# DETECTING PNEUMONIA IN CHEST X-RAYS USING CONVOLUTIONAL NEURAL NETWORKS

***Final Project Report***

*ABUBACKER, Mohamed Nafees*
DSBA
b00759248@essec.edu

*CARILLO, Jella Marie*
DSBA
b00767845@essec.edu

*DORVEAUX, Thomas*
DSBA
b00604063@essec.edu

## 1. Abstract

*Despite major progress in the fight against lung disease globally, Pneumonia is still a major cause of child mortality around the world. Identifying pneumonia in the patient's lungs is a key step to eradicate the disease. Among other methods like blood tests or pulse oximetry, doctors often study Chest X-rays to detect pneumonia. In such a context, Artificial Intelligence (AI) algorithms can make physicians gain precious time to detect the disease. They can also avoid misidentification of the illness, especially errors related to non-detections. In particular, Convolutional Neural Networks (CNN) seem the best tools for helping doctors scanning X-ray images.*

*In our study, we used typical CNN algorithms such as ResNet or AlexNet as well as custom CNNs to detect pneumonia on a sample of 5,863 X-Ray images provided by the Medical Center of Guangzhou in China. We also wanted to compare the efficiency of pre-trained models (from the ImageNet Dataset) and non-pre-trained models, to see if transfer learning could be of any utility in this matter. As the minimization of false negatives is one of the main requirements of any disease detection, we relied on the recall metric to evaluate our models.*

*Our results provided accuracy and recall superior to 80% and pre-trained models seemed to have the same performance, if not slightly better, than non-pretrained models. As the current literature on the subject already suggests, further study could be led with broader datasets from different hospitals (both training & testing datasets). Moreover, these CNN could be trained to spot not only one but several types of diseases on X-ray images (meaning the outputs of the neural networks will have more than 2 classes).*

## 2. Introduction & Motivation

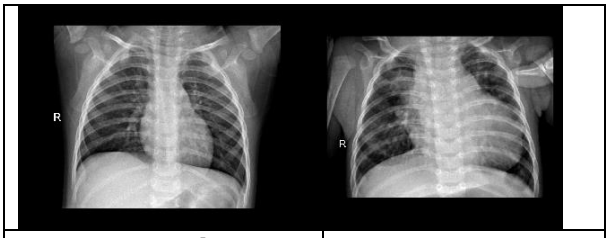
For 30 years, significant -if not spectacular- progress has been made to cure pneumonia. Several measures, including spreading the Pneumococcal Conjugate Vaccine (PCV3) and encouraging exclusive breastfeeding, have helped decreased substantially the mortality rate due to pneumonia.

Nonetheless, the global situation remains worrisome and lots of actions are still required. According to an article released by UNICEF on 28th January 2020, 9 million children could die within the next decade if the world doesn't boost up our ways to combat the deadly Pneumonia disease. If the current pace continues, 53 countries will not achieve the SDG3 target to end preventable child deaths.

In such circumstances, helping doctors to detect more easily and rapidly the illness seems all the more important. Indeed, early detection of pneumonia will decrease significantly the probability of death for the victims. The breakthrough of Artificial Intelligence (AI) and in particular Convolutional Neural Networks (CNNs) since 2010 provides new tools to improve this detection. Proving the efficiency of such tools for this task is the main motivation of this report.

## 3. Problem Definition

Pneumonia is caused by bacteria, viruses or fungi and mostly affects young children or old people. This disease fills the lungs of the patient with pus and fluid, making them struggling to breathe. There are multiple ways to diagnose this disease but the most commonly practised diagnosis is to analyse the X-Ray images of the patient's chest. The following image depicts the X-Ray images of normal and pneumonia affected lungs.



| Normal | Pneumonia |
|---|---|

There is a clear distinction between the two cases as the Pneumonia affected X-Ray has fluid filled in the left lung. Although some types of pneumonia can be prevented with vaccines and can be easily treated with low-cost antibiotics if properly diagnosed, tens of millions of children are still unvaccinated – and one in three children with symptoms do not receive essential medical care.

