Classifying Chest X-Ray COVID-19 images via Transfer Learning

Ethics and Explainability for Responsible Data Science (EE-RDS) 2021

Jimut Bahan Pal

Department of Computer Science
Ramakrishna Mission Vivekananda Educational and Research Institute

Nilayan Paul

Department of Physics
University of Calcutta

Contents

Contents

Introduction

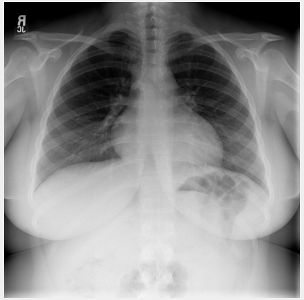
- COVID-19 has changed the way humans interact with the world. It is one of the events in history where humans has thrown almost everything to fight the pandemic with science and technology.
- This study deals with the application of Transfer Learning in classifying Chest X-Ray COVID-19 images with high Accuracy, Sensitivity and Specificity.
- Medical sectors also have a tremendous opportunity in applying Artificial Intelligence and Deep Learning for leveraging the diagnosis process via automation.
- Standard known Deep Learning architectures were used as backbones to classify the given dataset.†
- The models used here were previously trained on the ImageNet [2] dataset and were fine-tuned to get desired results.

†Dataset Retrieved from https://cxr-covid19.grand-challenge.org/Download/

Contents

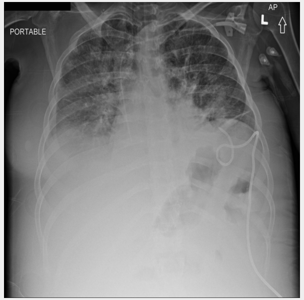
Dataset

- There are **3** classes present in this dataset [11] as shown in the right.
- There are **5273** Pneumonia images, **7966** COVID-19 images, and **8151** Normal images of sizes **512x512** and **1024x1024** (both 3 channel, and grayscale).
- There is a minor class imbalance.
- The distribution of Training and Validation dataset is shown in the Figure (on right).
- The images were rescaled to **360x360** size with **3** channels with intensities, normalised between **0** and **1** before passing to the model.

