UNSW
AUSTRALIA

SELF SUPERVISED FRAMEWORK TO ADDRESS LIMITED DATA FOR CANCER DIAGNOSIS

Hamish Haggerty

Supervisor: Dr Rohitash Chandra

School of Mathematics and Statistics
UNSW Sydney

June 2023

Submitted in partial fulfilment of the requirements of the degree of
Master of Statistics

Plagiarism statement

I declare that this thesis is my own work, except where acknowledged, and has not been submitted for academic credit elsewhere.

I acknowledge that the assessor of this thesis may, for the purpose of assessing it:

- Reproduce it and provide a copy to another member of the University; and/or,
- Communicate a copy of it to a plagiarism checking service (which may then retain a copy of it on its database for the purpose of future plagiarism checking).

I certify that I have read and understood the University Rules in respect of Student Academic Misconduct, and am aware of any potential plagiarism penalties which may apply.

By signing this declaration I am agreeing to the statements and conditions above.

Signed: _____ Date: _____

Acknowledgements

Thank you to my supervisor, Dr Rohitash Chandra for his advice and support at all stages of this work. He encouraged me to explore an applied project that has been both rewarding and interesting. I am grateful for his time, patience and guidance.

Hamish Haggerty, 21 April 2023.

Abstract

Cancer diagnosis is a well studied problem in machine learning. The early detection of cancer is often the determining factor in prognosis. Supervised deep learning achieves excellent results in cancer image classification, usually through transfer learning. These models still require large amounts of labelled data and for several types of cancer large datasets do not exist. Self-supervised learning (SSL) is an approach to pre-training models for downstream transfer that does not require labelled data, and has recently shown excellent performance on large datasets such as *ImageNet*. In this work, we demonstrate that a model pretrained using an SSL algorithm called *Barlow Twins* can outperform the conventional supervised transfer learning pipeline. We select an appropriate skin lesion dataset to demonstrate an application where limited data is available during training, i.e the training set is much smaller than the test set. We achieve a mean test accuracy of 66% for supervised transfer and 70% for self-supervised transfer learning. This work may be applicable to improving performance of cancer image classification models in the low labelled data regime, for example oral cancer diagnosis.

Contents

Chapter 1

Introduction

Cancer is the second leading cause of death worldwide, with almost 10 million deaths estimated in 2020 [20]. For many types of cancer (e.g. skin, oral, pancreatic), early diagnosis is the major determining factor in prognosis [37, 21, 40]. If cancer is detected sufficiently early, survival rates may be above 90% [59, 39, 78]. On the other hand, the prevalence of several cancer types is increasing [78], and this is particularly true in poorer communities in the developing world [67]. This is of particular concern since individuals in such communities have lower (if any) access to the expert clinicians traditionally needed to diagnose such cancer. There is thus an urgent need to develop cheaper and more data efficient methods of diagnosis that can be deployed globally. The use of machine learning and artificial intelligence has the potential to provide automated online diagnosis, breaking barriers internationally [16]. Such systems can guide or be leveraged in decision making of medical practitioners [38] which can provide huge benefits to disadvantaged communities, such as rural areas of developing countries [26].

In recent years, there has been significant interest in applying machine learning to cancer image diagnosis [49]. This includes lung, breast and several other forms of cancer, and involves classifying clinical images into categories ("malign" or "benign") or more fine grained classification [45, 15, 80]. There has been some challenges in the use of deep learning methods since they require large and class balanced datasets to function properly. The availability of timely data with proper organisation has been a challenge in the medical domain due to restrictions in archive access given patient confidentiality and ethical approval concerns [71]. In general, the collection of labelled datasets is a nontrivial issue and requires significant time and money to collect and annotate [14]. This is even more pronounced for medical images, which require expert knowledge or even medical testing (e.g. via biopsy) to classify, dramatically increasing the cost [33, 13]. Thus for several types of cancer, image datasets suitable for machine learning research are lacking. For example Sengupta et al. [63] demonstrated that there is a dearth of publically available oral cancer datasets. Although there has been published research on oral

