COVID 19 Detection Through Radiograph Images Using Deep Learning Algorithm

Alpas, Michael
Department of Computer Science, UIUC
(309)533-8994
malpas2@illinois.edu

Ahmad, Wasique
Department of Computer Science, UIUC
(414)630-6361
Wasique2@illinois.edu

Singh, Mohit
Department of Computer Science, UIUC
(309)750-7110
Mohits3@illinois.edu

Yadav, Upendra Singh
Department of Computer Science, UIUC
(215)439-4579
Usyadav2@illinois.edu

1. ABSTRACT

The COVID-19 pandemic until this write-up is still causing major outbreak in all countries around the world. The COVID-19 has led to a dramatic loss of human life worldwide and presents an unprecedented challenge to public health, food systems and the world of work caused by the infection of individuals by the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). The economic and social disruption caused by the pandemic is devastating tens of millions of people are at risk of falling into extreme poverty, while the number of undernourished people is Constantly increasing.

A critical step in the fight against COVID-19 is effective screening of infected patients, such that those infected can receive immediate treatment and care, as well as be isolated to mitigate the spread of the virus. The main screening method used for detecting COVID-19 cases is reverse transcriptase-polymerase chain reaction (RT-PCR) testing, which can detect SARS-CoV-2 RNA from respiratory specimens (collected through a variety of means such as nasopharyngeal or oropharyngeal swabs). While RT-PCR testing is the excellent, it is also very time-consuming, laborious, and complicated manual process that is also in short supply.

Uncovering this disease from radiography and radiology images is perhaps one of the efficient alternative ways to diagnose the patients. Our team was inspired by earlier works from fellow students, professors, and medical professionals.[15] For this project, we used chest x-rays images from publicly available datasets in which several licensed radiologists evaluated initial images that demonstrated COVID-19 disease.

Due to recent development in the field of Machine Learning Models and also in medical imaging field, deep learning methods can be applied to improve the accuracy of diagnosis of COVID-19 compared with the gold RT-PCR test and also to provide valuable insight for pre diagnose of patient outcomes.

In this paper we have proposed a solution which analyzes a broad range of Xray images [19] of patients using deep learning techniques such as Convolutional Neural Network, pre trained CNN models ((ResNet50, ResNet101, ResNet152 and InceptionV3) and PyTorch in Amazon SageMaker. We have also hosted our built model in Amazon SageMaker and exposed it as a web service to be consumed by any external third-party system. For simplicity of our studies, we have also developed a web application which act as an interface to detect COVID. Any User can upload his Xray image in web portal which internally consumes Amazon SageMaker hosted model and can find results in few minutes.

2. Introduction

The outbreak of unknown disease started since December 2019 which has spread from Wuhan, China.

It was completely unknown at first, but several specialists diagnosed its symptoms similar to those of coronavirus infection. [16]. As of April 2021, coronavirus cases reached to almost 142 million which has a death toll of close to 3 million and around 121 million recovered from the disease worldwide. [17]

Despite the worldwide research efforts over the past few months, early detection of COVID-19 remains a challenging issue due to limited resources and the amount of data available for research.

At present, this pandemic remains to challenge medical systems, governments, and businesses worldwide in many aspects – demands for health care workers, availability of effective vaccines, shortage of medical equipment while many healthcare workers and government officials have themselves been infected and continues to threaten everybody despite of good progress in managing it.

Chest radiography imaging is being used as an alternative screening method and done in parallel to PCR viral testing. The accuracy of Chest X-ray (CXR) diagnosis of COVID-19 infection strongly relies on radiological expertise due to the complex morphological patterns of lung involvement which can change in extent and appearance over time. If these patterns are detected with high accuracy, it can enable rapid triaging for screening, diagnosis, and management of patients with suspected or known COVID-19 infection.

Deep learning techniques, in the particular Convolutional Neural Networks (CNN), have been beating humans in various tasks of computer vision and other video processing tasks in recent years. Deep learning algorithms have already been applied for the detection and classification of Pneumonia and other diseases on radiography.

