Chest X-ray Disease Diagnosis – Final Report

Team #A2: Angel Vazquez

Abstract

Pneumonia is an infection of the lungs that is one of the worldwide top causes of child mortality. The analysis of Chest X-ray images is a cost-effective and suitable method to diagnose these diseases. However, it requires the analysis of experts' radiologists, which are not always available, especially in underdeveloped areas. The goal of this study is to develop a CNN-based model that can assist the diagnosis of pneumonia. This work presents a strategy to localize the regions that define the lungs in the X-ray images. Then only those partial images are used, instead of the whole, for training the models using a two-stage transfer learning approach. The proposed methodology achieved good results with an AUC score of 97.81 for diagnosing Pneumonia on the test data. Additionally, the proposed method achieved an average AUC of 87.15 in chest X-ray images as normal, viral pneumonia, and bacterial pneumonia.

Project presentation slides: here.
Video presentation: https://youtu.be/lB0AJTjmtnw.
Link to code repository in Github: https://github.gatech.edu/avazquez33/big-data-project.

1 Introduction

Pneumonia is an infection of the lungs that is one of the worldwide top causes of child mortality. Particularly in sub-Saharan Africa and Southern Asia, 15% of the deaths amongst children were due to pneumonia in 2019 [1]. The analysis of Chest X-Ray images is one of the most cost-effective and suitable methods to diagnose pneumonia or other lung infections [2]. However, such diagnoses tend to be expensive as they require the presence of experts' radiologists. Nevertheless, these experts' diagnoses are prone to error because of the similarity between pneumonia and other lung diseases. Each particular infection requires a specific treatment. Thus, misdiagnoses can result in the death of the patient. Therefore, there is a growing demand to develop computer systems that aid in the diagnose of pneumonia [3].

This work explores the use of Convolutional Neural Networks (CNN) by presenting a two-stage transfer learning approach for the automated diagnosis of pneumonia in children. The first stage consists of fine-tuning the convolutional weights of the selected network (ResNet50), in addition, to replacing and retraining the top layer to distinguish between a normal chest X-ray image and one with pneumonia. The second phase consists of extracting the features from the fine-tuned CNN to build a new neural-network-based classifier for three-class classification: normal, bacterial pneumonia, and viral pneumonia.

2 Related Work

There are many studies in which research have attempted to diagnose pneumonia and other lung diseases using deep CNN models.

Irvin et al. [4] introduced CheXpert, a large chest radiograph dataset with uncertainty labels. Along with the introduction of this dataset, the authors experimented with several CNN architectures, specifically ResNet152, DenseNet121, Inception-v4, and SE-ResNeXt101. They used different strategies to label the dataset, and ultimately achieved an Area Under the ROC (AUROC) curve greater than 0.9 for diagnosing certain pathologies like Consolidation, Edema, and Pleural Effusion.

Wang et al. [5] presented the Chest X-ray dataset. It contains over 110K images from 32,717 unique patients. They also performed some benchmarks using several pre-trained CNN to classify each pathology type. They measured the performance of these models with the multi-label classification ROC curves on 8 pathology classes and found that the AUROC varies depending on the pathology. One of the classes for which the models' performed worst was Pneumonia. The best model achieved an AUROC of 0.63 for the model with the best performance (ResNet-50).

Rajpurkar et al. [6] developed a 121-layer CNN model called CheXnet. They used 112,120 frontal-view X-ray images that were downscaled to 224x224 and normalized based on the mean and standard deviation of images in the ImageNet training set. The model was trained to make classification of 14 different pathologies, improving the AUROC of all of them compared to other previous studies. They also found that the proposed model exceeded the average performance of radiologists in detecting pneumonia.

Kermany et al. [7] used transfer learning for training a CNN model to detect pneumonia in chest X-ray images. The approach achieved an area under the ROC curve for detection of pneumonia from normal of 0.968 and the AUROC for distinguishing bacterial and viral pneumonia was 0.94.

