Pontifícia Universidade Católica
do Rio de Janeiro

Antônio Moreira Pinto

Fine-Tuning Self-Supervised Model With Siamese Neural Networks for Covid-19 Image Classification

Dissertação de Mestrado

Dissertation presented to the Programa de Pós–graduação em Informática, do Departamento de Informática of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Informática.

Advisor: Prof. Paulo Ivson Netto Santos

Rio de Janeiro
September 2024

Pontifícia Universidade Católica
do Rio de Janeiro

Antônio Moreira Pinto

Fine-Tuning Self-Supervised Model With Siamese Neural Networks for Covid-19 Image Classification

Dissertation presented to the Programa de Pós–graduação em Informática, do Departamento de Informática of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Informática. Approved by the Examination Committee:

Prof. Paulo Ivson Netto Santos
Advisor
Departamento de Informática – PUC-Rio

Prof. Alberto Barbosa Raposo
PUC-Rio

Prof. Anselmo Cardoso de Paiva
UFMA

Rio de Janeiro, September 26th, 2024

Antônio Moreira Pinto

Computer Science Bachelor by Federal University of Maranhão.

Bibliographic data

Pinto, Antônio Moreira

Fine-Tuning Self-Supervised Model With Siamese Neural Networks for Covid-19 Image Classification / Pinto, Antônio Moreira; advisor: Santos, Paulo Ivson Netto. – 2024.

52 f: il. color. ; 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática, 2024.

Inclui bibliografia

1. Informática – Teses. 2. Aprendizado Auto Supervisionado. 3. Redes Neurais Siamesas. 4. Masked Autoencoders. 5. Radiografias. I. Santos, Paulo Ivson Netto. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

CDD: 004

Aos meus pais e irmãos,
pela confiança.

Acknowledgments

To my advisor, Paulo Ivson, who provided me with the means by which i have accomplished my degree. Without his support and guidance i would not have been able to accomplish this work.

To CNPq and PUC-Rio, for the financial support granted, without which this work could not have been accomplished.
This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

Abstract

Pinto, Antônio Moreira; Santos, Paulo Ivson Netto (Advisor). **Fine-Tuning Self-Supervised Model With Siamese Neural Networks for Covid-19 Image Classification**. Rio de Janeiro, 2024. 52p. Dissertação de Mestrado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

In recent years, self-supervised learning has demonstrated state-of-the-art performance in domains such as computer vision and natural language processing. However, fine-tuning these models for specific classification tasks, particularly with labeled data, remains challenging. This thesis introduces a novel approach to fine-tuning self-supervised models using Siamese Neural Networks, specifically leveraging a semi-hard triplet loss function. Our method aims to refine the latent space representations of self-supervised models to improve their performance on downstream classification tasks. The proposed framework employs Masked Autoencoders for pre-training on a comprehensive radiograph dataset, followed by fine-tuning with Siamese networks for effective feature separation and improved classification. The approach is evaluated on the COVIDx dataset for COVID-19 detection from frontal chest radiographs, achieving a new record accuracy of 98.5%, surpassing traditional fine-tuning techniques and COVID-Net CRX 3. The results demonstrate the effectiveness of our method in enhancing the utility of self-supervised models for complex medical imaging tasks. Future work will explore the scalability of this approach to other domains and the integration of more sophisticated embedding-space loss functions.

Keywords

Self-supervised Learning; Siamese Neural Networks; Masked Autoencoders; Radiographs.

Resumo

Pinto, Antônio Moreira; Santos, Paulo Ivson Netto. **Ajuste Fino de Modelo Auto-Supervisionado usando Redes Neurais Siamesas para Classificação de Imagens de Covid-19**. Rio de Janeiro, 2024. 52p. Dissertação de Mestrado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Nos últimos anos, o aprendizado auto-supervisionado demonstrou desempenho estado da arte em áreas como visão computacional e processamento de linguagem natural. No entanto, ajustar esses modelos para tarefas específicas de classificação, especialmente com dados rotulados, permanece sendo um desafio. Esta dissertação apresenta uma abordagem para ajuste fino de modelos auto-supervisionados usando Redes Neurais Siamesas, aproveitando a função de perda semi-hard triplet loss. Nosso método visa refinar as representações do espaço latente dos modelos auto-supervisionados para melhorar seu desempenho em tarefas posteriores de classificação. O framework proposto emprega Masked Autoencoders para pré-treinamento em um conjunto abrangente de dados de radiografias, seguido de ajuste fino com redes siamesas para separação eficaz de características e melhor classificação. A abordagem é avaliada no conjunto de dados COVIDx 9 para detecção de COVID-19 a partir de radiografias frontais de peito, alcançando uma nova precisão recorde de 98,5%, superando as técnicas tradicionais de ajuste fino e o modelo COVID-Net CRX 3. Os resultados demonstram a eficácia de nosso método em aumentar a utilidade de modelos auto-supervisionados para tarefas complexas de imagem médica. Trabalhos futuros explorarão a escalabilidade dessa abordagem para outros domínios e a integração de funções de perda de espaço de embedding mais sofisticadas.

Palavras-chave

Aprendizado Auto Supervisionado; Redes Neurais Siamesas; Masked Autoencoders; Radiografias.

Table of contents

List of figures

List of tables

List of Abreviations

ViT – Vision Transformer

CNN – Convolutional Neural Networks

RNN – Recursive Neural Networks

BCE – Binary Cross-Entropy

SSL – Self Supervised Learning

TP – True Positive

FP – False Positive

TN – True Negative

FN – False Negative

Não, a vida não me foi dada senão uma vez,
e não quero esperar esta felicidade universal.
Antes de tudo eu quero viver, do contrário
seria melhor não existir.

Fiodor Dostoievski, *Crime e Castigo.*

1
Introduction

Machine Learning is a fundamentally important artificial intelligence tool that sets unprecedented importance within our current technological landscape. It powers search engines, provides text transcriptions, surgical planning, advertisements, facial recognition. These applications have been increasing in complexity in order to develop an abstract understanding of data and, therefore, unique and original associations, similar to human consciousness. Previously, machine learning required careful feature extraction to provide separable features correlated to the task at hand. Although we are not close to achieving general artificial intelligence, deep learning has surpassed conventional machine learning by enabling reliable automatic feature extraction.

Deep learning consists of current numerical methods used to develop pattern recognition by stacking learnable non-linear operations. It develops increasingly more abstract features which are, in turn, also feed into even more non-linear operations, which can assimilate any other continuous function given enough parameters via back-propagation (CYBENKO, 1989). This means that with deeper feature representations, we are able to highlight relevant aspects and suppress superfluous features. For example, a learnable convolutional filter may remove any background from the original filter. After that, learn to aggregate shapes within the convolution filter window. Subsequent layers, in turn, associate each feature map across their neighboring previous layer. The general goal is that the deep learning procedure develops proper understanding of the underlying data complexity by itself, reducing human bias and approximating it to general intelligence.

Artificial Intelligence, therefore deep learning, is conceptually set into two main fields. Supervised learning and self-supervised learning. They are, in essence, applied learning algorithms which either expert annotated data or no data labeling. For example, a supervised learning algorithm, set as $f(x)$ is comprised of w_i weights. Said weights, can be optimized by many numerical methods in order to provide the minimum or maximum cost, defined by some loss function $Loss(f(x), y)$. We can numerically estimate the optimal gradient of said Loss function is order to decrease our current $f(x)$. Our goal is that, with enough samples, our loss function space provides is learnable by our function weights and reach minimal cost to our task over any not-observed samples, achieving proper generalization at the selected task.

For the most part supervised learning has been the standard machine

learning method. Since the introduction of convolutional neural networks as the state-of-art classification tool (KRIZHEVSKY; SUTSKEVER; HINTON, 2012) scalable deep learning models have been developed and became the current benchmark. However, the increasingly need to annotated data is set to be the main bottleneck from which machine learning applications have difficulty overcoming. Solving this problem is what encourages the development of self-supervised learning.

The current classification state of art has been shifting towards self supervision, as most self supervised frameworks provide a generic methodological pipeline to retrain supervised models without annotations. Algorithms such as Masked Autoencoders (MAE) (HE et al., 2022), Knowledge Distillation (HINTON; VINYALS; DEAN, 2015) and Generative Neural Networks (GOODFELLOW et al., 2020) have show to be able to increase our current neural networks models by providing non-labeled, iterative representation learning (BENGIO; COURVILLE; VINCENT, 2013) processes.

A mixed, semi-supervised approach, can be seen applied to our currently most used AI assistants, such as ChatGPT (OPENAI, 2024). Their pre-trained model achieves such generalization by providing reinforcement learning tasks, such as predicting sentences next words and masking input tokens, which teaches the model how to properly understand the underlying grammatical structure of sentences. After that, their current weights are set into a supervised learning training loop, called Reinforcement Learning from Human Feedback. This method requires the users to provide prompts and rate the model responses, after which it starts to grasp which are the appropriate responses.

One explanation of why self-supervision improves supervised models performances is models pre-trained on an appropriate SSL task can encode this signal through learned representations that can solve downstream tasks with linear classifiers (KUMAR et al., 2022). What it means is that the feature encoder, as in any intermediate operation of a deep learning model with ℓ modules (e.g: convolution, self-attention), $(\ell_i, i < n-1 \in \mathbb{R})$ of the SSL task already sets our output to easily separable classification inputs. As of now, the current state-of-art classification models leverage the usage of SSL tasks in order to improve generalization and accuracy.

Self supervision hinges on the fact that the unsupervised approach can turn a set of inputs into a separable distribution. In turn, a weak learner is able to divide the embedding space output into a classification context. This is part of the reason methods such as Masked Autoencoders (HE et al., 2022) provide reasonable results through just linear probing and Kumar et al. (2022) argues that fine-tuning can distort pretrained features. However, this introduces a set

of weakness, on which it is not possible to maintain the same feature extractor quality while retraining said model, as the task at hand, while separable, may not be satisfied by the current function.

To solve this weakness, it is required to propose a new method to further refine the pretrained distribution without discarding the learned features.

1.1 Objectives

Our main goal is to provide a new semi-supervised method borrowing concepts from Siamese Neural Networks (KOCH et al., 2015), in order to further refine the deep features produced via our self-supervised process. That way, we are able to adjust self supervised models to better understand classification tasks without losing the current embedding space representation.

It is expected that our triplet (HADSELL; CHOPRA; LECUN, 2006) margin is able to further space out the learnable features as they are inherently a good embedding-space representation after a self-supervised task. As each learnable triplet is further spaced within Positive and Negative, their already projected deep features require fewer adjustments. We intend to further boost pretrained features performance using siamese neural networks at classification tasks.

Our goals are: (1) to provide a self-supervised COVID-19 radiography deep learning model; (2) evaluate the self-supervision improvements over a binary classification setting; (3) demonstrate the usage of embedding-space loss functions over the retention of our self-supervised embedding-space deep features.

