COVID19 Diagnosis using AutoML from 3D CT scans

Talha Anwar
Independent Researcher
chtalhaanwar@gmail.com

Abstract

Coronavirus is a pandemic that affects the respiratory system causing cough, shortness of breath, and death in severe cases. Polymerase chain reaction (PCR) tests are used to diagnose coronavirus. The false-negative rate of these tests is high, so there needs a supporting method for an accurate diagnosis. CT scan provides a detailed examination of the chest to diagnose COVID, but a single CT scan comprises hundreds of slices. Expert and experienced radiologists and pulmonologists can diagnose COVID from these hundreds of slices, but this is very time-consuming. So an automatic artificial intelligence (AI) based method is required to diagnose coronavirus with high accuracy. Developing this AI-based technique requires a lot of resources and time, but once it is developed, it can significantly help the clinicians. This paper used an Automated machine learning (AutoML) technique that requires fewer resources (optimal architecture trials) and time to develop, resulting in the best diagnosis. The AutoML models are trained on 2D slices instead of 3D CT scans, and the predictions on unknown data (slices of CT scan) are aggregated to form a prediction of 3D CT scan. The aggregation process picked the most occurred case, whether COVID or non-COVID from all CT scan slices and labeled the 3D CT scan accordingly. Different thresholds are also used to label COVID or non-COVID 3D CT scans from 2D slices. The approach resulted in accuracy and F1-score of 89% and 88%, respectively. Implementation is available at github.com/talhaanwarch/mia-covid19

1. Introduction

COVID-19 is a pandemic disease that mainly affects the respiratory system. The best way to prevent COVID is to isolate the infected peoples as it spread from patients to others very rapidly. This disease is usually diagnosed using reverse transcription-polymerase chain reaction (RT-PCR) test, but research showed that suspected patients should be examined with CT scans due to the high false-negative rate of PCR test. If a CT scan report is positive, no matter PCR is negative; the patient should not be removed from isolation [17]. Studies also showed that patients should not be diagnosed using CT scan only; instead, PCR test should also be performed to enhance the confidence level [17]. CT-scan can be used as a quick diagnostic method to categorize patients into "probably positive" and "probably negative" cohorts [4]. The bottom line is to use both PCR and CT-scan for the correct diagnosis of COVID patients.

Several studies have been performed to diagnose COVID from CT-scan images [1, 2]. Researchers tried to extract features from CT scans using convolution layers and fed these features to the recurrent neural networks to maintain the connection between slices[15, 14, 16]. In this section, only studies performed on the dataset under study are considered. D. Kollias *et al.,* introduced MIA-COV19 (COV19-CT-DB) dataset (dataset used for this study) and achieve a baseline macro F1-score of 70% using 3D CNN-RNN network [13]. Miron *et al.,* used Inflated 3D ResNet50 on 3D image size of 128x256x256. The authors used fold cross-validation and label-smoothing with cross-entropy and Sharpness Aware Minimization to avoid overfitting in 3D models [19]. Hsu *et al.,* performed slice level and CT scan level classification. For slice level, Swin transformer to get the different distribution of positive and negative classes and significant slices from the middle of CT-scan. Wilcoxon signed-rank test is used to find positive and negative examples during inference. For 3D CT scan classification, authors introduced CT scan-Aware Transformer (CCAT comprised of Within-Slice-Transformer (WST) and Between-Slice-Transformer (BST) [9]. W.Tan and J.Liu used 3D CNN with BERT to extract features and MLP classifiers to classify the data [24]. Linag proposed a sampling approach to sample fix number of slices from all 3D scans. The author used CNN comprised of squeeze excitation network to extract features and passed the features to transformer block and fully connected layer for classification [18]. Trinh ad Nguyen used ResNet and DenseNet in parallel to extract features and fully connected blocks for classification [26]. Qi *et al.,* used 3D RegNet with different optimizer and learning rate [20]. Teli introduced a custom CNN for classification [25]. Gao used vision transformer