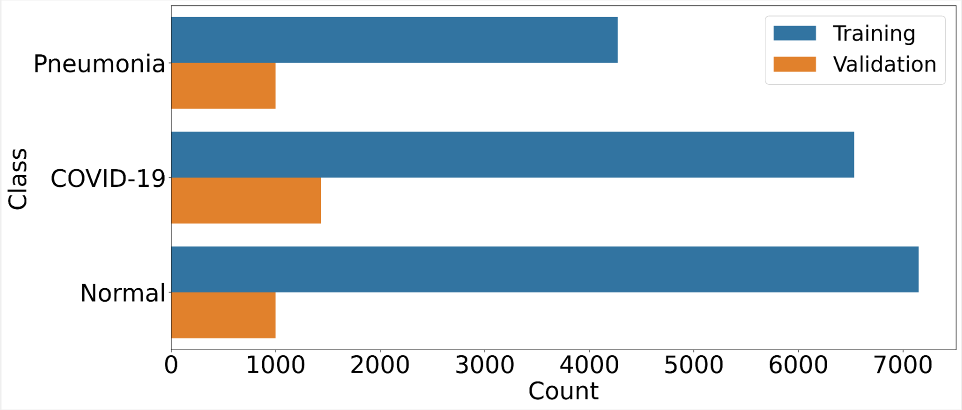


Normal

Covid-19



Pneumonia



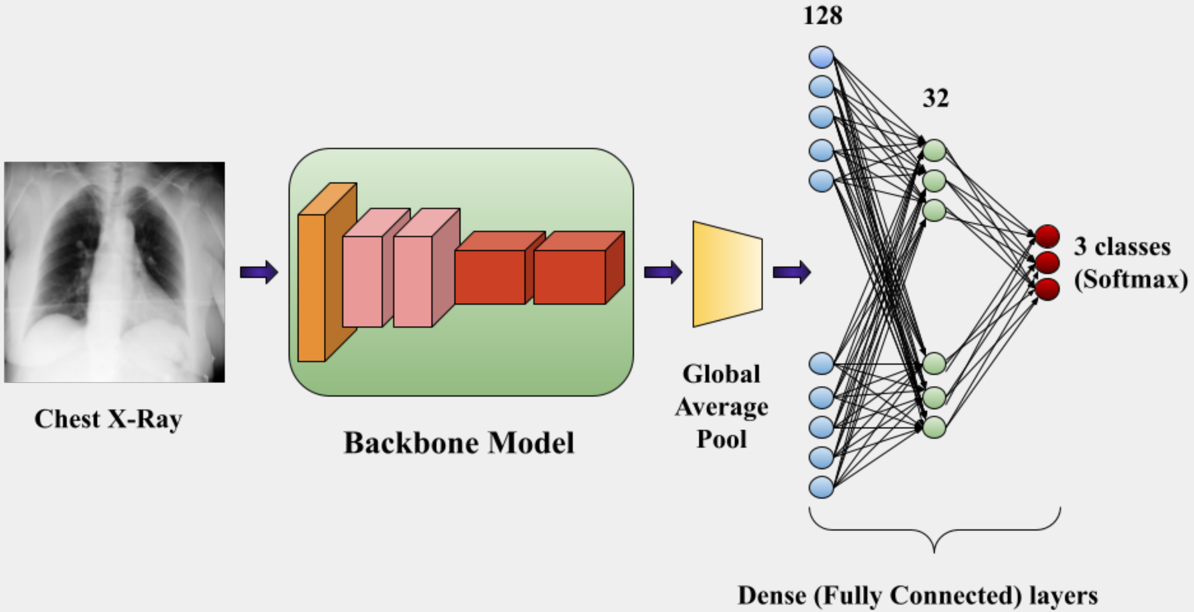
Data distribution across different classes

Contents

Model Structure

- The architecture of the model which was used to check the performance of validation dataset is shown in the figure.

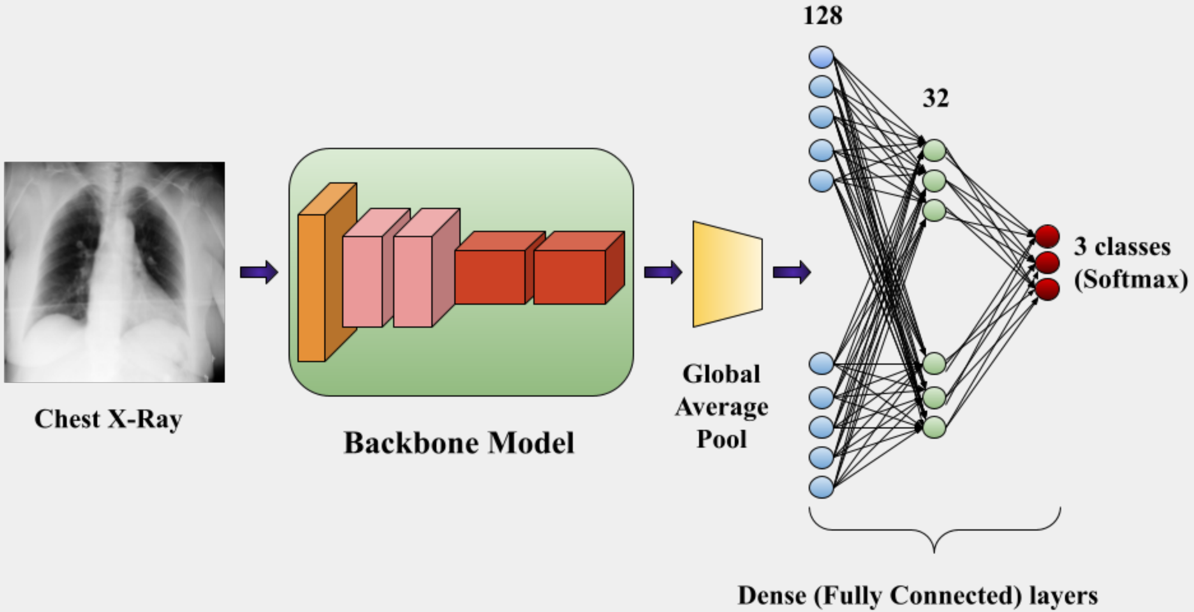
- We use the backbone model as the convolutional part of all the standard architectures that were taken into consideration for the study.



