DeepCOVIDExplainer: Explainable COVID-19 Predictions Based on Chest X-ray Images

Md. Rezaul Karim
Fraunhofer FIT, Aachen, Germany
RWTH Aachen University, Germany

Till Döhmen
Fraunhofer FIT, Aachen, Germany
RWTH Aachen University, Germany

Dietrich Rebholz-Schuhmann
German National Library of Medicine,
University of Cologne, Germany

Stefan Decker
Fraunhofer FIT, Aachen, Germany
RWTH Aachen University, Germany

Michael Cochez
Dept. of Computer Science, Vrije
Universiteit Amsterdam, Netherlands

Oya Beyan
RWTH Aachen University, Germany
Fraunhofer FIT, Aachen, Germany

ABSTRACT

Amid the coronavirus disease (COVID-19) pandemic, humanity experiences a rapid increase in infection numbers across the world. A challenge hospitals are faced with, in the fight against the virus, is the effective screening of incoming patients. One methodology is the assessment of chest radiography (CXR) images, which usually requires expert radiologists' knowledge. In this paper, we propose an explainable deep neural networks (DNN)-based method for automatic detection of COVID-19 symptoms from CXR images, which we call *'DeepCOVIDExplainer'*. We used 16,995 CXR images across 13,808 patients, covering normal, pneumonia, and COVID-19 cases. CXR images are first comprehensively preprocessed, before being augmented and classified with a neural ensemble method, followed by highlighting class-discriminating regions using gradient-guided class activation maps (Grad-CAM++) and layer-wise relevance propagation (LRP). Further, we provide human-interpretable explanations of the predictions. Evaluation results based on hold-out data show that our approach can identify COVID-19 confidently[1] with a positive predictive value (PPV) of 89.61% and recall of 83%, improving over recent comparable approaches. We hope that our findings will be a useful contribution to the fight against COVID-19 and, in more general, towards an increasing acceptance and adoption of AI-assisted applications in the clinical practice.

KEYWORDS

COVID-19, Biomedical imaging, Deep neural networks, Explainability, Grad-CAM++, Layer-wise relevance propagation.

1 INTRODUCTION

The ongoing coronavirus pandemic has had an devastating impact on the health and well-being of the global population already [10, 32]. As of April 10, 2020, more than 1.6 million infections of COVID-19 and 97,000 fatalities due to the disease were reported[2]. Recent studies show that COVID-19, caused by severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) [5], often, but by no means exclusively, affects elderly persons with pre-existing medical conditions [2, 8, 12, 26, 34][3]. While hospitals are struggling with scaling up capacities to meet the rising number of patients, it is important to make use of the screening methods at hand to identify COVID-19 cases and discriminate them from other conditions [32].

The definitive test for COVID-19 is the reverse transcriptase-polymerase chain reaction (RT-PCR) test [5], which has to be performed in specialized laboratories and is a labour-intensive process. COVID-19 patients, however, show several unique clinical and para-clinical features, e.g., presenting abnormalities in medical chest imaging with commonly bilateral involvement. The features were shown to be observable on chest X-ray (CXR) and CT images [12], but are only moderately characteristic to the human eye [26] and not easy to distinguish from pneumonia features.

AI-based techniques have been utilized in numerous scenarios, including automated diagnoses and treatment in clinical settings[18]. Deep neural networks (DNNs) have recently been employed for the diagnosis of COVID-19 from medical images, leading to promising results [12, 26, 32]. However, many current approaches are "black box" methods without providing insights into the decisive image features. Let's imagine a situation where resources are scarce, e.g., a hospital runs out of confirmatory tests or necessary radiologists are occupied, where an AI-assisted tools could potentially help less specialized general practitioners to triage patients, by highlighting critical chest regions can lead to the automated diagnosis decision [32]. A fully automated method without the possibility for human verification would, however, at current state-of-the-art, be unconscionable and potentially dangerous in a practical setting. As a first step towards an AI-based clinical assistance tool for COVID-19 diagnosis, we propose *'DeepCOVIDExplainer'*, a novel diagnosis approach based on neural ensemble methods. The pipeline of *'DeepCOVIDExplainer'* starts with histogram equalization enhancement, filtering, and unsharp masking of the original CXR input images, followed by the training of DenseNet's, ResNet's, and VGGNet's in a transfer learning (TL) setting, creating respective model snapshots. Those are incorporated into an ensemble, using Softmax class posterior averaging (SCPA) and prediction maximization (PM) for the best performing models.

Finally, class-discriminating attention maps are generated using gradient-guided class activation maps (Grad-CAM++) and layer-wise relevance propagation (LRP) to provide explanations of the predictions and to identify the critical regions on patients chest. We hope that *'DeepCOVIDExplainer'* will be a useful contribution towards the development and adoption of AI-assisted diagnosis applications in general, and for COVID-19 in particular. To allow for reproduction of results and for derivative works, we will make the source code, documentation and links to used data publicly available. The rest of the paper is structured as follows: Section 2 outlines related works and points out potential limitations. Section 3

---

[1]Findings are not externally (e.g., with radiologists) validated yet.
[2]https://www.worldometers.info/coronavirus/
[3]List of abbreviations can be found in the end of this paper.

describes our proposed approach, before demonstrating experiment results in section 4. Section 5 summarizes the work and provides some outlook before concluding the paper.

2 RELATED WORK

Bullock et al. [4] provides a comprehensive overview of recent application areas of AI against COVID-19, mentioning medical imaging for diagnoses first, which emphasizes the prevalence of the topic. Although PCR tests offer many advantages over CXR and CT [2], shipping the sample of patients is necessary, whereas X-ray or CT machines are readily available even in rather remote areas. In a recent study by K. Lee et al. [34], CXR and CT images from nine COVID-19 infected patients were analyzed by two radiologists to assess the correspondence of abnormal findings on X-rays with those on CT images. The proportion of patients with abnormal initial radiographic findings was 78.3% to 82.4% for SARS and 83.6% for MERS, while being only 33% for COVID-19 cases [34]. Chest CT images, in contrast, showed double lung involvement in eight out of nine patients. In other words, X-ray may not be the best imaging method for detecting COVID-19, judging by the small cohort of nine patients [34]. Another study by Yicheng Fang et al. [8], however, supports those findings and argues in favour of the effectiveness of CT over X-ray. CT should hence cautiously be considered as the primary imaging source for COVID-19 detection in epidemic areas [2]. Nevertheless, the limited patient cohort size leaves room for statistical variability and, in contrast to those findings, a few other studies have reported rather promising results for the diagnosis based on CXR imaging [9, 25, 32].

Narin et al. [25] evaluated different convolutional neural networks (CNN) architectures for the diagnosis of COVID-19 and achieved an accuracy of 98% using a pre-trained ResNet50 model. However, the classification problem is overly simplified by only discriminating between healthy and COVID-19 patients, disregarding the problem of discriminating regular pneumonia conditions from COVID-19 conditions. Wang et al. [32] proposed COVID-Net to detect distinctive abnormalities in CXR images of COVID-19 patients among samples of patients with non-COVID-19 viral infections, bacterial infections, and healthy patients. On a test sample, containing 10 positive COVID-19 cases among approx. 600 other cases, COVID-Net achieve a PPV of 88.9% and sensitivity of 80%. The small sample size does not enable generalizable statements about the reliability of the method, yet. The highlighted regions using 'GSInquire' are also not well-localized to critical areas. Overall, training on imbalance data, lack of thorough image preprocessing, and poor decision visualization have hindered this approach.

