COVID - 19 Automated Detection using Convolutional Neural Networks and Generative Adversarial Networks

A thesis submitted
by

Ultan Kearns

in partial fulfillment of the requirements for the degree of
Master of Science in Computing in Big Data Analytics and Artificial Intelligence

Supervisor: Dr Paul Greaney

Submitted to Quality and Qualifications Ireland (QQI)

Dearbhú Cáilíochta agus Cáilíochtaí Éireann
October 2022

Contents

Declaration

I hereby certify that this material, which l now submit for assessment on the programme of study leading to the award of Master of Science in Computing in . . . , is entirely my own work and has not been taken from the work of others except and to the extent that such work has been cited and acknowledged within the text of my own work. No portion of the work contained in this thesis has been submitted in support of an application for another degree or qualification to this or any other institution. I understand that it is my responsibility to ensure that I have adhered to LYIT's rules and regulations.

I hereby certify that the material on which I have relied on for the purpose of my assessment is not deemed as personal data under the GDPR Regulations. Personal data is any data from living people that can be identified. Any personal data used for the purpose of my assessment has been psuedonymised and the data set and identifiers are not held by LYIT. Alternatively, personal data has been anonymised in line with the Data Protection Commissioners Guidelines on Anonymisation.

I give consent for my work to be held for the purposes of education assistance to future Computing students at LYIT and it will not be shared outside the Department of Computing at LYIT. I understand that my assessment may be shared with any other third party and will be held securely in LYIT in line with the Institute's Records Retention Policy.

Signed: Ultan Kearns

Date: Wednesday 12$^{\text{th}}$ October, 2022

Acknowledgements

I would first of all like to thank my supervisor during this project Dr. Paul Greaney, he was a fantastic help throughout the course of writing this thesis and this work could not have been completed without his input and help.

I would also like to thank the lecturers who taught me so much during my postgraduate course, both Doctors Karla Muñoz-Esqueival and Shagufta Henna provided a fantastic introduction into many areas of Artificial Intelligence and the knowledge they imparted has helped me a lot throughout the course of conducting this research.

I would also like to thank Andrew Ng, his deep learning courses provided a great foundation into the realm of deep learning and Artificial Intelligence.

Acronyms

- AI - Artificial Intelligence
- ANN - Artificial Neural Network
- CNN - Convolutional Neural Network
- GAN - Generative Adversarial Network

List of Figures

Abstract

This paper aims to analyze the applications of Generative Adversarial Networks in overcoming issues of data-shortages in relation to COVID-19. There are many COVID-19 data-sets compiled but some suffer from lack of data-quality and data shortages[1][2]. In this paper I aim to create and train a convolutional neural network or CNNs to analyze X-Rays of patients lungs to automate the detection of COVID-19. The CNN will be trained with a number of images generated from different GAN architectures to determine which will prove most efficient in automating the detection of COVID-19. I also aim to use the GANs in conjunction with one and other to try out different combinations to see if feeding images generated by one GAN to other GANs will produce more accurate results when training the model.

Chapter 1

Introduction - An Overview of GANs, CNNs, & What I Aim To Accomplish in This Paper

1.1 What is A Generative Adversarial Network(GAN)

A generative adversarial network or GAN for short first appeared in a 2014 paper by Ian Goodfellow et al[3]. In this paper Goodfellow et al propose a new way to generate data via an adversarial process. The GAN essentially works as follows: two models are trained a generative model G which will generate the content from the data and another model D which will be the discriminator, judging if data created by the model came from the dataset rather than G. The goal of this training is to ensure data generated from G is realistic enough to fool our discriminator D into believing that our generated content came from the training set. It is in this way that we can create realistic "fake" data from our generative model.
There are a number of GAN architectures which are useful in different scenarios such as CycleGans[4] which are useful for translating images from a source domain $X- > Y$ in which Y is the target domain, StyleGan which was created by NVIDIA which allows more control over the generative process[5] & PixelRNN which can recreate images when given a fraction of the original and can generate new images based on probability[6].
In this paper I will use a number of different Generative Adversarial Network architectures and will use them in conjunction with each other by feeding content generated by one architecture into another to develop a more diverse training set for the final model.