In this paper, as an effort to improve the current COVID-19 detection using a limited number of publicly available CXR dataset. our team had the opportunity to apply some of the Deep Learning algorithms we learned from our class.

2.1 Benefit of using Xray images in detecting COVID19.

Rapid Triaging – CXR imaging enables rapid triaging of patients suspected of COVID-19 and can be done in parallel of viral testing (which takes time) to help relief the high volumes of patients especially in areas most affected where they have run out of capacity.

Availability of training data - CXR imaging is readily available and accessible in many clinical sites and imaging centers as it is considered standard equipment in most healthcare systems.

Portability - The existence of portable CXR systems means that imaging can be performed within an isolation room, thus significantly reducing the risk of COVID-19 transmission.

Motivated by the urgent need to develop solutions to aid in the fight against the COVID-19 pandemic and also inspired by the open source and open access efforts of the research community, and also as a deep learning project need, we thought of exploring various deep learning methods specifically a deep convolutional neural network design using PyTorch framework to detect COVID-19 cases from CXR images that are open source and available to the general public.

3. RELATED WORK

Different Artificial Intelligence tools have produced established and precise results in the applications that use either image-based or other types of data. Apostolopoulos and Mpesiana [4] performed one of the first studies on COVID-19 detection using X-ray images. In their research they evaluated the performance of state-of-the-art convolutional neural network architectures proposed over the recent years for medical image classification.

Experimental results show that their proposed model achieved an overall accuracy of 89.6%, and more importantly the precision and recall rate for COVID-19 cases are 93% and 98.2% for 4-class cases (COVID vs Pneumonia bacterial vs pneumonia viral vs normal). For 3-class classification (COVID vs Pneumonia vs normal), the proposed model produced a classification accuracy of 95%. The preliminary results of this study look promising which can be further improved as more training data becomes available.[14]

Their model achieved promising results on a small, prepared dataset which indicates that given more data, the proposed model can achieve better results with minimum pre-processing of data. Overall, the proposed model substantially advances the current radiology-based methodology and during COVID-19 pandemic, it can be very helpful tool for clinical practitioners and radiologists to aid them in diagnosis, quantification, and follow-up of COVID-19 cases.

4. DATA

While working in the project, we first used below IEEE "covid-chest X-ray dataset" (https://github.com/ieee8023/covid-chestxray-dataset)[2].

This data contains 225 COVID-19 chest X-ray images, which were obtained from Cohen data source and also available in above Github link.

This is the very first publicly available COVID-19 X-ray image data set; hence some patient's information is missing in few places.

Dataset contains multiple important metadata fields such as demographic features such as – sex, age and medical features such as temperature, WBC count, neutrophil count, lymphocyte count, O2 saturation, extubated, intubated, survival and RT_PCR positive. This dataset contains 131 male patients and 64 female patients and the average age for the group is 58.8±14.9 years.

However, while building the model, we realized that numbers of image data were quite less in this dataset, so we used few more datasets such as "X-Ray Image DataSet" from Ozturk et al. [21]. This dataset contains another 1127 images in three classes: 127 "Covid-19", 500 'No_findings', and 500 "Pneumonia".

During the model tuning and optimization phase, we added one more public image data sets and Chowdhury et al. [22] studies.

The final data set consists of 4412 frontal chest X-ray images in three classes: 698 "Covid-19", 1851 'No_findings', and 1863 "Pneumonia".

For finally model training, testing and optimization we have merged these three datasets [1], [21],[22]. After this, we used random 80% of dataset for training purpose and remaining 20% for testing purpose.

