ISSN 1870-4069

Convolutional Neural Networks Comparison in Covid Incidence Detection

Abel Robles Montoya[1], Alec Guerrero Gallardo[1], Jaqueline P.[1], David Alejandro T.[1], Victor G.[1], Volodymyr Ponomaryov[2]

[1] Tecnológico Nacional de México, campus Culiacán, Mexico

[2] Instituto Politecnico Nacional, Mexico

alx3416@hotmail.com

Abstract. This paper aims to compare the different state-of-the-art neural network performances in binary detection of the novel covid-19. We will be using Xception, VGG16, MobileNetV3, EficientNetB0, and NasNetMobile to make our comparison, working with approximately 4,000 Ches X-ray (CXR) images as our dataset. We obtain the quality measures (Accuracy, Precision, Recall, F1-score), confusion matrix, the receiver operating characteristic curve (curve), and saliency maps to observe the legitimacy of each network output. Xception gives us the best performance (at the expense of having the highest parameter volume) with a general accuracy of 0.97, a recall of 0.97 on the positive class, a F1-score of 0.97, and Area Under the Curve (AUC) of 0.995 on the ROC curve. EfficientNetB0 obtain a marginal 0.01 difference with respect to Xception in both recall and F1-score archiving a general accuracy of 0.97, a recall of 0.96 on the positive class, a F1-score of 0.96, and an AUC of 0.996, with the huge benefit of having about 17.3 millions fewer parameters. We present a comparison with previous works, where we can observe that our best model obtain a relatively reliable performance taking into consideration the bigger number of CXRs compared to the rest of the researches. Finally, we conclude with a discussion and future related work.

Keywords: Transfer learning, machine learning, deep learning, COVID-19.

1 Introduction

The appearance of the new coronavirus disease (covid-19) that was detected for the first time in Wuhan, China on December 31, 2019, and that has caused thousands of infections around the world have incentivizes the use of multiple new technologies in order to detect news incidences. To carry out the detection of covid-19, different techniques have been implemented such as nucleic acid

detection (poly-merase chain reaction or PCR)[14], antigen detection and antibody detection. The most widely used novel coronavirus (COVID-19) detection technique is real time PCR. However, considering its high cost and confirmation time, the rapid spread of the disease, the increment of demand for tests and the fact that the less sensitivity version of real time PCR provides high false-negative results, have caused other alternatives to be sought to complement the detection of covid 19. To resolve this problem, radiological imaging techniques such as chest X-rays (CXR), computed tomography (CT) [13] and the implementation of artificial intelligence has been used not only for detection but also to study and understand the behavior of this disease.

Deep learning is successfully used as a tool for machine learning, where a neural network is capable of automatically discern features. This makes a big contrast versus traditionally hand-crafted features. Among deep learning techniques, deep convolutional networks are actively used for medical image analysis. This includes applications such as segmentation, abnormality detection, disease classification, computer-aided diagnosis, and retrieval[4]. These techniques have been used to address the problem of discriminating benign cysts from malignant masses in breast ultrasound (BUS) images based [34] and for covid-19 detection using CXR [8].

This project aims to implement different Convolutional Neural Network (CNN) architectures to detect covid-19 using CXR images and to make a comparison between them to know which of them is better for this type of data and highly reduce the number of false negatives. For this proposes we implement different pre-trained models: MobileNet V3, EfficientNetB0, NasNet-Mobile, Xception, and VGG16. Previous studies have implemented models of deep learning through transfer learning to detect covid-19 such as in [17], where five pre-trained CNN-based models were proposed for detection of coronavirus pneumonia-infected patients using CXR radiographs. Three different binary classifications with four classes (COVID-19, normal (healthy), viral pneumonia, and bacterial pneumonia) were implemented by using five-fold cross-validation. Considering the performance results obtained, it has been seen that the pre-trained ResNet50 model provides the highest classification performance. Research [2] presented a novel method to improve the performance of transfer learning when the target domain data is not only scarce, but it also has a slightly different modality. This technique uses an ensemble of deep learning models that are modified and hierarchically fine-tuned to the target domain. They tested their technique by using pre-trained models of natural images (ImageNet) and transferring them to the domain of chest radiography images. The study [28] aimed at evaluating the effectiveness of the state-of-the-art pre-trained CNNs on the automatic diagnosis of COVID-19 from CXRs. The results showed that deep learning with X-ray imaging is useful in collecting critical biological markers associated with COVID-19 infections. In [6], the performance of state-of-the-art CNNs architectures proposed over the recent years for medical image classification was evaluated. The results suggest that Deep Learning with X-ray imaging may extract significant biomarkers related

to the Covid-19 disease. Another work where transfer learning was used to detect covid 19 via CXR was in project[13] by using the extreme version of the Inception (Xception) model. Extensive comparative analyses show that the proposed model performs significantly better as compared to the existing models. And also [19] proposed a model that depends on the working of deep learning-based CNN known as nCOVnet. it can classify the COVID-19 patient correctly with 97.97 % confidence.

