining base. The weights of this convolutional base were fixed during model training. Convolutional base 2 served as a feature extractor for the classifier. Figure 12 shows the architecture of convolutional base 2.

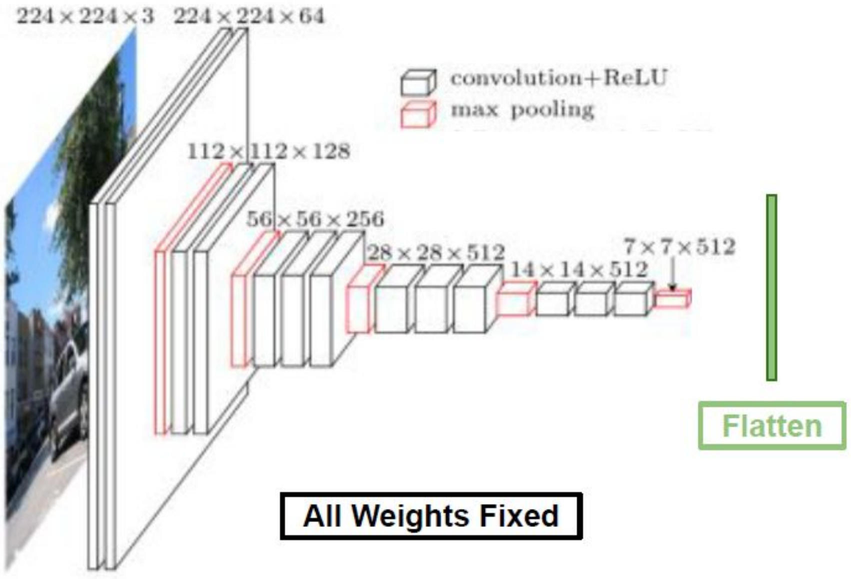


Figure 12: Architecture of convolutional base 2. All weights in the convolutional base were fixed during training. The original, unmodified source image was obtained from heuritech.[7]

Base 3: VGG16 with Fine-Tuning

Convolutional base 3 utilized the VGG16 model's base initialized with weights pre-trained on ImageNet. The VGG16 model was obtained using Keras. The top layers that serve as VGG16's classifier were excluded, and a flatten layer was added to the remaining base. The weights of the bottom three convolutional blocks were fixed during model training. The weights of the top two convolutional blocks were not fixed, and were updated during training. Figure 13 shows the architecture of convolutional base 3.