This document is structured as follows. Chapter 2 explains the theoretical background surrounding our method. In Chapter 3 we present previous relevant supervised and self supervised learning approaches. In Chapter 4 we explain our proposed method. In Chapter 5 our main findings. Finally, in Chapter 6 we present our conclusion and future work.

2
Theoretical Background

This chapter selects and defines most concepts and techniques related to this dissertation. At Section 2.1 we explain the most fundamental background surrounding artificial neural networks, which are the foundation of modern deep learning. After this, how modern feature extractors are related to the development of deep learning at Section 2.2. Then, Section 2.3 details on how Masked Autoencoders work. Finally, Sections 2.4 and 2.5 detail the mathematical formulation of our neural network of choice, the Vision Transformer (ViT).

2.1
Artificial Neural Networks

Artificial Neural Networks are machine learning graph mechanisms developed to simulate neurological functions. Their main concept being the activation. In biological terms, neuron activation refers to the threshold that a neuron must overcome to pass an electrical signal to connected neurons through the receptive fields of dendrites and axons. This concept is reused by activation functions. Activation functions are functions which, given the sum of all previous outputs, computes an new output. Figure 2.1 further illustrates this behavior, similar to the actual structure of a neuron.

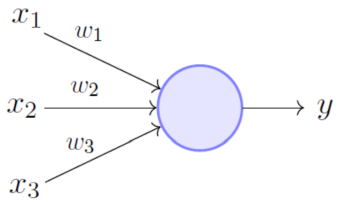


Figure 2.1: The Artificial Neuron Diagram (MINSKY; PAPERT, 1969).

A neuron, or perceptron, can be defined as a matrix of weights W, an input x and the activation θ, which is usually any differentiable non-linear function. Therefore, the output of an perceptron is given by,

$$\hat{y} = \theta(\sum W_i x_i + bias) \tag{2-1}$$

Said activation is what composes the non-linearity stacking that combines, suppresses and highlights the underlying concepts within the neural network propagation logic.

These weight vectors act as the input-output settings of the network. By calculating the appropriate weights, we can determine the optimal solution to our problem. This is done by numerically estimating the gradient of the cost related to a bad output. Assuming this cost function to be $y - \hat{y}$, $w_i \leftarrow w_i - lr(y - \hat{y})x_i$ and $b \leftarrow lr(y - \hat{y})$. We can refine this formulation into a simple multilayer perceptron (MLP), since we are able to update any graph via back-propagation as an application of the chain rule. It now was multiple neurons and these neurons can be layered, allowing the network do posses depth. Stacking multiple layers allow the network to simulate the behaviors of more complex and non-linear functions beyond the activation itself.

$$\hat{a}_i^l = \theta(\sum W_{ij}^l a_i^{l-1} + bias_j^l) \tag{2-2}$$

Where:

– a is the previous input.
– $\hat{a}$ is the activation output of the node i in layer l.

These update rules and mechanisms are, now, conveniently computed by automatic differentiation tools (BAYDIN et al., 2018). It is a tool to efficiently compute the gradient of any function by overriding the operation with their know derivative. Then, as every small operation is differentiable, we are able to update the gradient of every graph by recursively accessing their respective derivative, by chain rule. Given that $L(f(x), \hat{y})$ is differentiable, we estimate a form of landscape, called hyperplane, over which we can descent in order to find the steepest weight arrangement, close to the average expected minimum and optimally solving our classification problem.

In practice, it is enables the usage of algorithms like stochastic gradient descent. The mini-batch stochastic gradient descent algorithm samples a subset of inputs which are feed into our neural network. Then, we compute the average gradient, subtracting it from the weight matrix, pointing the neural network into reducing our loss $L(f(x), \hat{y})$. This process is repeated until the is no further decrease in error.

2.2 Feature Extraction

Before deep learning, most neural networks suited to image classification were feed carefully crafted image processing features that would highlight aspects correlated to expert knowledge and the nature of the data. These range from color, texture, shape and other heuristics that might be elicited. Then, the network would adjust these weights in order to grasp the correct correlation between the inputs and expected output.

One of the shortcomings were,

– Effective feature engineering required too much time and knowledge of the domain;
– These features were mostly suited only to said task and would not be easily transferred to other applications;
– The features often failed to provide complex patterns.

As neural network can provide the approximation of any continuous numerical function, and a given feature is the computation of the function, the optimal solution of the neural network is able to contain the feature extractor itself. Before deep learning, feeding raw data would only provide over-fitting, since the landscape of the hyperspace provided by feeding the output of an image is too complex for the data points to shape the optimal solution.

Deep learning are a set of methods that provide automatic feature extraction without needing of manual feature engineering. They are, as the name suggests, very deep neural networks. Before, it was expected that heavily parameterized models would lead to overfit. After the introduction of CNNs by LeCun et al. (1989) and popularization of AlexNet (KRIZHEVSKY; SUTSKEVER; HINTON, 2012), convolution based neural networks developed the landscape of deep neural networks as general automatic feature extractors and classifiers.

The convolution operation involves the usage of a small sliding learnable filter that maps the current matrix into a new matrix, called feature map. For an 2D input I and a filter F, the convolution $(I \cdot F)(x, y)$ at (x, y) is,

$$(I \cdot F)(x, y) = \sum_{i=0}^{m} \sum_{j=0}^{n} I(x + i, y + j) \cdot F(i, j) \tag{2-3}$$

Where,

– m and n are the height and width of the convolutional filter.

By stacking convolutional operations with pooling and activation functions, we are able to develop small automatic feature extractors that combine small local features and huge contextual relationships. Figure 2.2 shows how these feature maps are compartmentalized across the network layers.

A deep-learning architecture is a multilayer stack of simple modules, most of which are subject to learning, and many of which compute non-linear outputs. By stacking these modules it is possible to develop complex feature extractors that are both invariant to irrelevant inputs and can capture fine details.

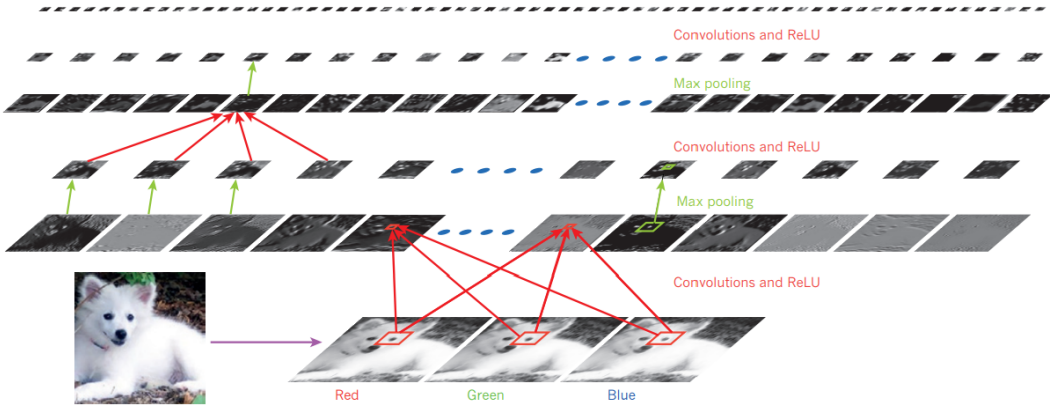


Figure 2.2: Convolutional Neural Networks Feature Maps (LECUN; BENGIO; HINTON, 2015).

2.3 Masked Autoencoders

MAE are a form of deep learning pre-training frameworks that leverages self-prediction in order to develop high quality and transferable pre-training features. They manage to provide efficient computation of masked regions by selecting patches of the standard transformer architecture and masking. Then, our auto-encoder goal is to extrapolate the missing pieces. This has shown to greatly improve downstream tasks via usage of non-labeled samples.

One of the main advantages of using MAE is that we do not need to attribute masking tokens to omitted regions. They are simply removed. As such, in a context of 75% masking ratio, we are able to severely reduce the computation cost of the pre-training process. The MAE decoder, on other hand, reintroduces the masked region into the network output. Now, with a full set of tokens, a series of transformer blocks are able to re-scale out output to the original unmasked dimension, as shown at Figure 4.1.

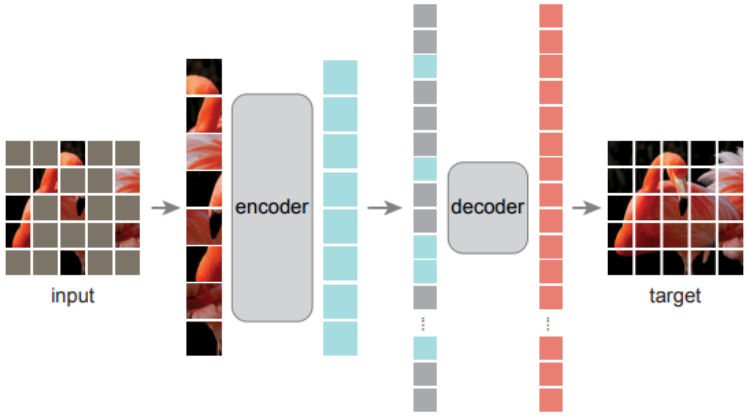


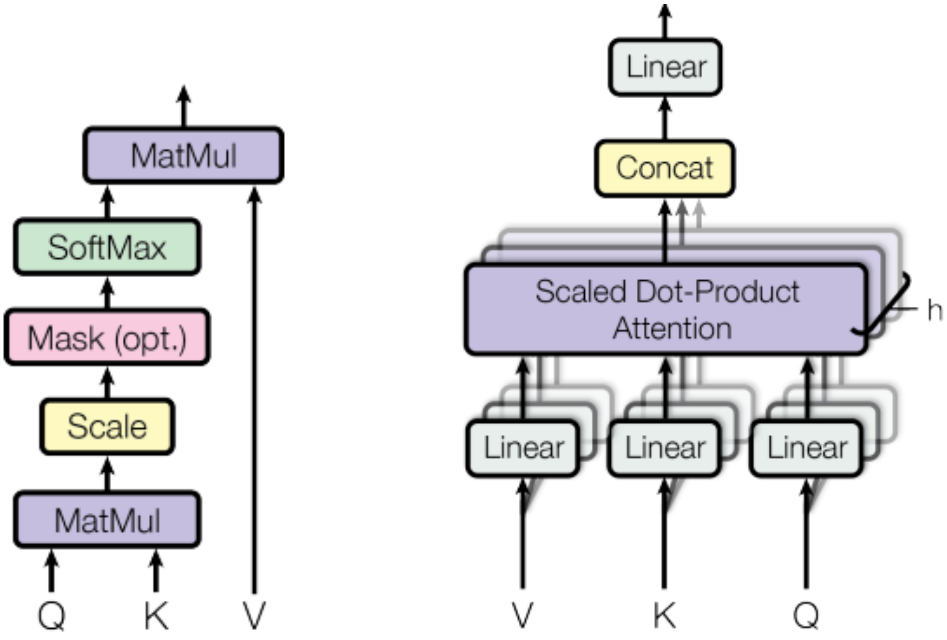
Figure 2.3: MAE, Input and Output regression (HE et al., 2022).