Author	F1-score	Technique
Miron [19]	90.06 %	Inflated 3D ResNet50
Hsu [9]	88.74 %	CT scan-Aware Transformer
Tan [24]	88.22 %	BERT and MLP
Our Approach	87.77 %	AutoML
Liang [18]	78.86 %	SE CNN and transformer
Trinh [26]	78.13 %	Res Dense Net
Qi [20]	71.83 %	3D-RegNet
Teli [25]	70.86 %	TeliNet (custom CNN)
Gao [3]	70.5 %	COVID-ViT
Kollias [13]	70%	CNN RNN

Table 1. Macro F1-score on test set using COV19-CT-DB database

Data		Train set	Validation set
COVID	2D slice	153681	35016
	3D image	690	165
Non COVID	2D slice	181991	40516
	3D image	780	209

Table 2. Number of 2D and 3D images in train and validation data

(ViT) based on attention models and DenseNet to classify COVID, and non-COVID CT scan images [3]. The current paper discussed the AtuoML approach to diagnose COVID from CT-scan images. Table 1 shows the technique and macro F1-score on the COV19-CT-DB gold standard test database.

2. Methodology

2.1. Dataset

The COV19-CT-DB database is used in this study and is composed of three sets *i.e.* sets training, validation, and test data [13]. There are 1560, 374, and 3455 3D scans in training, validation, and test set, respectively. Each 3D scan ranges from 50 to 700 2D slices. Training data has 690 COVID cases and 780 non-COVID cases. Similarly, validation data has 165 3D CT scans of COVID patients and 209 CT scans of non-COVID subjects. The test set has 3455 3D CT scans with unknown labels that need to be predicted. The dataset is labeled by four specialists, two radiologists, and two pulmonologists instead of relying on PCR tests.

2.2. Deep Learning

Instead of using a 3D convolution neural network (CNN), 2D CNN is used to classify COVID vs. non-COVID cases. 2D CNNs are used because of limited GPU resources, as 3D CNNs require a high-end GPU system for training. Second, the slices in 3D images are not of the same length. It ranges from 50 to 700 slices in each 3D scan. So to use 3D CNNs, CT scans need to be truncated or padded. There are chances of noise addition while padding and information loss while truncating the data. So, 2D CNNs are trained on slice level instead of 3D volume level. Evaluation is made on slice level as well as the 3D volumetric level validation data. For 3D level prediction, predictions are made on 2D slices, and then the most occurred prediction class is assigned as 3D CT scan prediction. For example, if a CT-scan has 100 slices, 51 are predicted as COVID and 49 as non-COVID, the 3D image is labeled COVID. The threshold level technique is also applied. For example, if 1% of slices are predicted as COVID, the CT-scan is labeled as COVID. The threshold value of 1%, 5%, 10%. 20%, 30%, 40%, and 50% are used to classify the validation data, and the threshold value that yielded the best results is used to produce test predictions.

2.3. AutoML

AutoML powered by AutoGluon is used to carry out the experiments for the classification of COVID vs. non-COVID 3D CT scans [5]. AutoML makes the classification pipeline automatic, avoiding the hustle of preprocessing and hyper-parameters tuning. For experimentation, Tesla P100 GPU is used. The batch size is set to 16, and the learning rate to 0.01. The number of epochs is set to 15, but if training time reached 8 hours, training is stopped. This helps to train more models in less time. The only preprocessing applied is to resize all images to 224*244. To get a prediction of 3D scans from 2D slices, different threshold techniques are used. In the max threshold technique, the class label is chosen based on maximum occurrence of a class. In percentage threshold, the class label is chosen if positive predictions are more than a specific percentage, then CT scan is labeled as positive.

2.4. Deep learning Architectures

Different 2D CNN pre-trained models are used such as

2.4.1 VGG

VGG is one of the oldest deep learning architecture proposed in 2014 and has a depth ranging from 16-19 [23]. VGG19_bn is 19th layers modified architecture with batch normalization layer added [22]. The concept of batch normalization is introduced in 2015, and almost all deep learning architectures after that used this layer [12]. Batch normalization layers reduce training time and overfitting issues of deep architectures.