Biraja G., et al. [9] employed uncertainty estimation and interpretability based on Bayesian approach to CXR-based COVID-19 diagnosis, which shows interesting results. The results may, however, be impaired by a lack of appropriate image preprocessing and the resulting attention maps show rather imprecise areas of interest. To overcome these shortcomings of state-of-the-art approaches, our approach first enriches existing datasets with more COVID-19 samples, followed by a comprehensive preprocessing pipeline for CXR images and data augmentation. The COVID-19 diagnosis of *'DeepCOVIDExplainer'* is based on snapshot neural ensemble

method with a focus on fairness, algorithmic transparency, and explainability, with the following assumptions:

- By maximum (or average) voting from a panel of independent radiologists (i.e., ensemble), we get the final prediction fair and trustworthy than a single radiologist.
- By localizing class-discriminating regions with Grad-CAM++ and LRP, we not only can mitigate the opaqueness of the black-box model by providing more human-interpretable explanations of the predictions [19] but also identify the critical regions on patients chest.

3 MATERIALS AND METHODS

In this section, we discuss our approach in detail, covering network construction and training, followed by the neural ensemble and decision visualizations.

3.1 Preprocessing

Depending on the device type, radiographs almost always have dark edges on the left and right side of the image. Hence, we would argue that preprocessing is necessary to make sure the model not only learns to check if the edges contain black pixels or not but also to improve its generalization. We perform contrast enhancement, edge enhancement, and noise elimination on entire CXR images by employing histogram equalization (HGE), Perona-Malik filter (PMF), and unsharp masking edge enhancement. Since images with distinctly darker or brighter regions impact the classification [27], we perform the global contrast enhancement of CXR images using HGE. By merging gray-levels with low frequencies into one, stretching high frequent intensities over high range of gray levels, HGE achieves close to equally distributed intensities [1], where the probability density function $p(X_k)$ of an image X is defined as [1]:

$$p(X_k) = \frac{n_k}{N}, \tag{1}$$

where k is the grey-level ID of an input image X varying from 0 to L, n_k is the frequency of a grey level X_k appearing in X, and N is the total number of samples from an input image. A plot of n_k vs. X_k is specified as the histogram of X, while the equalization transform function $f(X_k)$ is tightly related to cumulative density function $c(X_k)$ [1]:

$$f(X_k) = X_0 + (X_L - X_0)c(X_k) \tag{2}$$

$$c(X_k) = \sum_{j=0}^{k} p(X_j). \tag{3}$$

Output of HGE, $Y = Y(i, j)$ is finally synthesized as follows [1]:

$$Y = f(X) = \{f(X(i, j)) | \forall X(i, j) \in X\}. \tag{4}$$

Image filters 'edge enhances' and 'sharpen' are adopted with the convolution matrices as kernel $g(.)$. PMF is used to preserve the edges and detailed structures along with noise reduction as long as the fitting diffusion coefficient $c(.)$ and gradient threshold K are separate [16]. As a non-linear anisotropic diffusion model, PMF smoothens noisy images $\theta(x, y)$ w.r.t. partial derivative as [16]:

$$\frac{\partial u}{\partial t} = div(c(|\nabla u(x, y, t)|)\nabla u(x, y, t)), \tag{5}$$

where $u(x, y, 0)$ is the original image, $\theta(x, y)$, $u(x, y, t)$ is a filtered image after t iteration diffusion; div and ∇ are divergence and gradient operators w.r.t spatial variables x and y, where the diffusion coefficient $c(.)$ is computed as [28]:

$$c_1(|\nabla I|) = exp\left(-\left(\frac{|\nabla I|}{K}\right)^2\right) \tag{6}$$

$$c_2(|\nabla I|) = \frac{1}{1 + \left(\frac{|\nabla I|}{K}\right)^2}. \tag{7}$$

To determine whether the local gradient magnitude is strong enough for the edge preservation, diffusion coefficient function $c(.)$ is computed as follows [28]:

$$c_3(|\nabla I|) = \begin{cases} \frac{1}{2}\left[1 - \left(\frac{|\nabla I|}{K\sqrt{2}}\right)^2\right]^2, & |\nabla I| \leq K\sqrt{2} \\ 0, & |\nabla I| > K\sqrt{2} \end{cases} \tag{8}$$

where c_3 is the Tukey's biweight function. Since the boundary between noise and edge is minimal, c_3 is applied as the fitting diffusion coefficient [16]. Further, we attempt to remove textual artefacts from CXR images, e.g., a large number of images annotate right and left sides of chest with a white 'R' and 'L' characters. To do so, we threshold the images first to remove very bright pixels and the missing regions were in-painted. In all other scenarios, image standardization and normalization are performed. For image standardization, the mean pixel value is subtracted from each pixel and divided by the standard deviation of all pixel values. The mean and standard deviation is calculated on the whole datasets and adopted for training, validation and test sets. For image normalization, pixel values are rescaled to a [0,1] by using a pixel-wise multiplication factor of 1/255, giving a set of grey-scale images. Further, CXR images are resized $224 \times 224 \times 3$ before starting the training.

3.2 Network construction and training

We trained VGG, ResNet, and DenseNet architectures and create several snapshots during a single training run with cyclic cosine annealing (CAC) (see fig. 2)[22], followed by combining their predictions to an ensemble prediction [13, 17]. We pick VGG-16 and VGG-19 due to their general suitability for image classification. Based on the dense evaluation concept [30], VGG variants convert the last three fully connected layers (FCLs) to 2D convolution operations to reduce the number of hyperparameters. We keep last 2 layers fixed to adopt a 1×1 kernel, leaving the final one equipped with a Softmax activation. However, owing to the computational complexity of VGG-16 due to consecutive FCLs, the revised VGG-19 is trained with a reduced number of hidden nodes in first 2 FCLs.

Next, we pick ResNet-18 [33]) and ResNet-34 [11]) architectures. Apart from common building blocks, two bottlenecks are present in the form of channel reduction in ResNets. A series of convolution operators without pooling is placed in between, forming a stack. The first conv layer of each stack in ResNets (except for the first stack) are down-sampled at stride 2, which provokes the channel

difference between identity and residual mappings. ResNets are lightweight stack-based CNNs, with their simplicity arising from small filter sizes (i.e., 3×3) [30]. A series of convolution operators without pooling is then placed in between and recognized as a stack, as shown in fig. 1. However, w.r.t regularisation, a 7×7 conv filter is decomposed into a stack of three 3×3 filters with non-linearity injected in between [30]. Lastly, DenseNet-161 and DenseNet-201 architectures are picked. While ResNets merge feature-maps through summation, DenseNets concatenate additional inputs from preceding layers, which not only strengthens feature propagation and moderates information loss but also increases feature reusing capability by cutting down numbers of parameters [14].

