ISSN: 2689-1204

Research Article

Journal of Applied Material Science & Engineering Research

Interpreting the Predictions of Deep Network Build to Identify Early Detection of COVID-19 in X-Ray Images

Vipin Bansal[1*], Amit Jain[2]

[1]*Research Scholar Chandigarh University, India*

[2]*Associate Professor Chandigarh University, India*

[*]Corresponding author
Vipin Bansal, Research Scholar, Chandigarh University, India

Submitted: 13 May 2022; **Accepted:** 25 May 2022; **Published:** 31 May 2022

Citation: *Vipin Bansal, Amit Jain. (2022). Interpreting the Predictions of Deep Network Build to Identify Early Detection of COVID-19 in X-Ray Images. J App Mat Sci & Engg Res, 6(1), 05-11.*

Abstract

AI is proven technology which is currently serving many different industries. Weather forecasting, recommendation system, autonomous car is few of the examples where AI driven solutions are successfully used. Availability of intensive computing makes it possible to design and develop highly complicated deep learning architecture which is desire to reach human level of accuracy. Because of this reason it become possible to utilize AI technology in healthcare industry where accuracy is utmost important.

Healthcare industry generates various types of Electronic Health Records (EHR) like patient medical history, hospital administration data, biological data, radiological data etc. These type of EHR data can have huge potential in diagnosis various diseases and potentially avoiding any critical risk. AI is also contributing significantly on drug discovery, understanding genetic disorder, cancer detection and many more. All such complex use case needs a complex AI-Deep Neural Network (DNN). Due to its complex architecture these DNN models considered as black box and it became difficult to explain the outcome of such models. Entrust on such solutions considered as a major concern area. Various techniques have been evolved that try to explain the reasoning behind the outcome of such DNN. On this paper two such explanation techniques LIME and LRP is used on explain the prediction made on custom build CNN model. Custom CNN model is trained on Covid-19 patients X-RAY images. The main objective of this paper is to present the explanation difference made by LIME and LRP. Later domain experts can analyze the model predictions and facilities in improving the explanation techniques.

Keywords: Explainable Artificial Intelligence, Trusted AI, LIME, LRP, CNN.

Introduction

Many industries are successfully using various AI-ML solutions. AI ML solutions can analyze past data and predict the sales of the company and companies can control their manufacturing. Weather predictions helps farmer in better irrigation. Erroneous prediction in such cases conceivably harms the monetarily. But in case of Risk-sensitive system like health care, security, surveillance etc. a single wrong prediction can cost someone a huge and that's the reason it's difficult to trust a automate solution even with 99% of accuracy.

It raises an open question, how to build a trustable AI solution? How to earn the confidence of users who are using AI based solution? These are some of the questions, which can be answered by Explainable AI (XAI). XAI refers to an idea where various tools and techniques used to provide the explanation behind the predictions made by ML model. Such explanations also facilitate in building robust system and promote the research in right direction. This will generate more confidence in its user [1,2]. Even as per the EU General Data Protection Regulation (GDPR) clause, explanation is mandatory for any decision made by automated system [3]. Intention of writing this paper is to present two AI Explanation techniques Local Interpretable Model-Agnostic Explanation (LIME) and Layer-wise Relevance Propagation (LRP) which works on different principle and try to present the insight of the predictions by augmenting the Heatmap generated by these explanability tools. Heatmap represents the key pixels which probably used by the model in its prediction. Here we have used custom build Convolutional Neural Network (CNN) based deep learning ML model for expandability. This ML model trained on X-Ray images of COVID-19 infected patients. Domain specialist can analyze the outcome of two different explainable AI tools and facilitate the development of XAI tools in right direction.

Motivation

CNN has been proven very effective in terms of identifying features from the images. Many CNN architectures like YOLO, VGG etc [4-6]. have been successfully used for object classification and object identification tasks. Lots of medical applications

already using these AI-ML architectures for extracting features from the different type of medical dataset like cough audio data, X- ray/CT-scan images and helping in the diagnosis of various ailment. "CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning" have used chest X-ray images for Pneumonia detection and they have used CNN architecture [7]. Another paper "ChestX-Ray8: Hospital-Scale Chest X-Ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases" used X-Ray images for its study on Thorax disease [8].