2.4.2 ResNet

ResNet family of deep learning architectures came out in 2015 [6]. ResNet152 has dept up to 152 layers, but its complexity is quite less as compared to VGG models. ResNet introduced a concept called a shortcut connection that skips one or more layers and helps to avoid vanishing gradient problems caused by deeper length.

2.4.3 DenseNet

Almost after eight months of ResNet, DenseNet models were introduced [11]. DensetNet used a block called a Dense block in which a layer is connected to all subsequent layers in that block. DenseNet is deeper and has three times fewer parameters than ResNet.

2.4.4 MobileNet

MobileNet models were basically introduced for mobile devices with low computational powers in 2017 [8]. MobileNet is based on depth-wise separable convolution. It is a combination of depth-wise convolution and separable convolution. In this approach, 1x1 convolution is applied to all channels of input data. MobileNetV3 introduced in 2019 used AutoML to create best possible architecture [7]. It also used Squeeze and Excitation Networks (SENet) introduced in 2017 [10]. Squeeze is global average pooling. Excitation is two fully connected layers having RelU in between and Sigmoid at the end.

2.4.5 Se_ResNext

ResNext is similar to ResNet [27]. In ResNet, channels size is reduced by using 1x1 by convolution layer, for example, from 256 to 64 and then back to 256. In ResNext, channels size is reduced from 256 to 32 features maps, each having four channels and then aggregated back to 256. Applying convolution of 4 channels is much cheaper than on 64 channels. Se_ResNext is introducing SENet architecture in ResNext.

2.4.6 ResNest

ResNet was introduced in 2020, and it is a modularized architecture, which applies channel-wise attention on different feature map groups. [28].

2.5. External data

A portion of large external data [21] is also used to test the generalization of the models. The dataset comprised of 9776 non-COVID and 2282 COVID slices of different CT scans.

3. Results

All the experimentation discussed below is performed using the NVIDIA TESLA P100 GPU. Fig 1 shows macro F1-score on validation data using different threshold levels *i.e.* max threshold and percentage threshold. 1% mean that if 1% COVID slices are present, then assign 3D CT-scan image as COVID case. Similarly, with other percentage threshold levels but in the max threshold technique, if the maximum number of predicted slices labels belongs to COVID, the 3D scan is assigned COVID else non-COVID. The worst validation score is obtained when a CT scan is labeled as COVID based on 1% occurrence of coronavirus diagnosed slices. Results are also poorer when CT scan is predicted corona positive based on 5% and 10% threshold level. Best results are obtained when the 3D scan is labeled positive based on 30% or above slices positive.

Table 3 shows the accuracy, precision, recall, and F1 score obtained on the validation set using the max threshold technique. The highest macro F1score and accuracy of 85% is achieved in ResNest14 architecture for 2D image data. On 3D CT-scan, ResNest14 lead to the highest accuracy and F1-score of 89% and 88%, respectively. The lowest score is obtained using DenseNet201, where the macro F1-score on the 2D slice level and 3D scan is 81% and 84%, respectively. The lowest precision is 81% when the MobileNetv3 model is evaluated on 2D slices, and the lowest recall is 81% using DenseNet201. Table 3 shows the model size on disk and time for training and inference on the test set. Inference time is computed on the test set having 3455 Ct-scans 3D images instead of the validation set. MobileNetV3 has the smallest models' weights size and took around 6 hours 21 minutes for training and 2 hours 37 mutes for inference. VGG19 with batch normalization has the largest size of 1 GB.

On external data macro F1 score of 76% and accuracy of 84% is achieved. Recall measures the model's ability to detect Positive samples, and the model achieved a recall of 75%. The test set uses max threshold techniques and the ensemble of all models, 78.78%, 87.77%, and 96.75%, F1 COVID, F1 non-COVID, and macro F1 score, respectively is achieved. On the other hand, using percentage threshold techniques F1-score is low as compared to the max threshold technique. 61.8%,74.4% and 80.08% macro F1 score is