To avoid possible overfitting, L_2 weight regularization, dropout, and data augmentation (by rotating the training CXR images by up to $15°$) were employed. We did not initialize networks weights with any pretrained (e.g., ImageNet) models. The reason is that ImageNet contains photos of general objects, which would activate the internal representation of network's hidden layers with geometrical forms, colorful patterns, or irrelevant shapes that are usually not present CXR images.

We set the number of epochs (NE), maximum learning rate (LR), number of cycles, and current epoch number, where initial LR and NE are two hyperparameters. CAC starts with a large LR and rapidly decreases to a minimum value before it dramatically increases to the following LR for that epoch [13, 17]. During each model training, CAC changes the LR aggressively but systematically over epochs to produce different network weights [13]:

$$\alpha(t) = \frac{\alpha_0}{2}\left(\cos\left(\frac{\pi \bmod (t-1, \lceil T/C \rceil)}{\lceil T/C \rceil}\right) + 1\right), \tag{9}$$

where $\alpha(t)$ is the LR at epoch t, α_0 is the maximum LR, T is the total epoch, C is the number of cycles and mod is the modulo operation. After training a network for C cycles, best weights at the bottom of each cycle are saved as a model snapshot (m), giving M model snapshots, where $m \leq M$.

3.3 Model ensemble

Especially when a single practitioner makes a COVID-19 diagnosis, the chance of a false diagnosis is given. In case of doubt, a radiologist should, therefore, ask for a second or third option of other experts. Analog to this principle, we employ the principle of model ensembles, which combine the 'expertise' of different predictions algorithms into a consolidated prediction and hereby reducing the generalization error [13]. Research has shown that a neural ensemble method by combining several deep architectures is more effective than structures solely based on a single model [13, 17].

Inspired from [13, 31], we apply both ASCP and the PM of best-performing models from the list of snapshot models, ensemble their predictions, and propagate them through the Softmax layer, where the class probability of the ground truth j for a given image x is inferred as follows [31]:

$$P(y = j|\mathbf{x}) = \frac{\exp\left[\sum_{m=1}^{M} \hat{P}_m(y = j|\mathbf{x})\right]}{\exp\left[\sum_{k=1}^{K} \sum_{m=1}^{M} \hat{P}_m(y = k|\mathbf{x})\right]}, \tag{10}$$

where m is the last snapshot model, M is the number of models, K is the number classes, and $\hat{P}_m(y = j|\mathbf{x})$ is the probability distribution.
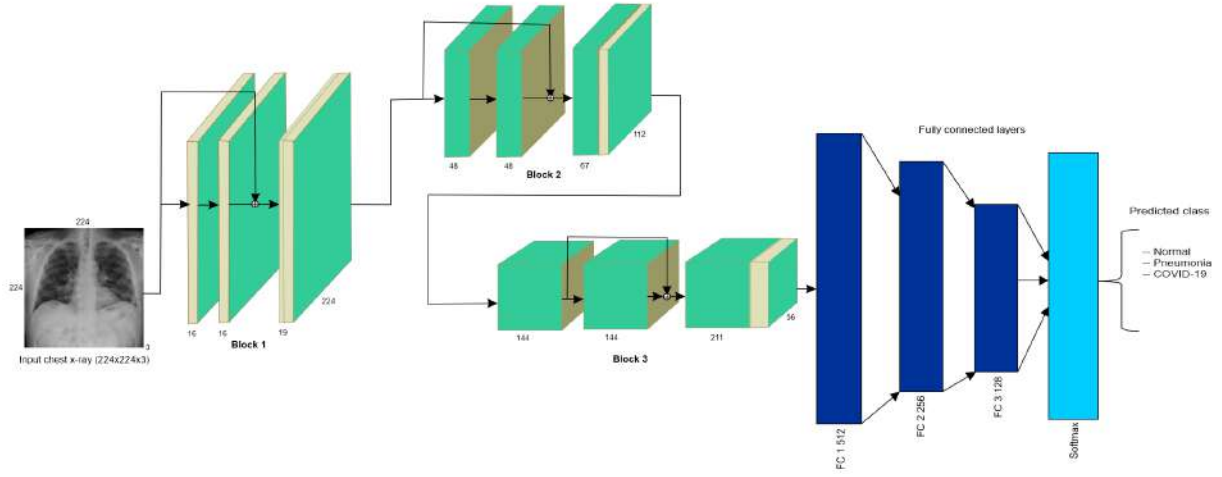
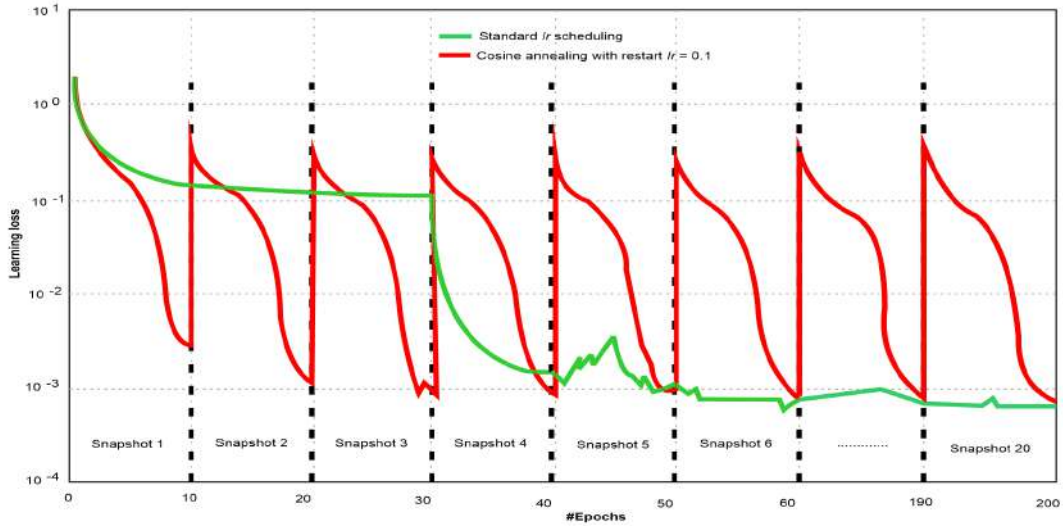**Figure 1: The classification with ResNet-based networks**

Figure 2: Training loss of VGG-19 network with standard learning rate (green) and cosine annealing cycles (red), the intermediate models, denoted by the dotted lines form an ensemble at the end of training