5. APPROACH

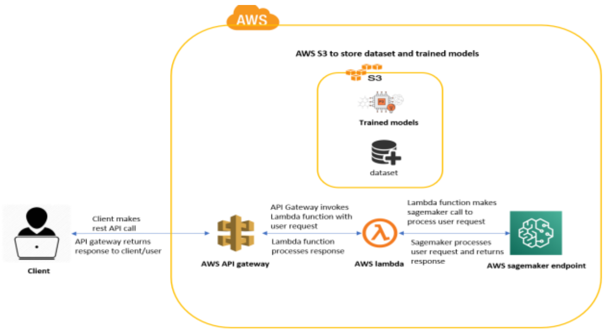
The entire project development cycles is divided into two stages. Feature Engineering using CNN (Convolutional Neural Network) and Model Training.[5]

- For PyTorch Model Training, we have used ResNet-50, ResNet152 and InceptionV3 pre trained models.[6]
- For building the last layer of binary or multi-level classification, we are using below classifier algorithms.
  - Logistic Regression.[7]
  - Boosted Random Forest Classification.[8]
  - Decision Tree.[9]
- DL Frameworks [10]
  - PyTorch and PySpark

5.1 High Level Architecture

Here is high level architecture for the project, we used amazon simple storage Service (AWS S3) to store dataset and trained models. These models were built and trained using ResNet-50 and InceptionV3. This model(ResNet-50) is hosted in Amazon

SageMaker[23] and SageMaker's endpoint is used in lambda function[24] which are externalize using Amazon API gateway as a REST endpoints.



5.2 Methodology Implementation

Models ResNet-50, short for Residual Networks, is a 50-layer-deep classic convolutional neural network. It has 5 stages each with a convolution block of 3 convolution layers and an identity block of 3 convolution layers. Like other ResNet models, ResNet-50 uses skip connection to add the output from an earlier layer to a later layer to mitigate the vanishing gradient problem.

InceptionV3 is another powerful convolutional neural network. This was developed from Google net, a deep learning model from Google. This mainly focuses on reducing computation and increasing optimization by applying different techniques such as factorized convolutions, regularization, dimension reduction, and parallelized computations. We used the ResNet-50 and InceptionV3 pretrained models. These two models have established pretrained weights and biases from the ImageNet classification training and have many well-built convolutional, RELU, and Dropout layers.

5.3 PyTorch and AWS SageMaker

PyTorch natively supports ResNet50 and InceptionV3 pretrained models. We used PyTorch to build the classification models. We adapt the concept of transfer learning, in which the models use all but the last fully connected layer as a fixed feature extractor and then trained a classifier.[18]. The data include images in various sizes and type formats, we resized all the images to the size of 224x224 for ResNet-50 and 299x299 for InceptionV3, normalized them with 0.485 mean and 0.229 standard deviation, and converted them into multidimensional tensors using torchvision.transform package.

We mixed up the dataset images and divided them into batches of 20 using the DataLoader function to feed the PyTorch-based machine learning model. In addition to the two pretrained models, for the PyTorch framework, we used a cross-entropy loss function to measure the deviation of the predictions of each class. We also used Stochastic Gradient Descent (SGD) as the optimizer with a learning rate of 0.001 that updates the values of the parameters of each layer and generates probabilities for the target class labels via its SoftMax function behind the scenes.
However, the results were not expected. We then changed the optimization technique to Adaptive Moment Estimation (Adam) method with a learning rate of 3e-5. We applied the train, evaluation, and visualization functions.

We used AWS SageMaker's GPU powered integrated Jupyter authoring notebook instance to build and train both ResNet-50 and InceptionV3 models using custom PyTorch code.

With our current analysis we have found that the ResNet-50 model performed better than the InceptionV3 model, we chose to host the ResNet-50 model using SageMaker hosting services to provide inferences. We used AWS SageMaker Neo to compile and deploy the trained model on a GPU powered real time inference instance.
We used AWS S3 storage to store the dataset to train the model, the best trained model and the SageMaker Neo compiled model.