Lack of required COVID-19 testing kit offers an opportunity to explore AI technology for health domain and give us an opportunity to serve the society. Proposed arrangement can be utilized for early detection of COVID-19 patients. Rustic zones where health care services are extremely negligible can be potentially an early COVID-19 detection kit. Integration of such arrangement with mobile app makes it reachable to remote area too. Augmenting the AI Explainability tool with the X-Ray images can facilitate the doctors in review their decision as well.

Dataset Description

This section incorporates the detailed description about the dataset used for training, data pre- processing and the different deep learning architectures explored.

Data Collection

Severe Acute Respiratory Syndrome Corona virus-2 (SARS CoV 2) is a new infectious virus in Corona family which spreads the COVID-19 disease and the first patient traced in China on late 2019 [9]. This paper has used COVID-19 patient X-ray images available on GitHub [10]. It's a collection of 158 X-Ray pictures (at the hour of composing this paper), conveyed as referenced in Fig. 1.

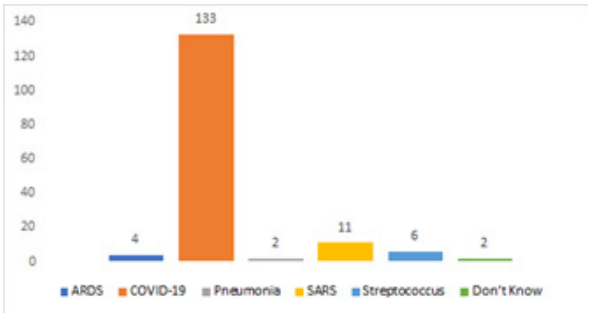

Figure 1: X-ray Image distribution.

Proposed CNN architecture needs front view chest X-ray images and that's the reason some of the images of this data source isn't suitable for model training. As a data clean-up activity, we dropped side view X-ray images and CT-scan images. All such images can cause a negative impact on model training. X-ray images further reduced to 123. Out of them 99 are of COVID-19 patients.

Severe Acute Respiratory Syndrome (SARS) virus also belongs to the same coronavirus family. Patient infected with this virus can show flu-like symptoms and gradually to pneumonia. Symptoms of acute COVID-19 is quite similar with pneumonia [11,12]. That's the reason, ideally pneumonia X-ray images can be used as a COVID-19 sample. But, considering the concept of Siamese Network and Triplet loss function which says always keep similar type of negative glass images on training data for better fine tuning of the model weights, that's the reason we kept them into the Non-Covid-19 class [13].

For Non-COVID-19 class, 500 pneumonia images have been used from the different sources and that makes a total count as 600 images [14]. Whole data collection and preparation pipeline has been explained in Fig. 2.

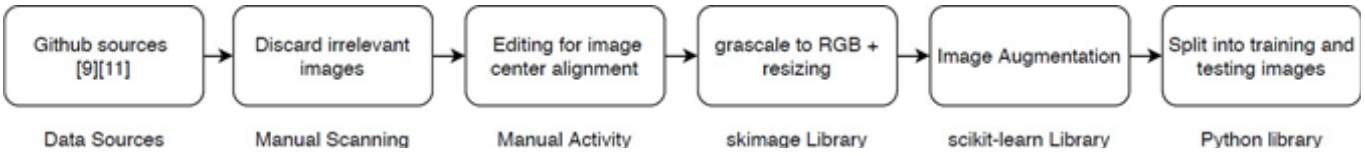

Figure 2: Data preparation and pre-processing

Data Preprocessing

Common issues observed with the image dataset are:

1) Alignment issue
2) Consistency issue in dimension
3) Channel variability

Images are manually visualized and filtered out those images which are not aligned properly. Further, such images have been rectified to ensure that all images should be centrally aligned and in RGB format. Also, image dimensions (224*224*3) set before feeding to the ML model. Selected dataset is skewed as it has 100:500 of COVID-19 Vs Non-COVID-19 classes respectively. Skewed data sometimes leads to over fit the model towards the majority class. Used different image augmented techniques to increase the count of COVID-19 images to 500 [15]. Now total image count is 1000 divided equally between the two different classes.