Model Structure

- After the backbone model, a global average pooling was used before passing it to fully connected (dense) layer comprising of **128-32-3** neurons.

- The output layer has **3** neurons corresponding to the **3** classes with **Softmax** as activation function.



Contents

Metrics

- Categorical Cross-Entropy [4] is used as loss function which can be written as:

$$CE = -\sum_{i}^{C=3} t_i log(s_i)$$

- Here, t_i is the actual class and s_i is the predicted class.

- Accuracy, Sensitivity and Specificity are written as follows:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad Specificity = \frac{TN}{TN+FP}$$

$$Sensitivity = \frac{TP}{TP+FP}$$

- Here, **TP** is True Positive, **TN** is True Negative, **FP** is False Positive, **FN** is False Negative.

- Adam optimizer was used with a learning rate of **1e-04**.

- All the models were made using Tensorflow and Keras framework in Python3 language.

Results on the Validation Datasets

- Standard architectures have been used and the results were tested on the validation dataset by using the same model as shown before.

- The results obtained from training is shown in the Tables (in right).

- Due to the limitations in memory different batch sizes were selected, and the details of each of them are shown in the Table (in right).

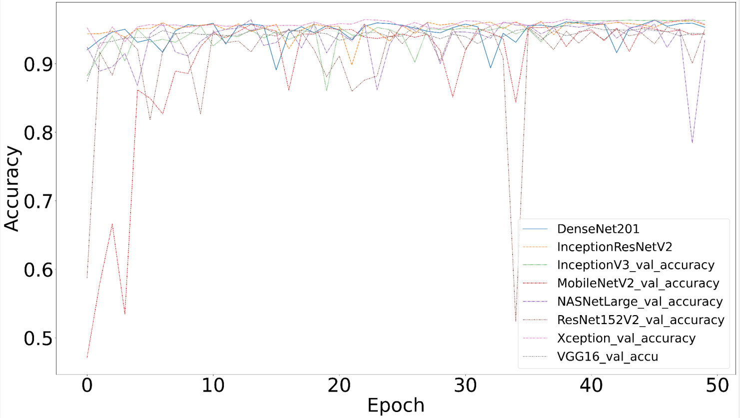
- It is worth noting that Inception V3 performs better than all of the other models in the validation dataset by training on “Training dataset”.

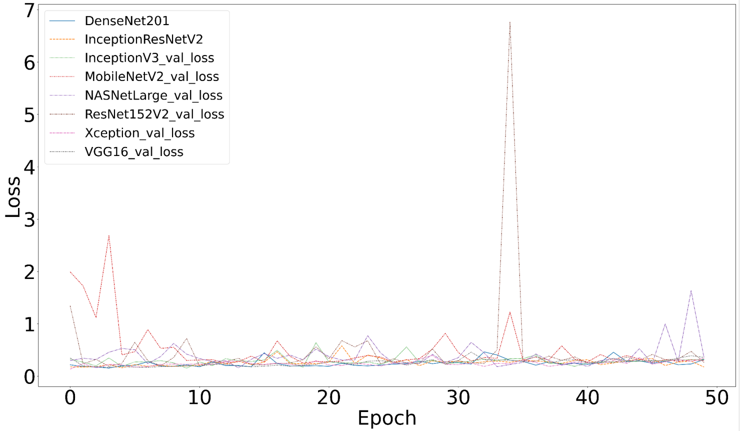
- The graph across 50 epochs are shown in the next slide for training of individual models.