3.4 Decision visualizations

To improve the COVID-19 detection transparency, class-discriminating regions on the subjects chest are generated by employing Grad-CAM [29], Grad-CAM++ [6], and LRP [15]. The idea is to explain where the model provides more attention for the classification. CAM computes the number of weights of each feature map (FM) based on the final conv layer to calculate the contribution to prediction y_c at location (i, j), where the goal is to obtain L^c_{ij} that satisfies $y^c = \sum_{i,j} L^c_{ij}$. The last FM A_{ijk} and the prediction y_c are represented in a linear relationship in which linear layers consist of global average pooling (GAP) and FCLs: i) GAP outputs $F_k = \sum_{i,j} A_{ijk}$, ii) FCL that holds weight w^c_k, generates the following output [21]:

$$y^c = \sum_k w^c_k F_k = \sum_k w^c_k \sum_{i,j} A_{ijk} = \sum_{i,j} \sum_k w^c_k A_{ijk}, \qquad (11)$$

where $L^c_{ij} = \sum_k w^c_k A_{ijk}$ [21]. Due to the vanishing of non-linearity of classifiers, CAM is an unsuitable method. Hence, we employ Grad-CAM to globally average the gradients of FM as weights instead of pooling. While heat maps (HM) are plotted, class-specific weights are collected from the final conv layer through globally averaged gradients (GAG) of FM instead of pooling [6]:

$$\alpha^c_k = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A^k_{ij}}, \qquad (12)$$
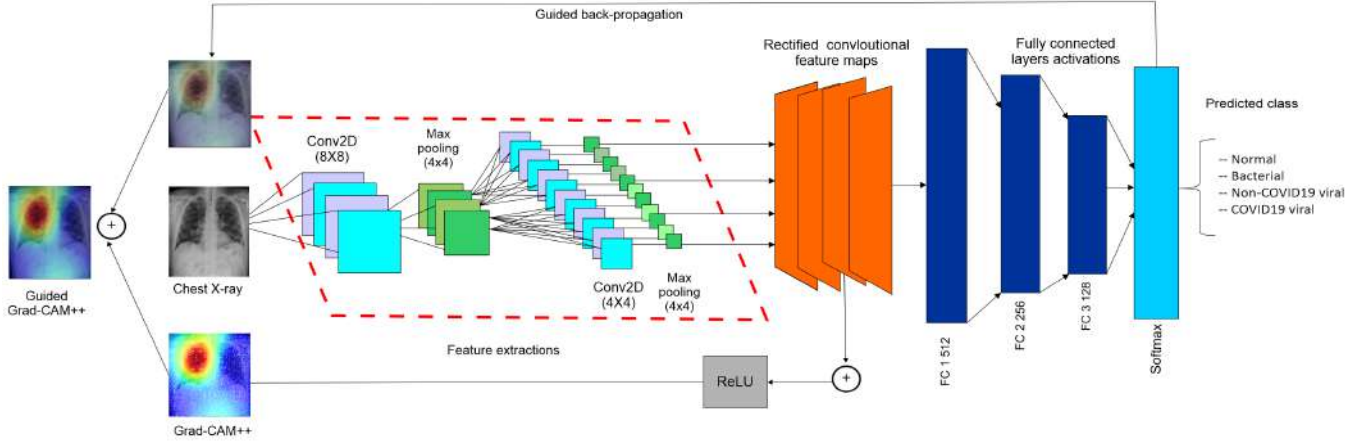
Figure 3: Classification and decision visualization with CNN-based approach

where Z is the number of pixels in an FM, c is the gradient of the class, and A_{ij}^k is the value of kth FM. Having gathered relative weights, the coarse saliency map (SM), L^c is computed as the weighted sum of $\alpha_k^c * A_{ij}^k$ of the ReLU activation. It introduces linear combination to the FM as only the features with a positive influence on the respective class are of interest [6] and the negative pixels that belong to other categories in the image are discarded [29]:

$$L^c = \text{ReLU}(\sum_i \alpha_k^c A^k), \tag{13}$$

Grad-CAM++ (see fig. 3) replaces the GAG with a weighted average of the pixel-wise gradients as the weights of pixels contribute to the final prediction w.r.t the following iterators over the same activation map A^k, (i, j) and (a, b).

$$w_k^c = \sum_i \sum_j \alpha_{ij}^{kc} \cdot \text{ReLU}(\frac{\partial y^c}{\partial A_{ij}^k}) \tag{14}$$

$$y^c = \sum_k w_k^c \cdot \sum_i \sum_j A_{ij}^k \tag{15}$$

$$\alpha_{ij}^{kc} = \frac{\frac{\partial^2 y^c}{(\partial A_{ij}^k)^2}}{2\frac{\partial^2 y^c}{(\partial A_{ij}^k)^2} + \sum_a \sum_b A_{ab}^k \frac{\partial^3 y^c}{\{(\partial A_{ij}^k)^3\}}}. \tag{16}$$

Even though CXR images rarely contain multiple targets, revealing particular image parts that contributed to the prediction, rather than the entire chest area is still helpful. CAM variants back-propagate the gradients all the way up to the inputs, are essentially propagated only till the final conv layer. Besides, CAM methods are limited to specific architectures, where an average-pooling layer connects conv layers with an FCL.

LRP is another robust technique of propagating relevance scores (RSs) and, in contrast to CAM, redistributes proportionally to the activation of previous layers. LRP assumes that the class likelihood can be traced backwards through a network to the individual layer-wise nodes of the input [15]. From a network of L layers, 1, 2, ..., N nodes

in layer l, 1, 2, .., M nodes in layer $l + 1$, the RS, $R_n^{(l)}$ at node n in layer l is recursively defined [15]:

$$R_n^{(l)} = \sum_m \frac{a_n^{(l)} w_{n,m}^{+(l)}}{\sum_{n'} a_{n'}^{(l)} w_{n',m}^{+(l)}} R_m^{(l+1)}. \tag{17}$$

The node-level RS for negative values is calculated with ReLU activation as [15]:

$$R_n^{(l)} = \sum_m \frac{x_n^{(l)} w_{n,m}^{(l)} - b_n^{(l)} w_{n,m}^{+(l)} - h_n^{(l)} w_{n,m}^{-(l)}}{\sum_{n'} x_{n'}^{(l)} w_{n',m}^{(l)} - b_{n'}^{(l)} w_{n',m}^{+(l)} - h_{n'}^{(l)} w_{n',m}^{-(l+1)}}. \tag{18}$$

The output layer RS is finally calculated before being back-propagated as follows [15]:

$$R_n^{(L)} = \begin{cases} z_t^{(L)} & n = t \\ 0 & \text{otherwise} \end{cases} \tag{19}$$

First, an image x is classified in a forward pass, where LRP identifies important pixels. The backward pass is a conservative relevance (i.e., $R_t^{(L)}$) redistribution procedure with back-propagation using deep Taylor decomposition [24], to generate a relevance map R_{lrp}, for which the nodes contributing most to the higher-layer, also receive most relevance. Finally, heat maps for all the test samples are generated based on the trained models, indicating the relevance for the classification decision for each.

4 EXPERIMENT RESULTS

In this section, we discuss the evaluation results both quantitative and qualitatively.