In order to allow the hosted model to be accessible from external applications, we integrated the hosted model with an Internet facing RESTful API built using Amazon API Gateway and Amazon Lambda function written in Python programming language. API Gateway acts as the 'Entry point' to our serverless User Interface facing web application built in React using AWS Amplify. The User Interface client application is a single page web application which provides users the ability to upload their Chest X-ray image.

Once user uploads the image, the request is sent to the POST RESTful API hosted on AWS API Gateway which then triggers the Lambda function. The Lambda function forwards the request to the hosted model. The hosted model then provides the data back to the Lambda function which is then forwarded to the API Gateway and finally to the User interface client application.

5.4 Model overview

5.4.1 CNN

In deep learning, a convolutional neural network (CNN, or ConvNet) is a class of deep neural networks, most commonly applied to analyzing visual imagery. The name "convolutional neural network" indicates that the network employs a mathematical operation called convolution.
CNNs are used for image classification and recognition because of its high accuracy. Examples of CNN in computer vision are face recognition, image classification etc.
Convolutional networks are a specialized type of neural networks that use convolution in place of general matrix multiplication in at least one of their layers. It follows a hierarchical model which works on building a network, like a funnel, and finally gives out a fully connected layer where all the neurons are connected to each other and the output is processed.
The main purpose of Conv. layers is to extract features from input images. In this study, different Conv. Layers can be applied to extract different types of features such as edges, texture, colors, and high-lighted patterns from the images.
A convolution layer is simply a feature detection layer. Every convolution layer has a specific number of channels; each channel detects a specific feature in the image. Each feature to detect is often called a kernel or a filter. The kernel is of a fixed size, usually, kernels of size 3 x 3 are used.

In a CNN, the input is a tensor with a shape: (number of inputs) x (input height) x (input width) x (input channels). After passing through a convolutional layer, the image becomes abstracted to a feature map, also called an activation map, with shape: (number of inputs) x (feature map height) x (feature map width) x (feature map channels). A convolutional layer within a CNN generally has the following attributes:

- Convolutional filters/kernels defined by a width and height (hyper-parameters).

- The number of input channels and output channels (hyper-parameters). One layer's input channel must equal the number of output channels (also called depth) of its input.

- Additional hyperparameters of the convolution operation, such as: padding, stride, and dilation.

5.4.2 Logistic Regression

Logistic regression is a supervised, classification algorithm which is used when the value we are going to predict can be divided into categories. Logistic regression transforms its output using the logistic sigmoid function, calculates its weights and biases to return a probability value which can then be mapped to two or more discrete classes.
As pneumonia and COVID detection is a classification problem, we can use logistic regression to return a probability score between 0 and 1, with 0 representing a normal lung and 1 representing infected lungs.

5.4.3 Random Forest

Random forest, like its name implies, consists of a large number of individual decision trees that operate as an ensemble. It uses bagging and feature randomness when building each individual tree to try to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree. [11]

A Boosted Random Forest is an algorithm, which consists of two parts; the boosting algorithm: AdaBoost and the Random Forest classifier algorithm [12] which in turn consists of multiple decision trees. A decision tree builds models that are similar to an actual tree. The algorithm divides our data into smaller subsets, simultaneously adding branches to the tree. The outcome is a tree consisting of leaf nodes and decision nodes. A decision node has two or more branches representing the value of each feature (like age, symptom1, etc.) tested and the leaf node holds the result value on the patient's prospective condition (target value).

Multiple classifier decision trees (ensemble of classifiers) eliminate the risk of failure of a single decision tree to correctly predict the target value. Thus, the random forest averages the result provided by multiple trees to provide the final result.

5.4.4 PyTorch

PyTorch is an open-source machine learning library based on the Torch library that implements a dynamic computational graph, which allows you to change the way your neural network behaves on the fly and capable of performing backward automatic differentiation. It provides two high-level features:[6]

- Tensor computing (like NumPy) with strong acceleration via graphics processing units (GPU)

- Deep neural networks built on a type-based automatic differentiation system.