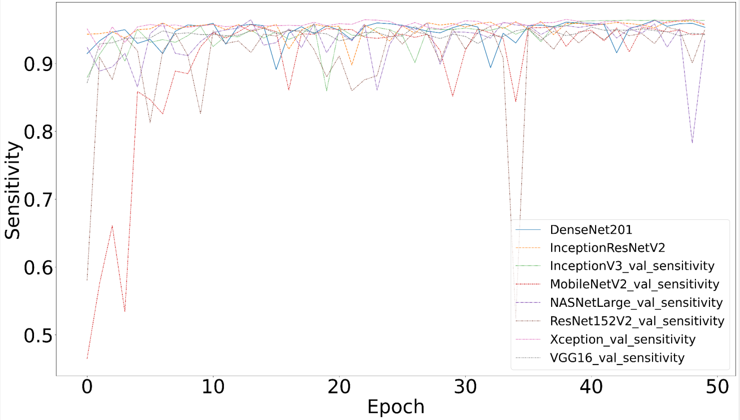
Model Name	**Accuracy (in %)**	**Sensitivity (in %)**	**Specificity (in %)**
MobileNetV2 [14]	96.29	94.09	97.36
VGG16 [9]	96.85	94.75	97.74
DenseNet201 [12]	96.87	94.79	97.75
Xception [13]	97.16	95.34	97.99
InceptionResNetV2 [23]	97.11	95.30	97.94
ResNet152V2 [10]	96.69	94.51	97.61
NASNetLarge [16]	95.69	92.61	96.63
InceptionV3 [11]	**97.57**	**95.98**	**98.27**

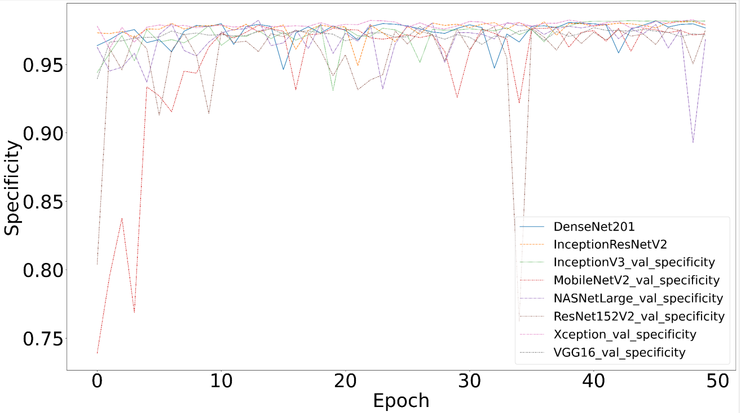
Model Name	**Total Parameters (in Million)**	**Batch Size**	**Avg Time per epoch (in sec)**
MobileNetV2 [14]	2.42	32	224
VGG16 [9]	14.78	32	303
DenseNet201 [12]	18.57	16	366
Xception [13]	21.12	16	489
InceptionResNetV2 [23]	54.53	16	438
ResNet152V2 [10]	58.59	8	590
NASNetLarge [16]	85.43	8	1443
InceptionV3 [11]	92.06	32	218