Finally, for training 80% of the images being used and rest 20% used for testing. These 20% images are kept separately before performing the image augmentation; otherwise, it might be possible that testing data may have some of the images in similar context of training images.

Deep Learning Architectures Used

Broadly explored two different architectures for model training:

1) Customized multilayer CNN architecture
2) Customized with VGG19 pretrained architecture

Customized Multilayer CNN Architecture

Customized multilayer CNN architecture explained in Fig. 3. Initially, we started with a very few numbers of CNN and fully connected layers. Also considered a small number of Kernels and Neurons for basics model training. But as expected, results were not appropriate. Generated ML model was underfitted [16].

And if we add more layers, increase Kernels and Neurons it was getting overfitted [16]. With Relu activation function, we also faced Dying-Relu problem [17].

To overcome all such issues, we defined a mid-size complex architecture. Also introduced Dropout, L2-regularization, and Early Stopping techniques to handle over fitting issue [18, 19]. Leaky-Relu as an activation function used in proposed architecture [20]. After trying multiple combinations of parameters, we come up with the following values of hyper parameters: base Dropout value is 10% which is increased to 30% for intermediate layers, L2 Regularization with 0.001 as a Learning-rate. SAME Padding and Stride value as [1*1] used across the whole architecture.

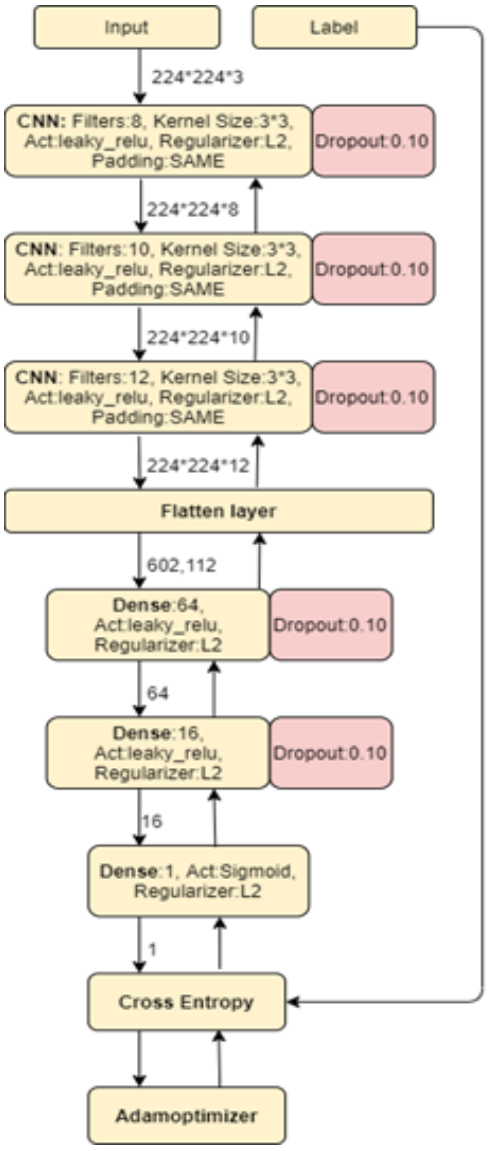

Figure 3: Customized CNN Architecture.

Customized with VGG19 [6] Pertained Architecture

In this architecture, we explored the technique of transfer learning and used pertained VGG19 architecture as described in Fig. 4. For better fitment with our requirement further customization has been done. Outcome of VGG19's last fully connected layer "vgg19/drop7/dropout/Mul1:0" connected with the newly defined fully connected layer of 64 neurons followed by the Logit layer which is having 1 neuron. Incremental training performed on the pretrained weights of VGG19 model which are downloaded from GitHub [21].