cancer image classification, the data used is closed source - our attempt at obtaining data by contacting primary authors was not successful. Oral cancer is diagnosed through biopsy, but the decision to biopsy is made by visual inspection of the mouth [6], and early detection is crucial for prognosis. In the United States, the 5 year survival with early detection is 85%, but only 28% of cases are detected at this stage. In later stages, when the cancer has spread, 5 year survival drops to 40% [9]. The lack of available datasets is concerning given that the prevalence of oral cancer is increasing,[78] particularly among poorer parts of the developing world, where expert medical care is less available. As an additional point, certain kinds of cancer have extremely low prevalence. For example chordoma has a prevalence of only 0.18-0.84 per million [2], so constructing a large labelled dataset in such cases is a long way off. Developing more data efficient techniques is therefore essential.

One oral cancer study is by Song et al [67]. The dataset in their study involved 3851 cheek mucosa images, labelled into 4 categories (2417 normal, 1100 premalignant, 243 benign, 91 malign).[1] Like other work, they utilised a network pretrained from ImageNet as their backbone model, fine tuning it on the current dataset. The baseline accuracy of this model was 81% whereas the baseline balanced accuracy was 62%. The reason for this discrepancy is that the dataset is not balanced. To improve the performance the authors use a combination of undersampling of the majority class and oversampling of the minority classes (through data augmentation). After doing so the accuracy is still 81% but the balanced accuracy increased to 80%. Brinker et al. [8] considered a balanced training set of 4204 melanoma and nevi mole images, where the classification had been determined by biopsy. Note that although this training set is smaller than other studies, it is a binary classification problem. They also utilised a transfer learning approach, via a ResNet-50 pretrained on ImageNet.[2] On the metrics of sensitivity and specificity, they found that the trained network outperformed dermatologists. The sensitivity and specificity was around (67.2%, 62.2%) for dermatologists and (82.3%,77.9%) for the neural network, a significant margin.

Typically, supervised deep learning involves first randomly initialising the weights of a neural network, and then iteratively updating the weights through backpropagation and stochastic gradient descent [23]. A very well known problem with this approach is that for challenging high dimensional problems, large amounts of labelled data are required to achieve good performance. Transfer learning is a machine learning technique that attempts to solve this problem. The idea is to initialise the neural network weights with those of a network that has already been

[1]Note that chance accuracy on this problem is 25%.

[2]ResNet-50 is also the architecture primarily used in the present work.

trained (traditionally in a supervised fashion) on a related task [87]. A common strategy in computer vision applications is to take the initial network to be one that has been pretrained on ImageNet [87]. It has been demonstrated that transfer learning can lead to significant performance gains compared to training from random initialisation [68].

Self-supervised learning (SSL) is a machine learning technique that involves training a model on unlabelled data in order to learn a good internal representation for downstream tasks [36]. The model uses the data itself to develop a surrogate target, which is the main distinguishing feature from unsupervised learning, although the boundary is not well defined. With this definition, denoising autoencoders [76] would be self-supervised models, whereas principal component analysis (PCA) [55] is unsupervised but not self-supervised. SSL has had huge success in natural language processing (NLP) problems such as language translation, topic modelling and sentiment analysis[48]. At a high level, the approach is to take an input sentence (or sequence) and "blank out" some of the text. The goal is to predict the missing text from the available text, usually using a transformer architecture and trained in an autoregressive fashion [74, 47]. Hence, large language models such as *generalised pre-trained transformers* (GPT-x) [57], and machine translation engines [79] such as *Google Translate* are based on self-supervised learning. SSL has historically been less successful in computer vision tasks [47], since visual images are inherently of much higher dimension. It is possible to represent a probability distribution over words exactly (there are $\sim$ 60k words in the English language), but doing this for natural images remains a challenge [47]. Despite this, in the last few years a particular form of SSL known as 'joint embedding architecture' applied to computer vision tasks has had significant success [28, 83, 25], rivaling supervised approaches in some cases. Therefore, SSL has a huge potential for medical diagnosis that has image data with certain limitations such as class imbalance and limited training data.