Results on the Validation Datasets





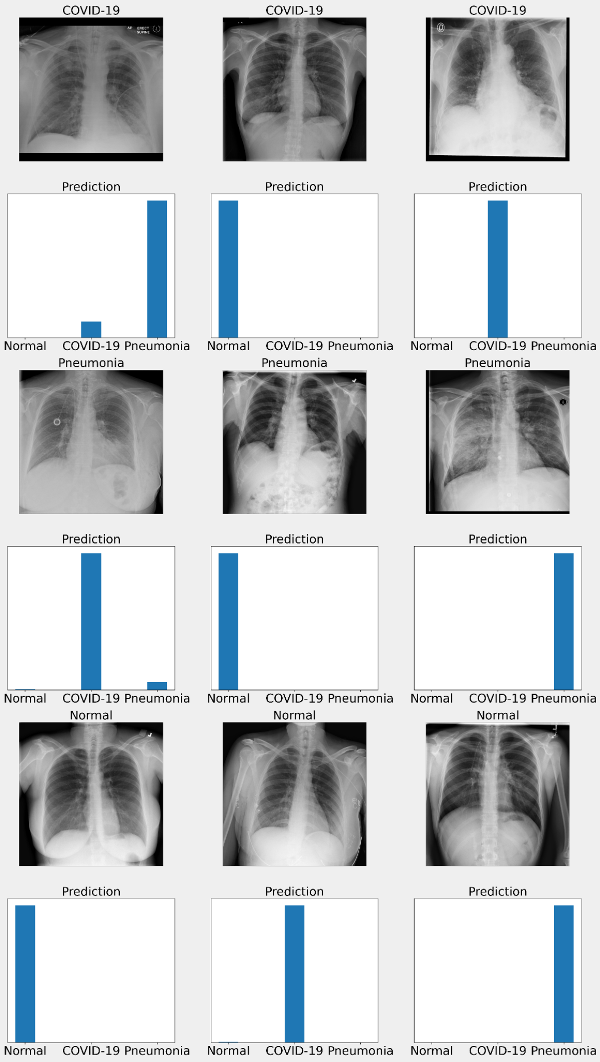




Contents

Confidence of Deep Learning Models

- Since the VGG-16 model has a relatively good result, it has been selected to check the confidence by providing different images.
- Each of the samples shows whether a class is classified correctly or is misclassified when provided to the VGG-16 model.
- Whenever the model is making any wrong prediction, it confidence lies in the wrong class.
- This makes it challenging to see what actually the neural network model learns and what exactly motivates the model to make a particular decision.
- This will help medical practitioners to check the authenticity of the predictions of Deep Learning architectures.



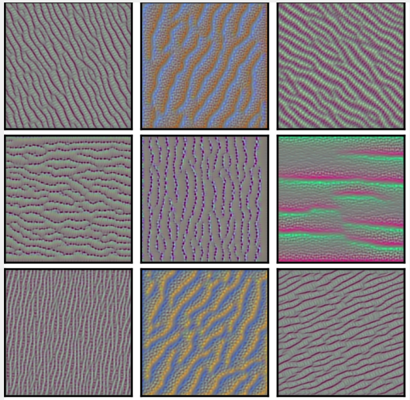
Visualisation of Filters Learnt

- To visualise the difference between the filters learnt, the weights of VGG 16 model before transfer learning were used i.e., Figures (a) and (c), and after transfer learning, i.e., Figures (b) and (d).

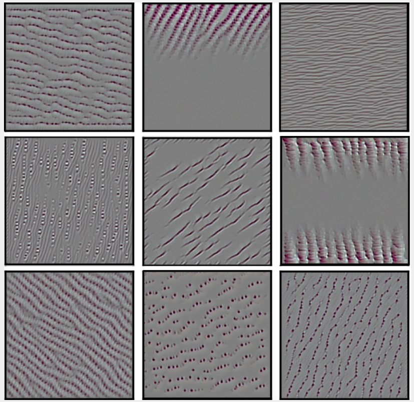
- The figures show that the filters change from recognising textures and patterns of the natural imagery [13] to problem specific images, i.e, Chest X-Ray image features.

- Even the colours change to grayscale (related to X-Ray images).

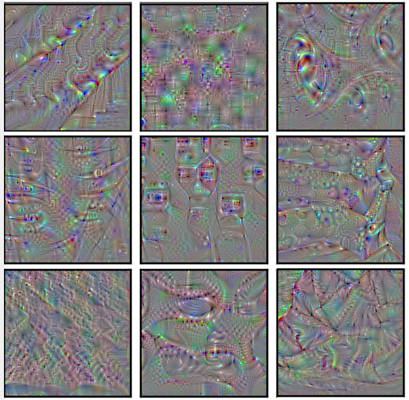
- The initial layers learn textures and the final layers learn patterns which describe the class as a whole.



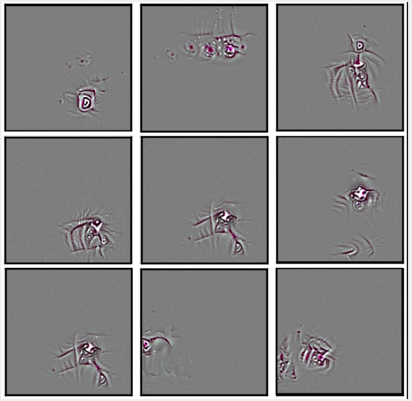
(a)



(b)



(c)