In [5], the-state-of-the-art CNN called MobileNet was employed and trained from scratch to investigate the importance of the extracted features for the classification task. The results suggest that training CNNs from scratch may reveal vital biomarkers related but not limited to the COVID-19 disease, while the top classification accuracy suggests further examination of the X-ray imaging potential. In study[24] the deep learning based methodology is suggested for detection of coronavius-infected patient using X-ray images. In [29] the proposed approach was designed to provide fast and accurate diagnostics for COVID-19 diseases with binary classification (COVID-19, and No-Findings) and multi-class classification (COVID-19, and No-Findings, and Pneumonia). The proposed method achieved an accuracy of 97.24%, and 84.22% for binary class, and multi-class, respectively. In the study [31], developed COVID-Net a deep CNN was designed tailored for the detection of COVID-19 cases from chest CXR images that is open source and available to the general public. And also, deep learning framework have been developed; namely COVIDX-Net to assist radiologists to automatically diagnose COVID-19 in X-ray images[15].

However, our work differs from some of the studies. We use a dataset with 2000 positive images and 2000 negative images, we don't use data augmentation, and our models are binary classifiers. Our research will help us to know more information about CNN pre-training models, which of these architectures give us more precise results, and which of them are more efficient according to the parameters it needs and its performance to solve these kinds of problems.

This paper is organized as follows. In section 2, we describe the state-of-the-art, previous studies that are related to this work. In section 3, the methodology that we follow to carry out this project. The results and the comparison of the quality measures of the different models are described in section 4. And finally, in section 5, we present the conclusions.

2 State of the Art

Previously comparisons have been made for different image classification techniques, and on some occasions, results obtained by different network architectures were directly compared, [30] develop a research project where InceptionV3, VGG16, ResNet50, and Xception networks were evaluated, which although they are not exactly the networks that will be compared below if data were obtained that helped as a starting point for this research. [30] classifies objects from the Cifar10 dataset using deep learning and implemented in an embedded device such as the Nvidia Jetson TX2, to compare the TensorFlow, Caffe, and Pytorch

frameworks for deep learning. This research compares the aforementioned architectures, concluding that the Xception network had a higher precision (93.3%) than Inception V3 network (85.8 %), similar to VGG16 (91.9%), but lower than ResNet50 (98.6%) [30].

In 2018, [34] made a comparison between some models, and consist in evaluate three transferred models InceptionV3, ResNet50, and Xception, a CNN model with three convolutional layers (CNN3), and traditional machine learning-based model with hand-crafted features were developed for differentiating benign and malignant tumors from breast ultrasound data. The transfer learning model with the pre-trained InceptionV3 network scored the best performance with a maximum accuracy of 85.13%. The accuracy obtained with the ResNet50 and Xception models was slightly lower, 84.94% and 84.06%, respectively. The authors concluded that the transferred InceptionV3 model achieved the best accuracy among the models compared.

In 2020, [3] proposed a more sophisticated transfer learning that not only systematically modifies the original network architecture, but also hierarchically refines it on augmented target domain data. In addition, they use a set of modified networks to make a prediction. They established the effectiveness of their technique with the ChestXray-14 radiography data set.

Their experimental results showed more than 50% reduction in the error rate using their method as compared to the baseline transfer learning technique. After that, they applied their technique to a COVID-19 data set for binary and multi-class classification tasks. Their technique achieved 99.49% accuracy for the binary classification, and 99.24% for multi-class classification.

With COVID-19 dataset, they compared four models, DenseNet201, ResNet50, Inception-V3 and VGG-16, and an ensemble of them. The accuracy mentioned above was from the ensemble of models.

Following a brief description of the principal networks we are using to conduct our experiment.

The NasNetMobile model is a mobile version of NasNet search which is a scalable CNN architecture consisting of blocks optimized for reinforced learning[16], this mobile version consists of 12 blocks with 5.3 million parameters[23], in this work we will use this version of NasNet with the weights obtained when training with the ImageNet dataset. MobileNet V3 it is the new entry in the MobileNet series and it is also based on network architecture search (NAS). Implementing complimentary search techniques and a novel architecture, this network promises to give the best results yet [16].

MobileNetV3 has two versions, which target are different resource use cases. In this work, we will use the MobileNetV3-Large version. Xception architecture is entirely based on depthwise separable convolution layers with residual connections, which is an extreme interpretation of the Inception model, its structure makes it easy to define and modify, and in some comparations performed with the JFT and ImageNet dataset, Xception shows a better performance in classification.[9] Despite this architecture not represent a big change from which

is based, their characteristics make it a good option to compare with other state of art architectures for image classification.