We feel that doctors can spend their time and energy on treating their patients rather than doing monotonous and repetitive work such as analyzing the X-rays and identifying whether a person is affected by Pneumonia or not. Therefore, we can state our problem as follows: testing different algorithm architectures to help diagnose pneumonia on X-ray images. More precisely, we want to test different CNNs to find the one that minimizes the recall (i.e. minimizes the number of False Negatives). Basically, our algorithms have to be able to classify any X-ray images of lungs as healthy or unhealthy (trace of pneumonia). This can be achieved by building algorithms which could classify the X-Rays without the doctor's intervention.

Of course, the underlying idea behind this approach is that our CNN will perform better than human eyes, even as experienced as physician's eyes. Or at least, that the support of already AI-diagnosed X-Rays images combined with the expertise of physicians will contribute significantly to the identification and the eradication of pneumonia.

## 4. Related Work

In [1], Rajpurkar et al analyzed the performance of a CNN called CheXNeXt, a 121 layer DenseNet architecture, in detecting abnormalities in chest x-rays and compared it to the diagnoses of radiologists. In their study, they looked at the detection of 14 various pathologies such as pneumonia, cardiomegaly, and emphysema. The algorithm they developed, which can locate parts of the chest-xray that are most indicative of each disease, managed to exceed the performance of radiologists on 11 out of 14 pathologies. Compared to the labeling time of 240 minutes of radiologists on 420 images, the algorithm managed to do the same in 1.5 minutes, offering a lot of time savings. The training process is robust to incorrect labels, as there are 2 stages: one in which multiple networks were trained to predict the probabilities for the presence of each of the 14 pathologies, and second in which an ensemble of a subset of these networks relabeled based on the mean of the predictions of each individual network.

In [2], three publicly available datasets (Indiana, JSRT, Shenzhen) were used in multiple deep convolutional networks (DCNs) to detect 20 different heart diseases. After exploring Alex-Net, VGG-Net, and ResNet, it was discovered that higher accuracy was achieved as the number of convolution layers increased. Like in [1], the portion of interest in identifying abnormalities was localized. This was done using heat maps from occlusion sensitivity. Also similar to [1], they have found that ensemble methods improve classification.

In [3], the subpar quality of images and annotations were addressed using a Contrast Induced Attention Network (CIA-Net), in which localization information is captured by contrasting positive and negative images. The paper concentrates more on the processing of the images more than the previous two papers, as they develop a learnable alignment module that corrects the mages in terms of scales, angles, and displacements of x-ray images. The combination of the CIA-Net and this alignment module results in accurate localization information.

In [4], the paper aimed to detect pneumonia in chest x-ray images using a simple CNN that was trained from scratch, as opposed to relying on transfer learning. Data augmentation was used to improve the accuracy of the model, as the dataset (consisting of images of chest x-rays from retrospective pediatric patients) has limited images. They discovered that smaller sized transformed images showed better validation accuracy.

## 5. Methodology:

1. *Main Idea: Classifying images thanks to CNN algorithms*

The strategy we adopted to tackle our problems is to classify images thanks to CNN algorithms.

The Convolutional Neural Networks are a particular type of multilayer perceptrons. They are primarily designed and used for image processing, even though they can have other applications in Natural Language Processing (NLP) or financial time series. They use convolutional and pooling layers that helped process the information contained in the pixels and learn the features of different categories of images.

CNN became famous when they began to be used for the ImageNet challenge. ImageNet is huge image dataset than have been built since 2006 and is still growing today. Each image is labeled according to defined categories (elephant for instance if the image contains an elephant). In 2011, the CNN Alexnet outperforms all the competitors and increased the accuracy by almost 10% compared to the previous years. This marked a real boom for the CNN who still are the reference algorithms for image classification today.

Due to their inherent quality to classify images, CNN are also widely used in medical imaging and are becoming reference tools in the domain. For these reasons, we naturally chose CNN algorithms to classify the X-ray images.

## *2. Preprocessing*

The dataset is classified into three different folders: train, test and validation. The train and test folders have two subfolders containing images labeled as normal and pneumonia. As a whole, we have 5863 images labeled by two categories. The output of the validation set will be used to get an accuracy score in Kaggle in the later stages of the project.

It is important to notice that this dataset comes from the same hospital in China, the medical centre of Guangzhou. A recent paper published in the Korean Journal of Radiology [5] has underlined the importance of training the dataset on external sources. For this study, we nevertheless only used the Guangzhou medical centre dataset.