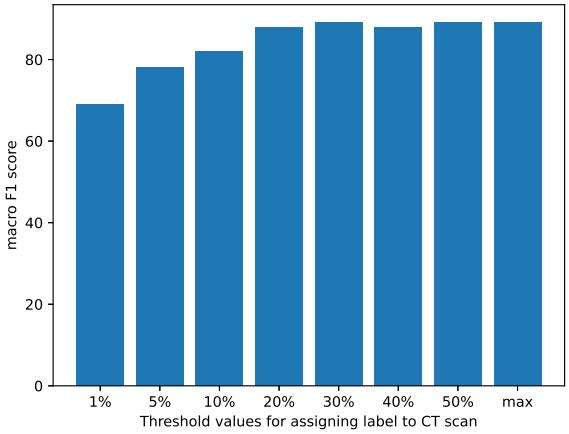

Figure 1. F1-score on validation data using different threshold values

	2D CT-scan slice				3D Volumetric Image			
Model name	**F1-score**	**Precision**	**Recall**	**Accuracy**	**F1-score**	**Precision**	**Recall**	**Accuracy**
ResNet152	80%	82%	80%	81%	85%	89%	85%	86%
DenseNet201	81%	83%	81%	82%	84%	88%	84%	86%
ResNest14	85%	86%	85%	85%	88%	90%	88%	89%
ResNext50	83%	83%	83%	83%	88%	89%	89%	88%
Se_ResNext50	84%	84%	83%	84%	87%	89%	87%	88%
MobileNetV3	80%	81%	88%	81%	86%	88%	86%	87%
VGG19_bn	84%	84%	84%	84%	87%	89%	87%	88%

Table 3. Results on the validation set using max threshold

Model name	**Model Size**	**Training Time**	**Inference Time**
ResNet152	472 MB	7 hours 41 minutes	3 hours 15 minutes
DenseNet201	166 MB	5 hours 21 minutes	2 hours 48 minutes
ResNest14	93 MB	7 hours 24 minutes	2 hours 22 minutes
ResNext50	203 MB	6 hours 31 minutes	2 hours 7 minutes
Se_ResNext50	222 MB	6 hours 10 minutes	2 hours 49 minutes
MobileNetV3	63 MB	6 hours 21 minutes	2 hours 37 minutes
VGG19_bn	1 GB	7 hours 56 minutes	2 hours 42 minutes

Table 4. Model size and execution time. Infer time is on test set

achieved on 10%, 20% and 30% threshold.

4. Conclusion

AutoML is the automatic way of doing any machine learning or deep learning task. We used the AutoGluon framework that leads to accurate diagnostic of COVID cased from 3D volumetric images. The advantage of AutoGluon (AutoML) is that there is no need to worry about preprocessing, architecture implementation, and evaluation. It does all process internally and yield good results with hyper-parameters selected. Using ResNest14 architecture, accuracy and F1-score of 89% and 88% are obtained. 87.77% macro F1 is achieved on the test set comprised of 3455 CT scan 3D images. Future work includes feeding input images of different sizes instead of fixed input of 224*224. Fixing the input size for all models results in loss of information as different models learned differently on different input sizes.

References

[1] Talha Anwar and Seemab Zakir. Deep learning based diagnosis of covid-19 using chest ct-scan images. In *2020 IEEE 23rd International Multitopic Conference (INMIC)*, pages 1–5, 2020.

[2] Hanqiu Deng and Xingyu Li. Ai-empowered computational examination of chest imaging for covid-19 treatment: A review. *Frontiers in Artificial Intelligence*, 4:96, 2021.

[3] Xiaohong Gao, Yu Qian, and Alice Gao. Covid-vit: Classification of covid-19 from ct chest images based on vision transformer models. *arXiv preprint arXiv:2107.01682*, 2021.

[4] Hester A Gietema, Noortje Zelis, J Martijn Nobel, Lars JG Lambriks, Lieke B Van Alphen, Astrid ML Oude Lashof, Joachim E Wildberger, Irene C Nelissen, and Patricia M Stassen. Ct in relation to rt-pcr in diagnosing covid-19 in the netherlands: a prospective study. *PloS one*, 15(7):e0235844, 2020.