Rajaraman et al. [8] present an approach in which they find regions of interest in the X-ray image and used partial images defined by those regions instead of the whole, to train the models. They used a VGG16 CNN architecture and observed that the customized model achieved 96.2% and 93.6% accuracy in detecting the pathology and distinguishing between bacterial and viral pneumonia respectively.

Ayan et al. [9] trained several CNN models on a chest X-ray scan database from pediatric patients from one to five years of age. They selected the three most successful ones ResNet-50, Xception, and MobileNet to propose an ensemble CNN model. The proposed method achieved an AUROC curve of 0.95.

3 Materials and methods

The proposed methodology combines some of the ideas exposed in Rajaraman et al. [8] and Ayan et al. [9]. Regions of interests containing the lungs were identified in the images (as exposed in Rajaraman et al. [8]). The training data went through a data augmentation process in which some additional normal-labeled images were generated from the available ones, as proposed by Ayan et al. [9]. The following sections describe the methodology at detail.

3.1 Data collection and preprocessing

The dataset is provided by Kermani and Goldbaum [10]. It is composed of chest X-ray radiography (CXR) gathered at the Guangzhou Women and Children's Medical Center from pediatric patients ages one to five years.
The data is already labeled. It has been split in train and test. The train set is composed of 5232 images and the test set has 624 images in total. Table 1 shows the class distribution.

Table 1. Class distribution on training and test set.

Class	Train	Test
Normal	26%	38%
Pneumonia Viral	26%	24%
Pneumonia Bacterial	49%	39%

These CXRs contain regions other than the lungs that do not contribute to diagnosing pneumonia. An algorithm based on anatomical atlases [11] to automatically detect the lung regions was used to remove these regions. The images were cropped to the size of a bounding box to include only the lung pixels that constitute the region of interest. The cropped CXRs were resampled to 1024 x 1024 pixel dimensions. The detected lung boundaries for the sample CXRs are shown in Figure 1.

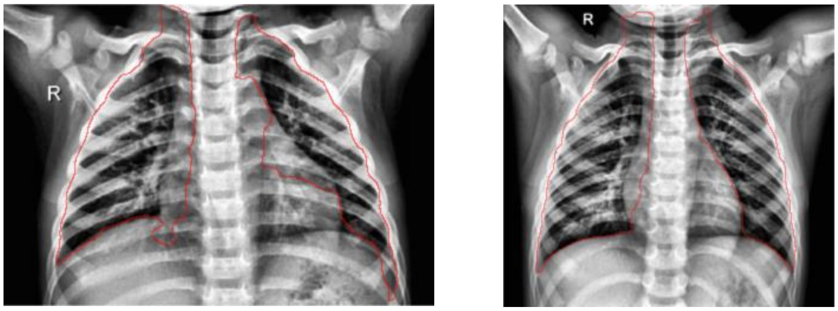


Figure 1. Detected lung boundaries in sample CXRs

The training data went through a data augmentation process to balance the existing classes in the dataset. The normal-labeled images in the training set were randomly augmented by applying rotation (range of angle +10, -10), zooming (range, 1-1.2), and horizontal flipping. With this, 2,534 new augmented CXRs were obtained. Therefore, achieving the class-based balance for the training dataset.
Additionally, while training the models in the first transfer learning stage, real-time data augmentation (shifting, rotating, zooming, and flipping) methods were used to prevent overfitting.

3.2 Data modeling – Transfer Learning

The goal is to develop a multi-class classification neural network model that can distinguish between normal, bacterial pneumonia, and viral pneumonia. The strategy to achieve this was to use a two-stage transfer learning process. The first stage consists of fine-tuning the convolutional weights of the selected CNN for the specific task of distinguishing between a healthy chest X-ray image and one with pneumonia. In the second stage, the output from the last convolutional layer of the fine-tuned model was extracted and transformed into feature vectors that were stored in text files. This data served to train the multi-class classification model. Having this data in files induced faster training times and allowed testing more hyperparameters and architectures to tune the network.