In 2019, [27] proposed a new way of thinking in model Scaling for CNNs introducing a novel method that uniformly scales all dimensions of depth/width/resolution(DWR). One of the observations the authors make state that: *Scaling up any dimension of network width, depth, or resolution improves accuracy, but the accuracy gain diminishes for bigger models.* This lead the to think that, in order to maximize accuracy and efficiency, it crucial to balance the DWR of the network. The result was the family of networks knows as EfficientNets, which score a state-of-the-art 84.3% top-1 accuracy on ImageNet on the 7th version while being smaller and faster than most convolutional networks. Finally VGG16 model was proposed to investigate the effects of the convolutional network depth and the performance on the localization and classification tasks. The authors Karen Simonyan and Andrew Zisserman presented this architecture in [22] in 2014, winning first and second place in the localization and classification categories, which proves to be an excellent vision model architecture [26].

Briefly, we use 5 models: MobileNetV3 Large, EfficientNetB0, Xception, NasNetMobile and VGG16. We used these models because the libraries that implement them run natively in Google Colaboratory, those libraries are TensorFlow and Keras. Despite having mentioned the ResNet50 network and seeing that it has been one of the best performing networks, we decided not to use that model, because we realized that the ResNet50 network has already been extensively tested against other networks, and we wanted to test the performance of networks that were not as widely used and that were already implemented in Google Colaboratory.

3 Metodology

Now, we will describe how we select the data, the general process and how our neural networks were set up. In this project, we work with pre-trained models and we modify some configurations of the different architectures until adapting them with the data which we decide to work, some parameters were modified to get better performance, such as the optimization function, the layers of the neural networks, the activation function, the loss function, and some other parameters that will be shown below.

3.1 Dataset

This dataset was developed by Lida Wang, Zhong Qiu Lin, and Alexander Wong, and its name is COVIDx, it is comprised of a total of 13,975 CXR images across 13,870 patient cases. And it was generated combining and modifying five different publicly available data repositories[32, 12, 11, 10, 21, 18, 33]: 1) COVID-19 Image Data Collection. 2) COVID-19 Chest X-ray Dataset Initiative. 3) ActualMed COVID-19 Chest X-ray Dataset Initiative, established in collaboration with ActualMed. 4) RSNA Pneumonia Detection Challenge dataset (which used publicly

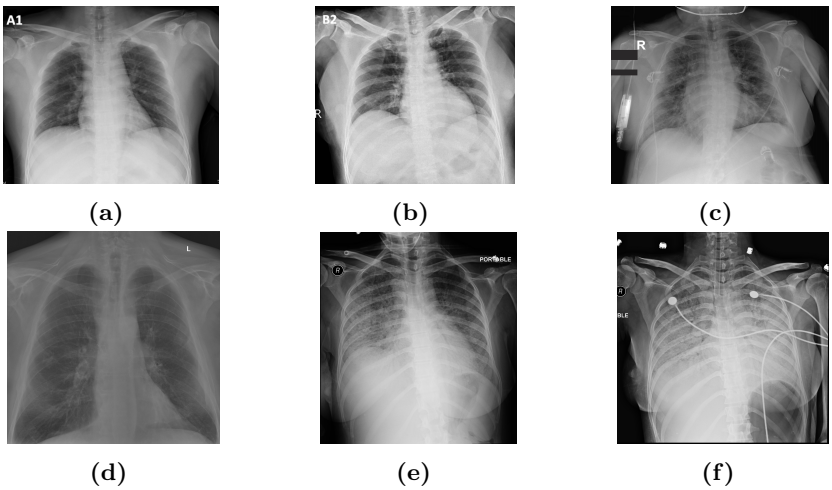


Fig. 1. Sample dataset images labeled with negative results of covid-19 with, a), b) and c) are positive, d), e) and f) are negative.

available CXR data from hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases). 5) COVID-19 radiography database.

The choice of these five datasets from which to create COVIDx, the authors says, is guided by the fact that all five of the datasets are open source and fully accessible to the research community and the general public, and as datasets grow they will continue to grow COVIDx accordingly. More specifically, Wang et. al. combined and modified the five data repositories to create the COVIDx dataset by leveraging the following types of patient cases from each of the data repositories [32, 12, 11, 10, 21, 18, 33]: • Non-COVID19 pneumonia patient cases and COVID-19 patient cases from the COVID-19 Image Data Collection. • COVID-19 patient cases from the COVID-19 Chest X-ray Dataset Initiative. • COVID-19 patient cases from the ActualMed COVID-19 Chest X-ray Dataset Initiative. • Patient cases who have no pneumonia (i.e., normal) and non-COVID19 pneumonia patient cases from RSNA Pneumonia Detection Challenge dataset. • COVID-19 patient cases from COVID-19 radiography database.

We used only 4000 images from COVIDx dataset because of hardware limitations, moreover we only had 2000 negative case images in the chosen dataset, so we took the same number of positive case images to make it balanced. The input images were of 224 pixels per side and RGB, so their dimensions were 224x224x3.

Below, in figure 1, is a sample of what images labeled with positive and negative results look like in the dataset.