4.1 Experiment setup

Experiments were carried out on a machine having an Intel(R) Xeon(R) E5-2640, 256 GB of RAM, and Ubuntu 16.04 OS. All the

Table 1: Train and test set distribution in COVIDx datasets

COVIDx version	Training	Test
v1.0	5,344	654
v2.0	11,744	5,032
v3.0	11,896	5,099

Table 2: The class distribution of COVIDx datasets

Version	Normal	Bacterial	Non-CoVID19 viral	COVID-19
v1.0	1,583	2,786	1,504	76
Version	**Normal**	**Pneumonia**		**COVID-19**
v2.0	8,066	8,614		190
v3.0	8,066	8,614		259

programs[4] were written in Python, where the software stack consists of scikit-learn and Keras with the TensorFlow backend. The LRP-based visualization and relevance calculation are generated using the iNNvestigate toolbox[5]. Networks were trained on an Nvidia Titan Xp GPU with CUDA and cuDNN enabled to make the overall pipeline faster. When we create snapshots, we set the number of epochs to 200, maximum LR to 1.0, and the number of cycles to 20, giving 20 snapshots for each model. For 6 architectures, we would get 120 snapshot models in total, on which we construct the ensemble. The best snapshot model is used for the decision visualizations, which we choose using WeightWatcher.

To tackle class imbalance, we apply class weighting to penalize a model when it misclassifies a positive sample. Although accuracy is an intuitive evaluation criterion for many bio-imaging problem, e.g., osteoarthritis severity prediction [3], those evaluation criteria are most suitable for balanced class scenarios. Given the imbalanced class scenario, with widely varying class distributions between different classes, we report precision, recall, F1, and positive predictive value (PPV) that are produced through random search and 5-fold cross-validation tests, i.e., for each hyperparameter group of the certain network structure, five repeated experiments are conducted.

4.2 Datasets

We consider 3 different versions of COVIDx datasets. The COVIDx v1.0 dataset had a total of 5,941 CXR images from 2,839 patients based on the COVID-19 image dataset curated by Joseph P., et al. [7] and Kaggle CXR Pneumonia dataset[6] by Paul Mooney. It is used in some early works, e.g., [32]. However, Kaggle CXR images are of children. Therefore, to avoid possible prediction bias (e.g., the model might be prone to predict based on mere chest size), we enriched *'COVIDx v2.0'* (with CXR images of adult subjects from the RSNA Pneumonia Detection Challenge[7], original and augmented versions of COVID-19 examples[8]), which we called *'COVIDx v3.0'* is used in our approach. Additional 59 CXR images are collected from: i) Italian Radiological Case[9], and ii) Radiopaedia.org (provided by Dr. Fabio Macori)[10]. *'COVIDx v3.0'* images are categorized as normal, pneumonia, and COVID19 viral. Table 1 and table 2 show the distributions of class, images, and patients.

---

[4]https://github.com/rezacsedu/DeepCOVIDExplainer
[5]https://github.com/albermax/innvestigate
[6]https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia
[7]https://www.kaggle.com/c/rsna-pneumonia-detection-challenge
[8]https://github.com/ieee8023/covid-chestxray-dataset
[9]https://radiopaedia.org/articles/covid-19-3?lang=us
[10]https://www.sirm.org/category/senza-categoria/covid-19/

4.3 Performance of individual model

Overall results are summarized in table 3. As seen, VGG-19 and DenseNet-161 performed best on both balanced and imbalanced datasets, while VGG-16 turns out to be the lowest performer. In direct comparison, the diagnosis of VGG-19 yields much better results than VGG-16, which might be explainable by the fact that a classifier with more formations requires more fitting of FMs, which again depends on conv layers. The architecture modification of VGG-19 by setting 2 conv layers and the filter size of 16, visibly enhances the performance. ResNet-18 performed better, although it's larger counterpart ResNet-34 shows quite unexpected low performance. Evidently, due to structured residual blocks, the accumulation of layers could not promote FMs extracted from the CXR images.

Both DenseNets architectures show consistent performance owing to clearer image composition. DenseNet-161 outperforms not only DenseNet-201 but also all the other models. In particular, DenseNet-161 achieves precision, recall, and F1 scores of 0.94, 0.95, and 0.945, respectively, on balanced CXR images. On imbalanced image sets, both DenseNet-161 and ResNet-18 perform consistently. Although VGG-19 and ResNet-18 show competitive results on the balanced dataset, the misclassification rate for normal and pneumonia samples are slightly elevated than DenseNet-161, which poses a risk for clinical diagnosis. In contrast, DenseNet-161 is found to be resilient against the imbalanced class scenario. Hence, models like DenseNet-161, which can handle moderately imbalanced class scenarios, seem better suited for the clinical setting, where COVID-19 cases are rare compared to pneumonia or normal cases. The ROC curve of the DenseNet-161 model in fig. 4 shows consistent AUC scores across folds, indicating stable predictions, signifying that the predictions are much better than random.

Nevertheless, bad snapshot models can contaminate the overall predictive powers of the ensemble model. Hence, we employ WeightWatcher [23] in two levels: i) level 1: we choose the top-5 snapshots to generate a full model, ii) level 2: we choose the top-3 models for the final ensemble model. In level 2, WeightWatcher is used to compare top models (by excluding VGG-16, ResNet-34, and DenseNet-201) and choose the ones with the lowest log norm and highest weighted alpha (refer to section 4 in supplementary for details), where a low (weighted/average) log-norm signifies better generalization of network weights [23]. Figure 5 shows choosing the better model between VGG-16 and VGG-19 with WeightWatcher in terms of weighted alpha and log norm.

4.4 Model ensemble

We perform the ensemble on following top-3 models: VGG-19, ResNet-18, and DenseNet-161. To ensure a variation of network architectures within the ensemble, VGG-19 is also included. As presented in table 4, the ensemble based on the SCPA method outperforms the ensemble based on PM methods moderately. The reason is that the PM approach appears to be easily influenced by outliers with high scores. To a great extent, the mean probabilities for each class affect the direction of outliers. For the SCPA-based ensemble, the combination of VGG-19 + DenseNet-161 outperforms other ensemble combination.

The confusion matrix of the best ensemble's performance on balanced data is shown in fig. 6. The results show that a majority

Table 3: Classification results of each model on balanced and imbalanced datasets

Network	Balanced dataset			Imbalanced dataset		
	Precision	Recall	F1	Precision	Recall	F1
VGG-16	0.77	0.75	0.761	0.72	0.75	0.734
ResNet-34	0.87	0.85	0.861	**0.84**	**0.86**	**0.851**
DenseNet-201	0.89	0.91	0.905	0.79	0.76	0.783
ResNet-18	0.91	0.93	0.921	0.86	0.82	0.839
VGG-19	**0.92**	**0.93**	**0.925**	0.85	0.83	0.845
DenseNet-161	**0.94**	**0.95**	**0.945**	**0.88**	**0.86**	**0.875**
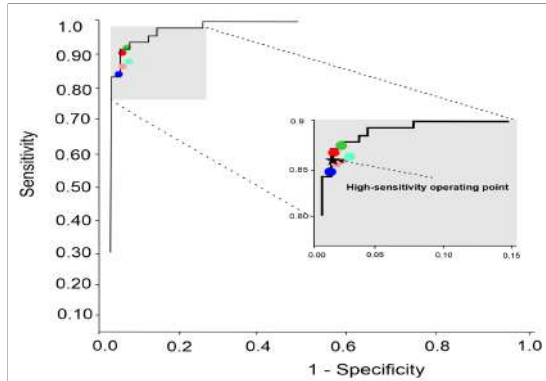
Figure 4: ROC curves of the ensemble model (black lines) for the detection of infection types on test set, color circles mean different folds, showing stable convergence

of samples were classified correctly, with precision, recall, and F1 scores of 0.926, 0.917, and 0.925, respectively, using the PM ensemble method. For the SCPA-based ensemble, precision, recall, and F1 are even slightly higher, yielding 0.931, 0.940, and 0.935, respectively. Additionally, we report the class-specific measures in table 5 to give a better view in both the balanced and imbalanced scenario.