3.2.1 Fine-Tuning for binary classification

The selected CNN architecture for the first stage was ResNet-50. The weights of the convolutional layers come from pre-trained ImageNet weights. The model was truncated at the deepest convolutional layer and added with a global average pooling and dense layer with two outputs (the two classes) followed by the SoftMax activation function.

The strategy to fine-tune the first model consisted of Freeze/Pre-train, and Fine-tune. The Freeze/Pre-train step involved training only the last dense layer of the model. This training process was done until convergence, meaning that the added fully connected layer achieved some stability. The rationale is that if one mixes randomly initialized trainable layers with trainable layers that hold pre-trained features, the randomly initialized layers will cause large gradient updates during training, which will destroy the pre-trained features. Therefore, it is preferred to train the top layer until it reaches stability, before unfreezing the rest of the network for further training. Finally, the fine-tuning step consisted of unfreezing the pre-trained weights and training the complete network with a small learning rate.

The optimized parameters during this stage were the learning rate and the L2 regularization of the model. Additionally, two optimizers were tested, Adam and SGD. From the training data, 20% of the records were used as a validation set. A grid-search over a range of values for the hyperparameters was performed to find the configurations that reduce overfitting and maximize generalization.

3.2.1 Feature extraction for multiclass classification

The fine-tuned ResNet-50 from the previous step was used. The dense layer at the top was removed, and only the global average pooling layer was left. Next, each of the images in the data was read, scaled, transformed. Then, these processed images were fed to the model to perform a forward pass and get the output from the CNN. The output from this forward pass consisted of an array of 2048 features, which was stored in a text file, along with the corresponding label (normal, pneumonia viral, or pneumonia bacterial). The architecture and the learning rate of the Neural Network used in this stage were optimized to improve performance and reduce overfitting. The best results were achieved with the architecture shown in Figure 2.

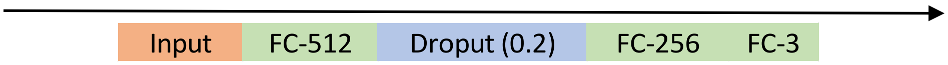


Figure 2. Neural Network Architecture for multiclass classification

The dropout layers were added to induce some regularization and reduce overfitting [12]. The first two fully connected layers are followed by sigmoid activation function. The last layer is followed by SoftMax activation.

3.2 Evaluation criteria

In the study, models were trained on the training data, the hyperparameters were tuned on the validation data (sampled from the training set), and the performance evaluation was done on the test data. The classification performance of the proposed methods was evaluated using the accuracy, precision, recall, f1-score, and the area under the ROC curve (AUC). The AUC can be a reliable criterion in datasets with an imbalanced distribution. The rest of the metrics serve for a more comprehensive analysis of the results and allow to make comparisons between the results of this study and the results of the authors that inspired it.

4 Experimental setup

In this work, a CNN model (ResNet50) was fine-tuned for classifying chest X-ray images. The region that comprises the lungs was localized for each CXR image. Then, cropped images that contain only the lungs region were used to train and fine-tune the CNN. Next, the output from the last convolutional layer of the fine-tuned model was extracted and transformed into features that served to train the final model.