The omitted masking tokens and shuffling of visible patches are a great method for regularization, as the removal of context conditions the model to compute features regardless of position that may provide relevant information to the feature decoder. In fact, He et al. (2022) states that introducing the masking token to the encoder input severely diminishes accuracy in context of linear probing. In other words, our latent space representation provided MAE is strongly indicated to be easily separable over classification tasks.

At the time of publication, this method outperformed contrastive learning methods such as MoCoV3 and BEiT at fine tuning. While BEiT provides higher linear probing accuracy, it is outperformed by fine tuning MAE on ViT-L model. Notably, MAE provide much better results when performing partial fine-tuning, from 73,5% accuracy up to 81%.

2.4 Transformer

The Transformer is the backbone of modern sequence to sequence models. Originally introduced by (VASWANI, 2017) it has been widely adopted into both computer vision, NLP and speech processing. Their main mechanism, called self-attention, was proposed with the purpose of replacing RNN models, due to memory restrictions of the recursive computing of a long sequence and inefficient parallel processing inherent to recursion. The Transformer and its self-attention mechanism have led to significant advancements in sequence to sequence processing, including models like BERT and GPT. These models have demonstrated state-of-the-art performance in a wide range taks and have been further adapted in this work via the usage of ViT.



(a) Scaled Dot-Product Attention (b) Multi-Head Self Attention

Figure 2.4: The multi-head attention diagram, from Vaswani (2017). The left side describes the attention mechanism operations at each Q (Query), K (Key) and V (Value), while the right side is a parallel application of the scaled dot-product attention.

Lets say there is an input $X = [x_1, x_2, \ldots, x_n]$, where each x is a d-dimensional vector. Each attention head applies a different linear transform to obtain a triplet called: queries, keys and values. That makes it so there are tree learnable weight matrices per attention head, h. These are illustrated at Figure 2.4a.

$$Q_h = XW_h^Q, K_x = XW_h^K, V_x = XW_h^V \tag{2-4}$$

$Att(Q_h, K_h, V_h)$ computes the probability distribution over d by applying $Softmax$ over $\frac{Q_h K_h^\top}{\sqrt{d_k}}$. We arrive at,

$$Att(Q_h, K_h, V_h) = softmax(\frac{Q_h K_h}{\sqrt{d_k}})V_h \tag{2-5}$$

Then, each attention head is concatenated, as seen at Figure 2.4b and multiplied by another weight matrix, amounting to,

$$MSA(X) = (Att(Q_1, K_1, V_1), Att(Q_2, K_2, V_2), \ldots, Att(Q_H, K_H, V_H))W \tag{2-6}$$

By forcing the attention mechanism to estimate queries and probability distributions, we can set them to focus on specific features and correlations. This enables the module complexity to increase with depth. It creates a context aware feature encoder.

2.5 Vision Transformers

ViTs have become reliable standard deep learning models since Dosovitskiy et al. (2020). The paper provides us with with an encoding and decoding method that makes it possible for the transformer architecture to retain visual information. In order to to that, the images are split into multiple patches, that are subsequently flattened and linear-probed into embeddings, now able to hold it as a sequence. We can see this feature patching and encoding at Figure 2.5.

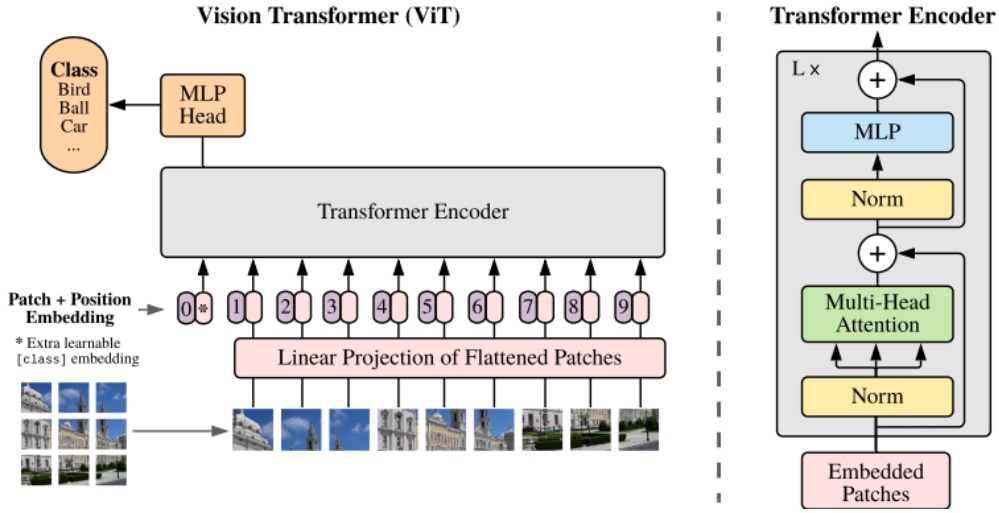


Figure 2.5: The ViT overview. The images are split into patches, which are subsequently embedded and processed by the transformer encoder. After this, the MLP head turns it into a Classification token (DOSOVITSKIY et al., 2020).

In other words, an image $x \in R^{(H,W,C)}$ is split int $x \in R^{N\times(P^2,C)}$, where P^2 are the patches of an (H, W) resolution sample and $N = HW/P^2$, which is the new sequence length for the transformer. Each patch is turn into a embedding projection by a learnable linear matrix called E. Each learnable embedding, z_i is the learnable representation of each i-th patch, $z_i = x_i E_i, E \in \mathbb{R}^{(P^2,C)\times D}$.

The N sequence patch embeddings are $Z = [z_0, z_1, \ldots, z_n]N \in \mathbb{R}^{(P^2,C)\times C}$. In short, $Z = [x_1E, x_2E, \ldots, x_nE]$. Then, to retain positional encoding we add E_{pos} to our embeddings. $E_{pos} \in \mathbb{R}^{(N+1)\times D}$. Then, these tokens are processed through multiple self-attention and linear layers. These layers are stacked through four main operations: Multi-Head Self-Attention (MSA); Layer Normalization (LN); MLP and Residual Connections. Equations 2-7 to 2-10 describe the output of each transformer block by layer ℓ.

$$z_0 = [x_{class}; x_p^1 E; x_p^2 E; \cdots ; x_p^N E] + E_{pos} \tag{2-7}$$

$$\begin{aligned} z'_\ell &= MSA(LN(z_{\ell-1})) + z_{\ell-1} \\ &\ell = 1 \dots L \end{aligned} \tag{2-8}$$

$$\begin{aligned} z_\ell &= MLP(LN(z'_\ell)) + z'_\ell \\ &\ell = 1 \dots L \end{aligned} \tag{2-9}$$

$$y = LN(z_L^0) \tag{2-10}$$

The usual ViT is based on formulating reasonably good patch encoding and computing the relationship between the correspondingly image subareas via self-attention. It is set up to be a good embedding feature extractor, yielding remarking results over transfer learning setups, both at supervised (DOSOVITSKIY et al., 2020) and specially over self-supervised contexts (HE et al., 2022). Transformer based networks are able to exceed convolutional neural network performance, provided there is enough compute power and data.

2.6 Siamese Neural Networks

Siamese Neural Networks can be seen as an extension of the pair concept that we see on unsupervised contrastive learning. However, as implied by name, we do not form pairs, but triplets. The semi-hard triplet loss for a triplet (a, p, n) consisting of an anchor a, a positive example p, and a negative example n is defined as:

$$\mathcal{L} = \sum_{i=1}^{N} \left[\max \left(0,\ \|a_i - p_i\|_2^2 - \|a_i - n_i\|_2^2 + \alpha \right) \right] \tag{2-11}$$

Where:

- $\|x\|_2^2$ denotes the squared Euclidean distance.
- α is the margin parameter.
- N is the number of triplets.

The goal, over the epochs, is to approximate our positive to the anchor, which belong to the same class, while distancing itself from the negative. As such, in other to fit to our new distribution, it is necessary to improve our

triplet selection process when training the siamese model. With this purpose, different sampling strategies are employed. For example, in semi-hard triplet loss, the negative example n_i is selected such that:

$$\|a_i - p_i\|_2^2 < \|a_i - n_i\|_2^2 < \|a_i - p_i\|_2^2 + \alpha \tag{2-12}$$

The goal is to select triplets where the negative is further than the anchor, but closer than the positive. This method can stabilize training and improve efficiency by guaranteeing a boundary between each embedding representation without collapsing most samples into a single point.

Any ViT architecture is able to provide visual compression through the patch embeddings and these embeddings, in turn, can be projected into contrastive learning setups. Our aim is to provide a method which can reliably fit self-supervised tasks, MAE, therefore enhancing the easily separable classification embeddings over the supervised setting. These embeddings are transferred to an linear-probe hyper-sphere whose embedding projection is adjusted through triplet loss. Due to triplet loss margin, we are able to retain most of it's original properties, while also fitting in small datasets, which is an intersection of both techniques usual application.

3
Related Work

This Chapter overlooks the current state of machine learning classification algorithms and their development across: (1) Image Classification; (2) Self Supervision; (3) Contrastive Learning; (4) Self Prediction.

3.1
Image Classification

Chronologically, the first CNN models to fully demonstrate their potential for large-scale image classification was AlexNet (KRIZHEVSKY; SUTSKEVER; HINTON, 2012). It was the winner of ImageNet Large Scale Visual Recognition Challenge 2012 by a significant margin, providing a top-5 error of 15.3%, 10.9% bellow the second best performing model.

After this, (SIMONYAN; ZISSERMAN, 2014) proposes VGG, demonstrating further performance increases by stacking small convolutional filters. Their improvements further developed the state of image recognition in a sense that not only convolution was a scalable feature extractor, but that scaling model depth would provide increasingly better results, consolidating deep learning as the state of art large scale classification models. They achieved 7.3% top-5 error at Imagenet.

ResNet (HE et al., 2016) introduces the usage of residual connections to circumvent the vanishing gradient problem, inherent to deep neural networks, allowing for even deeper neural networks and, in turn, new image classification benchmarks. Their largest proposed model achieved a top-5 Imagenet error rate of 3.57%. Similar to the ResNet, DenseNet (HUANG et al., 2017) also proposes the usage of skip connections, but instead of adding previous activation maps into subsequent ones, their proposition concatenates every previous layer into the next one. And, as another contribution, it was stated to reduce the number of necessary parameters around 5 times when compared to ResNet. This network architecture achieved a top-5 error rate of 3.74%. While not the state of art, their innovative design was praised and highly referenced.

CheXNet (RAJPURKAR, 2017) utilizes the DenseNet-121 architecture to produce a large-scale model capable of classifying 14 common chest diseases. It demonstrated that an AI-assisted system could perform at the level of a radiologist, achieving an AUC of 0.76. It highlighted the potential of deep learning models to assist radiologists in diagnostic tasks and consolidated

DenseNet as the convolutional neural network of choice for radiography multi-label classification.