1.0.1 Overview of paper

The use of transfer learning is common in the medical diagnosis literature concerning images [87] such as skin cancer [42] and lung cancer diagnosis [77]. Usually the models involved have been pretrained on ImageNet in a *supervised* fashion. This raises the question - can we do better? Since many studies are based on this framework, any improvement has significant clinical implications. This is of particular interest in the low data regime.

In this paper, we apply SSL and transfer learning to the problem of skin cancer diagnosis based on image data, which is imbalanced and has a limited set of training

data. We compare models pretrained the usual way via supervised learning on ImageNet, to networks pretrained through self-supervised learning, also on ImageNet. We use *International Skin Imaging Collaboration (ISIC)* [34] image-based dataset with training set of 2554 instances and a test set of size 19423 instances. As such, the training set mimics the typical low data setting of such cases.

Our work is organised as follows. We first give a brief review of convolutional networks, ImageNet and residual networks for context. Next follows a discussion of related work on image based medical classification (primarily cancer) through transfer learning, including a review of the technique of transfer learning. Following this is an extensive discussion of self-supervised learning. We give a general overview of the technique of SSL, with discussion on both a formal and intuitive level and discuss several modern algorithms. The discussion pays particular attention to Barlow Twins, since that is the algorithm primarily used in our work. Next we describe our experimental methodology, including the dataset and implementation details. We then discuss our results and consider directions for future work.

Chapter 2

Background and Related Work

2.0.1 Convolutional Neural Networks

Convolutional neural networks are feedforward networks built by stacking convolution layers, max pooling layers and standard fully connected layers [46]. Convolution layers are the main innovation of CNN's and are motivated by the notion of receptive field in visual neuroscience [32]. Such layers involve several filters each of which are convolved across the input, taking the Frobenius inner product with each region. These outputs are then stacked into channels. If the input to the layer has shape $n \times m \times c$ (e.g. an input image of shape $n \times m$ with c colour channels), and there are f filters of shape $a \times b$[1] then the output of the layer will have shape $((n-a)+1) \times ((m-b)+1) \times f$. (This assumes zero padding and a sliding window of 1). In this way, convolution layers use far less parameters than fully connected layers, and CNNs can be seen as a regularised version of multi layer perceptrons. Max pooling layers downsample their input, but with the innovation of skip connections are not used as much in such residual architectures [29].

2.0.2 ImageNet

ImageNet is a large annotated database of over 14 million natural images [70]. ImageNet 1k is a subset of about 1.2 million images distributed across 1000 categories [60, 14]. (Henceforth, when we refer to ImageNet we mean ImageNet 1k). Some example categories are 'magpie', 'taxi', 'crane'. In transfer learning applications it is common to take the initial network to be one trained on ImageNet in a supervised fashion [68]. The size of the dataset along with the large number of categories means such networks must learn informative features in order to perform ImageNet classification. Transfer learning is discussed more in the following sections.

2.0.3 Residual networks

The backbone architecture used in our experiments is a ResNet-50, which is a residual CNN with 50 layers. Residual networks were motivated by the curious empirical finding that adding a sufficient number of layers to a deep network eventually led

[1] technically the shape of a filter is $a \times b \times c$, i.e. a filter is composed of c 2d filters.

to an increase in *training* error, as well as test error [29]. Hence the test error increase is not due to overfitting alone. Note that if the additional layers simply learned an identity function, then the deeper network can learn the same function as a shallower network.

A solution to this problem is the introduction of residual skip connections, as depicted in 2.1. The output x of an earlier layer is input to the next layer, and is also added to the output of a later layer. If the dimensionality of the additions does not match, the identity mapping in 2.1 may be replaced by a matrix multiplication W with learnable parameters. Part of the motivation for this block is that it is easier to learn an identity function by driving $\mathcal{F}$ to the zero function than by modelling an identity function explicitly. Said another way, if the function we want to learn is $\mathcal{G}$, then it is easier to infer it from data by modelling the *residual*:

$$\mathcal{F}(x) \equiv \mathcal{G}(x) - x.$$

Kaiming et al. [29] demonstrated that the introduction of residual blocks enabled stable training of deeper networks, and state of the art performance on ImageNet in 2015. Their seminal paper is the most cited neural network paper of this century.