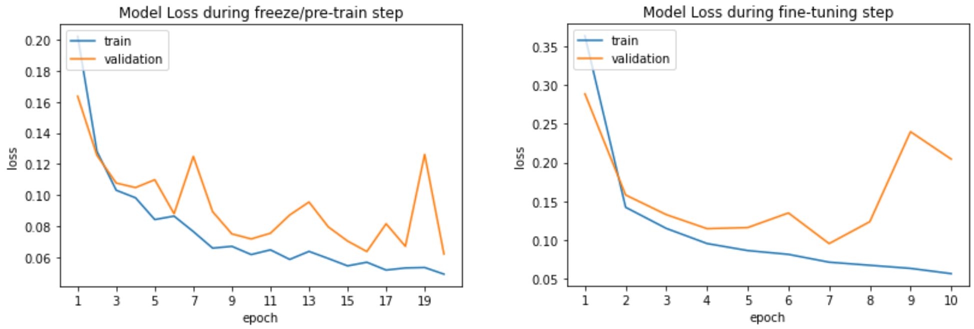
The data augmentation performed during the data processing phase was done in Python via the Augmentor [13] library. An algorithm based on anatomical atlases was applied to localize the regions of the lung. An existing implementation of the algorithm written in MATLAB was used [14]. All Neural Network-based models were trained on Python using Keras library. The ETL process for feature extraction of the ResNet50 fine-tuned model was done in a spark cluster setup in the machine used and leveraging the PySpark API. All experiments were performed in a Virtual Machine provided by Google Cloud Platform. The machine had 16GB Ram and Nvidia Tesla T4 graphic card with Debian 10 operating system.

5 Results

5.1 Fine-Tuning for binary classification

During the Freeze/Pre-train step, the best parameters found were a learning rate 1e-3 with Adam optimizing for categorical cross-entropy loss function. The L2 regularization did not help to improve performance, so it was set to 0. For the Fine-tune step, all network weights were unfrozen and re-trained at a learning rate of 1e-4 using SGD for optimization. Figure 3(a) shows the evolution of the loss curve at each epoch, for train and validation sets, during training for the Freeze/Pre-train step (which only updated the weights of the top layer). The loss for the validation kept decreasing during the entire training process (20 epochs) but show some instability towards the end that suggest some overfitting might be occurring.

Figure 3(b) shows the loss curve during the fine-tuning step in which the entire network was unfrozen, and all weights were recomputed with a low learning rate. This model was trained for 10 epochs only.



(a) **(b)**

Figure 3. Loss curves for binary classification models

The validation loss for the fine-tuning step shows that the model was achieving good generalization during the first 7 epochs, but then it started to diverge, showing some signs of overfitting which suggests that further hyperparameter tuning or regularization is required. From the generated models (one for each epoch), the model with the lowest validation loss was recorded as the best model.

The final model achieved an accuracy of 0.9230, and an AUC of 97.81 on the test set for the binary classification task (Normal vs Pneumonia). Table 2 shows results for other performance metrics of the final model. Additionally, Figure 4 shows the confusion matrix for the model.

Class	Precision	Recall	f1-score	Test samples
Normal	0.9745	0.8162	0.8884	234
Pneumonia	0.8995	0.9872	0.9413	390

Table 2. Performance metrics on final binary classification model

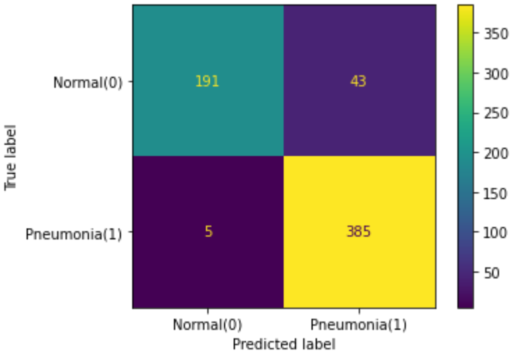


Figure 4. Confusion matrix for binary classification

The achieved AUC of the model is comparable to results from studies from other authors. However, this result was at the expense of low recall for the "Normal" class. The model is miss-classifying a considerable amount of "Normal" labeled examples as "Pneumonia". Nevertheless, the model captured almost all true instances of Pneumonia which is why the recall of that class is high. Considering that failing in diagnosing the affliction can result in the death of a patient, then this model shows some good properties.

Figure 5 shows the ROC curve for the model. The current prediction threshold is set at 0.5, as it was the value used during training, and no tuning was performed for it. The figure shows what would be the best threshold to balance the false positive rate and true positive rate.