After 8 years of consistent usage of CNNs, (DOSOVITSKIY et al., 2020), applies a new feature extractor called transformer, in place of convolution, leveraging mechanisms of natural language processing encoding and sequence to sequence computing, achieving competitive performance with fewer inductive biases in comparison to CNNs. While, from scratch, the ViT did not surpass the original CNN models, with retraining at larger datasets like ImageNet-21k the ViT Large achieved 88.55% top-1 accuracy, outperforming the best CNNs (85.8%).

3.2 Self Supervision

This Section details on works related to self supervision, which in recent years, has revolutionized image classification by enabling models to learn rich representations from unlabeled data. This shift began with methods like contrastive learning, which leveraged image pairs to learn embeddings. Building on this foundation, researchers introduced more advanced techniques such as MAEs, which further enhanced representation learning.

3.2.1 Contrastive Learning

Contrastive learning is a tool on which we can project the features of images into the latent space and use similarities and dissimilarities to further distance or approximate these samples. In the context of self-supervision, it assumed that every sample, x, can be paired with another similar sample without labels, called $\hat{x}$ and it should provide a similar representation. There are called co-occurrences and may be extrapolated by things such as how frames are near each other in a video. In text, if they are neighbors. Instead of a classification label, there are only loosely defined similarities.

SimClr (CHEN et al., 2020a) self-supervised approach is an application of contrastive learning processes. In essence, our positive pairs are composed of two augmented views of the same image, while negatives consists of views of different types of images. The projection head then turns our neural network output into a n dimensional embedding. The loss function for a positive pair (i, j) is defined as:

$$\ell_{i,j} = -\log \frac{\exp(sim(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} 1_{k \neq i} \exp(\mathrm{sim}(z_i, z_k)/\tau)} \tag{3-1}$$

Where:

– $\text{sim}(u, v)$ is the cosine similarity between vectors u and v.

– τ is a temperature parameter.

– $2N$ is the total number of augmented examples in a batch.

After this pre-training process, SimClr was fine tuned to various downstream classification tasks and, at the time, achieved 76% Top-1 accuracy ImageNet, providing a simple and efficient self supervision method and achieving near supervised learning baselines.

SimCLR has inspired numerous subsequent works in self-supervised learning and contrastive learning. It has paved the way for more advanced methods and improvements, such as Bootstrap Your Own Latent (BYOL) Grill et al. (2020). BYOL, puts forward changes to the current SimClr pipeline, stated to posses shortcomings related to the need of negative pairs. Their core idea hinges on the usage of two neural networks, called target network and online network.

The BYOL selects a sample x. Then, we reproduce two augmented view of x, called x_1 and x_2. y_1 and y_2 are $f_{online}(x_1)$ and $f_{target}(x_2)$. After that, their respective projection head maps y_1 and y_2 to z_1 and z_2. Then, the target network applies another predictor to transform each network embedding space representation from z_1, z_2 to p_1, p_2. Our loss becomes, then,

$$L = ||p_1 - z_2||_2^2 - ||p_1 - z_2||_2^2 \tag{3-2}$$

However, only the online network is minimized through this loss function. f_{target} is updated by the exponential moving average of f_{online}. As the target network is mainly trying to copy the target network output under different augmentations, it bootstraps their own latent space representation. This work was able to achieve 74.3% top-1 accuracy on ImageNet, showing that a contrastive loss approach could also be performed without negative pairs.

Caron et al. (2021) uses a teacher-student framework with ViTs where both networks are updated independently. The student network is trained to match the teacher's pseudo-labels produced under different augmentations, similar to BYOL but applied to ViTs. Achieved 80.1% top-1 linear evaluation performance on ImageNet and also laid the groundwork for self-supervision to ViT, which became the next state-of-art classification on the following year.

3.2.2 Self-Prediction

Self-Prediction SSL methods are techniques which revolve around the model predicting parts of itself from an incomplete representation. They range

from denoising, completing missing images patches and many other pretext-tasks. It is expected that we are able to link out pretext to the downstream task to ensure good representations.

Vincent et al. (2008) presents denoise auto-encoders. Denoising auto-encoders are a type of neural network designed to compress an image into a latent space representation and decode it without some sort augmentation/-corruption. The goal is to decrease the distance from $f(\hat{x}_1)$ to x_1, therefore $L = ||f(\hat{x}_1) - x_1||_2^2$. The paper's contributions have had a lasting impact on the field of unsupervised learning and continue to influence research in robust feature extraction. This method prevents the model from learning trivial identity functions and encourages the extraction of meaningful patterns, which may happen if the current task does not fit an auto-encoder setting.

Auto-Encoding Variational Bayes (KINGMA, 2013), also know as Variational Autoencoder (VAE), is an autoencoder optimized by Evidence Lower Bound. It laid the groundwork for subsequent research and improvements in generative modeling. VAE provides a probabilistic framework for learning latent representations that capture the underlying distribution of the data. Unlike traditional variational inference methods, VAEs are scalable to large datasets due to their neural network architecture. Still, while influential, Kingma (2013) did not focus on achieving state-of-the-art results on specific benchmarks but instead aimed to demonstrate the effectiveness and versatility of the VAE framework. However, the authors tested the model on several standard datasets for generative modeling and image reconstruction tasks, showing that VAEs can learn meaningful latent representations.

Self prediction methods mainly resurfaced beside contrastive learning after MAEs, by He et al. (2022), provided an efficient semi-supervised framework that leverages the usage of Vision Transformers with a variety of pretext tasks. The masking of visual representation is meant to provide hard to learn features, similar to what large language models have already been doing for many years in order to grasp grammar concepts (DEVLIN, 2018). After training by masking, the auto-encoder decoder is disconnected from the base ViT, which now holds a refined latent space representation. Such representation was able to achieve state-of-art imagenet accuracy through both linear-probe and fine tuning, over which the paper provides a wide parameter selection detailing experiment configurations leading to the optimal results.

3.3 COVID-19 Classification

COVID-19 has put a heavy strain across the global healthcare systems, developing an increased effort in many research field in order to search better, faster and efficient diagnosis techniques. Within this topic, deep learning plays a role into providing accurate diagnosis through image classification.

Gazda et al. (2021) has achieved remarkable COVID-19 classification performance by applying contrastive learning to radiography classification. At the time, it is stated by the author that every COVID-19 classification task was only pretrained by imagenet backbones, and applied their contrastive learning pipeline through four different CRX datasets. While not achieving state of art COVID detection, they successfully achieved higher accuracy then imagenet pretrained backbones with small amounts of data, as little as 1% of their current available samples, which thoroughly validates the usage of self supervision in the context of radiography classification, of +7.2%. Besides the performance increase, due to data availability, SSL was a sensible choice by the authors, being generally considered one of the best approaches when dealing with small datasets. Still, by modern standards, 80% accuracy does not provide a competitive model.

Works like (WANG; LIN; WONG, 2022) provide state of art COVID detection, at 98.25% accuracy, with a custom convolutional neural network. They proposed a new network architecture specifically tailored for the detection of COVID-19 from chest radiographs. Their dataset, COVIDx, includes a curated collection of publicly available chest radiographs and is regarded as a quite diverse dataset, being frequently updated over the recent years, just as the COVID-Net model.

(CONSTANTINOU et al., 2023) uses the COVID-QU posterior and anterior chest x-rays for classification across four different deep convolutional neural networks, namely ResNet50, ResNet101, DenseNet121, DenseNet169, and InceptionV3. This dataset discriminates their own samples into three classes, being healthy, bacterial/viral pneumonia and COVID-19 positive samples, containing images from 6 other public sources and up to 33,920 samples. They were all fine tuned from imagenet weights and are stated, at the time, to be state of art models. Their best performing model achieved 96% accuracy according to their tests.

Xiao et al. (2023) applies self supervision at COVID-19 classification thought a similar dataset, COVIDx 9, which is the previous version of the currently dataset provided by the COVID-Net Open Source Initiative. Differently from other previous work, both contrastive learning and self prediction are

applied, through MoCo (HE et al., 2020) and MAE (HE et al., 2022). While not surpassing the 98.3% state-of-art performance provided by COVID-Net-CRX-3 model, their best performing network achieves 96.3% accuracy while being only 1/10th of the approximate computational cost, being a surprisingly competitive option that might only require further model scaling in order to surpass the current state-of-art.

Also, their comparative analysis over the COVIDx dataset provides insights that are often lacking on modern COVID classification benchmarks, however, their applied metrics refer incorrectly to a binary classification report published by (WANG; LIN; WONG, 2022), which is implied to be performed at COVIDx 9B, not A. This does not impact the model COVID-19 recall, however, their accuracy may be under the actual value since COVIDx 9B does not include the pneumonia class, meaning their model may have achieved better results. Also, one of the few shortcomings of this work is that it was published around the same time as the expansion of the COVIDx CRX-3 dataset, which would not leave enough time for it to be reproduced with a substantially larger training set, but would only be suited to binary COVID-19 classification. Also, it is stated that they apply the state-of-art multi-label thorax disease detection, DenseNet-121. While this can be argued since most state of art CheXpert (IRVIN et al., 2019) leaderboard are DenseNet-121 ensembles, the performance cannot be strictly attributed to network topology. And the best performing model at the CheXpert dataset is the "Large-scale robust deep auc maximization" paper, from Yuan et al. (2021).

(WANG et al., 2023) proposes a custom ViT method called PneuNet, based on ResNet18 backbone and ViT, was developed for COVID-19 diagnosis from chest X-ray images. PneuNet uses channel-based attention, achieving a 94.96% accuracy in three-category classification, no pneumonia, normal pneumonia, and COVID-19, and a 99.30% accuracy in binary classification, with pneumonia and COVID-19. The study highlights the effectiveness of channel-based attention in feature recognition and image classification for COVID-19 diagnosis. One small problem with the statement that it achieved 99.3% accuracy on a binary classification problem is that the appropriate experimental setup would have been to include normal samples into a prior stage of processing or into the negative class, which is not the case, as the expected subject is not certain to be in either class.

(FEDORUK et al., 2023) study examines the impact of data augmentation techniques on the classification of COVID-19, viral pneumonia, lung opacity, and healthy lungs from chest X-ray images found that classical augmentation techniques were more effective than GAN-based augmentation. The

study employed two pre-trained CNN models, EfficientNet-B0 and Inception-v3, and compared the results of classical and GAN-based augmentation to a baseline of no augmentation. The results indicate that the EfficientNet-B0 model, without augmentation, achieved the best performance, with 90,2% accuracy. Even when considering that this method includes a wider range of classification settings, if we treat it as in a binary classification of COVID-19 and non COVID-19, this network has a 89,9% recall.