We downloaded the X-ray images directly from Kaggle, imported them in a shared drive and used Google Colab Notebooks to process them. They were already divided in three folders: train, validation, and test sets, each folder being subdivided into two subfolders, one containing healthy lungs and the other unhealthy lungs.

To preprocess the image, we implemented a class function. This function first resized the images into 256*256 squares before cropping them into 224*224 matrixes. The reshaping of the images into 224*224 matrices was essential to make them compatible inputs with the CNN. After being resized and cropped, the image arrays are turned into tensors before being normalized. Then the class function attributed the right label (1 for pneumonia and 0 for a healthy person) thanks to the path of the images as they were classified into subfolders according to their status (pneumonia or not pneumonia).

To optimize the quality and the speed of our training, we then shuffled the training set and used mini-batch of size 10 for training, validation and test sets.

## *3. Algorithms used*

We trained three existing CNN on our datasets : AlexNet, ResNet18 and VGG16 as well as two custom CNNs.

*AlexNet:*

AlexNet uses 8 layers - 5 convolutional and 3 fully connected. It uses Rectified Linear Units (ReLUs) as activation function. As mentioned above, AlexNet has been one of the first CNN to be popularized for image recognition.

*ResNet18:*

ResNet18 has 18 layers and is the deepest network we are using. It belongs to the ResNet CNN family. The ResNet algorithms have been built to avoid certain drawbacks of deeper networks. As the number of layers increases, the back propagation becomes longer and the gradient smaller at each step, decreasing the accuracy and the quality of the model. This is the problem of “vanishing gradients”. To tackle this issue, the ResNet architecture allows to skip some layers during the forward and backward propagation. By doing so, it reduces the vanishing gradient problem as well as accelerate the training.

To see the potential impact of transfer learning, we decided to use a pre-trained version of the ResNet18 algorithm.

*VGG16:*

VGG16 is deeper than AlexNet with 13 convolutional and 3 fully connected layers. Consisting of more than 100 Millions of parameters, it is more precise than AlexNet on the ImageNet dataset but also more computationally intensive. We have chosen this model as this has produced an accuracy of 92.7% for the ImageNet dataset where the images are resized to 224*224 (similar to our data set).

Understanding the simplistic nature of our images we also had the motivation to try shallow custom made CNNs and compare their performances with already existing CNNs. Indeed, our dataset is made with black and white images; the need for deep architecture is therefore to be demonstrated.

*Custom made Architecture #1:*

In the first custom made architecture, we used two convolution networks with two fully connected networks. We also coupled them with dropout functions and with two different kinds of activation functions. We tried Leaky ReLu activation function for the convolution networks and normal ReLu activation for the fully connected ones. We initially evaluated this model with a train and validation data sets and obtained this loss curves for 30 epochs.

*Custom made Architecture #2:*

In the second custom architecture, we used 2 convolutional layers with one fully connected layer coupled with 2*2 max-pooling and one activation function(ReLu).

We trained all the algorithms for 30 epochs. For each model, we drew a train-validation loss graph with the number of epochs on the X-axis as well as a confusion matrix to compute our recall score. We used SGD optimizer (rather than Adam optimizer) and a cross entropy loss function as we found out they provided the best results for our models.

### *4. Limits and difficulties*

We discovered that the VGG 16 was really a time consuming algorithm. The entire saved model after a 30 epochs training is more than 500 MB. Despite using several attempts to exploit GPU on from the Google Cloud Platform, we could not compute our accuracy on the test set and therefore were not able to provide a confusion matrix.

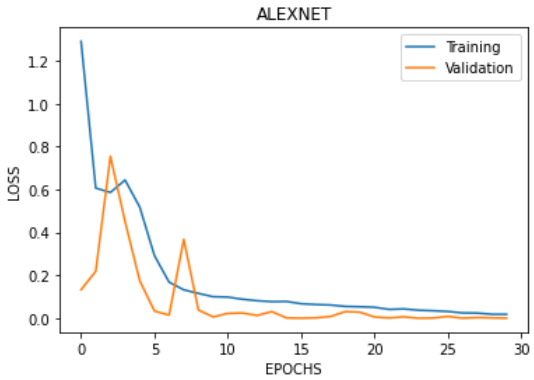
Other methods could have been used to improve the quality of our prediction. In particular, given the relative imbalance between normal images and pneumonia images, using data augmentation could have been helpful to improve the performance of our CNNs. Using a combination of datasets from different hospitals instead of just one may also provide a solution that can generalize better.

