The Effect of Artificial Amalgamates on Identifying Pathogenesis

This paper was downloaded from TechRxiv (<https://www.techrxiv.org>).

LICENSE

CC BY 4.0

SUBMISSION DATE / POSTED DATE

10-10-2021 / 15-10-2021

CITATION

Naveen, Philip (2021): The Effect of Artificial Amalgamates on Identifying Pathogenesis. TechRxiv. Preprint.
<https://doi.org/10.36227/techrxiv.16779292.v1>

DOI

[10.36227/techrxiv.16779292.v1](https://doi.org/10.36227/techrxiv.16779292.v1)

The Effect of Artificial Amalgamates on Identifying Pathogenesis

Philip Naveen

October 10, 2021

Abstract

The purpose of this research was to define acceleration in diagnostic procedures for airborne diseases. Airborne pathogenicity can be troublesome to diagnose due to intrinsic variation and overlapping symptoms. Coronavirus testing was an instance of a flawed diagnostic biomarker. The levels of independent variables (IV) were vanilla, sparse, and dense amalgamates formed from multilayer perceptrons and image processing algorithms. The dependent variable (DV) was the classification accuracy. It was hypothesized that if a dense amalgamate is trained to identify Coronavirus, the accuracy would be the highest. The amalgamates were trained to analyze the morphological patches within radiologist-verified medical imaging retrieved from online databanks. Using generative cross-validations, the DV was consulted for each amalgamate. Self-calculated t-tests supported the research hypothesis, with the dense amalgamate achieving 85.37% correct classification rate. The null hypothesis was rejected. Flaws within the databanks were possible sources of error. A new algorithm developed here identified Coronavirus, Mycobacterium, Carcinoma, and Pneumonia from 96-99% accuracy. Future enhancements involve tracking osteopenia/osteoporosis with the algorithm.

1 Introduction

1.1 Background

Significant medical advances in the past century have led to improvements to diagnostic procedures and technology, such as biomarkers, which are biological alterations within organisms indicative of pathogenicity. They vary in cost and implementation. While biomarkers are dependable, they are costly to develop, and require extended testing. The low success rate of manufacturing successful biomarkers accompanied by the long timeframe of development leads to the spread of pandemics [1].

Coronavirus has three spike protein variants: SARS-CoV, SARS-CoV-2, and MERS-CoV [1]. The main side effects are very similar to Influenza, Tuberculosis, and Pneumonia on the surface [2]. Throughout 2020 and early 2021, the United States had incredible difficulty differentiating infected and healthy individuals. Hospitals were overloaded beyond their carrying capacities. Many people waited weeks for their results due to the extensive analysis of the samples required for proper diagnosis [3]. Luckily, the vaccinations are currently fixing the issue. Other pandemics could yield similar effects due to the slowness of testing. The Delta and Lambda variants are examples of how intrinsic mutation can be troublesome for developing biomarkers

[2, 3]. If better diagnostic procedures could be implemented, the curve-flattening process would be much easier.

Generally, airborne diseases have prominent pathogenic effects in the lungs [2]. Computed tomography (CT) and x-ray scans are used by radiologists to image anatomical structures within humans non-invasively [4]. CT scans and x-rays can show inflammatory patches in those regions [5]. Unfortunately, there are only an estimated 34,000 radiologists in the USA, meaning that using the aforementioned diagnostic procedures may be difficult depending on the area. Theoretically, if a computational application can identify Coronavirus using medical imaging, then testing for diseases will become more widespread and efficient [6]. An artificial intelligence and computational biology approach may be successful.

1.2 Experimental Design

The independent variable (IV) is the type of amalgamate used. Amalgamates used here are different permutations of multilayer perceptrons (MLP) concatenated with image processing via graph theory and computer code. MLPs, also known as neural-networks, are a framework of neurons connected with synaptic weights being finetuned using backpropagation, which is a process in which every synaptic weight is adjusted across many iterations [7]. Vanilla MLPs are the regular formation of backpropagation, which involves the calculation of gradient and loss to optimize the arrangement to solve problems such as regression and classification. Sparse variants have certain nodes with their synapses severed, and dense variants have additional layers [7]. The processing algorithms here cropped the image and lessened the pixel density, since those operations are conventionally used when classifying images [8].