5.5 Performance Criteria

A variety of performance criteria are used to evaluate the performance of classification models, namely classification accuracy, sensitivity, specificity, precision, recall, and F1 score.
Sensitivity and specificity are two suitable criteria that can be used to report model performance. These criteria are also widely used in the medical field.

The cross-validation estimator can be used and also can be created a confusion matrix, as shown in table below.

Table 2. Confusion Matrix.

	Predicted Positive	Predicted Negative
Actual Positive	TP	FN
Actual Negative	FP	TN

FN, false negative; FP, false positive; TN, true negative; TP, true positive.

- TP (true positive) is a number of anomalies and was identified with the correct diagnosis.
- TN (true negative) is an incorrectly measured number of regular instances.
- FP (false positive) is a Type 1 error, a set of regular instances that are classified as an abnormality diagnosis.
- FN (false negative) is a Type 2 error, a list of abnormalities observed as an ordinary diagnosis.

Accuracy is defined by the rate of correctly classified images.

$$Accuracy = TP + TN/(TP + TN + FP + FN)$$

Precision is used to give relationship between the TP predicted values and FP predicted values.

$$Precision = TP/(TP + FP)$$

Recall is defined as the ratio of the total number of correctly classified positive patients to the total number of positive patients. It should be as high as possible.

$$Recall = TP/(FN + TP)$$

F1 score: It is difficult to compare two models with low precision and high recall, or vice versa. Thus, to render them comparable, we use the F1 score. It allows for the measurement of the Recall and Precision values at the same time. It uses the harmonic mean in place of the arithmetic mean by penalizing the extreme values more.

$$F1\ score = 2 * (precision * recall) / (precision * recall)$$

5.6 Experimental Setup

For this experiment, we are using the following technology stack – Docker, Jupyter Notebook and PyTorch. To manage the dependencies, we are planning to leverage containerized environment using Docker. This container will be responsible for hosting a Jupyter notebook environment and for our base image, we will install Python libraries such as PyTorch, Numpy and Pandas. The Docker image will be saved in teams' Docker Hub. The advantage of this kind of set-up, team members will allow to have a ready-to-go consistent working environment so each member can focus on the assigned tasks.

The containerized set-up is a simple solution, however, in the event that a greater processing power is needed due to limitation of team member's individual laptops – we discussed to possibly leverage AWS Elastic Container Service or AWS Fargate for managed serverless compute for containers.

6. RESULTS

To evaluate the efficacy, accuracy of our model we used F1 score, accuracy, sensitivity, and confusion matrix.

We are still trying to optimize our results, however our draft versions results are below, we tested binary classification at first with ResNet50 trained model and then with InceptionV3 model and compared these two models.

Covid Calssification Results				
Used Models	Precision (COVID Image Only)	Recall (COVID Image Only)	F1 Score (COVID Image Only)	Accuracy
ResNet-50 Small dataset	0.94	0.93	0.92	0.93
InceptionV3 Small Dataset	0.88	0.89	0.87	0.88
ResNet-50 Full dataset	0.95	0.97	0.96	0.95
InceptionV3 Full Dataset	0.91	0.92	0.91	0.9

7. DISCUSSION

As per above results, both CNN trained models ResNet-50 and InceptionV3 are showing promising results. We are getting approximately 95% accuracy in ResNet-50 model and 90% accuracy in InceptionV3 model.

The above results indicate that ResNet-50 model is more effective and accurate in predicting results.

This could be explained by underlining architecture of Inception V3 and ResNet-50 models. InceptionV3 model has issue with vanishing gradient problem and on the other end ResNet50 uses skip connection to propagate information over layers and enhances the detection of smaller objects in the image, which led to improve accuracy in detecting the results.

While developing this project, at first we thought of hosting this project locally in a docker image, but due to very slow performance and umpteen issues related to environment finally we decided to move to AWS environment. Our teammates learnt most of the AWS services such as AWS SageMaker, AWS S3 etc.