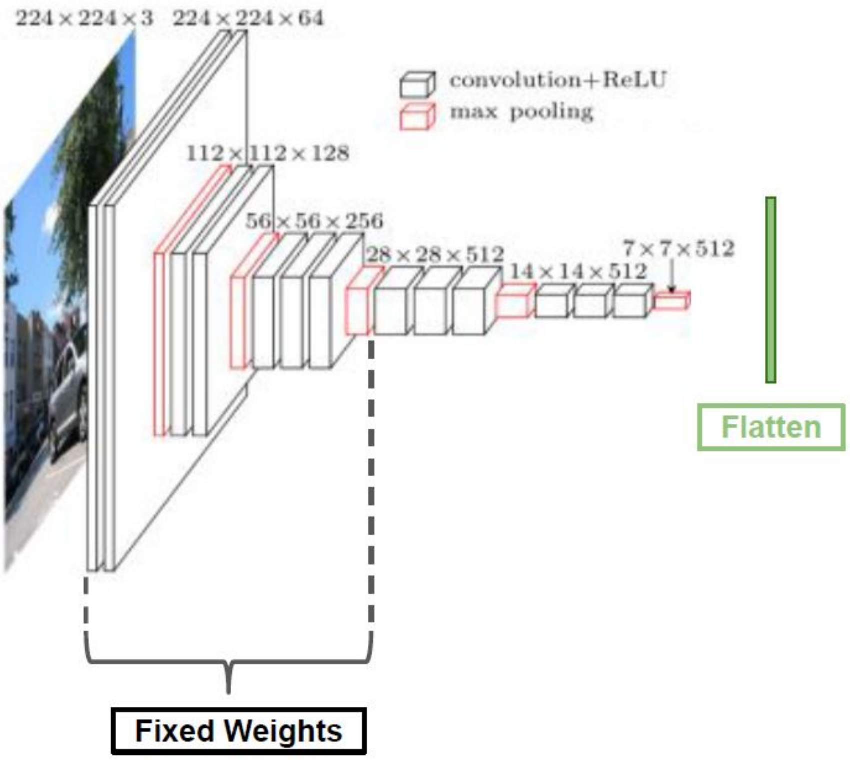


Figure 13: Architecture of convolutional base 3. All weights in the bottom three convolutional blocks were fixed during training. The original, unmodified source image was obtained from heuritech.[7]

Appendix C - Classifier

Neural Network Classifier

The NN classifier takes input from the convolutional base and produces a class probability distribution as output. The NN is composed of two fully connected (FC) layers, each followed by a dropout layer, and an output layer. The hidden FC layers use ReLU activation, and the output layer uses softmax activation. Figure 14 shows the architecture of the NN classifier.

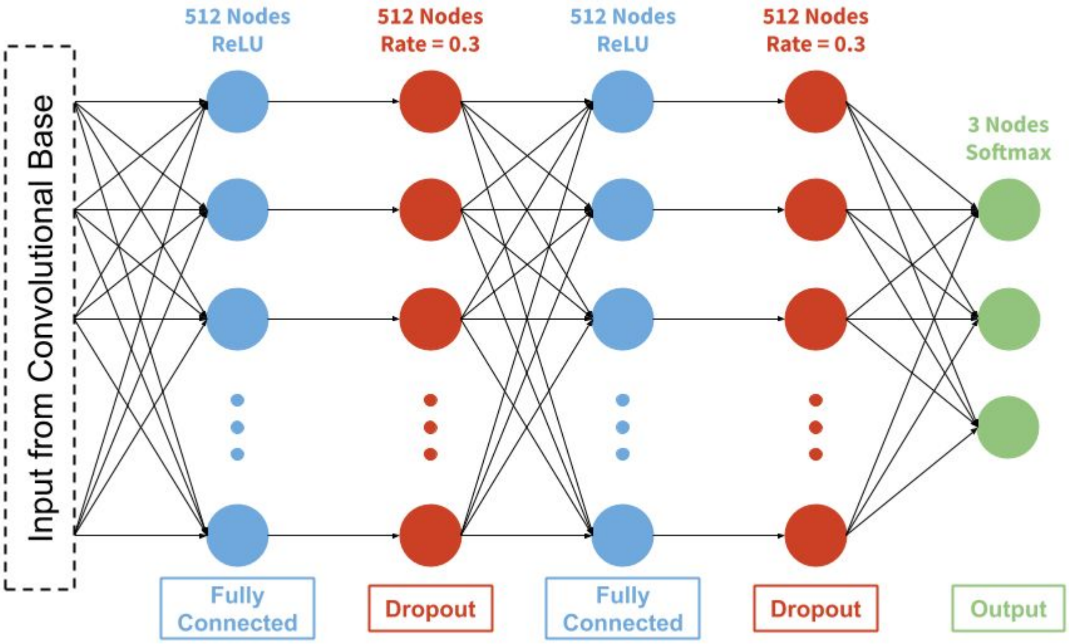


Figure 14: Architecture of the Neural Network classifier.

Appendix D - Project Code

This project's GitHub repository can be found here.

The script used to process the raw data for this project can be found here.

The script containing the functions that were used for this project to load the processed data and prepare the feature and target arrays for analysis can be found here.

The script used to analyze the general image properties for this project can be found here.

The script containing the functions that were used to create the models for this analysis can be found here.

The following scripts were used to fit the models for this analysis: model 1, model 2, and model 3. These model fitting scripts require functions from two additional scripts, found here and here.

The script used to evaluate the models for this analysis can be found here. This model evaluation script requires functions defined in another script found here.

The script used to perform the binary classification analysis for this project can be found here.