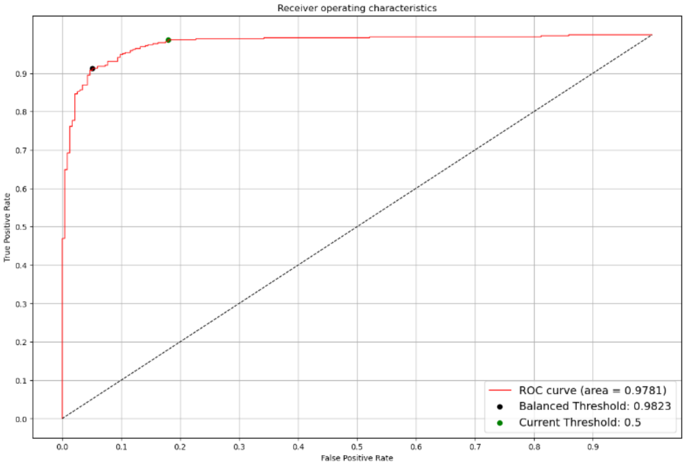


Figure 5. ROC curve for binary classification

Changing the decision threshold for "Pneumonia" from 0.5 to 0.9823 improves the accuracy of the model, but just marginally (from 0.9230 to 0.9262). This new threshold decreases the recall of the positive class, by comparison to the previous results, as shown in the confusion matrix in Figure 6.

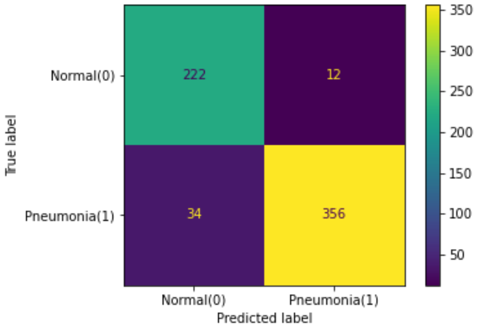


Figure 6. Confusion matrix for binary classification using the "balanced" decision threshold.

Ultimately, since the threshold was not tuned during training, the value used for measuring the performance was 0.5. So, the final results of the model are those shown in Figure 4 and Table 2.

5.1 Feature extraction for multiclass classification.

After extracting the features from the images using the fine-tuned model. The multiclass classification model was trained using the architecture described in previous sections. The optimizer used was SGD, and best results were achieved with a learning rate of 1e-3. The fraction of the inputs to drop by the Dropout Layer was set to 20%. Figure 7 shows the loss curve for this multi-class classification model. The training was performed for 60 epochs, and the model with the lowest validation loss was selected.

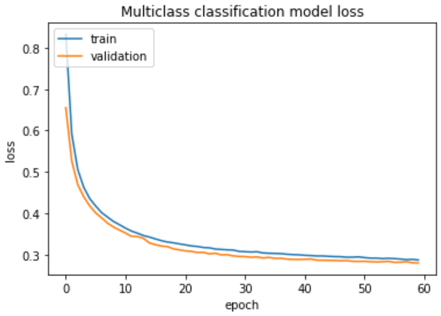


Figure 7. Loss curve for multiclass classification model

The train and validation losses decrease at a similar rate and stay very close of each other during training. This provides evidence that the model is achieving good generalization.

The final model achieved an accuracy of 0.8365, and an average AUC of 87.15 on the test set for the multiclass classification task (Normal vs Viral vs Bacteria). Table 3 shows results for other performance metrics of the final model. Additionally, Figure 8 shows the confusion matrix for the model.