4.5 Quantitative analysis

Since we primarily want to limit the number of missed COVID-19 instances, the achieved recall of 83% is still an acceptable metric, which means that a certain fraction of all patients who test positive, will actually not have the disease. To determine how many of all infected persons would be diagnosed positive by the method, we calculate the positive predictive value (PPV). Out of our test set with 77 COVID-19 patient samples, only six were misclassified as pneumonia and two as normal, which results in a PPV of 89.61% for COVID-19 cases, slightly outperforming a comparable approach [32]. In our case, results are backed up by a larger test set, which contributes to the reliability of our evaluation results. It is to note that the PPV was reported for a low prevalence of COVID-19 in the cohorts. In a setting with high COVID-19 prevalence, the likelihood for false-positives is expected to shrink further in favour of correct COVID-19 predictions.

4.6 COVID-19 predictions and explanations

Precise decisive feature localization is vital not only for the explanation but also for rapid confirmation of the reliability of outcomes,

especially for potentially false-positive cases [6]. Attention map highlighting of critical regions on the chest advocate transparency and trustworthiness to clinicians and help them leverage their screening skills to make faster and yet more accurate diagnoses [32]. In general, the more accurate a model is, the more consistent the visualizations of Grad-CAM and Grad-CAM++ will be. Key features can then easily be identified based on where the activation maps are overlapping. The critical regions of some CXR images of COVID-19 cases are demonstrated in fig. 7, fig. 8, and fig. 9, where class-discriminating areas within the lungs are localized.

As seen, HMs generated by Grad-CAM and Grad-CAM++ are fairly consistent and alike, but those with Grad-CAM++ are more accurately localized. The reason is that instead of certain parts, Grad-CAM++ highlights conjoined features more precisely. On the other hand, although LRP highlights regions much more precisely, it fails to provide attention to critical regions. It turned out that Grad-CAM++ generates the most reliable HM's when compared to Grad-CAM and LRP. To provide more human-interpretable explanations, let's consider the following examples (based on ResNet-18):

- **Example 1**: the CXR image is classified to contain a confirmed COVID-19 case with a probability of 58%, the true class is COVID-19, as shown in fig. 7.
- **Example 2**: the CXR image is classified to contain a confirmed COVID-19 case with a probability of 58%, the true class is COVID-19, as shown in fig. 8.
- **Example 3**: the CXR image is classified to contain COVID-19 case with a classification score of 10.5, true class is COVID-19, as shown in fig. 9.

4.7 Discussion and diagnosis recommendations

Based on the above analyses, *'DeepCOVIDExplainer'* disseminates the following recommendations: firstly, even if a specific approach does not perform well, an ensemble of several models still can outperform individual models. Secondly, since accurate diagnosis is a mandate, models trained on imbalanced training data can provide distorted or wrong predictions during inference time, due to possible overfitting during the training. In this case, even a high accuracy score can be achieved without predicting minor classes, hence might be uninformative.

Thirdly, taking COVID-19 diagnosis context into account, the risk resulting from a pneumonia diagnosis is much lower than for a COVID-19 diagnosis. Hence, it is more reasonable to make a decision based on the maximum score among all single model predictions. Fourthly, due to the nature of neural networks, decision visualizations cannot be provided based on ensemble models, even
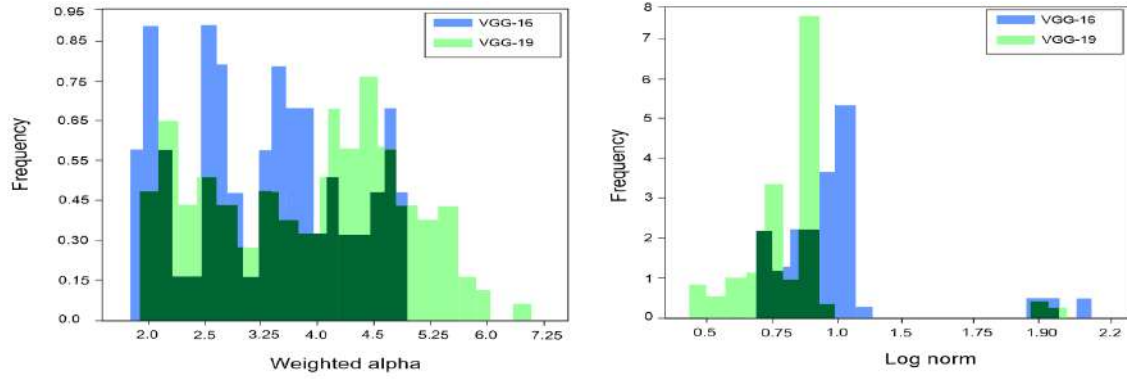
Figure 5: Choosing VGG model with WeightWatcher: a) w.r.t weighted alpha, b) w.r.t log-norm

Table 4: Classification results for ensemble methods on balanced dataset

Architecture combination	Prediction maximization			Softmax posterior averaging		
	Precision	Recall	F1	Precision	Recall	F1
ResNet-18+DenseNet-161	0.905	0.913	0.909	0.925	0.94	0.933
VGG-19+DenseNet-161	**0.926**	**0.917**	**0.925**	**0.931**	**0.94**	**0.935**
VGG-19+ResNet-18	0.895	0.915	0.905	0.915	0.93	0.923
DN-161+VGG-19+ResNet-18	0.915	0.892	0.901	0.924	0.937	0.931

Table 5: Classwise classification results of ensemble model on chest x-rays

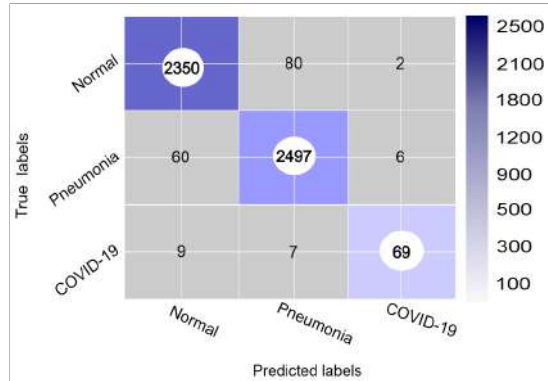| Infection type	Balanced dataset			Imbalanced dataset		
	Precision	Recall	F1	Precision	Recall	F1
Normal	0.93	0.92	0.925	0.89	0.87	0.865
Pneumonia	0.90	0.91	0.905	0.85	0.84	0.845
COVID-19	0.84	0.83	0.835	0.82	0.80	0.816

Figure 6: confusion matrix of the ensemble model

though their usage contributes to decision fairness and reliability. For the decision visualization, therefore, it is recommended to pick the single best model as a basis and to employ Grad-CAM++ for providing the most reliable localization.