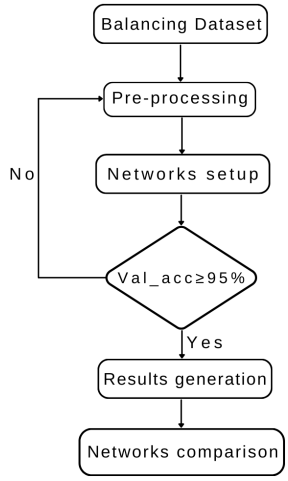


Fig. 2. General process of the experiment.

Next, in figure 2, we present the general flow we used to conduct this experiment. As previously mentioned, we balance the data set in two sets of approximately 2000 samples for class, then we pre-process it as the network requires it (0-1 or 0-255 or No normalization), and set up the network,(this will be described in much more detail in the next section). Continuing, we train the networks making use of callbacks for monitoring the metric valLoss, to change the learning rate or early stop the training if no substantial learning was archive in the few last epochs. We set a threshold of ValAccuracy $\geq 95\%$ in order to proceed to the next step in the process, if it did not fulfill the condition, we return to the image pre-processing, searching for new ways to represent the information or we re-tune the network. When such threshold was cover, we proceed to estimate our quality measures, confusion matrix and ROC curve for further comparison.

3.2 Setting up the networks

`MobileNet V3 Large:` We used the large version of `MobileNet V3` and we configured this model with the initial weights from ImageNet and after the base model, we added the layers, first we added a `GlobalAveragePooling2D` layer, after that a `Dropout` layer of 50%, then a Flatten layer, and finally a `Dense` layer with 1 neuron and a *sigmoid* activation function. We use an `Adam` optimizer, which combines the best of the `AdaGrad` and `RMSProp` optimizers, with a learning rate of 0.001 and the function loss was `binary crossentropy`, this architecture gives us a total of params of 4,227,713 which 4,203,313 are trainable. This model was trained with a generator as a training dataset of 3884 images which were resized to 224x224 pixels and are `RGB` and a batch size of 60. During the fitting, we use a callback to get a view on internal states and statistics of the model.

`EfficientNetB0:` We drop the top classifications layers and replace them with a `GlobalAveragePooling2D` layer, a `Dropout` (50%), `Flatten`, and finally a `Dense` 2 neurons layers with a *softmax* activation function. We train the whole network almost from scratch, reusing only the 2% of the ImageNet weights. Because the normalization process is a part of the network itself, we introduce the data into the network without any pre-processing (`int64`) in `RGB` mode and 60 image batches.

`Xception:` The model was configured with the initial weights from and after the base model, we added a `GlobalAveragePooling2D` layer, a dense layer with 256 neurons and `Relu` activation function, a `Dropout` layer of (50%) and at the end, a `Dense` layer with 2 neurons and `softmax` activation function. We use an `RMSprop` optimizer with a learning rate of 0.0001 and the function loss was `categorical crossentropy`, this architecture gives us total params of 21,386,538 which 21,332,010 are trainable. This model was trained with a generator as a training dataset of 3884 images which were resized to 224x224 pixels and are `RGB`, 60 of batch size, and class mode categorical. During the fitting, we use a callback to monitor the loss and reduce the learning rate where needed.

`NasNetMobile:` In this model, we use the ImageNet initial weights as well and implement the `NasNetMobile` base model without including the top layers, after the base model we added some layers like `Dropout` layer of 50%, `Flatten`, `BatchNormalization`, and `Dense` of 256 neurons, a total of 14 layers before the output layer which has 2 neurons and `softmax` activation function. This model had 17,858,582 total params which 13,483,842 are trainable and was compiled with `categorical crossentropy` as loss function.

`VGG16:` We configurated this model with initial weights from ImageNet, we added a `Flatten` layer, a `Dropout` layer of 50% which helps prevent overfitting, and a `Dense` layer with *sigmoid* activation, the function loss was `binary crossentropy` and we use an `Adam` optimizer. This architecture gives us a total of params of 14,739,777 which 25,089 are trainable. This model was trained with a generator as a training dataset of 3884 which were resized to 224x224 pixels `RGB` and a batch size of 60. During the fitting, we use a callback to monitor the loss and reduce the learning rate.

4 Results

In this section, we present the main result of the previously discussed methodology for each network, a comparison, and a summary of the best performance. To compare the models, we evaluate some quality measures like accuracy, precision, recall, F_1 score using the results of the confusion table and ROC curve of each neural network. These results will be shown below.

4.1 Evaluation Metrics

To present the results and compare the models, we consider the following quality measures: Accuracy is the percentage that indicates the ratio of correct predic-

tions compared to the test data, as shown in (1). Precision is the probability, given a positive label, of how many of them are actually positive, as shown in (2). Recall, or sensitivity, is the ratio of correctly predicted positive observations to the all observations in actual class, como se describe en (3). Finally, F_1 score is the weighted average of Precision and Recall, as shown in (4).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \tag{1}$$

$$Precision = \frac{TP}{TP + FP}, \tag{2}$$

$$Recall = \frac{TP}{TP + FN}, \tag{3}$$

$$F_1 score = 2 * \frac{Precision * Recall}{Precision + Recall}, \tag{4}$$

Where TP refers to true positives, TN to true negatives, FP to false positives and FN to false negatives.