Class	Precision	Recall	f1-score	Test samples
Normal	0.977	0.739	0.842	234
Viral	0.712	0.750	0.730	148
Bacteria	0.818	0.983	0.893	242

Table 3. Performance metrics on multiclass classification model

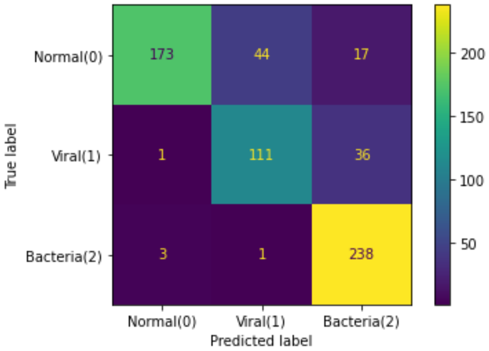


Figure 8. Confusion matrix for multiclass classification

The model has high precision for classifying “Normal” instances, but the recall is low. The model has difficulties accurately classifying “Viral” examples as the performance metrics for this class show low scores. Nevertheless, the model identified almost all instances of Pneumonia regardless of the type.

The AUC curves (Figure 9) for each of the classes confirm that the worst performance is for the Viral class. One reason for this could be that the X-ray images corresponding to the Viral Pneumonia class do not show patterns as clear as the Bacterial group for the model to perform well. So, it might be harder to identify.

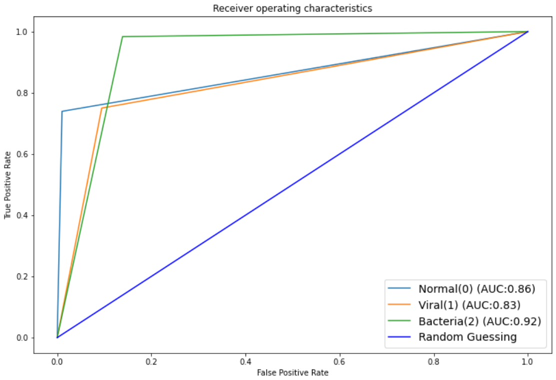


Figure 9. ROC curve for multiclass classification

6 Discussion

In this section, the obtained results in this work were compared with other studies that achieved successful results in the literature. This comparison is given in Table 4.

References	Classes	Accuracy	AUC	Sen.
Rajaraman et al. [9]	Normal vs Pneumonia	96.2	99	99.5
Ayan et al. [9]	Normal vs Pneumonia	95.83	95.21	97.76
Proposed Method	Normal vs Pneumonia	92.3	97.81	98.72
Rajaraman et al. [9]	Normal vs Viral vs Bacterial Pneumonia	91.8	93.9	-
Ayan et al. [9]	Normal vs Viral vs Bacterial Pneumonia	90.71	-	-
Proposed Method	Normal vs Viral vs Bacterial Pneumonia	83.65	87.15	-

Table 4. Comparison of proposed methodology with the literature studies

The proposed method achieved higher AUC than Ayan et al. [9]. Moreover, it showed higher recall (Sen.) for the Pneumonia class, which is desirable as it would diagnose the illness for most of the cases in which is truly present. The proposed method did not outperform the work made by Rajaraman et al. [8], as that model achieved higher AUC, accuracy, and Sen.
For the multiclass classification task, the proposed methodology did not achieve better results than the obtained by the referenced authors. Since this model uses features extracted from the one designed for binary classification, it may be inheriting biases from there that cause this new model to perform worse. Previous iterations of this methodology have shown that it is necessary to increase the performance of the base model (binary classifier) to achieve better results for the multiclass classification model, provided we still do feature extraction.

7 Conclusion

This work proposed a methodology to train models on the task of diagnosing pneumonia in chest X-ray images. The proposed method did improve results from previous studies in the literature on the binary classification problem but not for the multiclass classification. Nevertheless, the proposal shows some advantages as it is less complex and requires fewer resources. Transfer-learning via feature extraction provided be a fast way to build these types of models, while maintaining acceptable performance.
Some modifications can be applied to the current methodology which can help to improve results. A more complex CNN model, different from ResNet50, could capture more patterns from the CXRs that would allow for better classification. Additionally, it can be tested whether including more data that does not necessarily belong to pediatric patients can improve the performance of the classifiers.