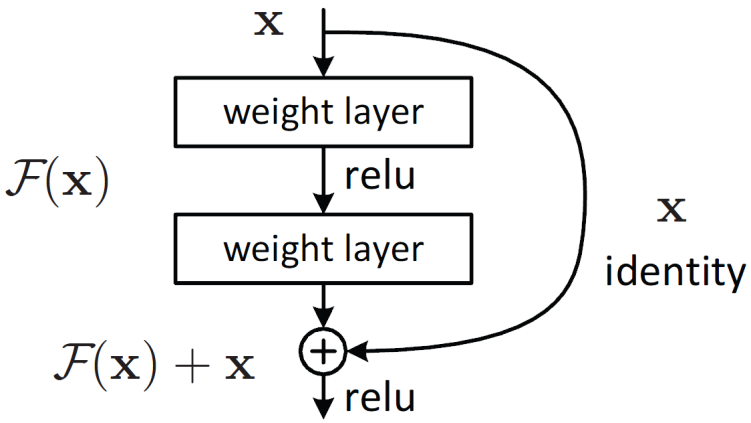

Figure 2.1: A residual block

2.0.4 Image-based cancer diagnosis

Skin cancer detection

The ISIC database is a large database of skin lesion images that has been carefully curated by experts [34]. The ISIC2018 training dataset had 10015 images, distributed among 7 categories (melanoma, basal cell carcinoma, melanocytic nevus, dermatofibroma, benign keratosis, actinic keratosis, and vascular lesion). As an example of work on this data, Guo and Ashour [27] took a supervised transfer learning

approach, utilising CNNs pretrained on ImageNet. Their model had an AUC of 72 on a validation set. In other work Majtner et al. [50] fine tuned two models on the 10015 labelled ISIC examples. The network architectures were VGG16 and GoogLeNet, and had both similarly been pretrained on ImageNet. The individual accuracy of the nets was 80.1% and 79.7% and the accuracy of the ensemble was 81.5%.

Maron et al. [51] trained a network on 11444 ISIC skin images. The architecture is a ResNet-50 pretrained on ImageNet. They evaluate the model on a test set of biopsy determined images under two tasks: classification into benign and malign; and classification into the correct category among 5 possibilities. They compare the performance to that of 112 dermatologists and find that it is generally superior. For the first task, the sensitivity and specificity for dermatologists was 74.4% and 59.8%. For the same sensitivity, specificity of the CNN was 91.3%. On the second task the performance of the network is superior except for the basal cell carcinoma category where it is on par. The accuracy of dermatologists for melanoma is 63% vs. 65% for the network.

Indeed, it is at this point known that deep learning can outperform dermatologists, given a large labelled dataset. Esteva et al. [18] trained an Inception v3 CNN on a huge dataset of 129450 lesion images with 2032 disease categories (e.g. instead of a label of 'melanoma' there is a more fine grained subclassification of amelanotic melanoma, lentigo melanoma etc). The network has been pretrained on ImageNet, and they fine tune all layers using the RMSProp optimiser, with an initial learning rate of 0.001 which is decayed every 30 epochs. They evaluate the network in several ways, and compare its performance to that of dermatologists. On the 3-class classification problem (benign, malign, non-neoplastic) the CNN has accuracy 72.1% whereas the dermatologists mean accuracy is 65.78%. They also compare the performance on two binary classification problems: malignant melanoma v.s. benign nevi; and keratinocyte carcinoma vs. seborrheic keratosis. The CNN generally outperforms the dermatologists mean performance on these problems as well.

Other medical detection problems

Singh et al. [62] compared 6 architectures VGG16, VGG19, ResNet-50, DenseNet, MobileNet and EfficientNet on the problem of breast image classification. The models had all been pretrained on ImageNet. The derived training dataset under consideration contained 31393 samples and the test set had 7849 samples, relatively balanced between malign and benign. The VGG19 performed best, with sensitivity and precision of 93.05% and 94.46% respectively. Ayana et al. [1] studied the problem of mammogram classification. Their study is on the DDSM dataset which

contains 13128 images, 5970 benign and 7158 malign. They use an 80:20 train-test split. While other works mentioned utilise CNNs, they use vision-transformers, training 3 different architectures: ViT, Swin-T, PVT. They compare the performance of these models under two regimes: training from scratch, and transfer learning. The transfer learning regime means the weights come from a network pretrained on ImageNet, as usual. Under the TL regime, all architectures achieve 100% accuracy. Conversely, when training from scratch the performance ranges from 72% - 78%, a huge difference. They also demonstrate that on this problem the vision transformer outperforms several CNN architectures, and so is worthy of further study.