4.2 Results

Confusion matrix: Figure 3 shows the confusion matrix obtained for each network where we can see the predictions compared with the actual values of the class to which the image belongs. In this context, we are looking to reduce to the minimum the false-positives (FP) and especially the false-negative (FN) outcomes as it could give the flawed belief that a person is non-contaminated and therefore exposing the community further to viral exposition. Considering this, Xception achieves the best performance as it only had seven FN and six FP, followed by EfficientNetB0 with nine FN and seven FP. The rest of the models, although having relatively high accuracy and recall, have a few more FN and FP, resulting in less reliable networks especially in the case of the mobileNetV3, which has the highest number of misclassifications.

Saliency Maps: The next two figures, show us the regions of the input image that contributes the most to the output value(label prediction), using the tool provied by [25], in order to corroborate the network correctness and its reliability. Figure 4 shows two positive cases (1 each row) and Figure 5 shows two negative cases, the first column in each figure is the original image and the rest belongs to each of the networks.

Comparison table: Table 1 show the quality measures previously mentioned and the total number of trainable parameters. Furthermore, we highlight the positive recall column as we considered the most meaningful result in this context, where a false negative could have far-reaching consequences. Here we can observe very similar outcomes, giving the central stage to VGG16 for archiving a relatively good performance with the lowest number of trainable parameters, approximately 3.97 million less than EfficientNetB0, the next network in the volume tier.

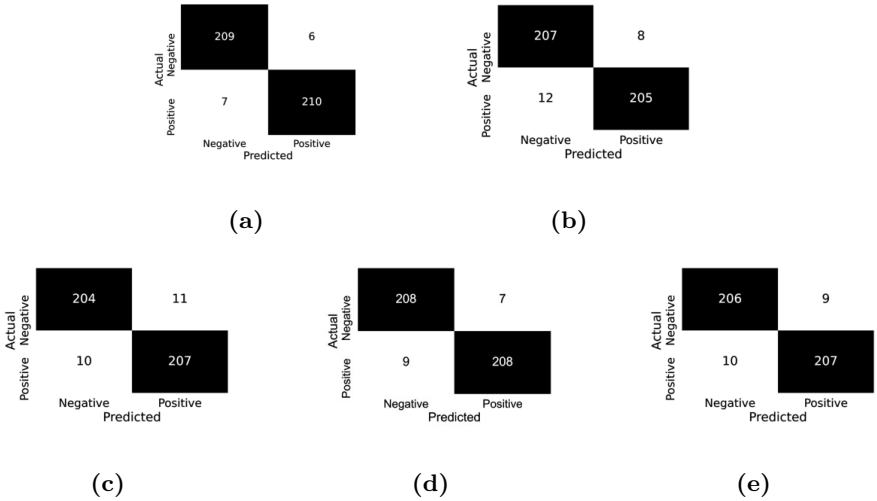


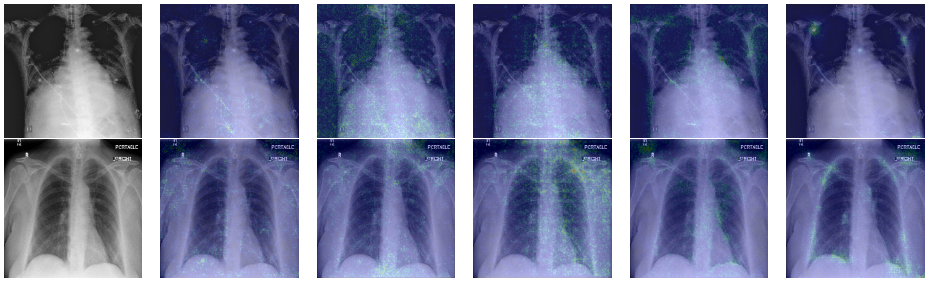
Fig. 3. This figure shows the confusion table for each network with, a) Xception, b) NasNetMobile, c) MobileNetV3, d) EfficientNetB0 and, e) VGG16

However, using the proper tuning and the fact that VGG16 were almost completely reused, it would be unfair to say, VGG16 had the best performance, as is possible to lower the other network's parameter number. In addition to this, we find that, oddly, being training with all its parameters, VGG16 falls behind. From this we consider that VGG16 has a promising reuse value in it is pre-train weights, as for the rest of the networks, they were, for the most part, trained from scratch so its reuse capability is off discussion in this work. We also examine that, considering the good achievement mobiles networks (EfficientNetB0, MobileNetV3Large, NasNetMobile) had relative to Xception (a large one), we can argue that, using this kind of models without transfer learning may be overkill for binary classification with the current number of samples.