3.4 Improvements

From 2020 to 2022, most influential self supervised pretrained frameworks were slightly shifted from contrastive learning (CHEN et al., 2020b), (HE et al., 2020), (GRILL et al., 2020), (CARON et al., 2021) to self prediction (HE et al., 2022), mainly MAE, which were, at the time of release the state of the art for multi-label classification. Taking this into consideration, our work focuses on self prediction with MAE as the pretext self supervised task.

Also, taking into consideration that most COVID-19 classification models applied to self supervision of radiographs outperform their imagenet counterparts, our pretext dataset is comprised of radiography reconstruction. This is feasible due to large amounts of Open Source available samples, such as CheXpert and RSNA Pneumonia Challenge (STEIN MD, 2018) dataset.

Our work also builds upon (GAZDA et al., 2021) and (XIAO et al., 2023), by updating our training set samples to a the current COVID-Net Open Source Initiative Dataset and studying applications regarding as to how retain best features separability, an often overlooked self supervision bottleneck (KUMAR et al., 2022), which is something we try to circumvent with an online semi-hard triplet loss. And, since works like (GAZDA et al., 2021) (XIAO et al., 2023) have already proven that self supervision outperforms imagenet pretrained backbones, we will not include this experiment in our comparison. (FEDORUK et al., 2023) also sheds a light into wether the usage of generative augmentation could provide performance increases, which is not the case.

4
Method

Our methodology is split into two stages, where our first step details on our ViT MAE training process, while the second step details our supervised step, through triplet loss. In order, our experiments are as following

1. Masked Auto-encoder;
2. Supervised cross-entropy approach;
3. Supervised cross-entropy loss transfer learning;
4. Supervised siamese neural network loss transfer learning;

4.1
Self-Supervised

Our first step is the application of a Masked Auto-encoder pipeline, where,

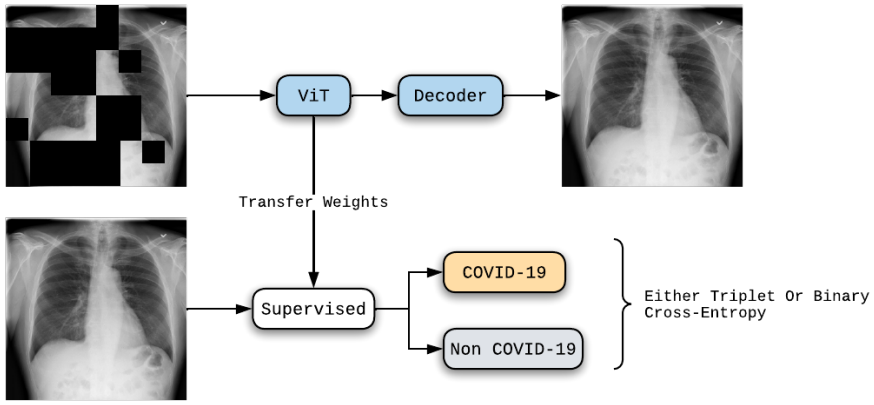


Figure 4.1: The first step, at the top, is the MAE pretraining process. After this, our weights are transferred to the supervised learning setting, over which we classify our samples between COVID-19 and non COVID-19.

The top part of our figure illustrates the masked encoding-decoding pipeline, where the goal of our ViT is to retain significant features across the first and second stages. These features, in turn, provide an latent space representation that should be separable. We aim to demonstrate differences in model performance by maintaining cohesive feature separation when performing transfer learning with self-supervised features. We expect reduced overfit by employing triplet margin into a downstream task.

4.2 Supervised

We employ semi-hard online triplet mining. This triplet loss is meant to calculate every permutation of positive, negative and anchor within the sampled subset. Then, by setting an small margin, only hard samples (i. e: near our classification boundary) are spaced out into their respective classification cluster. Our choice of implementation is based on Abadi et al. (2015) semi-hard triplet loss.

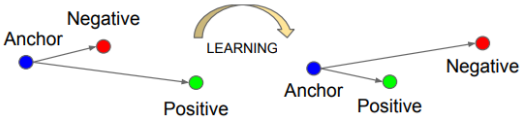


Figure 4.2: Triplet learning. The positive becomes closer to the anchor, while the negative distances itself from it (SCHROFF; KALENICHENKO; PHILBIN, 2015).

Figure 4.3 illustrates the triplet loss pipeline, where, after extracting our training set embeddings, the distribution is feed into a linear classifier. This classifier is a linear logistic regressor, meant to provide, in similar metrics to a sigmoid output layer, how separable are the current training embeddings of the learned data distribution. What that means is that our regressor is fit into $OPT(regressor(train_{embeddings}, \rightarrow\ train_{labels})$ and evaluated as $regressor(val_{embeddings})$.

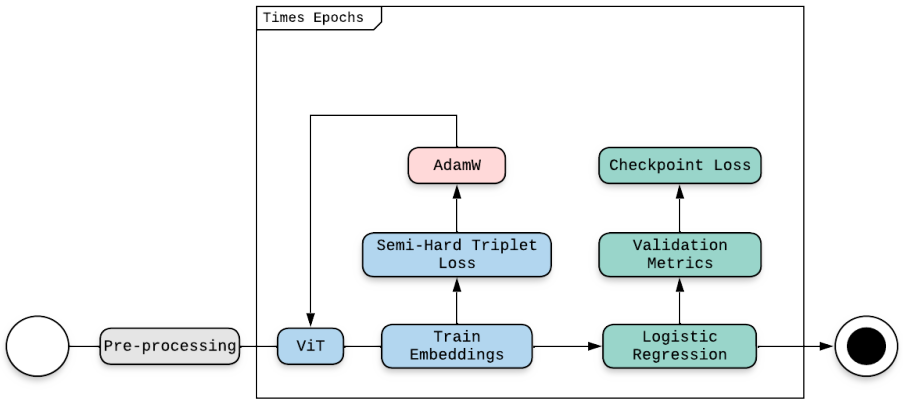


Figure 4.3: Our inputs are feed into our ViT, adjusted by Triplet Loss with AdamW (LOSHCHILOV, 2017). Then, every epoch, our fits our logistic regression algorithm samples through train embeddings, which is now able to classify our validation set. Every epoch the best scoring validation loss weight is registered.

This logistic regression method serves as the baseline for measuring the separability of our data points. That way, our training and validation subset

history across epochs can be compared to our regular BCE experiment.

4.3 Metrics

To evaluate the performance of our models, we used accuracy, F1 score (F1), Precision, Recall, Specificity. The accuracy is calculated with the following Equation 4-1,

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4-1}$$

TP is the True Positive, TN is the True Negative, FP is the False Positive, and FN is the False Negative.

Recall is defined by Equation 4-2; Specificity with 4-3; F1 score with 4-4 and Precision with 4-5.

$$recall = \frac{TP}{TP + FN} \tag{4-2}$$

$$specificity = \frac{TN}{TN + FP} \tag{4-3}$$

$$F1 = \frac{2 * precision * recall}{precision + recall} \tag{4-4}$$

$$precision = \frac{TP}{TP + FP} \tag{4-5}$$

For the binary classification setup, these metrics were directly extracted by the logits, while the siamese neural network generated a new embedding distribution within every epoch, which was applied to our linear regressor every validation step.

5
Results

This Chapter details on what datasets, experiments and measurements have been acquired through the application of the proposed method. Section 5.1, Experimental Setup, provides a brief overview of which experiments were executed over which datasets. Section 5.2 details acquired metrics with both MAE and without MAE applying BCE. Section 5.3, our Siamese Neural Network examples. The last Section, 5.4, presents improvements that may be inferred by results across every experiment.

5.1
Experimental Setup

Our experimental setup is intrinsically divided into two main stages. These are, a pre-training stage (1) and a classification stage (2-4). They are, then, split into 10 main experiments. These are,

1. MAE Reconstruction (1 Experiment): this is the reconstruction process, where our goal is to aquire a good embedding space representation of our domain.

2. BCE Classification (3 Experiments): here, we aim to set a baseline performance of our current classification model without any sort of pretraining. This is important in order to measure the influence of our MAE application.

3. BCE Classification with MAE weights (3 Experiments): this experiment sets what is currently expected from the standard classification process after fine tuning our MAE weights.

4. Siamese Neural Network with MAE weights (3 Experiments): our current proposition for better fine tuning compared to 3.

The first experiment is the self-supervision setting, where we employ the reconstruction set across the combined CheXPert (IRVIN et al., 2019), NIHCC (STEIN MD, 2018) and COVIDx (WANG; LIN; WONG, 2022) datasets. From step 2 to 4, the proposed experiments were reproduced via the usage of different training split sizes. This means that, for each of our supervised settings (2-4), we employ different sizes of training datasets. In total, we have executed 9 different supervised pipelines,

- BCE Classification,
    1. With 1% of the available training subset samples;
    2. With 10% of the available training subset samples;
    3. With 100% of the available training subset samples.
- BCE Classification with MAE weights,
    1. With 1% of the available training subset samples;
    2. With 10% of the available training subset samples;
    3. With 100% of the available training subset samples.
- Siamese Neural Network with MAE weights,
    1. With 1% of the available training subset samples;
    2. With 10% of the available training subset samples;
    3. With 100% of the available training subset samples.

Our goal with restricting the amount of available training data is to evaluate the robustness of the learned reconstruction features, as the learned representation should require fewer data to fit our test set distribution, at least when comparing the experiments that require pretrained weights to the others that do not. This is something that is usually performed to evaluate generalization with few-shot and no-shot self-supervised methods. The respective validation and test sets were not changed across each experiment.

5.1.0.1 Hyper-parameters

Overall, our choice of hyper-parameters were selected in order to fit according to both previous literature of masked auto-encoders and COVID classification models. This is done in order to provide a fair comparison between each approach, with either supervision and no self supervision.

Currently, many papers based on Dosovitskiy et al. (2020) simple ViT structure simulate their proposed architecture based on Base, Medium and Huge models. However, since works with similar goals have achieved promising results with even smaller ViTs when compared to base (), we can expect to deliver considerable performance with a "Small" ViT architecture. Tough some works refer to a ViT small as a consolidated network, there is not much consensus according to what is a ViT "Small" is. Our PyTorch (PASZKE et al., 2019) ViT Small encoder implementation is structured as following,

Table 5.1: ViT Small neural network structure layer by layer across each computing stage.