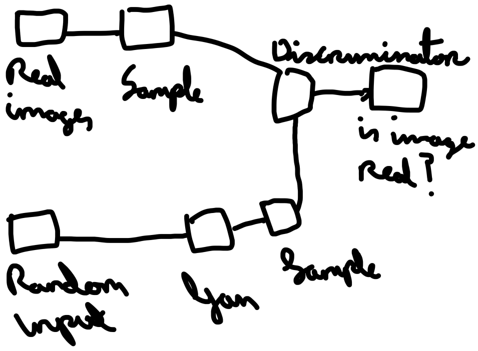

Figure 1.1: Basic Example of Generative Adversarial Network

As we can see from the image above we start the process by taking a sample of real images from our training data then passing it to the discriminator, we also take a sample from the GAN created images and pass that to the discriminator which will then determine if images are real are fake. After the discriminator determines if the image is real or fake backpropagation is then performed to train the model so that it can differentiate better between samples which came from our training-set and those which came from G.

1.2 What is An Artificial Neural Network? (ANN)

An Artificial Neural Network or ANN for short is a network of neurons / nodes which will be used for training a model to perform a certain task, they are made up of an input layer, N hidden layers, and finally an output layer. Each layer will have it's own activation function and will adjust it's weights and biases to determine the final output of the model.[7] These networks were heavily inspired by biological processes which occur in the brain.
Artificial Neural Networks are a general purpose model used to solve a number of common problems.

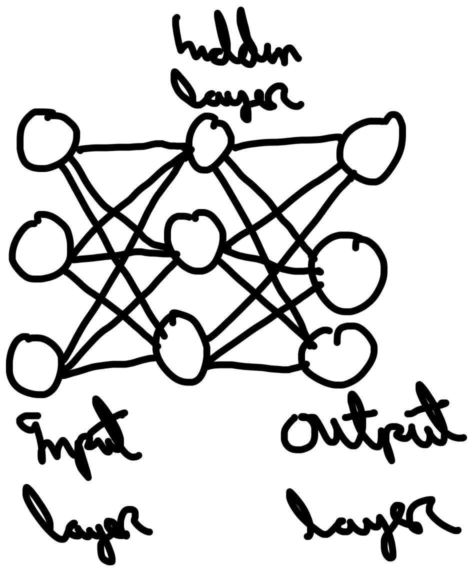

Figure 1.2: Basic Example of Artificial Neural Network

This is a basic example of an Artificial Neural Network1.2, as we can see from the image above the network has an input layer, a hidden layer and an output layer. Generally when creating these networks we determine the number of neurons in both the input and the output layers based on the different classifications we are trying to predict. The above network could be used to predict if an image is of a cat, a dog or a fish for example. There can multiple hidden layers in an ANN and the number of neurons in each layer can be adjusted. In reality Artificial Neural Networks will typically be far bigger than the example given above in terms of neurons and hidden layers but for illustrative purposes the above diagram will suffice. Each neuron will also have it's own weights, biases and activation function which will determine whether a neuron fires or not. Common activation functions include RELU(Rectified Linear Units), Sigmoidal function and tanh.

1.3 What is A Convolutional Neural Network? (CNN)

A Convolutional Neural Network or CNN for short, is a type of neural network which is primarily used for tasks involving image and pattern recognition[7] The structure is similar to an ANN in which we have an input layer, N hidden layers, and finally an output layer.

As with the Artificial Neural Network each of these layers will have an activation function and it's own weights and biases to determine the final output for a given input. The model will be fed an image which is made up of vectors(RGB) and from that image the model will determine certain patterns and classify the given image which in this context will be either COVID-19 Positive or COVID-19 Negative.