The dependent variable (DV) is the correct classification rate of Coronavirus returned when classifying Coronavirus-infected lungs. It will be represented as the accuracy. The DV will be measured using an advanced cross-validation simulation using data fabricated with an DAE, which is an unsupervised learning framework [9]. A confusion matrix will be used to assess the model [10]. For multi-class identification, confusion matrices can be fairly complex, so a one-vs-everything approach was taken here to simplify the statistics [10]. The four cases will be true positives, true negatives, false positives, and false negatives. The calculation of accuracy is the true positives and true negatives compared against all the predictions [3].

The purpose of this experiment is to determine the effect of different computational amalgamates on the diagnosis of Coronavirus by analyzing CT scans and x-rays of the lung. The research hypothesis is that if the dense amalgamate is trained on CT scans and x-rays of Coronavirus in the lungs, then it will score higher accuracy than the vanilla and sparse amalgamates. This is based on the fact that the dense MLPs have shown to be successful in image classification in the past. If the identification of Coronavirus is successful as was in other studies, then computer-based diagnosis may be promising in differentiating other airborne diseases [6, 5]. The knowledge gained from this experiment can be applied in the creation of a new hybrid multivariate model to accelerate diagnostic procedures for various diseases. The architecture of the simulation system and proposed model can be seen in figure 1 and figure 2.

2 Procedure

One Acer laptop with an i7 processor running the 64-bit edition of Windows was obtained. Using Oracle's Virtual Machine (VM) software, a Ubuntu distribution of Linux was forged

with a Windows host. Python3 was installed along with JupyterLab on the VM. Using Git-Bash, a series of Pip installations were used to acquire linear algebra and machine-learning dependencies.

Through various databanks available on Kaggle, medical CT scans and x-rays of Coronavirus were downloaded to the VM. Using Python3, image processing algorithms were implemented. The images were cropped to the same size, and had their visual acuity reduced to 100x100 pixels per image. The rank-3 tensors were reduced to rank-2 by eliminating the red-green-blue color channels. The tensors were subsequently compared for mean squared error.

Figure 1: Simulation System using Sparse Matrices and Autoencoders

Vanilla, sparse, and dense MLP models were concatenated to the processing algorithms to form the three amalgamates. Using the images, the amalgamates were trained for ten epochs. A specialized simulation was configured to test the amalgamate performance, and the complete workflow of the software simulation can be seen in figure 1. A denoising autoencoder (DAE) and sparse matrix overlays augmented images for simulation. A one-vs-everything adaptation of a confusion matrix in a 2x2 format was programmed to calculate prediction statistics. Each amalgamate had its performance tested via the simulation system. The data was recorded in the lab notebook. The most accurate of them was determined using t-tests.

Figure 2: Multivariate Model using Semi-Supervised Deep-Learning

A new hybrid multivariate model was developed here. The morphological patch representations of medical images were analyzed by augmenting the basic dense amalgamate with machine-learning algorithms. A feature extractor was fabricated using a convolutional generative adversarial network (GAN). The extractor was developed iteratively based on lightweight pretrained networks (see appendix) [11]. An isolation forest was configured to eliminate anomaly feature sets extracted from images through unsupervised dimensionality reduction. Through testing many different classification algorithms, an ideal design was determined (see appendix). The support vector machine (SVM) was chosen.

It was found that through splitting a multivariable problem into many binary ones, the computational cost could be lowered and the accuracy could be increased significantly. Through observing a medical professional while shadowing and researching the healthcare system, it was found that developing this algorithm in accordance with electronic health records (EHRs), which are software used by physicians to document information digitally. Through weeks of testing and training the framework, the algorithm was developed to be as integrative as possible (see appendix). It was programmed using a simplified variant of the software development life cycle to ensure streamline development. The final architecture of the engineered model can be seen in figure 2.

Using the same simulation process for measuring Coronavirus diagnosis, this new hybrid

model was tested. It was also expanded to identify Tuberculosis, Carcinoma, and Pneumonia. Further tests studied its capability in differentiating subtypes and malignancy of Carcinomas and identifying the pathogenicity sources of Pneumonia. These results were also recorded in the lab notebook. The amalgamates, simulation, and hybrid model were bidirectionally saved as PB and H5 files while the VM was disconnected from the host and were uploaded onto GitHub for future modification or cross-platform testing. This project was conducted under adult supervision and approval. For programming at night, 75% of blue light was filtered from monitor screens to prevent eye strain for safety reasons. All data used was open-source and retrieved from a source approved by McAfee.