[5] Jian Guo, He He, Tong He, Leonard Lausen, Mu Li, Haibin Lin, Xingjian Shi, Chenguang Wang, Junyuan Xie, Sheng Zha, et al. Gluoncv and gluonnlp: Deep learning in computer vision and natural language processing. *J. Mach. Learn. Res.*, 21(23):1–7, 2020.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. arxiv 2015. *arXiv preprint arXiv:1512.03385*, 2015.

[7] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1314–1324, 2019.

[8] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[9] Chih-Chung Hsu, Guan-Lin Chen, and Mei-Hsuan Wu. Visual transformer with statistical test for covid-19 classification. *arXiv preprint arXiv:2107.05334*, 2021.

[10] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[11] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[12] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal co-

variate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[13] Dimitrios Kollias, Anastasios Arsenos, Levon Soukissian, and Stefanos Kollias. Mia-cov19d: Covid-19 detection through 3-d chest ct image analysis. *arXiv preprint arXiv:2106.07524*, 2021.

[14] Dimitrios Kollias, N Bouas, Y Vlaxos, V Brillakis, M Seferis, Ilianna Kollia, Levon Sukissian, James Wingate, and S Kollias. Deep transparent prediction through latent representation analysis. *arXiv preprint arXiv:2009.07044*, 2020.

[15] Dimitrios Kollias, Athanasios Tagaris, Andreas Stafylopatis, Stefanos Kollias, and Georgios Tagaris. Deep neural architectures for prediction in healthcare. *Complex & Intelligent Systems*, 4(2):119–131, 2018.

[16] Dimitris Kollias, Y Vlaxos, M Seferis, Ilianna Kollia, Levon Sukissian, James Wingate, and Stefanos D Kollias. Transparent adaptation in deep medical image diagnosis. In *TAILOR*, pages 251–267, 2020.

[17] Inan Korkmaz, Nursel Dikmen, Fatma Oztürk Keleş, and Tayibe Bal. Chest ct in covid-19 pneumonia: correlations of imaging findings in clinically suspected but repeatedly rt-pcr test-negative patients. *Egyptian Journal of Radiology and Nuclear Medicine*, 52(1):1–9, 2021.

[18] Shuang Liang. A hybrid deep learning framework for covid-19 detection via 3d chest ct images. *arXiv preprint arXiv:2107.03904*, 2021.

[19] Radu Miron, Cosmin Moisii, Sergiu Dinu, and Mihaela Breaban. Covid detection in chest cts: Improving the baseline on cov19-ct-db. *arXiv preprint arXiv:2107.04808*, 2021.

[20] Haibo Qi, Yuhan Wang, and Xinyu Liu. 3d regnet: Deep learning model for covid-19 diagnosis on chest ct image. *arXiv preprint arXiv:2107.04055*, 2021.

[21] Mohammad Rahimzadeh, Abolfazl Attar, and Seyed Mohammad Sakhaei. A fully automated deep learning-based network for detecting covid-19 from a new and large lung ct scan dataset. *Biomedical Signal Processing and Control*, 68:102588, 2021.

[22] Marcel Simon, Erik Rodner, and Joachim Denzler. Imagenet pre-trained models with batch normalization. *arXiv preprint arXiv:1612.01452*, 2016.

[23] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[24] Weijun Tan and Jingfeng Liu. A 3d cnn network with bert for automatic covid-19 diagnosis from ct-scan images. *arXiv preprint arXiv:2106.14403*, 2021.

[25] Mohammad Nayeem Teli. Telinet, a simple and shallow convolution neural network (cnn) to classify ct scans of covid-19 patients. *arXiv preprint arXiv:2107.04930*, 2021.

[26] Quoc Huy Trinh and Minh Van Nguyen. Custom deep neural network for 3d covid chest ct-scan classification. *arXiv preprint arXiv:2107.01456*, 2021.

[27] Saining Xie, Ross B Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. corr abs/1611.05431 (2016). *arXiv preprint arXiv:1611.05431*, 2016.

[28] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Haibin Lin, Zhi Zhang, Yue Sun, Tong He, Jonas Mueller, R Manmatha, et al. Resnest: Split-attention networks. *arXiv preprint arXiv:2004.08955*, 2020.