There are a few ways in which CNNs differ from ANNs in that they are comprised of three layers which are the convolutional layer, the pooling layer and fully connected layers[7]. The convolutional layer is responsible for determining the output of a given input, the pooling layer will reduce the parameters of a given input by means of downsampling and finally the fully connected layers will then determine and classify the output for a given input. The convolutional layers parameters utilize learnable kernels, this layer also produces a $2D$ activation map which will be used to determine if a neuron fires or not for a given input. We can adjust hyper-parameters in the convolutional layer to greatly reduce the complexity of the model through optimization which can be achieved by adjusting the following hyper-parameters: depth, stride and zero padding.

Depth is related to the output volume produced by the convolutional layers in the model which can be manually set by adjusting the number of neurons in each layer. Reducing the depth of the model will greatly increase the training time but at the expense of performance.

Stride is related to the spatial dimensionality of the input which will determine our receptive field(every neuron is connected only to a small region of the input this region is referred to as the receptive field[7]), if the stride is set to a low integer we will produce extremely large activations and if it is set too high we won't produce enough activations.

Finally we have zero-padding this will pad the border of the images input to our model reducing their dimensionality, padding is useful for increasing the accuracy of the model as it can possibly eliminate areas of the image which are not useful for the model and can also improve training time times(in some use cases).[8]

Through the adjustment of the hyper-parameters mentioned above and through the utilization of different activation functions the Convolutional Neural Network's accuracy can be improved through a process of trial and error.

1.4 Supervised Learning

Supervised Learning is a type of learning involving labelled data to train the model[9]. The data is labelled manually by a data-scientist which can be a long and laborious process depending on a number of factors(size of the data, number of classes etc.) but offers many benefits when it comes to training models. Supervised learning performs extremely well at tasks involving classification(classifying data into a given category) and regression(understanding the relationships between independent & dependent variables).

1.5 Unsupervised Learning

Unsupervised Learning is a type of machine learning involving unlabelled data to train the model[9]. This type of machine learning requires no human intervention since the data is unlabelled and will detect relationships between data based on the raw data fed in to the model. This type of machine learning is used for the following tasks: Clustering(grouping data together based on shared characteristics or features), Association(Finding relationships between features), and Dimensionality reduction(Reducing the number of features in a given data-set without compromising the integrity of said data). The key differences between supervised and unsupervised learning are: labelled vs unlabelled datasets, and finding relationships in data(unsupervised) or trying to predict and classify data(supervised).
For this project I will be using a mix of labelled / unlabelled data-sets to train and test the model.

1.6 Tensorflow

Tensorflow is an open-source library used for machine-learning and Artificial Intelligence research worldwide[10]. Tensorflow provides numerous modules and classes which will be the foundation of building both the Generative Adversarial Network and the Convolutional Neural Network. There have been numerous case-studies proving the efficacy of Tensorflow in solving many AI / ML problems and the library is used by the likes of Google, airB&B, ARM, Coca Cola, Intel, and many more[11].
Given the renown and reputation of Tensorflow and after exploring the documentation online I thought that this would be a welcome library to include when implementing the complex models which will be needed to automate the detection of COVID-19 and to implement the Generative Adversarial Networks to increase the robustness of the Convolutional Neural Network.

1.7 Keras

Keras is a deep-learning framework for python which will provide a number of helpful functions / methods when creating and training the CNN[12]. Keras is built on top of Tensorflow and will help greatly with data-loading, pre-processing and the overall building of the model. Keras is commonly used by data-scientists and researchers due to the powerful methods it offers and the time it saves. The additional classes and modules Keras provides will greatly help with and reduce the time of building both the Convolutional Neural Network and the Generative Adversarial Network.
Like Tensorflow Keras has been used by a number of companies and is well recognised in

the Artificial Intelligence community. It's uses include Computer Vision, Natural Language Processing, Generative Deep-Learning and Reinforcement Learning just to name a few[13].