That being said, Xception was the model that gives us the best performance in terms of pure statistics, archiving a general accuracy of 0.97 %, a recall of 0.97 % on the positive class, a F1-score of 0.97 %, and AUC of 0.995 on the ROC curve, follow by EfficientNetB0 with accuracy 0.97 %, recall 0.96 % positive class, F1-score 0.96 %, and AUC 0.996.

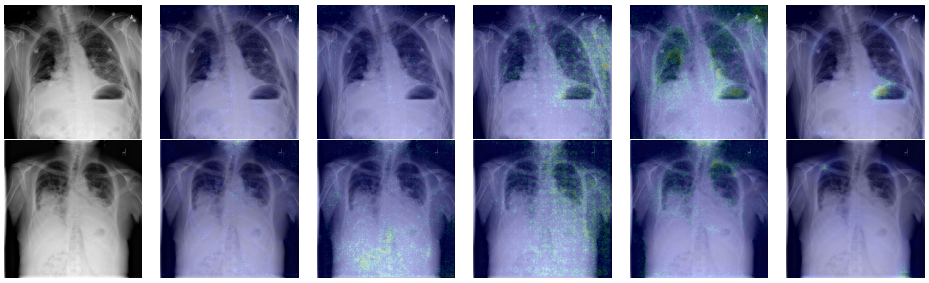
In table 1 we expose the main result of our experiment, showing the precision, recall, F1-score, general accuracy, and number of trainable parameters. We also highlight the positive recall column, as it is the quality measure of highest interest in this work.

ROC curve: Because our data set was well balanced, we did not see the necessity of the precision-recall plot as the ROC curve was a suitable indicator of the diagnostic ability of our classifier. Area Under the Curve (AUC) was calculated using the trapezoidal rule, supported in [20].



Original | MobileNetV3 | NasnetMobile | Xception | VGG16 | EfficientNetB0

Fig. 4. Saliency maps for covid clasification. Each row represents a positive example of covid and the activation of the points that contributed the most to the clasification with each of the networks.



Original | MobileNetV3 | NasnetMobile | Xception | VGG16 | EfficientNetB0

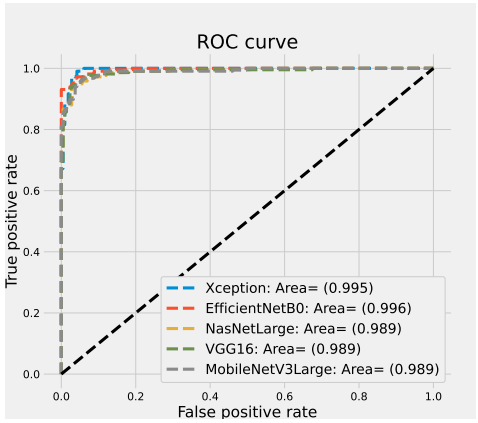
Fig. 5. Saliency maps for covid clasification. Each row represents a negative example of covid and the activation of the points that contributed the most to the clasification with each of the networks.

Table 1. Quality measures.

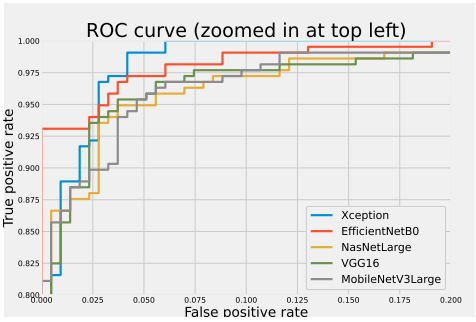
	Precision	**Recall**	F1-Score	Precision	Recall	F1-Score	Accuracy	Parameters (Millions)
Xception	0.97	**0.97**	0.97	0.97	0.97	0.97	0.97	21.3
EfficientNetB0	0.97	**0.96**	0.96	0.96	0.97	0.96	0.96	3.996
NasNetMobile	0.96	**0.95**	0.96	0.95	0.96	0.96	0.96	13.483
VGG16	0.96	**0.95**	0.96	0.95	0.96	0.96	0.96	0.02508
MobileNetV3	0.95	**0.95**	0.95	0.95	0.95	0.95	0.95	4,203

In figure 6a we expose the Receiver Operating Characteristic curve, where we can see the general performance of the diagnostic ability for each network, and in figure 6b we can observe a zoom in the top-left area for a better appreciation. Black serve as reference line. In general, we can see the same performance around all the models, this favorite of course the low parameters networks, due to the

gain in training time and the number of the parameters.



ROC



zoom

Fig. 6. Networks comparison a) ROC Curve b) ROC Curve zoomed in a top left.