5 CONCLUSION AND OUTLOOK

In this paper, we proposed *'DeepCOVIDExplainer'* to leverage explainable COVID-19 prediction based on CXR images. Evaluation results show that our approach can identify COVID-19 with a PPV of 89.61% and recall of 83%, outperforming a recent approach. Further, as Curtis Langlotz[11] stated "AI won't replace radiologists, but radiologists who use AI will replace radiologists who don't". In the same line, we would argue that *'DeepCOVIDExplainer'* is not to replace radiologists but to be evaluated in a clinical setting and is by no means a suitable replacement for a human radiologists. We would even argue that human judgement is indispensable when the life of patients is at stake. However, we hope our findings will be a useful contribution to the fight against COVID-19 and towards an increasing acceptance and adoption of AI-assisted applications in the clinical practice.

Lastly, we want to outline potential areas of enhancement: Firstly, since only a limited amount of CXR images for COVID-19 infection cases were at hand, it would be unfair to claim that we can rule out overfitting for our models. More unseen data from similar distributions is necessary for further evaluation to avoid possible out-of-distribution issues. Secondly, due to external conditions, we were yet not been able to verify the diagnoses and localization
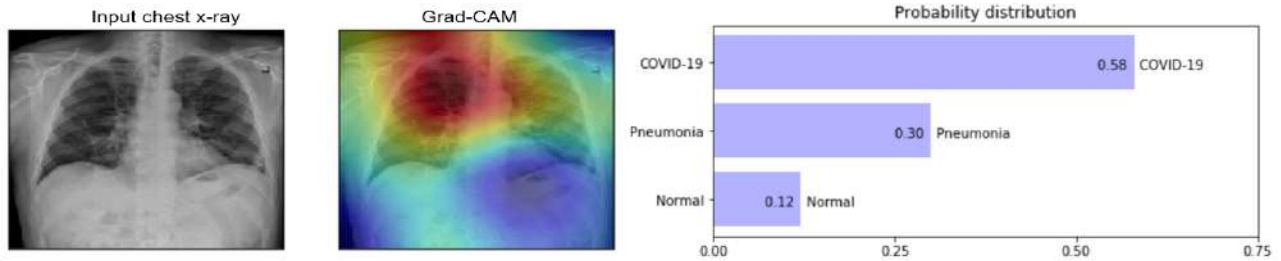
---

[11]https://www.nature.com/articles/d41586-019-03847-z

Figure 7: The input chest x-ray classification, decision visualization with Grad-CAM and explanation
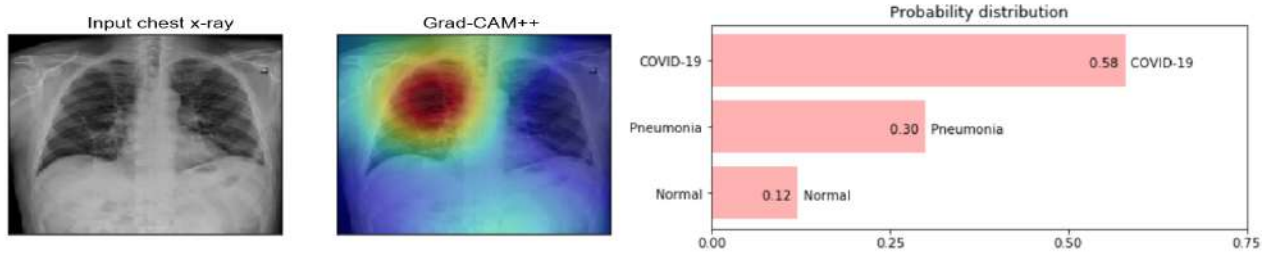

Figure 8: The input chest x-ray classification, decision visualization with Grad-CAM++ and explanation
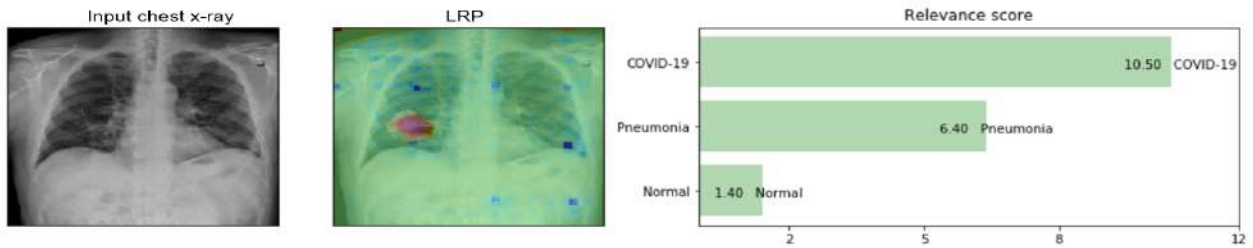

Figure 9: The input chest x-ray classification, decision visualization with LRP and explanation

accuracies with the radiologists. Thirdly, accurate predictions do not only depend on single imaging modalities, but could also build up on additional modalities like CT and other decisive factors such as, e.g., patients demographic and symptomatic assessment report [31]. Nevertheless, we would argue that explaining predictions with plots and charts are useful for exploration and discovery[20]. Explaining them to patients may be tedious and require more human-interpretable decision rules in natural language. In future, we intend to overcome these limitations by: i) alleviating more data (e.g., patient CT, phenotype, and history) and training a multimodal convolutional autoencoder, and ii) incorporating domain knowledge with neuro-symbolic reasoning to generate decision rules to make the diagnosis fairer.

ACKNOWLEDGMENTS

ACRONYMS

Acronyms and their full forms used in this paper are as follows:

AUC Area Under the receiver operating characteristic Curves
CAM Class Activation Maps
CXR Chest X-ray
CNN Convolutional Neural Network
CLRP Contrastive Layer-wise Relevance Propagation
COVID-19 Corona virus disease
CT Computed Tomography
CAC Cyclic Cosine Annealing
DNN Deep Neural Networks
DL Deep Learning
DenseNet Dense Convolutional Network
DTD Deep Taylor Decomposition
FCL Fully-Connected Layer
FCN Fully Convolutional Neural Network
FM Feature Maps
GAG Globally Averaged Gradients
GAP Global Averaged Pooling
Grad-CAM Gradient-guided class-activation maps
HM Heat Maps

HGE Histogram Equalization
ICU Intensive Care Unit
LR Learning Rate
LRP Layer-wise relevance propagation
ML Machine Learning
NE Number of Epochs
PMF Perona-Malik Filter
PM Prediction Maximization
PPV Positive Predictive Value
ROC Receiver Operating Characteristic
ResNet Residual Network
RS Relevance Score
RT-PCR Reverse Transcriptase-polymerase Chain Reaction
SARS-CoV-2 Severe Acute Respiratory Syndrome Coronavirus 2
SM Saliency Maps
SGD Stochastic Gradient Descent
SCPA Softmax Class Posterior Averaging
SGLRP Softmax-gradient LRP
TBF Tukey's biweight function
TL Transfer Learning.