3 Results

Table 1: The Effect of Artificial Amalgamates on Disease Diagnosis

Descriptive Information	Amalgamate		
	Vanilla	Sparse	Dense
Mean	81.69%	74.51%	85.37%
Range	1.60%	2.17%	2.75%
Minimum Value	81.07%	73.59%	84.13%
Maximum Value	82.67%	75.66%	86.88%
Variance	2.06e(-5)	2.98e(-5)	5.73e(-5)
Standard Deviation	4.54e(-3)	5.46e(-3)	7.57e(-3)
1SD	81.66-81.69	74.50-74.52	85.36-85.38
2SD	81.68-81.69	74.49-74.52	85.35-85.39
3SD	81.68-81.70	74.49-74.53	85.35-85.39
Number	25	25	25
Results of the t-test			
Vanilla vs. Sparse $t = 359.16$ $p < 0.01$			
Vanilla vs. Dense $t = 167.25$ $p < 0.01$			
Sparse vs. Dense $t = 475.79$ $p < 0.01$			
At $df = 48$, $\alpha = 0.01$, table- $t = 2.407$ for significance

Figure 3: The Descriptive and Inferential Statistical Analysis of The Effect of Artificial Amalgamates on Identifying Pathogenesis

The results of the procedure can be seen in figure 3. It was hypothesized the dense amalgamate would yield the highest accuracy when identifying Coronavirus. The mean comparisons of the vanilla (81.69% accuracy), sparse (74.51% accuracy), and dense (85.37% accuracy) amalgamates implied that the dense amalgamate had the highest accuracy, while the sparse one had the lowest. The variances and standard deviation for each level of independent variables were

calculated. All six values were very low, which implies the data is precise. There were some outliers when the 1SD, 2SD, and 3SD ranges were consulted. However, those were only considered outliers due to standard deviations being so low. These differential statistics showed the data was statistically significant.

Three t-tests calculated at a 0.01 level of significance with 48 degrees of freedom were used to compile the data collected. A null hypothesis stating the three amalgamates would have no significant difference in accuracy was formulated. A table-t value of 2.407 was used.

All comparisons of vanilla vs. sparse, vanilla vs. dense, and sparse vs. dense were statistically significant because the table-t value was less than all calculated-t values. The data was statistically significant, with a greater than 99% confidence in the results being void of chance and error. The null hypothesis was rejected in favor of the research hypothesis. The trend was the increased synaptic weights within the networks better compiled the images regardless of their pixel values.

A new multivariate model fabricated here had the best results. The bootstrap distribution, receiver operating characteristic, and loss function visualizations were omitted because of format restrictions. The model scored accuracy, sensitivity, specificity, and precision greater than 90% for identifying Coronavirus, Tuberculosis, Carcinoma and Pneumonia. The model was then also used to differentiate the subtypes and malignancy of Carcinomas and Pneumonia infection types. Accuracies between 96-99% were achieved for these situations.

Extraneous confusion matrix statistics were omitted from this paper because of the space limit. The model had very low false positive and false negative rates and had state-of-the-art performance. The model was also capable of parsing thousands of images in a timeframe of under an hour on both i3 and i7 computers without external graphics processing devices or cloud server access.

4 Conclusions

4.1 Value and Limits

A new multivariate model for identifying diseases was developed here by augmenting a dense amalgamate. The robust algorithm had state-of-the-art results in Coronavirus classification. It also identified Tuberculosis, Carcinoma and Pneumonia from 96-99% accuracy. Additionally, it was adept in differentiating lung cancer subtypes and malignancy as well as identifying the source of chest infections from 96-99% accuracy.

Possible sources of error in this project come from the databanks because there could have been errors in sorting them before publication. Lopsided data may have resulted in mild overfitting [12]. The DAEs could have been further optimized to have lower error, and perhaps generators from the GANs would have been a better fit for the simulation system [13, 14].

4.2 Future Applications and Research

This type of generative cross-validation may provide better insight on model performance as the entropy of the training set is controllable. Gated recovery units and long-short term memory frameworks are variants of recurrent neural-networks used to track time series data [15, 16] that could be used to extend the multivariate model for tracking disease progression.

Examples would be Alzheimer's in the brain or Osteoporosis in the vertebral column [4, 17]. Testing the model in electronic health records [18] to study "differential diagnosis" in clinics [19] would also be an interesting future study.