1.8 Background of Problem & Aims of This Paper

COVID-19 is a highly transmissible virus which has caused a worldwide pandemic and has sadly claimed many lives, there have been 616,951,418 cases worldwide and 6,530,281 deaths as of the 4th of October 2022[14]. During the pandemic Ireland alone had a total of over 1.6 million confirmed cases and nearly 8,000 deaths[15]. This has led many researchers to pursue the goal of automating the detection of COVID-19 to partially relieve the immense pressure put on medical staff throughout the pandemic.

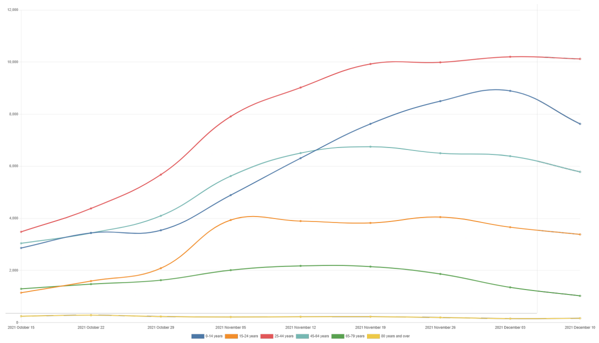

Figure 1.3: Graph of COVID-19 Statistics by age-range Ireland from October 2021 - December 2021 Courtesy of CSO[16]

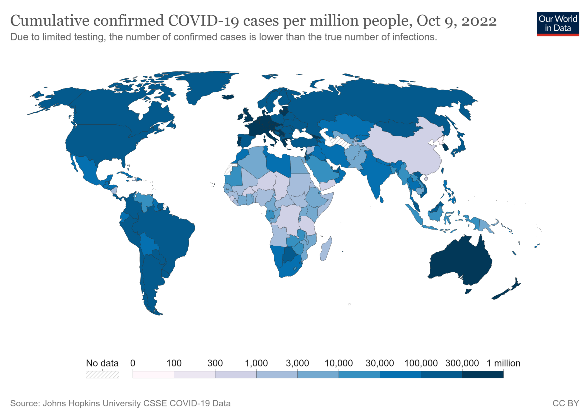

Figure 1.4: Cumulative cases of the COVID-19 virus world-wide courtesy of Our World in Data[17]

The main objective of this research is to develop a robust model which can accurately analyze X-Rays of patients and determine from said X-Rays if the patient is afflicted with

COVID-19. I plan to do this by utilizing a number of different GAN architectures which will create realistic "fake" data which will then be used to train a number of models, from this training I plan to compare and contrast the results when generating data with different architectures to determine the best configuration for data generation to train the CNN model. There has been some success in utilizing Convolutional Neural Networks to automate the detection of the virus[18][19]. It is my hope that through the use of data-augmentation utilizing a variety of GAN architectures that such Convolutional Models can be improved upon and made more accurate.
I plan on utilizing existing data sets which I will list in the next section when training the Generative Adversarial Models, through trial and error I plan on determining the best architecture of GANs to use for training the model for this use-case.

1.9 Data-sets Used in This Paper

- COVID-19 Radiography Database[20]
- COVID-19 Chest Xray[21]
- COVID-19 Pneumonia Normal Chest Xray PA Dataset[22]

1.10 About the data-sets

Before beginning the training of the model it is important to explore and understand each of the data-sets. There are a total of three data-sets which will be used in the course of this research, I will explain more about these data-sets below.

1.10.1 COVID-19 Chest X-Ray

The COVID-19 Chest X-Ray data-set is a data-set which is comprised of labelled X-Ray Images taken from a number of patients. This data-set contains 357 X-Ray Images of COVID Positive patients, and Chest X-Rays of those afflicted with another disease(MERS, SARS, and ARDS). This data-set also offers a metadata file listing the diagnosis of the patient along with a number of other features.