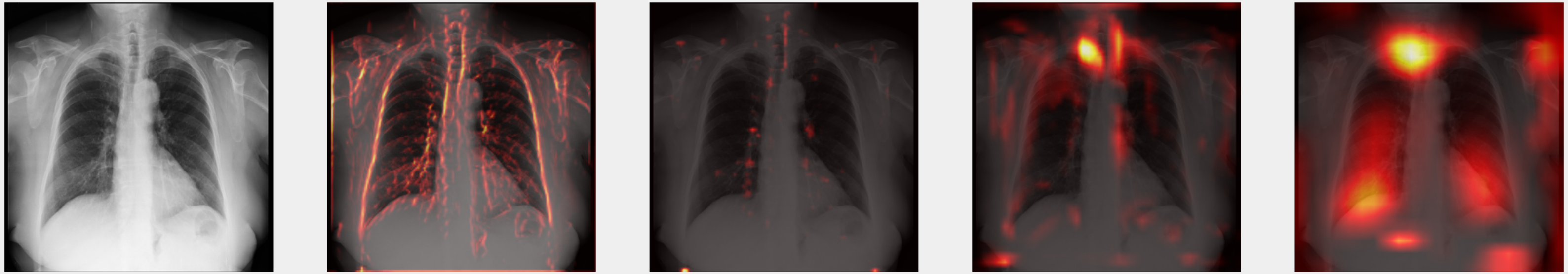


(d)

‘Belief’ of Deep Learning Models

- Sometimes the predictions of the Deep Learning models might be very confusing, and there may be very less information why the model selects a particular class for a particular image.

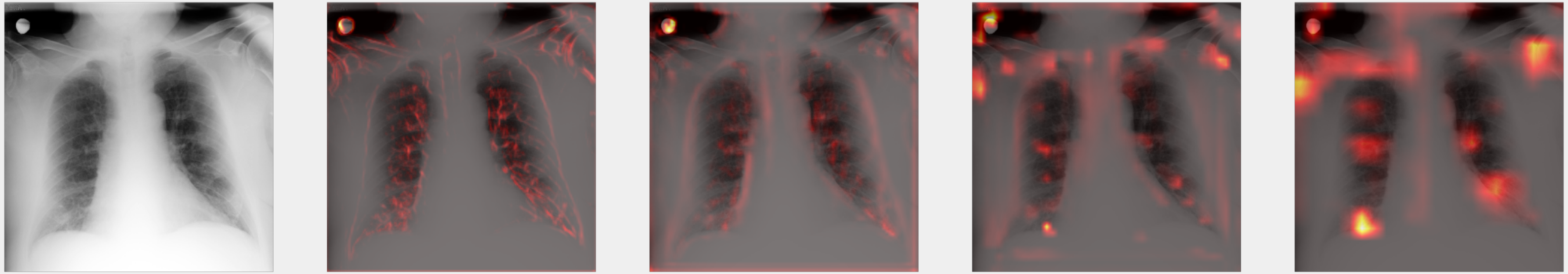
- In the next few slides some efforts have been made to justify the confidence or “belief” of the deep learning model via **GRAD-CAM [5]**.



(a) True positive: COVID detected as COVID

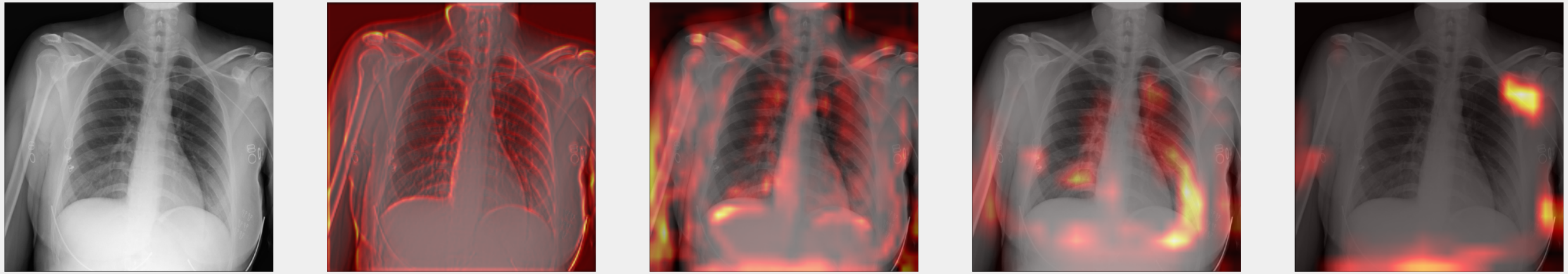
‘Belief’ of Deep Learning Models

- From the visualisations, it looks like the model is not taking into account the confidence from the final layers only, rather it is taking confidences from the individual layers, right from the beginning to the final layers.