Layer	**Output Shape**	**Details**
Input	(B, 3, 320, 320)	Input batch of images
Rearrange	(B, N, P)	N = number of patches, P = 20×20×3
Layer Norm	(B, N, dim)	Normalize patch embeddings
Linear	(B, N, 384)	Project patches to 384 dimensions
Layer Norm	(B, N, 384)	Normalize again
Pos Embedding	(B, N + 1, 384)	Add positional encoding and CLS token
Dropout	(B, N + 1, 384)	Apply dropout
Transformer x12	(B, N + 1, 384)	
- Layer Norm	(B, N + 1, 384)	Normalize input
- MSA	(B, N + 1, 384)	Compute attention weights and output
- Residual	(B, N + 1, 384)	Add input to attention output
- Feed Forward	(B, N + 1, 384)	MLP with two linear layers
- Residual	(B, N + 1, 384)	Add input to MLP output
Identity	(B, 384)	Pass through
Linear	(B, 1)	Final layer for classification

Our choice of input size is approximate to the state of the art chest radiograph classification models. Since COVID-19 binary classification is naturally less complex than an 14 class multi-label setting, like Yuan et al. (2021) (320x320), our choice of resolution should not be a bottleneck to our classification performance. Aside from this, the higher resolutions would require an even grater amount of computing power, which is a limiting factor when operating with more then 374.379 images.

As for our MAE encoder, we employ a two layer 2 layer transformer decoder with latent representation of dimensionality 512 and 75% masking ratio, following the decoder size recommendations according to He et al. (2022) for fine tuning MAE.

Aside from this, like (HE et al., 2022) and (CHEN et al., 2020a), our work applies the usage of regularized version of AdamW in order to better preserve our feature encoder distribution, AdamW (LOSHCHILOV, 2017). It is a popular version of Adam that studies have shown to achieve better performance in SSL tasks compared to those using other optimizers, particularly in scenarios where fine-tuning on downstream tasks is required, such as our task. Another important factor that should be made explicit is that, while training with a learning rate greater than 1-3e, none of our experiments would converge, neither across reconstruction or classification models. Both reconstruction and fine tuning were trained across 100 epochs with a batch size of 64.

5.1.1 Self-Supervised Dataset

Our current task of choice comes from radiography medical imaging. Over recent years, many radiography datasets have been created for the purpose of deep learning classification frameworks, such as CheXpert (IRVIN et al., 2019) (UK), RNSA (STEIN MD, 2018) (USA), COVIDx (WANG; LIN; WONG, 2020) (Global). Historically, they target mainly pneumonia and, most recently, COVID-19 automatic diagnosis. Due to uncertainty, hard to control clinical setting and biases, most of these datasets cannot be readily mixed into one simple supervised setting.

Our main proposition is to use the largest available x-ray datasets and repurposed them into self-supervision. After removing redundancies, such as RNSA samples inside COVIDx, and lateral radiography's from CheXpert, we arrive at 374.379 frontal radiography samples, ranging from healthy, pneumonia, COVID and many other diagnosis. 37.437 samples of our pretraining subset were randomly selected for our validation set in order to measure our reconstruction task convergence.

Dataset	Samples	Total
CheXpert	191.229	
RNSA	112.120	
COVIDx	71.030	374.379
Total Training	336.942	
Total Validation	37.437	374.379

Table 5.2: Dataset Summary

With this dataset, we are able to cover a wide range of diseases, countries of origin, acquisition methods, old and new samples. As of now, is the largest frontal radiography dataset. They are all public, providing even more transparency.

Figure 5.1 displays the learning curve of our model at the reconstruction task. We can see that, our model loss decreases steadily until epoch 50, while slowly entering a plateau between 50 and 100. With this we can infer that the reconstruction task was successfully learned by our reconstruction model and therefore, should contain a good embedding representation of our inputs after 100 epochs.

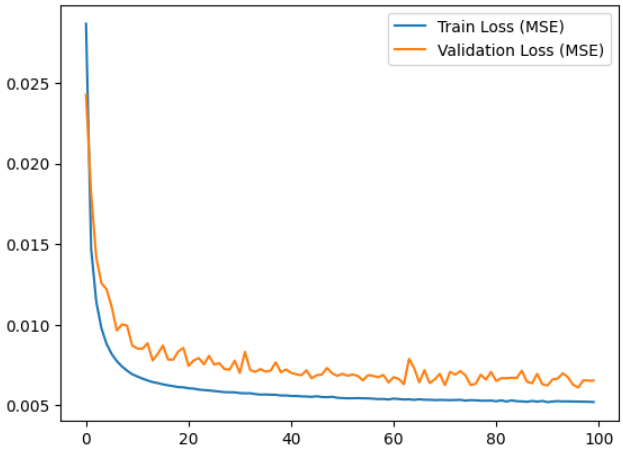


Figure 5.1: The history of our MAE train loss, MSE, across each epoch while performing the pretraining process. These are the training loss (blue) and validation loss (orange).

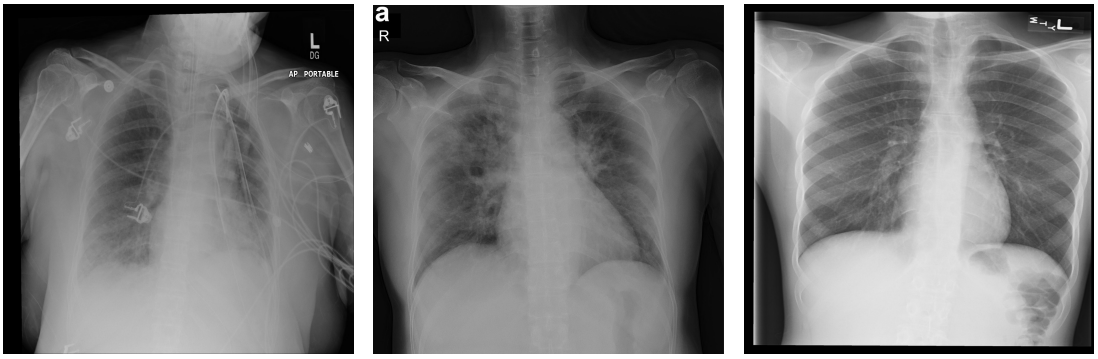
5.1.2 Supervised Dataset

After our pre-training process, our downstream task is set to COVID classification over the COVIDx-4 open source dataset. This dataset was selected as it is one, if not the most diverse and reviewed chest x-ray datasets for COVID-19 detection, while being the most recently updated version of the COVIDx dataset suited for binary classification. This dataset is a collection of different publicly available datasets. Namely,

- ActualMed COVID-19 Chest X-ray Dataset Initiative (WANG; LIN; WONG, 2020);
- RSNA Pneumonia Detection Challenge (STEIN MD, 2018);
- RICORD COVID-19 (TSAI et al., 2021);
- COVID-19 Image Data Collection (COHEN; MORRISON; DAO, 2020);
- BIMCV-COVID19 (VAYÁ et al., 2020);
- COVID-19 radiography database (CHOWDHURY et al., 2020).

Each of these sources have been updated across each year, whereas the current version of COVIDx CRX 4 is from October 2023. It provides 67.864 training samples and 8.482 validation samples. As for our test set of choice, the COVIDx 9B dataset was selected. Most peer-reviewed models are, currently, evaluated over this set of images, including the current state of art COVIDNet-CXR-3 model. This set of images include 200 non COVID-19 samples. These range from non-COVID-19 pneumonia and healthy samples. The other 200 samples are confirmed COVID-19 positives.

Some of the supervised dataset samples can be seen at Figures 5.2a, 5.2b and 5.2c, being respectively, non-COVID-19 pneumonia, COVID-19 positive and a healthy samples.



(a) Pneumonia (b) COVID-19 (c) Healthy

Figure 5.2: Samples from COVIDx 9B Dataset (WANG; LIN; WONG, 2020).

In summary, our supervised setting is composed of,

Table 5.3: Supervised Experiments Dataset.

Dataset	Total	Positive	Negative
Train	67.864	57.200	10.664
Validation	8.482	4.241	4.241
Test	400	200	200

5.2 Cross Entropy

We executed the training process using weighted resampling of the training dataset, ensuring that each batch was approximately balanced. This is done in both experiments in order to provide a fair comparison. Table 5.4 presents our current results for our ViT Small architecture within the COVIDx 9 test dataset.

Table 5.4: Results of training from scratch with BCE. The scores are based on COVIDx 9 test dataset. The percentage column refers to how much of our training subset was selected.

Percentage	Accuracy	F1	Recall	Specificity	Precision
100%	92.50%	92.35%	90.50%	94.50%	94.27%
10%	85.00%	84.29%	80.50%	89.50%	88.46%
1%	84.25%	84.29%	84.50%	84.00%	84.08%

There were not any particular changes to this experiments beside the loading of the MAE encoding network weights into our ViT Small. Table 5.5

refers to our current results for our ViT Small architecture within the COVIDx 9 test dataset by training with MAE weights.

Table 5.5: Results of pretraining with MAE and fine tuning with BCE. The scores are based on COVIDx 9 test dataset. The percentage column refers to how much of our training subset was selected.

Percentage	Accuracy	F1	Recall	Specificity	Precision
100%	97.50%	97.44%	95.00%	100.00%	100.00%
10%	98.00%	97.98%	97.00%	99.00%	98.98%
1%	97.25%	97.19%	95.00%	99.50%	99.48%

5.3 Siamese Neural Networks

Siamese neural networks require more specialized regularization. Their output needs to regularized in order to not fall out to infinity or converge another class to a single point cluster. This is effectively done by adding a dropout layer, 1D batch normalization layer and an l2 regularization custom layer. Other small adjustments include dropping the last batch of our iteration process, as it becomes statically likely to provide an unbalanced dataset if our batch size diminishes.

Table 5.6 refers to our results with our siamese neural network on the COVIDx 9 test dataset.

Table 5.6: Results of pretraining with MAE and fine tuning siamese neural networks. The scores are based on COVIDx 9 test dataset. The percentage column refers to how much of our training subset was selected.

Percentage	Accuracy	F1	Recall	Specificity	Precision
100%	98.50%	98.48%	97.00%	100.00%	100.00%
10%	97.00%	96.91%	94.00%	100.00%	100.00%
1%	97.25%	97.17%	94.50%	100.00%	100.00%

5.4 Discussion

The loss curve displayed at Figure 4.1 suggests that our MAE model was able to learn how to proper embedding representation, further corroborated by comparing 5.4 and 5.5, where even when utilizing as much as 1% of the original set, our pretrained model outperformed every training approach. Also, this model checkpoint was recorded at epoch 15 out of 100, while our binary cross-entropy model trained up until epoch 95. Considering that there were

many more batch iterations across the complete set, the first model trained order of magnitudes faster then the not pretrained model.

As seen on Table 5.6, our current siamese neural network approach surpasses the current most accurate COVID-Net model, from **98.25%** accuracy to **98.50%** over COVIDx 9 test set. It also provides a more efficient model, measuring 5.937.537.024 Multiply-Accumulate Operations (5.937 GMACs), approximately 20.4% of COVIDNet-CXR-3.

Also, while comparing our pretrained approaches to the not pretrained approach, we can see at Figure 5.3 that there is a huge gap in performance between using and not using MAE. This has been already reflected by the test result gap up to 5% accuracy, however, the regular approach possesses a notably slower training process.

When comparing our MAE fine tuned through BCE to our Siamese Neural Network, we can see at Tables 5.5 and 5.6 that it surpasses BCE performance by +1% at COVIDx 9 test set. We can see that this does not only extend to our test set, but to our validation set from COVIDx CRX 4. Figure 5.3 displays the validation score of each approach. The siamese neural network accuracy asymptotically dominates the BCE model curve.