References

- [1] He F, et al. (2020, March 2nd). Coronavirus disease 2019: What we know? Wiley Journal of Medical Virology. Retrieved on August 13th 2021 from doi: 10.1002/jmv.25766 "PDCA12-70 data sheet," Opto Speed SA, Mezzovico, Switzerland.
- [2] Liu Q, et al. (2020, October 21st). A Two-Dimensional Sparse Matrix Profile DenseNet for COVID-19 Diagnosis Using Chest CT Images. IEEEAccess. Retrieved on June 9th 2021 from URL: <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9268138>
- [3] Simoneux R. and Shafer S. (2020). Current State of COVID Testing. In the Know, 84, 8-9. Retrieved on August 13th 2021 from doi: 10.1097/01.asm.0000716944.70366.26
- [4] Sapthagirivasan V. and Anburajan M. (2013). Diagnosis of osteoporosis by extraction of trabecular features from hip radiographs using support vector machine: An investigation panorama with DXA. Computers in biology and medicine, 43(11). Retrieved on August 13th 2021 from doi: 10.1016/j.combiomed.2013.09.002
- [5] Wang L, et al. (2020, Nov 11th). COVID-Net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images. Scientific Reports, 10(1). Retrieved on August 13th 2021 from doi: 10.1038/s41598-020-76550-z
- [6] Rahimzadeh M. and Attar A. (2020, May 21st). A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of Xception and ResNet50V2. Informatics in Medicine Unlocked, 19, 360-369. Retrieved on August 13th 2021 from doi: 10.1016/j.imu.2020.100360
- [7] Baker B, et al. (2017, March 22nd). Designing Neural Network Architectures using Reinforcement Learning. arXiv of Cornell University. Retrieved on June 9th 2021 from URL: <https://arxiv.org/abs/1611.02167> R. E. Sorace, V. S. Reinhardt, and S. A. Vaughn, "High-speed digital-to-RF converter," U.S. Patent 5 668 842, Sept. 16, 1997.
- [8] Albawi S. and Mohammed T. (2017, August 21st). Understanding of a Convolutional Neural Network. International Conference on Engineering and Technology (ICET). Retrieved on June 9th 2021 from doi: 10.1109/ICEngTechnol.2017.8308186
- [9] Xu W, et al. (2019). Adversarially Approximated Autoencoder for Image Generation and Manipulation. IEEE Transactions on Multimedia, 21, 2387-2396. Retrieved on August 15th 2021 from doi: 10.1109/TMM.2019.2898777
- [10] Krizhevsky A, et al. (2012). ImageNet classification with deep convolutional neural networks. Communications of the ACM, 60, 84 - 90. Retrieved on August 15th 2021 from doi: 10.1145/3065386
- [11] Rice L, et al. (2020). Overfitting in adversarially robust deep learning. ICML. Retrieved on August 15th 2021 from URL: <https://arxiv.org/pdf/2002.11569.pdf>