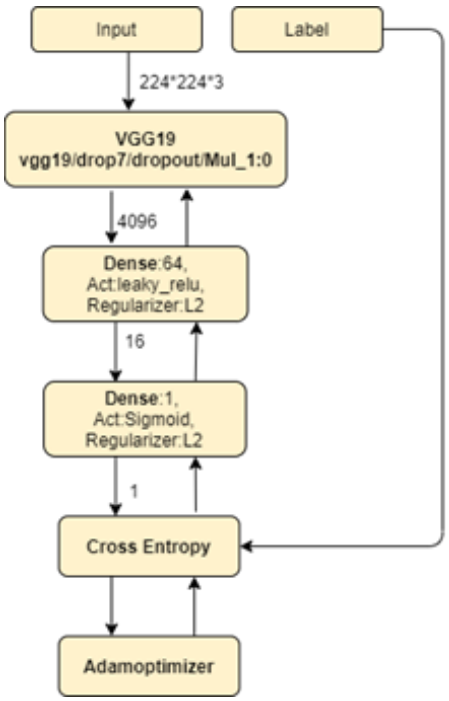

Figure 4: VGG19 with FC layer Architecture.

For health care use cases predicting any False Negative can be very risky and that's the reason higher Sensitivity is required for such solutions. "How Data Scientist can convince a doctor with his work" explaining the importance of sensitivity in healthcare [23]. Considering higher Sensitivity as an evaluation criterion of the model, we have considered option 1 and 4 as mentioned in Figure 5 & 6 for further exploration.

VGG19 based customized models observed to be over fitted, possibly because of its architecture complexity and lack of enough training data whereas Customized CNN architecture with original images having test results closed to training results. Also, we can see better training and testing loss consistency in customized architecture as presented in first graph of Figure 7. That's the reason AI Custom Trained CNN Architecture with original images has been considered further for AI Explanation.

		Accuracy	Sensivity	Precision	F1-Score
Customized CNN Architecture	1. Original Image Dataset	96.80%	100.00%	84.61%	91.66%
	2. With Augmented Image Dataset	97.87%	97.50%	98.23%	97.86%
Customized Model with Pretrained VGG19 Weights	3. Original Image Dataset	96.40%	95.45%	85.71%	90.32%
	4. With Augmented Image Dataset	98.50%	100.00%	97.08%	98.52%

Figure 5: Model results on Training data.

		Accuracy	Sensivity	Precision	F1-Score
Customized CNN Architecture	1. Original Image Dataset	96.96%	100.00%	78.57%	88.00%
	2. With Augmented Image Dataset	93.33%	80.95%	80.95%	80.95%
Customized Model with Pretrained VGG19 Weights	3. Original Image Dataset	95.95%	90.90%	76.92%	83.33%
	4. With Augmented Image Dataset	93.33%	85.71%	78.26%	81.81%

Figure 6: Model results on testing data.

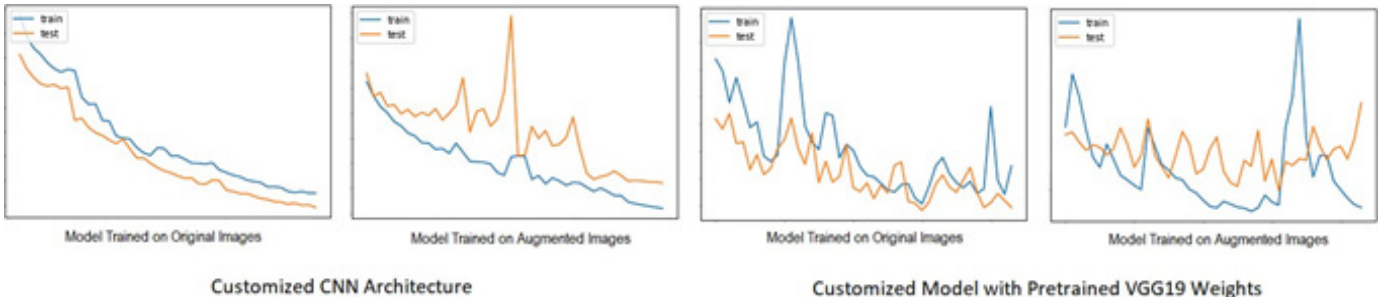

Figure 7: Loss graph of different trained models.