Wang et al. [77] apply transfer learning to CT lung classification, using residual CNNs. The training set has 2054 images distributed among 4 categories, and the test set has 168 images. They also utilise a pretrained network, but contrary to all other work mentioned, it is not pretrained on ImageNet. Rather, it has been pretrained on another lung image dataset, Luna16. To pretrain the network they use the Adam optimiser, with a fixed global learning rate of 0.001. Fine tuning is also done with Adam, but with a lower learning rate of 0.0001. The network achieves 85.71% accuracy.

Khan et al. [41] applied transfer learning to the problem of Alzheimer's disease multiclass classification from MRI images. There are four categories: healthy controls, early mild cognitive impairment, late mild cognitive impairment and Alzheimer's disease. The dataset has a range of 1882-4637 slices per category, with 70% for training and 15% for testing. ImageNet pretrained VGG16 and VGG19 networks are used in their study. The backbone models were frozen and several fully connected layers were appended to the networks. Hence they follow the 'linear evaluation' protocol (see 2.0.5) except the head is nonlinear instead of linear. The head consists of fully connected layers of sizes $(4096, 1000, 512, 4)$. The VGG19 performs slightly better than the VGG16 network, with accuracies of 98.47% and 97.12% respectively. They demonstrate that these networks are superior to several alternative architectures, such as AlexNet and ResNet-50.

Baldota et al. [3] study the problem of pancreatic cancer classification. Their study involves a large dataset of size 26469 among three categories: healthy, cancer and pancreatitis. An ImageNet pretrained DenseNet201 is fine tuned resulting in almost 100% accuracy of 99.87%. This study demonstrates the performance that is possible with a sufficient amount of labelled data.

Jabbar et al. [35] studied the problem of diabetic retinopathy classification. The training set had 35126 samples among 5 categories: normal, mild, moderate, severe,

proliferative. They compare several ImageNet pretrained networks, for which the accuracy ranges from 92.4% to 96.6%.

We can summarise that all the mentioned studies involved transfer learning. Specifically, the initial neural networks were pretrained in a *supervised* fashion, usually on ImageNet and with a CNN base architecture. In later sections we discuss an alternative pretraining methodology called *self-supervised* learning, which can yield superior transfer results.

2.0.5 Transfer learning

A common strategy in computer vision applications is to take the initial network to be one that has been pretrained on ImageNet. The final linear layer (which represents the scores for the 1000 ImageNet categories) is removed and a new linear layer is appended mapping to the correct number of categories for the present task. This 'supervised transfer learning' approach consistently outperforms training from random initialisation [68].

Transfer learning can be motivated based on insights into how humans learn. Knowledge in one domain can be used to assist learning in a new but related domain. This may exist on a cognitive level or a motor level. For example, visual knowledge required to identify horses can be used to identify zebras; or motor knowledge of how to play softball can be used to play baseball. Pan and Yang provide a formal definition of transfer learning in terms of probability distributions [53], see also [42].

The success of transfer learning in deep learning can also be understood by analogy with the primate visual system. Early layers of visual cortex are tuned to detect generic features of object, like edges. High level semantic knowledge happens at higher cortical levels [32]. Similarly, early layers of supervised CNNs tend to learn features resembling colour blobs or gabor filters, and this appears to be independent of the data they are trained on [81]. The final layers, conversely, are more tuned to the specific supervised problem at hand. Indeed, consider the last layers output representing the scores (normalised or unnormalised) for the categories under consideration. Permuting the labels (e.g. dog=0, cat=1 $\rightarrow$ cat=0, dog=1) does not change the nature of the classification problem. In this case, the trained network can remain the same except for multiplication of the final layer by a permutation matrix.