Project Takeaways

I did not have prior experience developing deep learning models or processing images. Some of the challenges faced were:

1. Learn tools and algorithms for image processing.
2. Set up a system with sufficient computing capacity to train the CNN models with an acceptable runtime.
3. Gain an understanding of deep learning libraries, along with novel techniques to improve the model's performance and reduce overfitting.

To overcome these challenges a depth review of the existing literature was required, along with tutorials to learn how to use the deep learning libraries leveraged to build the models.
The computing resources from local machines resulted insufficient to train the CNN models. So, it was necessary to deploy a virtual machine on the cloud to get the resources for training the models promptly.

References

1. WHO: Levels-and-Trends-in-Child-Mortality, Report 2019.
2. WHO: Standardization of Interpretation of Chest Radiographs for the Diagnosis of Pneumonia in Children. World Health Organization, Geneva (2001).
3. Shen, D.; Wu, G.; Suk, H.-I.: Deep learning in medical image analysis. Annu Rev Biomed Eng. 19, 221–248 (2017)
4. J. Irvin, P. Rajpurkar, M. Ko, Y. Yu, S. Ciurea-Ilcus, C. Chute, H. Marklund, B. Haghgoo, R. Ball, K. Shpanskaya, J. Seekins, D. A. Mong, S. S. Halabi, J. K. Sandberg, R. Jones, D. B. Larson, C. P. Langlotz, B. N. Patel, M. P. Lungren, and A. Y. Ng. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. arXiv preprint arXiv:1901.07031, 2019
5. X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 3462–3471. IEEE, 2017.
6. Rajpurkar, P.; Irvin, J.; Zhu, K.; Yang, B.; Mehta, H.; Duan, T.; Ding, D.; Bagul, A.; Langlotz, C.; Shpanskaya, K.: Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning (2017). arXiv preprint https://arxiv.org/abs/1711.05225.
7. Kermany, D.S.; Goldbaum, M.; Cai, W.; Valentim, C.C.; Liang, H.; Baxter, S.L.; McKeown, A.; Yang, G.; Wu, X.; Yan, F.: Identifying medical diagnoses and treatable diseases by image-based deep learning. Cell. 172, 1122–1131. e9 (2018).
8. Rajaraman, S.; Candemir, S.; Kim, I.; Thoma, G.; Antani, S.: Visualization and interpretation of convolutional neural network predictions in detecting pneumonia in pediatric chest radiographs. Appl Sci. 8, 1715 (2018).
9. Enes, Aayan; Bergen Karabulut; Halil Murat Unver: Diagnosis of Pediatric Pneumonia with Ensemble of Deep Convolutional Neural Networks in Chest X-Ray Images. King Fahd University of Petroleum & Minerals 2021.
10. Kermany, D.; Goldbaum, M.: Labeled optical coherence tomography (OCT) and Chest X-Ray images for classification. Mendeley Data. 2 (2018).
11. Candemir, S.; Jaeger, S.; Palaniappan, K.; Musco, J.P.; Singh, R.K.; Xue, Z.; Karargyris, A.; Antani, S.; Thoma, G.; McDonald, C.J. Lung Segmentation in Chest Radiographs Using Anatomical Atlases with Nonrigid Registration. IEEE Trans. Med. Imaging **2014**, 33, 577–590.
12. X. Colin, Wei; Sham Kakade; Tengyu Ma: The Implicit and Explicit Regularization Effects of Dropout (2018). arXiv preprint https://arxiv.org/abs/2002.12915.
13. https://augmentor.readthedocs.io/en/master/
14. https://lhncbc.nlm.nih.gov/LHC-downloads/downloads.html#chest-x-ray – Chest X-ray Screening System: Segmentation Module – v3
15. Xiaoran Chen: Image enhancement effect on the performance of convolutional neural networks. Blekinge Institute of Technology (2019).