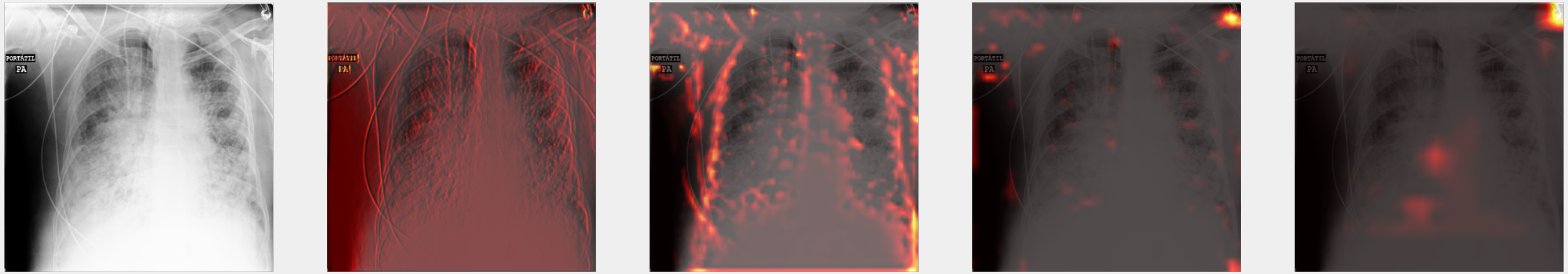


(b) False negative: COVID detected as Normal

‘Belief’ of Deep Learning Models



(c) False positive: Normal detected as COVID



(d) COVID detected as Pneumonia

Contents

Ablation Study

- For ablation study, input image size of 500x500, fully connected layers of 1024-1024-3 neurons, Dropouts or their combinations were used, with VGG16 and InceptionV3 models.

- For studies where the above were not used, input image size of 360x360 or fully connected layers of 128-128-3 neurons were used.

- The performance of the two models for different input features was noted and the results are shown in the next slide.

- The results show that it is not necessarily true that the overall performance of Deep Learning architecture will increase if individual sub-structures which gives better result for a particular setting are combined.

- Also, a particular structure might not give better result when the backbone model is changed, i.e., performance of model and structure is dataset dependent.

Ablation Study (Results on VGG-16 and Inception-V3 models)

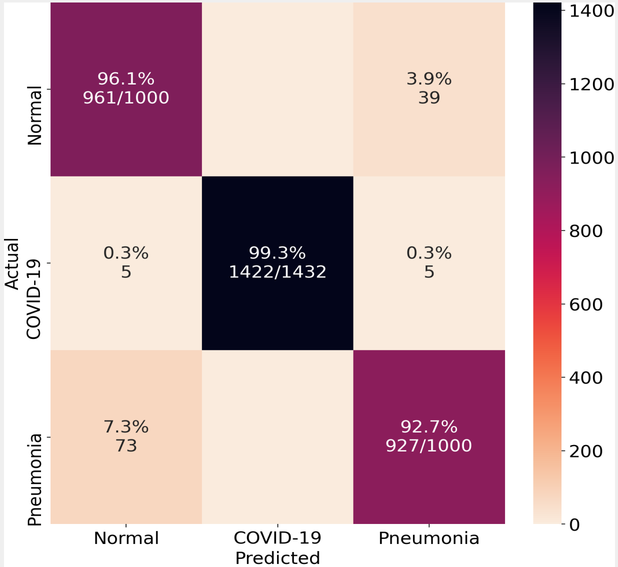
Model Name	500x500x3*	1024-1024-3†	Dropouts	Accuracy (%)	Sensitivity (%)	Specificity (%)
VGG 16	✓			96.87	94.83	97.77
		✓		**96.96**	**94.99**	**97.82**
			✓	96.87	94.91	97.77
	✓	✓		95.82	93.15	97.02
		✓	✓	95.82	92.95	97.00
	✓		✓	96.30	93.81	97.37
	✓	✓	✓	**96.96**	94.91	**97.82**
Inception V3	✓			97.51	95.80	98.24
		✓		96.95	94.99	97.83
			✓	97.20	95.31	98.00
	✓	✓		**97.63**	**96.03**	**98.32**
		✓	✓	97.29	95.58	98.08
	✓		✓	97.28	95.50	98.08
	✓	✓	✓	94.42	90.54	96.05

Contents

Results on Test Set

- When evaluated on unseen test, the results were a bit lower from the one obtained when trained on the training dataset and tested on validation dataset.
- The model which performed the best on Ablation study, was selected for the test set.
- Also, it is worth noting that when the train and validation dataset are combined for training the model and testing on test dataset, the performance decreases slightly. This might be due to the fact that the distribution of the combined train and validation dataset might be shifted from the individual train and test datasets, hence decreasing the performance.
- Hence the final model obtained was trained on just the train dataset and tested on unseen test dataset.