- [12] Zhang H, et al. (2017). StackGAN: Text to Photo-Realistic Image Synthesis with Stacked Generative Adversarial Networks. 2017 IEEE International Conference on Computer Vision (ICCV), 5908-5916. Retrieved on August 15th 2021 from doi: 10.1109/ICCV.2017.629
- [13] Radford A, et al. (2016). Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. arXiv of Cornell University. Retrieved on August 15th 2021 from URL: <https://arxiv.org/pdf/1511.06434.pdf>
- [14] Hochreiter S. and Schmidhuber J. (1997, November 15th). Long Short-Term Memory. *Neural Computation*, 9, 1735-1780. Retrieved on August 13th 2021 from doi: 10.1162/neco.1997.9.8.1735J. Padhye, V. Firoiu, and D. Towsley, "A stochastic model of TCP Reno congestion avoidance and control," Univ. of Massachusetts, Amherst, MA, CMPSCI Tech. Rep. 99-02, 1999.
- [15] Horvath A, et al. (2018). Spinal Cord Gray Matter-White Matter Segmentation on Magnetic Resonance AMIRA Images with MD-GRU. arXiv of Cornell University. Retrieved on August 15th 2021 from doi: 10.1007/978-3-030-13736-6
- [16] Lee J, et al. (2018, July 13th). Osteoporosis detection in panoramic radiographs using a deep convolutional neural network-based computer-assisted diagnosis system: a preliminary study. *Dento maxillo facial radiology*, 20170344. Retrieved on August 15th 2021 from doi: 10.1259/dmfr.20170344
- [17] Arbabshirani M, et al (2018). Advanced machine learning in action: identification of intracranial hemorrhage on computed tomography scans of the head with clinical workflow integration. *NPJ Digital Medicine*, 1. Retrieved on August 15th 2021 from doi: 10.1038/s41746-017-0015-z S. Zhang, C. Zhu, J. K. O. Sin, and P. K. T. Mok, "A novel ultrathin elevated channel low-temperature poly-Si TFT," *IEEE Electron Device Lett.*, vol. 20, pp. 569–571, Nov. 1999.
- [18] Radford A, et al. (2016). Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. arXiv of Cornell University. Retrieved on August 15th 2021 from URL: <https://arxiv.org/pdf/1511.06434.pdf>
- [19] Shen Z, et al. (2019). A systematic review on open source clinical software on GitHub for improving software reuse in smart healthcare. *Applied Sciences*, 9, 150. Retrieved on August 15th 2021 from doi: 10.3390/APP9010150
- [20] Hani C, et al. (2020). COVID-19 pneumonia: A review of typical CT findings and differential diagnosis. *Diagnostic and Interventional Imaging*, 101, 263-268. Retrieved on August 15th 2021 from doi: 10.1016/j.diii.2020.03.014 M. Shell. (2002) IEEEtran homepage on CTAN. [Online]. Available: <http://www.ctan.org/tex-archive/macros/latex/contrib/supported/IEEEtran/>
- [21] Xu J, et al. (2019, July 11th). Three-way confusion matrix for classification: A measure driven view. *Information Science*, 507, 772-794. Retrieved on August 13th 2021 from doi: 10.1016/J.INS.2019.06.064
- [22] Liu Y, et al. (2017). A method for multi-class sentiment classification based on an improved one-vs-one (OVO) strategy and the support vector machine (SVM) algorithm. *Information Sciences*, 394, 38-52. Retrieved on August 13th 2021 from doi: 10.1016/j.ins.2017.02.016

- [23] Xia Q, et al. (2020). Roadmap on emerging hardware and technology for machine learning. Nanotechnology. Nanotechnology, 32. Retrieved on August 15th 2021 from doi: 10.1088/1361-6528/aba70f
- [24] Mensi A, et al. (2021). Proximity Isolation Forests. 2020 25th International Conference on Pattern Recognition (ICPR), 8021-8028. Retrieved on August 15h 2021 from doi: 10.1109/ICPR48806.2021.9412322
- [25] Zhang J, et al. (2017, January 4th). Diagnostic Method of Diabetes Based on Support Vector Machine and Tongue Images. BioMed Research International. Retrieved on August 13th 2021 from doi: 10.1155/2017/7961494

5 Appendix

Prediction	Count
True Positives	539
True Negatives	5074
False Positives	36
False Negative	0

99.36% accuracy, 99.29% specificity, 100.00% sensitivity, 93.74% precision

Figure 4: Coronavirus Prediction Table

Prediction	Count
True Positives	3347
True Negatives	4763
False Positives	149
False Negative	114

96.93% accuracy, 97.09% specificity, 96.71% sensitivity, 95.74% precision

Figure 5: Tuberculosis Prediction Table

Prediction	Count
True Positives	861
True Negatives	403
False Positives	13
False Negative	0

98.81% accuracy, 96.86% specificity, 100.00% sensitivity, 98.13% precision

Figure 6: Carcinoma Prediction Table

Prediction	Count
True Positives	4214
True Negatives	5006
False Positives	67
False Negative	59

98.65% accuracy, 98.68% specificity, 98.62% sensitivity, 98.43% precision

Figure 7: Pneumonia Prediction Table

Prediction	Count
True Positives	118
True Negatives	405
False Positives	2
False Negative	11

97.57% accuracy, 99.51% specificity, 91.47% sensitivity, 98.33% precision

Figure 8: Adenocarcinoma Prediction Table

Prediction	Count
True Positives	102
True Negatives	411
False Positives	3
False Negative	4

98.65% accuracy, 99.28% specificity, 96.23% sensitivity, 97.14% precision

Figure 9: Squamous Cell Carcinoma Prediction Table

Prediction	Count
True Positives	69
True Negatives	406
False Positives	3
False Negative	10