Another challenge, we had in selection of CNN models. We selected multiple CNN models, trained them and tested and checked misclassification rate, it took a lot of time and efforts in deciding the correct CNN model.

In our next step, we will fine tune our model and use n-fold cross validation and will see how it will improve accuracy.

7.1 Optimization

For optimizing the results in PyTorch framework, at first, we have used a cross-entropy loss functions to measure the prediction of each class. Along with the cross-entropy function, we also used Stochastic Gradient Descent (SGD) as the optimizer with a learning rate of 0.001 which updates the value of parameter of each layer and hence generates probabilities for the target class labels via its SoftMax function.

After using cross-entropy loss function and also SGD optimization we could not achieve good accuracy, then we change the optimization technique to ADAM (Adaptive Moment Estimation) method with learning rate 3e-5. By using cross entropy Loss function and also ADAM optimizer with learning rate of 3e-5, we improved the results having more than 90% accuracy.

8. CONCLUSION

In this paper, The results showed that the convolutional neural network with minimized convolutional and fully connected layers are capable of detecting COVID-19 with good accuracy so far obtained results are promising however we are trying to optimize our models and improving it accuracy.

At last, we can say that deep learning algorithms such as CNN are capable to analyze Xray images and easily detection of COVID-19 decease. We used cloud computing architecture – AWS SageMaker and also demonstrated how it can be used to host machine learning model.

9. CONTRIBUTION

Each team member was given an opportunity to read several research papers which helped us build our ideas and strategy for the project. Individuals had a chance to analyze the dataset and developed the models that was needed, and we used the containerized environment. Also, there were several tasks assigned to individuals – Michael focus was making the container environment which was completed in the 1st week of April. Along with this, members had opportunity to read different research papers and shared learned approaches for our project. Mohit did coordination and draft were completed on April 20th.

Wasique and Upendra worked on Deep Learning algorithms based on the research papers we read and did setup of AWS.
After individual tasks, everyone contributed on the analysis and coding. We completed the coding and prepared draft.

10. REFERENCES

[1] Roberts, M., Driggs, D., Thorpe, M. *et al.* Common pitfalls and recommendations for using machine learning to detect and prognosticate for COVID-19 using chest radiographs and CT scans. *Nat Mach Intell* **3,** 199–217 (2021). (link: https://doi.org/10.1038/s42256-021-00307-0)

[2] Cohen, J. P. COVID-19 Image Data Collection. ArXiv. 2020, arXiv:2003.11597.

[3] https://doi.org/10.1177/2472630320958376

[4] Apostolopoulos, I. D., Mpesiana, T. Covid-19: Automatic Detection from X-Ray Images Utilizing Transfer Learning with Convolutional Neural Networks. Phys. Eng. Sci. Med. 2020, 43, 635–640

[5] Z. Zahisham, C. P. Lee and K. M. Lim, "Food Recognition with ResNet-50," 2020 IEEE 2nd International Conference on Artificial Intelligence in Engineering and Technology (IICAIET), Kota Kinabalu, Malaysia, 2020, pp. 1-5, doi: 10.1109/IICAIET49801.2020.9257825. (link: link: https://doi.org/10.1016/j.procs.2018.05.069)

[6] https://towardsdatascience.com/an-overview-of-resnet-and-its-variants-5281e2f56035https://towardsdatascience.com/an-overview-of-resnet-and-its-variants-5281e2f56035

[7] Peng, CY.J., So, TS.H., Stage, F.K. et al. The Use and Interpretation of Logistic Regression in Higher Education Journals: 1988–1999. Research in Higher Education 43, 259–293 (2002). https://doi.org/10.1023/A:1014858517172

[8] Mariana Belgiu, Lucian Drăguţ,Random forest in remote sensing: A review of applications and future directions, ISPRS Journal of Photogrammetry and Remote Sensing, https://doi.org/10.1016/j.isprsjprs.2016.01.011. (https://www.sciencedirect.com/science/article/pii/S0924271616000265)