There are several ways to perform transfer learning. Three examples are:

1. Freeze the pretrained model and train only the new linear layer. This is known as a linear probe or linear evaluation.
2. Do 1. for several iterations, and then unfreeze the backbone model and continue training as usual.

3. Train the network the standard way (no freezing involved).

If the distribution of the current dataset is similar to the pretraining distribution, then it is well known that 3. is superior to 1. In this case, standard fine tuning outperforms a linear probe. However, if the distributions are *not* similar Kumar et al. [44] demonstrated that 1. can outperform 3. Generally, 2. is superior overall. The problem with 3. in the out of distribution setting is that the pretrained model contains high quality features which may be destroyed at the start of training by aligning the body of the network with the head (with respect to the new dataset). This is a problem because the head is randomly initialised. Training the head only, with the backbone frozen for ~ 1 epoch is typically sufficient to align it with the body with respect to the current data so that the pretrained features are not lost. **Hence we follow scheme 2. in this work**.

Chapter 3

Self-supervised learning

In the context of computer vision, SSL involves training a model on a (typically *very*) large unlabelled image dataset. The model is trained to accept a high dimensional input x (i.e. an image) and output a vector r in $\mathbb{R}^d$. The vector r is the learned representation of the input x. Formally, the goal is to learn a map:

$$f_\theta : \mathbb{R}^n \to \mathbb{R}^d. \tag{3.0.1}$$

The function f_θ (the 'encoder' network) is a deep neural network with trainable parameters θ. Once the model has been trained, the representation can be used for downstream tasks. This means that instead of x being the input to a new task, the input is instead $r = f_\theta(x)$. One can freeze the representation, turning the gradients of θ off, or allow them to adapt to subsequent tasks. The simplest example of SSL is the case where one has access to a large amount of unlabelled data and only a small amount of labelled data, from the same distribution. The unlabelled data is used to pretrain the representation, which is then used to train a classifier using the labelled data. A standard way new SSL methods are evaluated is using this scheme on ImageNet:

1. Use the entire ImageNet dataset, without the labels, to pretrain a SSL representation.
2. Next, use the representation to train a classifier using a fraction (typically 1-10%) of labelled data.

Step 2 can be performed in two ways. The first is to train a linear classifier on top of the frozen representation, known as 'linear evaluation'. In this scheme the encoder is frozen, and only the head is updated through supervised learning. The head is a linear layer appended to the encoder, mapping from $\mathbb{R}^n$ to the number of categories. Alternatively, one can unfreeze the encoder and perform standard supervised learning, using the pretrained net as the initial weights. This is known as fine tuning, or semi-supervised learning.[1]

[1]Note that this is essentially transfer learning as discussed in the prior section.

The primary question of course, is how to learn the function in 3.0.1. Since we do not have access to labels, a natural approach is to enforce the intuitive condition that different distorted views of the same image should have similar representations. An image of a rotated, or blurry, or grayscale dog is still a dog.

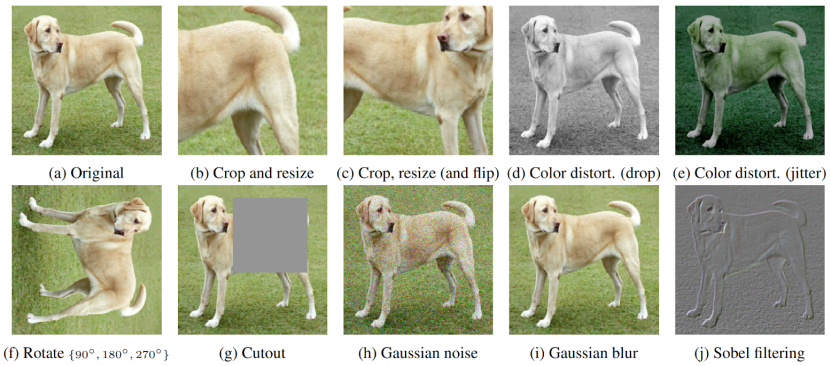

Figure 3.1: Positive samples [12]

In other words, the semantic meaning of the image is *invariant* to these kinds of augmentations. A trivial way to impose this constraint is to map every image to a constant, also known as a collapsed solution. Much of modern SSL involves techniques to enforce an invariance condition while preventing collapse, so that meaningful representations are learned.