1.10.2 COVID-19 Radiography Database

The COVID-19 Radiography Database is made up of 3,616 images of chest X-Rays taken from COVID positive patients, 10,192 Images of lung X-Rays taken from healthy patients, and 1,345 X-ray images of Viral Pneumonia positive patients. All Images in this data-set are

PNG(Portable Network Graphic) images and are at a resolution of height 299 pixels & width 299 pixels eliminating the need for pre-processing of the images, the data-set also includes metadata for each of the images in this data-set showing a number of features with the diagnosis of the patient as well. This data in this data-set was gathered by a team of researchers from Qatar University, Doha, Qatar, and the University of Dhaka, Bangladesh along with their collaborators from Pakistan and Malaysia.

1.10.3 COVID-19 Pneumonia Normal Chest Xray PA Data-set

The COVID-19 Pneumonia Normal Chest X-Ray PA data-set is comprised of a train set containing 74 Normal X-Ray Images taken from healthy Patients and afflicted with Pneumonia and a test-set containing a Normal set containing 20 chest X-Rays taken from healthy patients and a Pneumonia set containing 20 images. The images in this data-set are unlabelled and no metadata is offered, however the images are segragated into separate files listing the diagnosis.

1.10.4 How I Plan to Use These Data-sets

I plan to use each of these data-sets to train and test the model and use data-augmentation to increase the train and test-sets by utilizing Generalized Adversarial Networks. When using these data-sets in conjunction it is my hope that the GAN will have enough data to be effective when generating new sample images to train the final model.

1.11 Structure of This Thesis

This thesis is broken into 6 chapters in total, I will include the headings of the chapters and a brief summary of each chapter below:

1.11.1 Chapter 1 - Introduction

This section will offer the reader of this thesis a brief introduction to a number of core concepts which will be necessary to understand before diving deeper into this thesis. It is important that the reader has a basic understanding of Generative Adversarial Networks, Convolutional Neural Networks, Artificial Neural Networks, supervised & unsupervised learning and the overall question that this research proposes before discussing the implementation or discussing pertinent literature in this field.
In this section I will frame the research question, explain what a Generative Adversarial Network is it's function and how it works, I will also explain Artificial Neural Networks & Convolutional Neural Networks, and I will discuss the basic methodologies relating to

the implementation of this project. I will also discuss the libraries used to implement the practical artifact, data-sets used to train the model and give the reader of this thesis a clear understanding of the key aims of this research.

1.11.2 Chapter 2 - Literature Review

In this section I will review pertinent literature related to the problem domain and discuss the ideas / concepts presented in these papers, I will also review the results from the research conducted in these papers and use them as a metric to gauge the performance of my own model. The papers will also be compared and contrasted and I will discuss the findings and how useful these papers were when conducting my own research. It is very important to understand the problem domain before beginning implementation of this project to ensure that I am not "reinventing the wheel". This section will also provide the reader of this thesis with the most up-to-date progress made within the problem domain.

1.11.3 Chapter 3 - Implementation

In this section I will discuss the architecture of the Convolutional model, the various architectures of Generative Adversarial Networks implemented, how the models were trained and the overall design of the code implemented and the rationale behind certain design choices. I will also show the results from training the models and discuss how through trial and error I was able to improve the various models and will include code samples so that the models can be reviewed by the reader or re-implemented by them.

1.11.4 Chapter 4 - Results

In this section I will review the results achieved from training the best models and suggest how they may possibly be improved. I will be showing lots of graphs / tables in this section to gauge each model's test / dev set errors and I will also be comparing and contrasting the effects of the different GAN architectures implemented as well as discussing the results of the convolutional model.

1.11.5 Chapter 5 - Further Research

In this section I will discuss further research that may need to be done by any researchers who would like to build upon this research. I will also review where the models could be improved and what I'd do differently if I were to conduct this research again and common issues I faced during the implementation of this project and how I overcame them.

1.11.6 Chapter 6 - Conclusion

This will be the final section of the paper and will tie the entire thesis together. This section will be a summary of all the research conducted, the code, and my experience overall throughout the writing of this thesis.