## 6. Evaluation

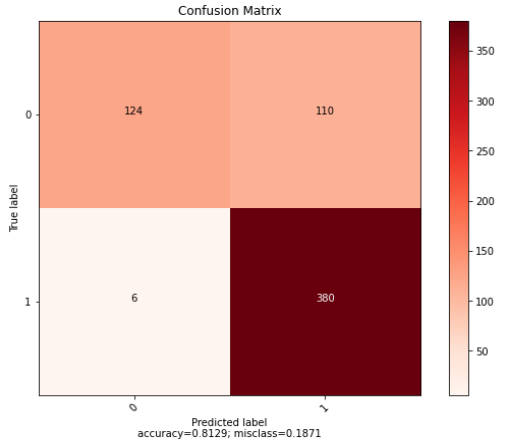
We used 3 pretrained architectures and two custom made ones to classify our preprocessed and cleaned images.

*AlexNet:*

We initially tried using the AlexNet in our image set. We segregated our images into train and validation data loaders with the batch size of 10 to evaluate the model before running it using the test data set. We added another additional layer to make the output binary.




The above picture depicts the train and validation data set loss curves for 30 epochs for the AlexNet architecture. We can see that both the train and validation losses tend to converge after the 30th epoch. We notice a lot of variations with the validation loss curve but the variations tend to stabilise after the 9th epoch. The training losses tend to show one major deviation but then follows a smooth trend of convergence. There is no indication of overfitting from the graph.

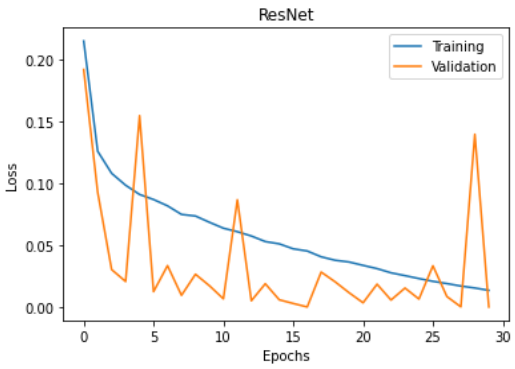



| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.95 | 0.53 | 0.68 | 234 |
| 1 | 0.78 | 0.98 | 0.87 | 386 |
| accuracy | | | 0.81 | 620 |
| macro avg | 0.86 | 0.76 | 0.77 | 620 |
| weighted avg | 0.84 | 0.81 | 0.80 | 620 |

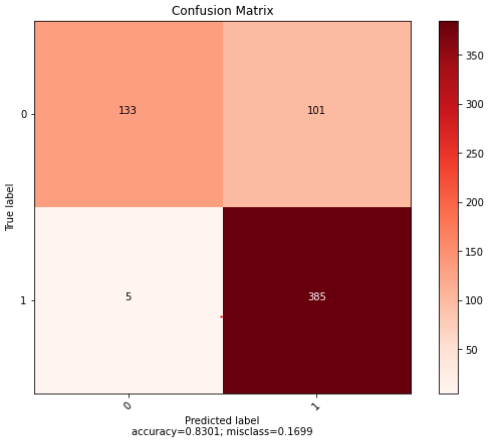
To evaluate the model we looked into the confusion matrix too. In the above-mentioned confusion matrix, label 1 indicates the images with Pneumonia whereas label 0 denotes normal X-rays. We obtained this confusion matrix from testing our trained model in the test set. This confusion matrix indicates that AlexNet works with an accuracy of 81.3% with a Pneumonia identification recall of 98%.

*ResNet 18:*

We performed a similar evaluation with ResNet 18 in our data set. We also added another additional layer to make the output binary. But unlike our last approach with AlexNet we used the pretrained parameters to train our model here. In AlexNet we only used the architecture and obtained the training parameters directly from the backpropagation of our train dataset.




In the above-mentioned loss curves, the training losses tend to take a longer time to converge, unlike our previous results. We expect the training losses to converge as we train our models with more epochs. We also notice a lot of variations in the validation losses in this model. This can denote the need for training the model for more epochs or it could be an indication for overfitting.