Model Name	**Accuracy (in %)**	**Sensitivity (in %)**	**Specificity) (in %)**
InceptionV3 (train)	94.67	94.67	94.78
InceptionV3 (train + val)	94.41	94.41	94.68



Confusion matrix for InceptionV3 obtained on validation dataset

Contents

Conclusion and Limitations

- Application of Transfer Learning shows that the performance of a deep learning architecture can be improved significantly without the use of data augmentation.
- Ablation studies showed that combining different substructures which performs good individually might not result in a better overall structure.
- It may be confusing for humans to understand what the Deep Learning architecture actually sees, hence more transparent way of seeing the “belief” of Deep Learning architectures are needed.
- The model performs best on the distribution of data which it was trained on, so bringing data from slightly different domain may result in degradation of performance. **Hence this cannot be used as a diagnostic tool.**

Contents

Future Work

- Since the model doesn't perform as well as when it was trained on training dataset and evaluated on validation dataset, so, combining data from different domains may help to learn domain invariant features. So, by performing **Domain Adaptation [14]** the performance of the existing model may be increased.
- **Neural Architecture Search [12]** can be used to find the best model for the given dataset, but it is a computationally expensive task.
- Using **Attention Module** and building different substructures might increase the performance of the existing models.

Contents

Acknowledgements

The authors are also thankful to the organizers of Chest X-Ray COVID-19 detection challenge for their efforts in sharing the datasets.

The authors are grateful to Swathy Prabhu Mj, Ramakrishna Mission Vivekananda Educational and Research Institute, for arranging a machine with an Asus RTX 2080 Ti (12 GB VRAM) and 64 GB RAM, to hasten the research.

Contents

References

1. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” Journal of Machine Learning Research, vol. 15, no. 56, pp. 1929–1958, 2014. [Online]. Available: http://jmlr.org/papers/v15/srivastava14a.html

2. A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in Advances in Neural Information Processing Systems, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., vol. 25. Curran Associates, Inc., 2012.

3. M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, “Tensorflow: A system for large-scale machine learning,” in 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), 2016, pp. 265–283. [Online]. Available: https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf

4. Z. Zhang and M. R. Sabuncu, “Generalized cross entropy loss for training deep neural networks with noisy labels,” in Proceedings of the 32nd International Conference on Neural Information Processing Systems, ser. NIPS’18. Red Hook, NY, USA: Curran Associates Inc., 2018, p. 8792–8802.

5. R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-cam: Visual explanations from deep networks via gradient-based localization,” in 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 618–626.

References

6. K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1409.1556

7. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2818–2826.

8. C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, “Inception-v4, inception-resnet and the impact of residual connections on learning,” in Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, ser. AAAI’17. AAAI Press, 2017, p. 4278–4284.

9. G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2261–2269.

10. F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1800–1807.

11. M. A. Akhloufi and M. Chetoui, “Chest XR Covid-19 detection,” https://cxr-covid19.grand-challenge.org/, August 2021, online; accessed September 2021.

References

12. B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, “Learning transferable architectures for scalable image recognition,” in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 8697–8710.

13. Zeiler M.D., Fergus R. (2014) Visualizing and Understanding Convolutional Networks. In: Fleet D., Pajdla T., Schiele B., Tuytelaars T. (eds) Computer Vision – ECCV 2014. ECCV 2014. Lecture Notes in Computer Science, vol 8689. Springer, Cham. https://doi.org/10.1007/978-3-319-10590-1_53

14. Yaroslav Ganin, Victor Lempitsky, “Proceedings of the 32nd International Conference on Machine Learning”, PMLR 37:1180-1189, 2015.

Thank You!

(For more queries contact: jpal.cs@gm.rkmvu.ac.in or nlpl931@gmail.com)