97.34% accuracy, 99.27% specificity, 87.34% sensitivity, 97.34% precision

Figure 10: Large Cell Carcinoma Prediction Table

Prediction	Count
True Positives	120
True Negatives	396
False Positives	20
False Negative	0

96.27% accuracy, 95.19% specificity, 100.00% sensitivity, 85.71% precision

Figure 11: Benign Carcinoma Prediction Table

Prediction	Count
True Positives	551
True Negatives	402
False Positives	21
False Negative	14

96.46% accuracy, 95.04% specificity, 97.52% sensitivity, 96.33% precision

Figure 12: Malignant Carcinoma Prediction Table

Prediction	Count
True Positives	1340
True Negatives	4944
False Positives	129
False Negative	2

97.96% accuracy, 97.46% specificity, 99.85% sensitivity, 91.22% precision

Figure 13: Viral Pneumonia Prediction Table

Prediction	Count
True Positives	4271
True Negatives	4768
False Positives	305
False Negative	2

96.72% accuracy, 93.99% specificity, 99.95% sensitivity, 93.33% precision

Figure 14: Bacterial Pneumonia Prediction Table

5.1 First Annexation: Denoising Autoencoders for Simulating Generative Cross-Validations

DAEs are a common form of unsupervised learning that operate by encoding the data and then artificially reconstructing the original array via a MLP. It was found that these are used to refine and manipulate data. The design of the ones used here were based on a denoising architecture that was repurposed for generation.

These networks were trained using backpropagation on the processed images of diseased scans. Healthy scans were drawn and given a randomly generated sparse matrix overlay in an attempt to artificially create noise within the scans. These deformed scans were then subsequently parsed through their respective DAE and refined into artificial scans in a pseudo-generation approach.

Paired with a confusion matrix, the accuracy, specificity, sensitivity, and precision were calculated using the true positive, false positive, true negative, and false negative values. Generally, confusion matrices assess all predictions. However, calculating such metrics even in a 3x3 formation is very complex [20]. To simplify the prediction statistics, it was also refined into the one-vs-everything adaptation [21]. The final design of the simulation was reduced from a single 11x11 matrix to eleven separate 2x2 matrices concatenated to their respective models. A GAN was suggested in the conclusion. These are actually two MLPs working together. This is described in the second annexation.

5.2 Second Annexation: Design of Convolutional Generative Adversarial Networks for Feature Extraction

In other research involving disease classification, a common trend is the use of convolutional MLPs to enhance model performance [2, 4, 5, 6]. This approach was harnessed here as well. However, these applications commonly utilize pretrained networks such as AlexNet, GoogLeNet, ResNet50, ResNet101, InceptionV3, VGG16, VGG19, etc. These engines are seemingly overcomplicated for the problem of disease classification via medical imaging since the databanks were all greyscale. A possible reason these are used so commonly is because they are readily available in MATLAB, which is one of the biggest deep-learning toolboxes available commercially.

X-rays and CT scans are considered rank-2 tensors opposed to rank-3 tensors since they are greyscale with one color channel opposed to RGB with three color channels [2]. Feature extraction for this type of data using the aforementioned networks would be overkill. The computational complexity would be extraneous. Therefore, it was determined a custom feature extractor specified for this problem would be ideal. The basic criteria for it was:

1. The extraction would have to be lighter than the previously mentioned pretrained network, which range from 8-101 convolutional layers deep.
2. The framework for this learning would be unsupervised.
3. The correct classification rate of Coronavirus must be higher than that of the basic dense amalgamate.

Using the aforementioned software development life cycle, a conservative extractor enhanced in conjunction with the Keras Tuner optimization tool. The process was iterative, with many minor adjustments being made to the architecture over weeks of working. This method was

used to finetune the synaptic weights between the convolution and pooling layers [7]. In other research, it was found that ADAM was ideal for optimizing such a classification task with sparse categorical crossentropy loss [2, 4]. Such a framework was modeled here with the loss function being swapped with regular binary crossentropy when the structure was refined into a one-vs-everything adaptation. Various activation functions were tested. Through this process the most successful combination was:

1. Swish between the convolution and pooling functions.
2. Rectified Linear for hidden dense layers.
3. Softmax for probability distribution in the output layer.