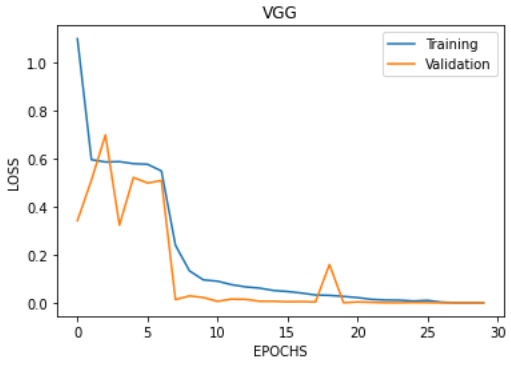



| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.96 | 0.57 | 0.72 | 234 |
| 1 | 0.79 | 0.99 | 0.88 | 390 |
| accuracy | | | 0.83 | 624 |
| macro avg | 0.88 | 0.78 | 0.80 | 624 |
| weighted avg | 0.86 | 0.83 | 0.82 | 624 |

When we evaluated the confusion matrix for the test data we obtained an accuracy of 83.01% with a Pneumonia labeled recall of 99%.

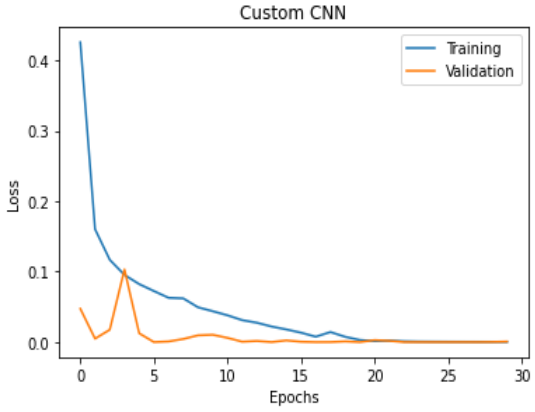
*VGG16:*

We also tried VGG 16 pre-trained architecture with our dataset. We observed that the training and validation loss curves converged much earlier when compared to the other pre-trained deep models. At the same time, we can see that there is a stepwise reduction in the losses as the epochs increase.

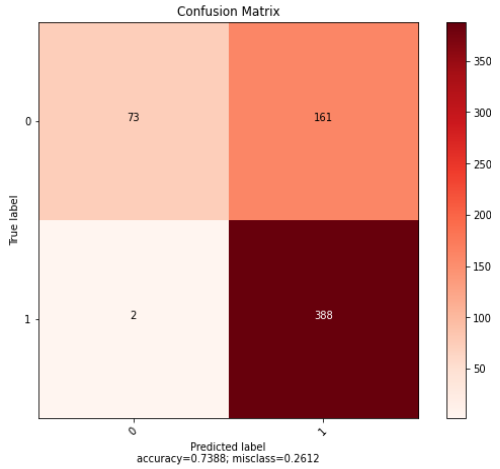


Due to computational limitations, we weren't able to run the model with the testing set to get the confusion matrix.

*Custom made Architecture #1:*



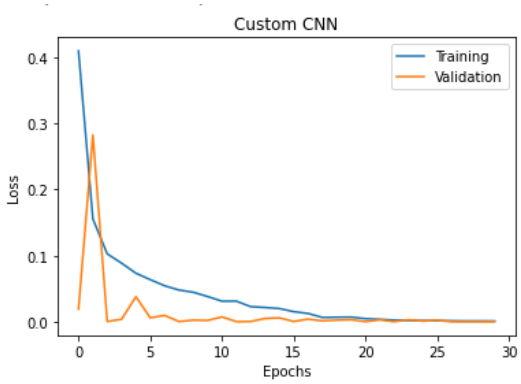

We can see that both training and validation loss curves converge very early unlike the other deep pre-trained architectures. Both of them tend to converge even before the 20th epoch. Then the losses remain stable near 0. Despite the minor variations in the validation loss curve in the early stage, the loss tends to stabilise early and converge.