REFERENCES

[1] Tarun Kumar Agarwal, Mayank Tiwari, and Subir Singh Lamba. 2014. Modified histogram based contrast enhancement using homomorphic filtering for medical images. In *2014 IEEE International Advance Computing Conference (IACC)*. IEEE, 964–968.

[2] Tao Ai, Zhenlu Yang, Hongyan Hou, Chenao Zhan, and Liming Xia. 2020. Correlation of chest CT and RT-PCR testing in coronavirus disease 2019 (COVID-19) in China: a report of 1014 cases. *Radiology* (2020), 200642.

[3] Alireza Baratloo, Mostafa Hosseini, and Gehad El Ashal. 2015. Simple Definition and Calculation of Accuracy, Sensitivity and Specificity. *Emergency (Tehran, Iran)* 3, 2 (2015), 48–49.

[4] Joseph Bullock, Katherine Hoffmann Pham, and Miguel Luengo-Oroz. 2020. Mapping the Landscape of Artificial Intelligence Applications against COVID-19. *arXiv:2003.11336* (2020).

[5] Makeda Moore Candace and Daniel. 2020. COVID-19. (2020). https://radiopaedia.org/articles/covid-19-3?lang=us Online; accessed April-July-2020.

[6] Aditya Chattopadhay and Anirban Sarkar. 2018. Grad-CAM++: Generalized gradient-based visual explanations for convolutional networks. In *Applications of Computer Vision(WACV)*. IEEE, 839–847.

[7] Joseph Paul Cohen, Paul Morrison, and Lan Dao. 2020. COVID-19 image data collection. *arXiv 2003.11597* (2020). https://github.com/ieee8023/covid-chestxray-dataset

[8] Yicheng Fang, Huangqi Zhang, Jicheng Xie, Minjie Lin, Lingjun Ying, Peipei Pang, and Wenbin Ji. 2020. Sensitivity of chest CT for COVID-19: comparison to RT-PCR. *Radiology* (2020), 200432.

[9] Biraja Ghoshal and Allan Tucker. 2020. Estimating Uncertainty and Interpretability in Deep Learning for Coronavirus (COVID-19) Detection. *arXiv:2003.10769* (2020).

[10] Ophir Gozes, Maayan Frid-Adar, Hayit Greenspan, and Eliot Siegel. 2020. Rapid AI development cycle for the coronavirus pandemic: Initial results for automated detection and patient monitoring using deep learning CT image analysis. *arXiv:2003.05037* (2020).

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proc. of the IEEE CVPR*. 770–778.

[12] Chaolin Huang, Yeming Wang, Xingwang Li, Lili Ren, and Xiaoying Gu. 2020. Clinical features of patients infected with novel coronavirus in Wuhan, China. *The Lancet* 395, 10223 (2020), 497–506.

[13] Gao Huang, Yixuan Li, Geoff Pleiss, and Kilian Q Weinberger. 2017. Snapshot Ensembles: Train 1, get m for free. *arXiv:1704.00109* (2017).

[14] Gao Huang, Zhuang Liu, and Kilian Q Weinberger. 2017. Densely connected convolutional networks. In *proc of the IEEE CVPR*. 4700–4708.

[15] Brian Kenji Iwana, Ryohei Kuroki, and Seiichi Uchida. 2019. Explaining Convolutional Neural Networks using Softmax Gradient Layer-wise Relevance Propagation. *arXiv:1908.04351* (2019).

[16] V Kamalaveni, R Anitha Rajalakshmi, and KA Narayanankutty. 2015. Image denoising using variations of Perona-Malik model with different edge stopping functions. *Procedia Computer Science* 58 (2015), 673–682.

[17] Ashiqur Karim, Md. Rezaul Rahman, João Bosco Jares, Stefan Decker, and Oya Beyan. 2019. A Snapshot Neural Ensemble Method for Cancer-type Prediction Based on Copy Number Variations. *Neural Computing and Applications* (2019), 1–19.

[18] Md Rezaul Karim, Oya Beyan, Achille Zappa, Ivan G Costa, Dietrich Rebholz-Schuhmann, Michael Cochez, and Stefan Decker. 2020. Deep learning-based clustering approaches for bioinformatics. *Briefings in Bioinformatics* (02 2020). https://doi.org/10.1093/bib/bbz170 bbz170.

[19] M. R. Karim, M. Cochez, O. Beyan, S. Decker, and C. Lange. 2019. OncoNetExplainer: Explainable Predictions of Cancer Types Based on Gene Expression Data. In *2019 IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE)*. 415–422.

[20] M. R. Karim, G. Wicaksono, and I. G. Costa et al. 2019. Prognostically Relevant Subtypes and Survival Prediction for Breast Cancer Based on Multimodal Genomics Data. *IEEE Access* 7 (2019).

[21] Bum Jun Kim, Gyogwon Koo, Hyeyeon Choi, and Sang Woo Kim. 2020. Extending Class Activation Mapping Using Gaussian Receptive Field. *arXiv:2001.05153* (2020).

[22] Ilya Loshchilov and Frank Hutter. 2016. SGDR: Stochastic Gradient Descent with Warm Restarts. *arXiv:1608.03983* (2016).

[23] Charles H Martin and Michael W Mahoney. 2019. Traditional and heavy-tailed self regularization in neural network models. *arXiv:1901.08276* (2019).

[24] Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. 2017. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition* 65 (2017), 211–222.

[25] Ali Narin, Ceren Kaya, and Ziynet Pamuk. 2020. Automatic Detection of Coronavirus Disease using X-ray Images and Deep Convolutional Neural Networks. *arXiv:2003.10849* (2020).

[26] Ming-Yen Ng, Elaine Y Lee, Jin Yang, and Pek-Lan Khong. 2020. Imaging profile of COVID-19 infection: radiologic findings and literature review. *Cardiothoracic Imaging* 2, 1 (2020).

[27] Sampada S Pathak, Prashant Dahiwale, and Ganesh Padole. 2015. A combined effect of local and global method for contrast image enhancement. In *International Conference on Engineering & Technology*. IEEE, 1–5.

[28] Pietro Perona and Jitendra Malik. 1990. Scale-space and edge detection using anisotropic diffusion. *IEEE Tras. on pattern analysis and machine intelligence* 12, 7 (1990), 629–639.

[29] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE ICCV*. 618–626.

[30] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556* (2014).

[31] Aleksei Tiulpin, Jérôme Thevenot, Esa Rahtu, Petri Lehenkari, and Simo Saarakkala. 2018. Automatic Knee Osteoarthritis Diagnosis from Plain Radiographs: A Deep Learning-based Approach. *Scientific reports* 8, 1 (2018), 1727.

[32] Linda Wang and Alexander Wong. 2020. COVID-Net: A Tailored Deep Convolutional Neural Network Design for Detection of COVID-19 Cases from Chest X-ray Images. *arXiv:2003.09871* (2020).

[33] Saining Xie, Ross Girshick, and Kaiming He. 2017. Aggregated residual transformations for deep neural networks. In *proc. of IEEE conference on computer vision and pattern recognition*. 1492–1500.

[34] Kate Madden Yee. 2020. X-ray may be missing COVID cases found with CT. *Korean Journal of Radiology* (2020), 1–7.