Comparison with state-of-the-art: Using the reports published by [17, 28, 1], we present table 2, which summarizes the best performance of different recent articles, using similar datasets of Covid-19 and the same binary problem, to compare this work achievement. The table also includes studies that tackle the problem of discerning Covid-19 incidences on CXR using CNNs from a three classes perspective, being the third-class CXR of patients who suffer viral or bacterial pneumonia. Furthermore, we introduce the sample number of Covid-19 CXR, Healthy or Normal CXR (a.k.a, no-findings), and viral or bacterial pneumonia CXR.

Best architecture with its accuracy (%) is also present for each research. Our method was characterized by containing a relatively high number of CXRs in both Covid-19 and healthy units and adopt a binary approach. We consider that our best model archive a relatively solid result compared with these other

networks, contemplating that our dataset was bigger. That being said, we do consider that there is still room for improvement, using a bulkier dataset, employing data augmentation or image enhancement techniques, and adding pneumonia CXR specimens for network validation.

Table 2. General comparison of this work best results with the state-of-the-art.

Study	No. of samples	Architecture	Number of clases	Accuracy (%)
[17]	50 Covid-19, 50 Normal	ResNet-50	2	98
[28]	1,200 Covid-19, 1,341 Normal, 1,345 viral Pneumonia	VGG-16	3	98.29
[7]	224 Covid-19, 700 Pneumonia, 504 Healthy	VGG-19	3	98.75
[15]	25 Covid-19, 25 Normal	VGG-19, DenseNet-121	2	90
[24]	25 Covid-19, 25 Normal	ResNet-50	2	95.38
[29]	1,050 Covid-19, 1,050 Normal	CapsNet	2	97.24
[19]	192 Covid-19, 145 Normal	nCOVnet	2	97.62
[31]	358 Covid-19, 8,066 Normal, 5,538 Pneumonia	COVID-Net	3	93.3
[13]	125 Covid-19, 500 Normal, 500 Pneumonia	Xception	3	97.40
[5]	224 Covid-19, 504 Normal, 700 bacterial Pneumonia	VGG-19	3	93.48
This work	**2,158 Covid-19, 2,158 Normal**	**Xception**	**2**	**97**

5 Conclusions

In this paper, a comparison was made between five neural network models, which were pre-trained with the ImageNet dataset and then individually configured to be trained with chest X-ray images to identify covid incidences which were obtained from the COVIDx dataset and only approximately 4000 were used due to hardware limitations, each network was implemented using the base model of the corresponding architecture, but in some cases with the necessary adaptations to obtain better results.

This comparison shows that, although the five models obtain close results in terms of the recall measure, which is the most relevant for the problem at hand, some models such as VGG16 achieve relatively good results with a very small number of parameters compared to those that obtained the best results but with a considerably larger number of parameters such as Xception which obtained the highest recall or EfficientNetB0 with very close results but a larger area under the curve exceeding Xception with 0.01, which demonstrates the impact that can be achieved using transfer learning before training the model for a specific dataset.

Depending on the problem, it may be worth sacrificing a certain percentage of efficiency to greatly improve the performance of the network. The main contributions are: first, the transfer learning used in each model, adapting the architecture and parameters to the specific problem. Second, the comparison between some of the most popular models evaluated quantitatively with quality metrics and the general performance as a classifier through the roc curve. Third, the implementation of these models for covid detection using radiographs.

In future work, we can analyze variance to ensure that with a smaller part of the dataset very similar results are obtained, also improve the architecture with fewer parameters preserving the results and improve the most relevant quality metrics by adding images of cases with pneumonia as it is often confused with the covid and even use a larger dataset or data augmentation to get the train the network with a larger number of samples, finally we would also add a preprocessing stage of the images where everything that is not relevant for classification is removed.

This work is available at the Github repository[3].

References

1. Altaf, F., Islam, S.M., Janjua, N.K.: A novel augmented deep transfer learning for classification of COVID-19 and other thoracic diseases from X-rays. Neural Computing and Applications 0 (2021), https://doi.org/10.1007/s00521-021-06044-0
2. Altaf, F., Islam, S.M., Janjua, N.K.: A novel augmented deep transfer learning for classification of covid-19 and other thoracic diseases from x-rays. Neural Computing and Applications pp. 1–12 (2021)
3. Altaf, F., Islam, S.M., Janjua, N.K.: A novel augmented deep transfer learning for classification of covid-19 and other thoracic diseases from x-rays. Neural Computing and Applications pp. 1–12 (2021)
4. Anwar, S.M., Majid, M., Qayyum, A., Awais, M., Alnowami, M., Khan, M.K.: Medical image analysis using convolutional neural networks: a review. Journal of medical systems 42(11), 1–13 (2018)
5. Apostolopoulos, I.D., Aznaouridis, S.I., Tzani, M.A.: Extracting Possibly Representative COVID-19 Biomarkers from X-ray Images with Deep Learning Approach and Image Data Related to Pulmonary Diseases. Journal of Medical and Biological Engineering 40(3), 462–469 (2020)
6. Apostolopoulos, I.D., Mpesiana, T.A.: Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. Physical and Engineering Sciences in Medicine 43(2), 635–640 (2020)
7. Apostolopoulos, I.D., Mpesiana, T.A.: Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks. Physical and Engineering Sciences in Medicine 43(2), 635–640 (2020), https://doi.org/10.1007/s13246-020-00865-4
8. Bassi, P.R., Attux, R.: A deep convolutional neural network for covid-19 detection using chest x-rays. Research on Biomedical Engineering pp. 1–10 (2021)
9. Chollet, F.: Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1251–1258 (2017)
10. Chung, A.: Actualmed covid-19 chest x-ray data initiative. https://github.com/agchung/Actualmed-COVID-chestxray-dataset (2020)
11. Chung, A.: Covid-19 chest x-ray data initiative. https://github.com/agchung/Figure1-COVID-chestxray-dataset (2020)