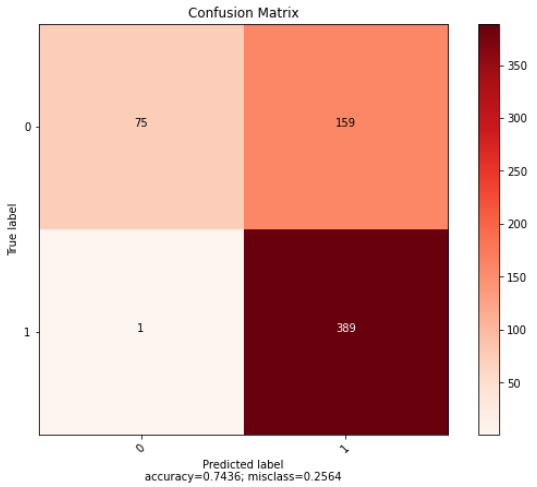

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.97 | 0.31 | 0.47 | 234 |
| 1 | 0.71 | 0.99 | 0.83 | 390 |
| accuracy | | | 0.74 | 624 |
| macro avg | 0.84 | 0.65 | 0.65 | 624 |
| weighted avg | 0.81 | 0.74 | 0.69 | 624 |

The confusion matrix signifies an accuracy of 73.8% with a Pneumonia labeled recall of 99%. But this has a relatively higher normal labeled precision when compared to the pre-trained architecture.

*Custom made Architecture #2:*




Observing the training and validation loss curves gives us an intuition that, the losses for this CNN converge early unlike the other deep pre-trained models. The training and validation loss curves tend to converge around 23rd epoch and remain stable after that. Just like the previous customised CNN, the validation losses show very high variation in the early stages and then later stabilises.




| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.99 | 0.32 | 0.48 | 234 |
| 1 | 0.71 | 1.00 | 0.83 | 390 |
| accuracy | | | 0.74 | 624 |
| macro avg | 0.85 | 0.66 | 0.66 | 624 |
| weighted avg | 0.81 | 0.74 | 0.70 | 624 |

The confusion matrix indicates that this model has an accuracy of 74.36% when tested with the test data with a Pneumonia labeled recall of 100%.

## 7. Conclusion:

To conclude, these are our inferences from our experiments with various models:

1. ResNet18 provides the best results so far with the accuracy of 83% and a recall of 99%.
2. But our custom model #2 gave us a 100% recall.
3. Shallow custom made CNNs have a slightly higher precision.

Since our main objective is to identify the Pneumonia cases from the chest X-rays, we believe that we need to give more importance to the Pneumonia labeled recall rather than the accuracy. We feel that it is imperative in this application to identify all true positives (X-Rays which indicate Pneumonia in real-time) with the allowance of a mismatch in identifying then true negatives (The X-ray indicates the patient not to have Pneumonia whereas our model indicates the opposite).
Hence we concur that the custom made model #2 with the highest recall (100%) is best suited for the practical situation.

As our different models perform quite well, we see our results as encouraging. Further studies could be led in the domain. In particular, by gathering X-ray images from different hospitals and on different diseases, we could train CNN to identify not only one but several illnesses. Data augmentation could be a useful tool to widen our datasets and improve the quality of such future work.

## 8. References


[1] Rajpurkar P, Irvin J, Ball RL, Zhu K, Yang B, Mehta H, et al. (2018) Deep learning for chest radiograph diagnosis: A retrospective comparison of the CheXNeXt algorithm to practicing radiologists. PLoS Med 15(11): e1002686. https://doi.org/10.1371/journal.pmed.1002686

[2] Islam, M. T., Aowal, M. A., et al. (2017) Abnormality Detection and Localization in Chest X-Rays using Deep Convolutional Neural Networks. arXiv:1705.09850

[3] Liu, J., Zhao, G., Fei, Y., Zhang, M., Wang, Y., & Yu, Y. (2019). Align, Attend and Locate: Chest X-ray Diagnosis via Contrast Induced Attention Network with Limited Supervision. The IEEE International Conference on Computer Vision (ICCV), 2019, pp. 10632-10641

[4] Stephen, O., Sain, M., et al. (2019) An Efficient Deep Learning Approach to Pneumonia Classification in Healthcare. Hindawi Journal of Healthcare Engineering. https://doi.org/10.1155/2019/4180949

[5] Kim, D. W., Jang, H. Y., Kim, K. W., Shin, Y., & Park, S. H. (2019). Design characteristics of studies reporting the performance of artificial intelligence algorithms for diagnostic analysis of medical images: results from recently published papers. Korean journal of radiology, 20(3), 405-410.