In computer vision, Saliency map is one of the frequently used techniques to visualize the importance of each pixel. Next section explains the integration of two different classes of XAI tools which provides its explanations as a heat map i.e., by highlighting the integral part of the image which is used for influencing the model prediction. We have used post-hoc approaches for ML model explain ability. Post-hoc approaches works on pretrained model. Two classes explained next.

Data Perturbation Based XAI Tool

Local Interpretable Model-Agnostic Explanation (LIME), SHapley Additive exPlanations (SHAP) etc [24, 25]. are some of the frameworks which works on the principle of data perturbation. From this category LIME is used for the integration. LIME is model agnostic tool that can be used for any type of ML architecture. "It's a local explanation tool which internally generates a perturbed data. This perturbed data are the instances which are surrounding the instance of interest. In case of image data, it uses Super-pixel's of an image for the data perturbation [26]. LIME generates collection of perturbed data by randomly setting OFF and ON of such super pixel. OFF means super pixel is absent and not considered whereas ON means super pixel considered for this sample. Such perturbed data are labeled using the trained black box model. Once it's done a new surrogate model gets trained on this perturbed data, where each Super-pixel value considered as a feature for this surrogate model. This surrogate model uses self-explanatory ML algorithms such as Logistic Regression, Decision Tree etc. Weights associated with each Super-pixel (feature) represent its importance and later they are masked with the actual image for presenting a heat map.

Causality based XAI Tools

Layer-wise Relevance Propagation (LRP), Integrated Gradients, Saliency Maps, Deep LIFT etc [27-30]. are some of the tools which works on the principal of causality. From this class, LRP is used for the integration. LRP is a model dependent tool and works for Deep Networks. It's a back propagation-based approach where the features of each layer back propagated till it reaches to input, and it starts from the output layer of the architecture. Algorithm proceeds backward in a layer-by-layer fashion by distributing the relevance score of each neuron and this process iterated till it reaches to the input layer. At the input layer all the relevance weights are sum up to represent the final relevance of each feature (pixels in image) during its prediction. This can be presented as a heat map where accumulated weights of pixel represent its relevance.

Integration of Tools and Results

GitHub project is used For Lime integration whereas for LRP we have used Deep Explain tool [24, 31].

A. When model predicts correctly

1) COVID Class refer Fig. 8:

Figure 8: Model predicts COVID classes correctly.

2) Non-COVID Class refer Fig. 9:

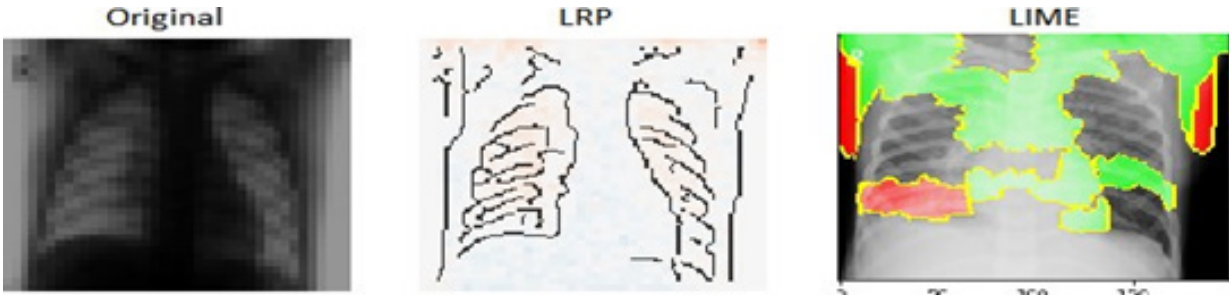

Figure 9: Model predicts Non-COVID classes correctly.