[3] https://github.com/Abel-RM/Deteccion-Covid.git

12. Cohen, J.P., Morrison, P., Dao, L., Roth, K., Duong, T.Q., Ghassemi, M.: Covid-19 image data collection: Prospective predictions are the future. arXiv preprint arXiv:2006.11988 (2020)
13. Das, N.N., Kumar, N., Kaur, M., Kumar, V., Singh, D.: Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID- 19 . The COVID-19 resource centre is hosted on Elsevier Connect , the company ' s public news and information (January) (2020)
14. Grupo Directivo de Organización Mundial de la salud: Pruebas diagnósticas para el sars-cov-2 (2020)
15. Hemdan, E.E.D., Shouman, M.A., Karar, M.E.: COVIDX-Net: A Framework of Deep Learning Classifiers to Diagnose COVID-19 in X-Ray Images (2020), http://arxiv.org/abs/2003.11055
16. Howard, A., Sandler, M., Chen, B., Wang, W., Chen, L.C., Tan, M., Chu, G., Vasudevan, V., Zhu, Y., Pang, R., Le, Q., Adam, H.: Searching for mobileNetV3. Proceedings of the IEEE/CVF International Conference on Computer Vision (2019)
17. Narin, A., Kaya, C., Pamuk, Z.: Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks. Pattern Analysis and Applications pp. 1–14 (2021)
18. Pan, I., Cadrin-Chênevert, A., Cheng, P.M.: Tackling the radiological society of North America pneumonia detection challenge. American Journal of Roentgenology 213(3), 568–574 (2019)
19. Panwar, H., Gupta, P.K., Khubeb, M., Morales-menendez, R.: Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID- 19 . The COVID-19 resource centre is hosted on Elsevier Connect , the company ' s public news and information (January) (2020)
20. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12, 2825–2830 (2011)
21. Radiological Society of North America: Covid-19 radiography database. https://www.kaggle.com/tawsifurrahman/covid19-radiography-database (2019)
22. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International journal of computer vision 115(3), 211–252 (2015)
23. Saxen, F., Werner, P., Handrich, S., Othman, E., Dinges, L., Al-Hamadi, A.: Face attribute detection with mobilenetv2 and nasnet-mobile. In: 2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA). pp. 176–180. IEEE (2019)
24. Sethy, P.K., Behera, S.K., Ratha, P.K., Biswas, P.: Detection of coronavirus disease (COVID-19) based on deep features and support vector machine. International Journal of Mathematical, Engineering and Management Sciences 5(4), 643–651 (2020)
25. Simonyan, K., Vedaldi, A., Zisserman, A.: Deep inside convolutional networks: Visualising image classification models and saliency maps. CoRR abs/1312.6034 (2014)
26. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)

27. Tan, M., Le, Q.V.: Efficientnet: Rethinking model scaling for convolutional neural networks. arXiv preprint arXiv:1905.11946 (2019)
28. Taresh, M.M., Zhu, N., Ali, T.A.A., Hameed, A.S., Mutar, M.L.: Transfer learning to detect covid-19 automatically from x-ray images using convolutional neural networks. International Journal of Biomedical Imaging 2021 (2020)
29. Toraman, S., Burak, T., Turkoglu, I.: Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID- 19 . The COVID-19 resource centre is hosted on Elsevier Connect , the company ' s public news and information (January) (2020)
30. Valderrama Molano, J.A., et al.: Clasificación de objetos usando aprendizaje profundo implementado en un sistema embebido. Universidad Autónoma de Occidente (2017)
31. Wang, L., Lin, Z.Q., Wong, A.: COVID-Net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images. Scientific Reports 10(1), 1–12 (2020), https://doi.org/10.1038/s41598-020-76550-z
32. Wang, L., Lin, Z.Q., Wong, A.: Covid-net: a tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images. Scientific Reports 10(1), 19549 (Nov 2020), https://doi.org/10.1038/s41598-020-76550-z
33. Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., Summers, R.M.: 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 (2017)
34. Xiao, T., Liu, L., Li, K., Qin, W., Yu, S., Li, Z.: Comparison of transferred deep neural networks in ultrasonic breast masses discrimination. BioMed research international 2018 (2018)