Older activation functions such as Sigmoid and TanH were tested as well, but didn't show as much potential. The model's speed was tested on laptops running i3 (2 cores) and i7 (4+ cores) processors without external graphics processing [22] to replicate the range of computational power the average physician is working with assuming their location is void of cloud servers.

It was found that the extractor was viable in both settings. The final architecture of the discriminator was only three layers deep excluding hidden layers. It accomplished accuracy rivaling that found in other research utilizing much more complex extractors.

Another thing that differentiates this methodology from others is the type of MLP used. A GAN enhanced with convolution was developed here. Such an architecture functions by training two MLPs:

1. The generator, which trains to fabricate data based on feedback from the discriminator [13].
2. The discriminator, which trains to differentiate data from the dataset and data fabricated by the generator [13].

Collectively, these two networks can create accurate data [12, 13] and predictions. Both of these algorithms were enhanced by convolutional layers [7]. The discriminator from the trained framework was utilized in this research to make binary classifications.

GANs are slightly unconventional for classification, but since the model works in an unsupervised binary framework, it was logically ideal for the problem statement since ensemble learning was used.

5.3 Third Annexation: Anomaly Detection and Data Conditioning using Forest-Based Algorithm

Similar to other models, this one trained on thousands of images [2, 4, 5, 6]. The images were retrieved from databanks on Kaggle. This research operates on semi-supervised learning, so labels were generated using nested iteration based on the directories in the ZIP files. A plot of the mean squared error between the images was observed to determine what conditioning would be optimal. The graph was omitted due to formatting guidelines. By observing the patterns, it was determined that:

1. The dataset was relatively inconsistent in samples between classes.
2. The error between the images was slightly higher than expected.

Many steps in processing the images and labels were taken in an attempt to remedy this issue. One such thing was reducing the resolution of each tensor along with cropping them to the same size. Pixel dimensions of 50x50, 75x75, 100x100, 125x125, and 150x150 were tested. The 50x50 arrangement had the lowest error, but it was found that the 100x100 and 150x150 ones had better accuracy. It was suspected that this was due to overfitting caused by the lack of inconsistencies in 50x50 images [11].

Paired with a semi-random and true-random bootstrap, an isolation forest was trained in an unsupervised framework. Isolation forests, much like random forests, are based on decision trees. In this instance of ensemble learning, it isolates the outliers by randomly selecting a datapoint from a specified pool and subsequently selecting a split in the range of maximum and minimum values the specified dataset. Recursively partitioning the datasets using progressively shrinking trees, the decision trees collectively act as an unsupervised classification method [23].

Though shown in the workflow, when applied, the image will not be parsed through the isolation forest for anomaly determination. The isolation forest was only used to partition the processed datapoints for training the multivariate model [11].

5.4 Fourth Annexation: Dynamics of Supervised Classification Methodology using Support Vector Machines

A thorough analysis of different classification algorithms was conducted for the ensemble approach. As previously mentioned, the model was trained to identify many other diseases than Coronavirus. The challenges of developing the multivariate model to do this were:

1. Keeping the computational cost low without sacrificing the high accuracy.
2. Two different radiograph types within the databanks.
3. Overlapping labels within the datasets for the disease subtypes.

Since the dense amalgamate is based on a MLP, it is capable of making classifications on it's own. However, other approaches were investigated due to the limitations of the kernel functions. Various machine-learning algorithms were concatenated to the ensemble. It was determined that three of these issues could be solved using two separate solutions:

1. Using an ensemble learning approach (aka. using multiple models strategically concatenated to collectively solve complex problems that would otherwise be difficult with a single model).
2. Splitting the model from a multidimensional one to many binary ones using a one-vs-everything adaptation.

Both of these were implemented. Calling the multivariate model one algorithm is misleading. It is more accurately a family of networks working together to classify diseases. Based on this idea, a classification algorithm was selected.

Initially, classification algorithms such as naive bayes, k-nearest-neighbors, k-mean clustering, etc. were researched. It was determined the SVM would be ideal because research showed that they are best for extreme cases [4, 24], which is perfect for the partitioned instances proposed by a one-vs-everything adaptation of this problem [21].

To simply put it, a SVM classifies information using one or many hyperplanes. Often these planes are linear. They are always defined by support vectors drawn from the dataset. They are often considered one of the most robust classification algorithms because they can operate based on non-linearly separable data [21, 24]. Concatenating a supervised kernel function for classification to unsupervised algorithms for feature extraction and anomaly reduction makes the model now semi-supervised.