When Model Predicts In-Correctly

1) COVID Class refer Fig. 10:

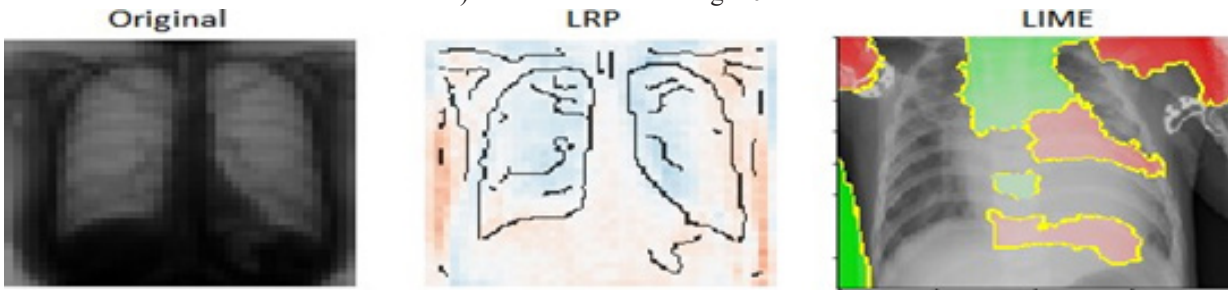

Figure 10: Model predicts COVID classes in-correctly.

2) Non-COVID Class refers Fig. 11:

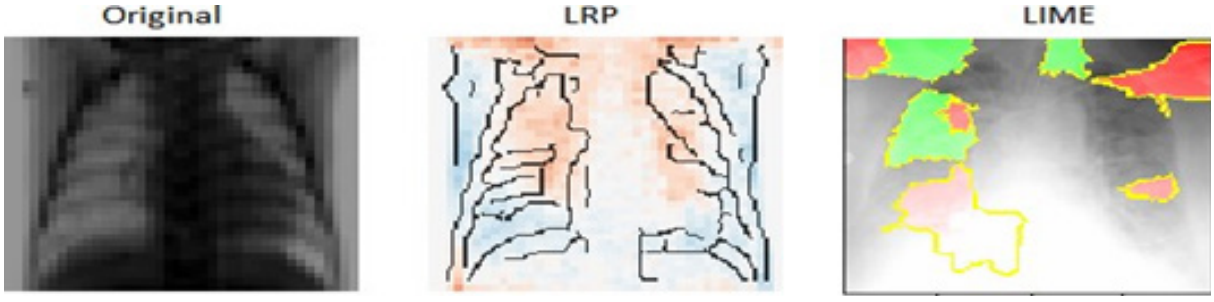

Figure 11: Model predicts Non-COVID classes in-correctly.

In all the cases, order of the images is, original Image followed by heat map of LRP and LIME respectively. LRP distributed the whole image in RED and BLUE color pixels. Every pixel turned out to be a feature and have certain weights. RED represents the positive weightage whereas BLUE represents negative weightage for the prediction. RED color pixel contributed positively to prediction whereas BLUE color pixel negatively contributed. For the presentation perspective results of LRP masked with original image.

LIME gives its super-pixel outcome in GREEN and RED color. GREEN region represents the image section, which is mostly contributed towards that prediction, whereas RED represents the area which is mostly contributed against the predicted class.

LIME trained a surrogate model on randomly generated perturbed data, that's the reason LIME can generate a different explanation even if the same sample is used again for the explanation. This effect has been minimized by taking enough size of perturbed samples and we have used 5000 entries for surrogate model training so that it always produces approximately the same explanation.

Conclusion

This work includes the AI Explainability of Risk- sensitive use case and presented heat-map as a visualization tool for representing most informative area of the X-ray images. We explored two different post-hoc approaches i.e., Perturbation and Causality base and generates an explanation by presenting a different heap-map. Using heatmap we tried to present a localized area of the X-ray images that influenced the ML model on its prediction. This localized area can be the probable area of COVID-19 infection. Integration of such type of AI-ML solutions with the mobile app can penetrate a huge population and can be easily reached to remote areas.

ML models predictions with visualized explanation can be used by healthcare professional in their clinical practices. Also, with the constant feedback loop these tools can be improved further and can help in producing more robust solution in future.

This is not a complete tool that can substitute the infrastructure problem. This can be a part of the solution that can be utilized with other tools or some other ML solutions as an Ensemble approach for improved and correct analysis.