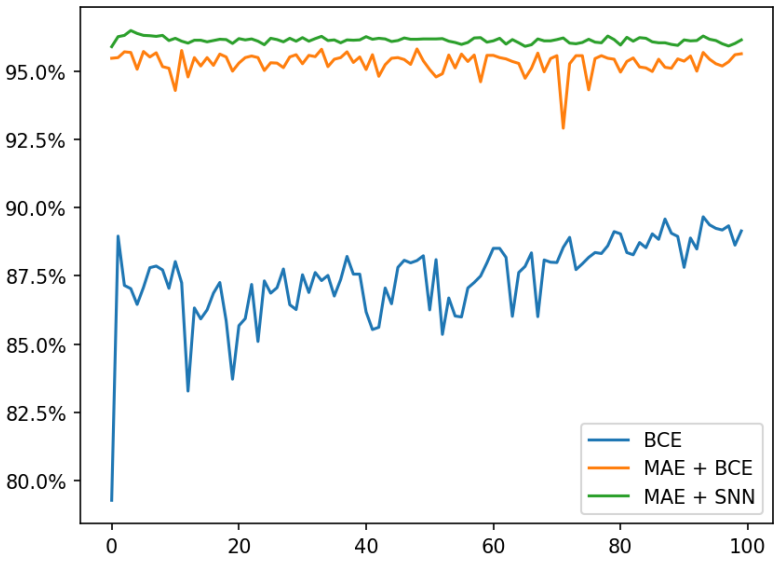


Figure 5.3: The history of the validation accuracy across each epoch when performing fine tuning with our proposed siamese neural network (Green), MAE with BCE (Orange) and no pretrained weights (Blue).

Also, we can see that the accuracy curve of our triplet loss model provides a more stable learning process, as local minimums are more frequent through our learning process, dipping the validation accuracy. Given that our goal is to build upon as much of our pretraining as possible, this is also a positive outcome from our Siamese approach.

The positive effects of our new approach were further compared by directly analyzing the linear separability of the embedding space through dimensionality reduction by t-SNE (MAATEN; HINTON, 2008). With t-SNE, we are able to project the previous embedding space distribution of our test set and see if after fine tuning we are able to preserve said features. Figures 5.4 and 5.5 displays the embedding space projection of our MAE after 100 epochs of training.

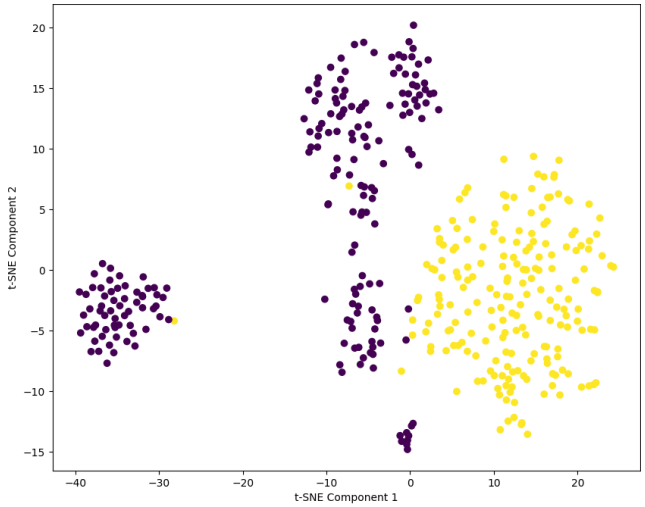


Figure 5.4: The t-SNE projection of the embedding space after training with MAE. Each sample is one of the COVIDx 9 400 samples test set. In purple, we have the negative COVID samples, while in yellow, COVID positive samples.

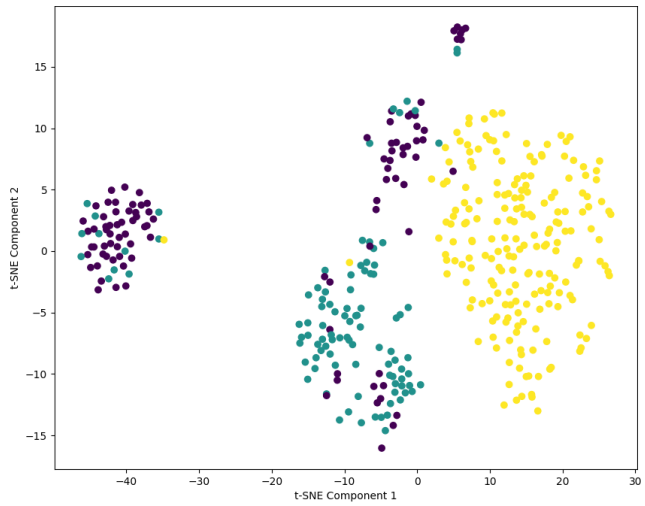


Figure 5.5: The t-SNE projection of the embedding space after training with MAE. Each sample is one of the COVIDx 9 400 samples test set. In purple, we have healthy samples, non-COVID pneumonia samples in green and, with yellow, COVID positive samples.

With the embedding space projections of Figure 5.4, we are able to see that the MAE provides us with separable embeddings, with a shared

boundary across component 1 that is still distinct across each cluster. It might be expected that, since there are three visible clusters, the non-COVID cluster should be split into either healthy and pneumonia positive samples. However, at Figure 5.5, we see that these labels are not visibly separable.

Figures 5.6 and 5.7 display the embedding space representation after fine tuning with our current SNN. What happened has that the components that were near the rightmost cluster were pushed further apart to the left, while samples that were notably inside one of the respective clusters were not shifted.

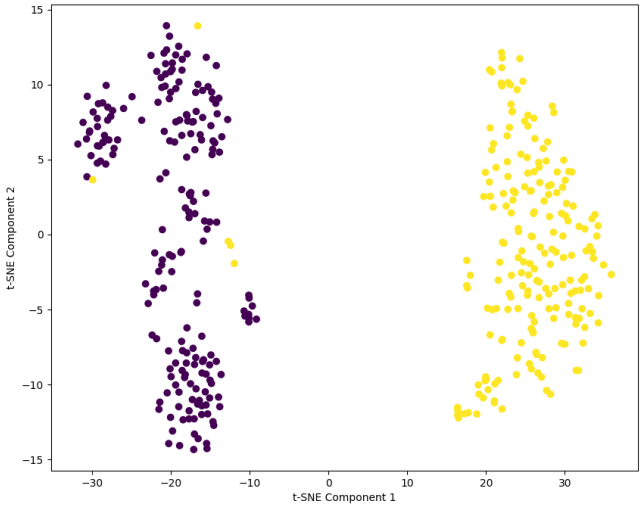


Figure 5.6: The t-SNE projection of the embedding space after fine tuning our SNN. Each sample is one of the COVIDx 9 400 samples test set. In purple, we have the negative COVID samples, while in yellow, COVID positive samples.

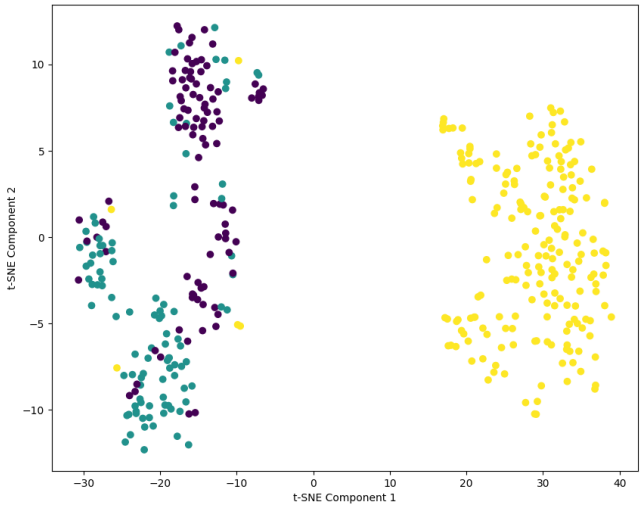


Figure 5.7: The t-SNE projection of the embedding space after fine tuning our SNN. Each sample is one of the COVIDx 9 400 samples test set. In purple, we have healthy samples, non-COVID pneumonia samples in green and, with yellow, COVID positive samples.

At Figure 5.7, we are able to see that even after fine tuning, the embedding space representation of the pneumonia samples were consistently at the same range across the y-axis, where most negative sample were split in two from top half to bottom half. This can be stated to be a positive results, as the new embedding space representation has roughly the same shape as the previous, indicating that our new model does not collapse our inputs into one single instance and should be able to keep most of our self-supervised learning.

With Figures 5.8 and 5.9 we can see some problems pertaining to directly fine tuning the ViT. While these results offer clearly separable embeddings, there is barely any cohesion left from the pretrained weights, providing a collapsed representation of the previous deep features. This is not a desirable outcome from a generalization standpoint, as the resulting model is likely to not generalize.

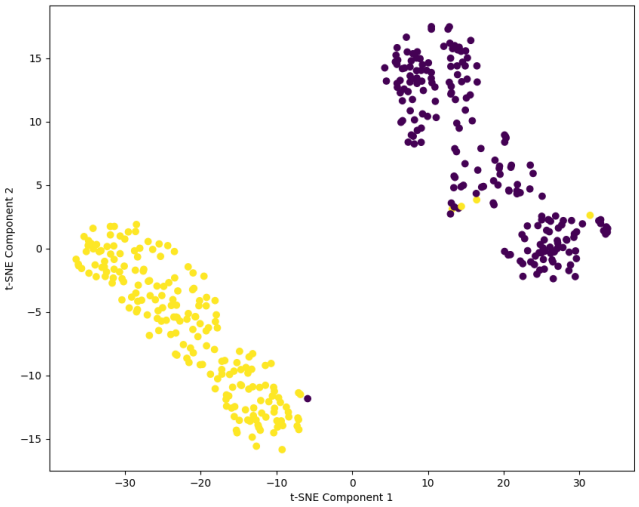


Figure 5.8: The t-SNE projection of the embedding space after fine tuning our MAE with BCE. Each sample is one of the COVIDx 9 400 samples test set. In purple, we have the negative COVID samples, while in yellow, COVID positive samples.

Table 5.7 displays the aforementioned results when compared to the state of art COVID detection model, COVID-Net.

Table 5.7: Comparison with the State Of Art.