[9] Yoo, S. H., Geng, H., Chiu, T. L., Yu, S. K., Cho, D. C., Heo, J., Choi, M. S., Choi, I. H., Cung Van, C., Nhung, N. V., Min, B. J., & Lee, H. (2020). Deep Learning-Based Decision-Tree Classifier for COVID-19 Diagnosis From Chest X-ray Imaging. Frontiers in medicine, 7, 427. https://doi.org/10.3389/fmed.2020.00427

[10] https://blog.usejournal.com/deep-learning-with-pytorch-zero-to-gans-week-1-592397473811

[11] Md. Zahangir Alam, M. Saifur Rahman, M. Sohel Rahman, A Random Forest based predictor for medical data classification using feature ranking, https://doi.org/10.1016/j.imu.2019.100180.(https://www.sciencedirect.com/science/article/pii/S235291481930019X)

[12] Iwendi Celestine, Bashir Ali Kashif, Peshkar Atharva, Sujatha R., Chatterjee Jyotir Moy, Pasupuleti Swetha, Mishra Rishita, Pillai Sofia, Jo Ohyun TITLE=COVID-19 Patient Health Prediction Using Boosted Random Forest Algorithm (link:https://www.frontiersin.org/article/10.3389/fpubh.2020.00357)DOI=10.3389/fpubh.2020.00357

[13] Khanday, A.M.U.D., Rabani, S.T., Khan, Q.R. et al. Machine learning based approaches for detecting COVID-19 using clinical text data. Int. j. inf. tecnol. 12, 731–739 (2020). https://doi.org/10.1007/s41870-020-00495-9

[14] Khan AI, Shah JL, Bhat MM. CoroNet: A deep neural network for detection and diagnosis of COVID-19 from chest x-ray images. Compute Methods Programs Biomed. 2020 Nov;196:105581. doi: 10.1016/j.cmpb.2020.105581. Epub 2020 Jun 5. PMID: 32534344; PMCID: PMC7274128.

[15] Wang W, Xu Y, Gao R, et al. Detection of SARS-CoV-2 in Different Types of Clinical Specimens. JAMA. 2020;323(18):1843–1844. doi:10.1001/jama.2020.3786

[16] N. Chen, M. Zhou, X. Dong et al., "Epidemiological and clinical characteristics of 99 cases of 2019 novel coronavirus pneumonia in Wuhan, China: a descriptive study," The Lancet, vol. 395, no. 10223, pp. 507–513, 2020.View at: Publisher Site |

[17] https://www.worldometers.info/coronavirus/

[18] Minaee, S., Kafieh, R., Sonka, M., Yazdani, S., & Soufi, G. J. (2020). Deep-covid: Predicting covid-19 from chest x-ray images using deep transfer learning. arXiv preprint arXiv:2004.09363.

[19] Cohen, J. P., Morrison, P., & Dao, L. (2020). COVID-19 image data collection. arXiv preprint arXiv:2003.11597

[20] Narin, A., Kaya, C., Pamuk, Z. (2020). Automatic Detection of Coronavirus Disease (COVID 19) Using X-ray Images and Deep Convolutional Neural Networks. arXiv preprint arXiv:2003.10849.

[21] Ozturk, T., Talo, M., Yildirim, E. A., Baloglu, U. B., Yildirim, O., & Acharya, U. R. (2020). Automated detection of COVID-19 cases using deep neural networks with X-ray images. Computers in Biology and Medicine, 103792.

[22] Chowdhury, M. E., Rahman, T., Khandakar, A., Mazhar, R., Kadir, M. A., Mahbub, Z. B., ... & Reaz, M. B. I. (2020). Can AI help in screening viral and COVID-19 pneumonia?. arXiv preprint arXiv:2003.13145.

[23] https://aws.amazon.com/sagemaker/

[24] https://docs.aws.amazon.com/lambda/latest/dg/welcome.html