For better efficacy, ML model needs to be re-trained frequently with the latest and updated data. Another approach based upon Siamese network along with the Triplet Loss function can be evaluated to differentiate COVID Vs Non- COVID X-rays when they have very fine similarity.

References

1. Holzinger, A., Biemann, C., Pattichis, C. S., & Kell, D. B. (2017). What do we need to build explainable AI systems for the medical domain?. arXiv preprint arXiv:1712.09923.
2. Holzinger, A., Langs, G., Denk, H., Zatloukal, K., & Müller, H. (2019). Causability and explainability of artificial intelligence in medicine. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 9(4), e1312.
3. Wikipedia .(2020). General Data Protection Regulation. 2020.
4. Khan, A., Sohail, A., Zahoora, U., & Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. Artificial intelligence review, 53(8), 5455-5516.
5. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).
6. Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
7. Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., ... & Ng, A. Y. (2017). Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. arXiv preprint arXiv:1711.05225.
8. Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., & Summers, R. M. (2017). Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2097-2106).
9. Whoint .(2020). WHO Timeline - COVID-19.
10. Cohen, J. P., Morrison, P., & Dao, L. (2020). COVID-19 image data collection. arXiv preprint arXiv:2003.11597.
11. Healthline .(2020). Coronavirus vs. SARS: How Do They Differ?
12. WebMD .(2020). Coronavirus and Pneumonia.
13. Hermans, A., Beyer, L., & Leibe, B. (2017). In defense of the triplet loss for person re-identification. arXiv preprint arXiv:1703.07737.
14. P. Mooney .(2018). Chest X-Ray Images (Pneumonia). Kaggle Datasets.
15. Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. Journal of big data, 6(1), 1-48.
16. Zhang, H., Zhang, L., & Jiang, Y. (2019, October). Overfitting and underfitting analysis for deep learning based end-to-end communication systems. In 2019 11th International Conference on Wireless Communications and Signal Processing (WCSP) (pp. 1-6). IEEE.
17. Lu, L., Shin, Y., Su, Y., & Karniadakis, G. E. (2019). Dying relu and initialization: Theory and numerical examples. arXiv preprint arXiv:1903.06733.
18. Jabbar, H., & Khan, R. Z. (2015). Methods to avoid over-fitting and under-fitting in supervised machine learning (comparative study). Computer Science, Communication and Instrumentation Devices, 70.
19. Ying, X. (2019, February). An overview of overfitting and its solutions. In Journal of Physics: Conference Series (Vol. 1168, No. 2, p. 022022). IOP Publishing.
20. Xu, B., Wang, N., Chen, T., & Li, M. (2015). Empirical evaluation of rectified activations in convolutional network. arXiv preprint arXiv:1505.00853.
21. GitHub .(2020). taehoonlee/tensornets
22. Zhu, W., Zeng, N., & Wang, N. (2010). Sensitivity, specificity, accuracy, associated confidence interval and ROC analysis with practical SAS implementations. NESUG proceedings: health care and life sciences, Baltimore, Maryland, 19, 67.
23. H. Harvey .(2017). How data scientists can convince doctors that AI works", Medium.
24. Ribeiro, M. T., Singh, S., & Guestrin, C. (2016, August). " Why should i trust you?" Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1135-1144).
25. Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. Advances in neural information processing systems, 30.
26. D. Jain .(2019). Superpixels and SLIC", Medium.
27. Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller,

K. R., & Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one, 10(7), e0130140.
28. Sundararajan, M., Taly, A., & Yan, Q. (2017, July). Axiomatic attribution for deep networks. In International conference on machine learning (pp. 3319-3328). PMLR.
29. Simonyan, K., Vedaldi, A., & Zisserman, A. (2014). Deep inside convolutional networks: Visualising image classification models and saliency maps. In In Workshop at International Conference on Learning Representations.
30. Shrikumar, A., Greenside, P., & Kundaje, A. (2017, July). Learning important features through propagating activation differences. In International conference on machine learning (pp. 3145-3153). PMLR.
31. Ancona, M., Ceolini, E., Öztireli, C., & Gross, M. (2017). Towards better understanding of gradient-based attribution methods for deep neural networks. arXiv preprint arXiv:1711.06104.