Method	Accuracy	*Recall*	*GMACs*
BCE	92.50%	90.50%	5.8701
BCE + MAE	98.00%	97.00%	5.8701
SNN + MAE	**98.50%**	**97.00%**	5.937
(WANG; LIN; WONG, 2020)	98.25%	**97.00%**	29.1

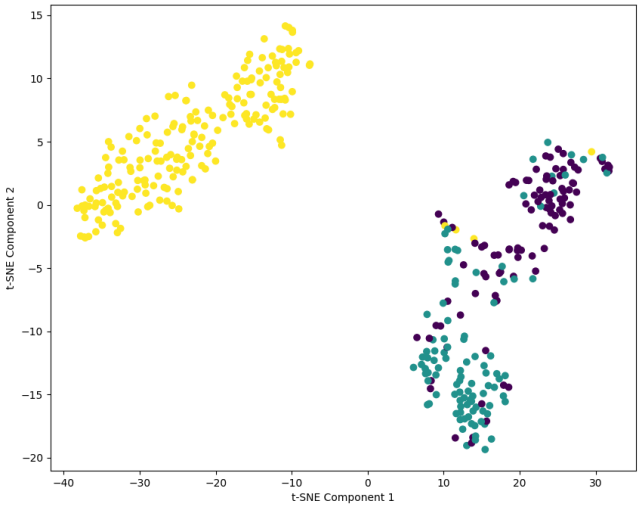


Figure 5.9: The t-SNE projection of the embedding space after fine tuning our MAE with BCE. Each sample is one of the COVIDx 9 400 samples test set. In purple, we have healthy samples, non-COVID pneumonia samples in green and, with yellow, COVID positive samples.

6
Conclusions

Within this work, we select from a wide range of open available datasets to pretrain visions transformer through MAE, to evaluate new fine tuning approaches. A small ViT is successfully develop and pretrained in over 330.000 frontal chest radiographies. To the author's knowledge, it currently surpasses all known peer-reviewed CRX MAE dataset experiments in size.

After this, we comparatively experiment over multiple subsets of COVIDx Open Source Initiative. These splits are set from 1%, 10% up until 100% of labeled data, in order to gauge how data availability affects our pretrained models. As expected, image availability decreased accuracy of non pretrained models up to -8.25% accuracy. However, the pretrained ViTs performed roughly the same regardless of the split.

As another contribution, this work surpasses the state of art deep learning models through a junction of MAE and siamese neural network. By leveraging the implicit separability of our pretrained embeddings, we were able to perform accurate fine tuning by targeting nearly separable embeddings, through semi-hard triplet sampling and margin. With this, our method overcomes the current most accurate model with 98.5% accuracy and surpasses our own BCE baseline, by +1%.

Additionally, our model offers further computational efficiency, requiring only around 20.4% of the Multiply-Accumulate Operations of the best-performing COVID-Net model.

As for future work, we expect to provide a harder classification task, so that there are more meaningful differences between each our siamese neural network and the cross-entropy based model. Besides this, further ViT scaling should yield more accurate results. And by employing a more sophisticated embeddings classification algorithm. Self supervising our validation distribution could further boost our available analysis.

We also intent to evaluate more intermediate embeddings to be able to make neural architectural search. It is expected that any intermediate computation of our ViT is sufficiently separable, therefore, the network may be pruned to minimize computation, parametrization and possible overfit.

7
Bibliography

ABADI, M. et al. **TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems**. 2015. Software available from tensorflow.org. Disponível em: <http://tensorflow.org/>.

BAYDIN, A. G. et al. Automatic differentiation in machine learning: a survey. **Journal of machine learning research**, v. 18, n. 153, p. 1–43, 2018.

BENGIO, Y.; COURVILLE, A.; VINCENT, P. Representation learning: A review and new perspectives. **IEEE transactions on pattern analysis and machine intelligence**, IEEE, v. 35, n. 8, p. 1798–1828, 2013.

CARON, M. et al. Emerging properties in self-supervised vision transformers. In: **Proceedings of the IEEE/CVF international conference on computer vision**. [S.l.: s.n.], 2021. p. 9650–9660.

CHEN, T. et al. A simple framework for contrastive learning of visual representations. In: PMLR. **International conference on machine learning**. [S.l.], 2020. p. 1597–1607.

CHEN, T. et al. **Big Self-Supervised Models are Strong Semi-Supervised Learners**. arXiv, 2020. Disponível em: <http://arxiv.org/abs/2006.10029>.

CHOWDHURY, M. E. et al. Can ai help in screening viral and covid-19 pneumonia? **Ieee Access**, IEEE, v. 8, p. 132665–132676, 2020.

COHEN, J. P.; MORRISON, P.; DAO, L. **COVID-19 Image Data Collection**. 2020. Disponível em: <https://arxiv.org/abs/2003.11597>.

CONSTANTINOU, M. et al. Covid-19 classification on chest x-ray images using deep learning methods. **International Journal of Environmental Research and Public Health**, v. 20, 2023. Disponível em: <https://api.semanticscholar.org/CorpusID:256216270>.

CYBENKO, G. Approximation by superpositions of a sigmoidal function. **Mathematics of control, signals and systems**, Springer, v. 2, n. 4, p. 303–314, 1989.

DEVLIN, J. Bert: Pre-training of deep bidirectional transformers for language understanding. **arXiv preprint arXiv:1810.04805**, 2018.

DOSOVITSKIY, A. et al. **An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale**. arXiv, 2020. Disponível em: <http://arxiv.org/abs/2010.11929>.

FEDORUK, O. et al. Performance of gan-based augmentation for deep learning covid-19 image classification. **ArXiv**, abs/2304.09067, 2023. Disponível em: <https://api.semanticscholar.org/CorpusID:258187484>.

GAZDA, M. et al. Self-supervised deep convolutional neural network for chest x-ray classification. **IEEE Access**, v. 9, p. 151972–151982, 2021.

GOODFELLOW, I. et al. Generative adversarial networks. **Communications of the ACM**, ACM New York, NY, USA, v. 63, n. 11, p. 139–144, 2020.

GRILL, J.-B. et al. Bootstrap your own latent-a new approach to self-supervised learning. **Advances in neural information processing systems**, v. 33, p. 21271–21284, 2020.

HADSELL, R.; CHOPRA, S.; LECUN, Y. Dimensionality reduction by learning an invariant mapping. In: IEEE. **2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)**. [S.l.], 2006. v. 2, p. 1735–1742.

HE, K. et al. Masked autoencoders are scalable vision learners. In: **Proceedings of the IEEE/CVF conference on computer vision and pattern recognition**. [S.l.: s.n.], 2022. p. 16000–16009.

HE, K. et al. Momentum contrast for unsupervised visual representation learning. In: **Proceedings of the IEEE/CVF conference on computer vision and pattern recognition**. [S.l.: s.n.], 2020. p. 9729–9738.

HE, K. et al. Deep residual learning for image recognition. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2016. p. 770–778.

HINTON, G.; VINYALS, O.; DEAN, J. Distilling the knowledge in a neural network. **arXiv preprint arXiv:1503.02531**, 2015.

HUANG, G. et al. Densely connected convolutional networks. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2017. p. 4700–4708.

IRVIN, J. et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In: **Proceedings of the AAAI conference on artificial intelligence**. [S.l.: s.n.], 2019. v. 33, n. 01, p. 590–597.

KINGMA, D. P. Auto-encoding variational bayes. **arXiv preprint arXiv:1312.6114**, 2013.

KOCH, G. et al. Siamese neural networks for one-shot image recognition. In: LILLE. **ICML deep learning workshop**. [S.l.], 2015. v. 2, n. 1, p. 1–30.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. **Advances in neural information processing systems**, v. 25, 2012.

KUMAR, A. et al. Fine-tuning can distort pretrained features and underperform out-of-distribution. **arXiv preprint arXiv:2202.10054**, 2022.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **nature**, Nature Publishing Group UK London, v. 521, n. 7553, p. 436–444, 2015.

LECUN, Y. et al. Handwritten digit recognition with a back-propagation network. **Advances in neural information processing systems**, v. 2, 1989.

LOSHCHILOV, I. Decoupled weight decay regularization. **arXiv preprint arXiv:1711.05101**, 2017.

MAATEN, L. van der; HINTON, G. Visualizing data using t-sne. **Journal of Machine Learning Research**, v. 9, n. 86, p. 2579–2605, 2008. Disponível em: <http://jmlr.org/papers/v9/vandermaaten08a.html>.

MINSKY, M.; PAPERT, S. An introduction to computational geometry. **Cambridge tiass., HIT**, v. 479, n. 480, p. 104, 1969.

OPENAI. **ChatGPT**. 2024. <https://www.openai.com/chatgpt>.

PASZKE, A. et al. Pytorch: An imperative style, high-performance deep learning library. In: **Advances in Neural Information Processing Systems 32**. Curran Associates, Inc., 2019. p. 8024–8035. Disponível em: <http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf>.

RAJPURKAR, P. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. **ArXiv abs/1711**, v. 5225, 2017.

SCHROFF, F.; KALENICHENKO, D.; PHILBIN, J. Facenet: A unified embedding for face recognition and clustering. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2015. p. 815–823.

SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. **arXiv preprint arXiv:1409.1556**, 2014.

STEIN MD, C. W. C. C. G. S. J. D. k. L. C. L. P. M. K. M. M. M. P. P. C. S. H. M. T. X. A. **RSNA Pneumonia Detection Challenge**. Kaggle, 2018. Disponível em: <https://kaggle.com/competitions/rsna-pneumonia-detection-challenge>.

TSAI, E. B. et al. The rsna international covid-19 open radiology database (ricord). **Radiology**, Radiological Society of North America, v. 299, n. 1, p. E204–E213, 2021.

VASWANI, A. Attention is all you need. **Advances in Neural Information Processing Systems**, 2017.

VAYÁ, M. D. L. I. et al. Bimcv covid-19+: a large annotated dataset of rx and ct images from covid-19 patients. **arXiv preprint arXiv:2006.01174**, 2020.

VINCENT, P. et al. Extracting and composing robust features with denoising autoencoders. In: **Proceedings of the 25th international conference on Machine learning**. [S.l.: s.n.], 2008. p. 1096–1103.

WANG, L.; LIN, Z. Q.; WONG, A. COVID-net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest x-ray images. v. 10, n. 1, p. 19549, 2020. ISSN 2045-2322. Number: 1 Publisher: Nature Publishing Group. Disponível em: <https://www.nature.com/articles/s41598-020-76550-z>.

WANG, L.; LIN, Z. Q.; WONG, A. **COVID-Net**. 2022. Https://github.com/lindawangg/COVID-Net. Accessed: 2024-10-29.

WANG, T. et al. Pneunet: deep learning for covid-19 pneumonia diagnosis on chest x-ray image analysis using vision transformer. **Medical & Biological Engineering & Computing**, v. 61, p. 1395 – 1408, 2023. Disponível em: <https://api.semanticscholar.org/CorpusID:256414929>.

XIAO, J. et al. Delving into masked autoencoders for multi-label thorax disease classification. In: **Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)**. [S.l.: s.n.], 2023. p. 3588–3600.

YUAN, Z. et al. Large-scale robust deep auc maximization: A new surrogate loss and empirical studies on medical image classification. In: **Proceedings of the IEEE/CVF International Conference on Computer Vision**. [S.l.: s.n.], 2021. p. 3040–3049.