Chapter 2

Literature Review - Examining Current Paradigms & Research into GANs, CNNs and Their Applications in The Automated Detection of Diseases

Chapter 3

Implementation

3.1 Libraries Used

3.2 GAN Architectures

3.3 CNN Design

Chapter 4

Results of Research

Chapter 5

Future Work and Research

Bibliography

[1] A. Binkheder *et al.*, "Covid-19 data explorer," [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8701465/.

[2] S. Maior *et al.*, "Convolutional neural network model based on radiological images to support covid-19 diagnosis: Evaluating database biases," [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0247839#sec025/.

[3] A. Goodfellow *et al.*, "Generative adversarial nets," 2014. [Online]. Available: https://arxiv.org/pdf/1406.2661.pdf.

[4] P. Zhu *et al.*, "Unpaired image-to-image translation using cycle-consistent adversarial networks," 2020. [Online]. Available: https://arxiv.org/pdf/1703.10593.pdf.

[5] A. Karras Laine, "A style-based generator architecture for generative adversarial networks," 2019. [Online]. Available: https://arxiv.org/pdf/1812.04948.pdf.

[6] K. Oord Kalchbrenner, "Pixel recurrent neural networks," 2016. [Online]. Available: https://arxiv.org/pdf/1601.06759.pdf.

[7] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," [Online]. Available: https://arxiv.org/pdf/1511.08458.pdf.

[8] M. Hashemi, "Enlarging smaller images before inputting into convolutional neural network: Zero-padding vs. interpolation," [Online]. Available: https://journalofbigdata.springeropen.com/articles/10.1186/s40537-019-0263-7.

[9] J. Delua, "Supervised vs. unsupervised learning: What's the difference?," [Online]. Available: https://www.ibm.com/cloud/blog/supervised-vs-unsupervised-learning.

[10] Multiple, "Tensorflow," [Online]. Available: https://www.tensorflow.org/.

[11] Multiple, "Tensorflow case studies," [Online]. Available: https://www.tensorflow.org/about/case-studies/.

[12] Multiple, "Keras," [Online]. Available: https://keras.io/.

[13] Multiple, "Keras examples," [Online]. Available: https://keras.io/examples/.

[14] Multiple, "Who coronavirus (covid-19) dashboard," [Online]. Available: https://covid19.who.int/.

[15] Multiple, "Who coronavirus (covid-19) dashboard ireland," [Online]. Available: https://covid19.who.int/region/euro/country/ie/.

[16] Multiple, "Covid-19 deaths and cases statistics," [Online]. Available: https://www.cso.ie/en/releasesandpublications/ep/p-covid19/covid-19informationhub/health/covid-19deathsandcasesstatistics/.

[17] O. W. I. Data(multiple), "Covid-19 data explorer," [Online]. Available: https://ourworldindata.org/explorers/coronavirus-data-explorer?tab=map&facet=none&Metric=Confirmed+cases&Interval=Cumulative&Relative+to+Population=true&Color+by+test+positivity=false&country=USA~ITA~CAN~DEU~GBR~FRA~JPN.

[18] a. o. Ahmed Wang, "Automated detection of covid-19 through convolutional neural network using chest x-ray images," [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0262052/.

[19] C. E. Belman-López, "Detection of covid-19 and other pneumonia cases using convolutional neural networks and x-ray images," [Online]. Available: http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0120-56092022000100108/.

[20] C. Rahman *et al.*, "Covid-19 radiography database," [Online]. Available: https://www.kaggle.com/datasets/tawsifurrahman/covid19-radiography-database.

[21] J. P. Cohen, P. Morrison, and L. Dao, "Covid-19 image data collection," *arXiv*, 2020. [Online]. Available: https://github.com/ieee8023/covid-chestxray-dataset.

[22] Asraf, "Covid19 with pneumonia and normal chest xray(pa) dataset," [Online]. Available: https://www.kaggle.com/datasets/amanullahasraf/covid19-pneumonia-normal-chest-xray-pa-dataset.