Another view on the collapse problem is that 3.0.1 needs also to ensure that different images have different representations (particularly semantically distinct images). There are two main approaches: *contrastive* and *non-contrastive*. Both these approaches involve a joint embedding architecture, but they handle the collapse problem in different ways. One view of a joint embedding architecture can be seen in 3.2. It involves the following steps:

1. Sample a batch of data. Typically the batch needs to be quite large, although as discussed later certain algorithms work well with smaller batches.
2. Sample two random data augmentations $t \sim T$ and $t' \sim T'$ and produce two distorted copies of the batch, $v = t(X)$ and $v' = t'(X)$. For example, each element of the batch may have some random amount of blur applied.
3. Compute the projected representations $z = p(f(v))$ and $z' = p(f(v'))$.
4. Compute a loss function $\mathcal{L}(z, z')$. Update the weights with backpropagation and stochastic gradient descent. Return to step 1 and repeat until convergence.

Once training is completed, throw away the projector network, and keep the encoder. The representation r is used as input to downstream tasks. The encoder is usually a CNN as is common in computer vision. The projector network is

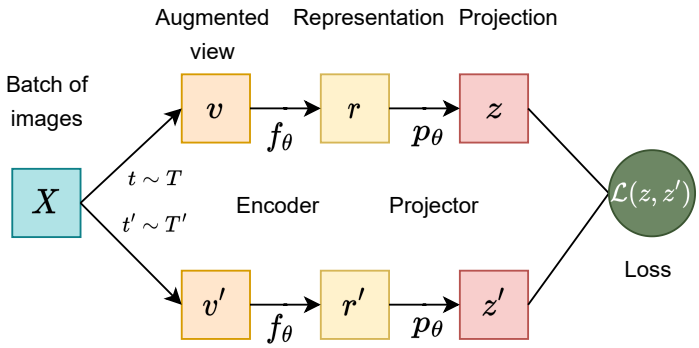

Figure 3.2: Joint embedding architecture

typically a standard feedforward net with several hidden layers. It has been found empirically that computing the loss function in projector space enables learning of better representations, compared to using the encoder [12]. We ocassionally call z the 'projected representation'.

3.0.1 Contrastive learning

Contrastive learning is one approach to defining the loss function in 3.2. It is motivated by the intuition that augmented views of different images should have dis-similar representations. These images are known as *negative samples*, in contrast to *positive samples* in 3.1. More formally, the set of augmented views of a given input image x are positive samples:

$$\{t(x) : t \sim \tau\},$$

where τ is a distribution of data augmentations. All other images in a batch are then viewed as negative samples. Note that positivity / negativity is an equivalence relation between pairs of distorted images.

3.0.2 SimCLR

SimCLR is the paradigmatic example of a contrastive learning algorithm [12]. Its loss function aims to increase the inner product of projected representations of $t(x)$ and $t'(x)$. Simultaneously, projected representations of $t(x)$ and $t'(y)$ are decreased, for all $x \neq y$. In other words, positive samples are driven together and negative samples apart:

$$\mathcal{L}_{simclr} = \frac{1}{N}\sum_{n=1}^{N} -\log \frac{\exp(\text{sim}(z_{2n-1}, z_{2n})/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k\neq 2n-1]} \exp(\text{sim}(z_{2n-1}, z_k)/\tau)} \tag{3.0.2}$$

Using the JEA 3.2, given a batch of N images two distorted views of the batch are produced, yielding $2N$ samples. The z_i are the projected representations and sim is the normalised dot product. $\mathbb{1}$ is the indicator function. The numerator in the sum aims to make inner products of positive samples large, with the denominator making inner products of negative samples small. As is typical of contrastive algorithms, SimCLR needs extremely large batch sizes (in excess of 4000) to work well.

On the other hand, non-contrastive approaches only use positive samples. These methods are mostly distinguished by the mechanism by which they prevent collapse. Bootstrap Your Own Latent and Barlow Twins are two such schemes.

3.0.3 BYOL

BYOL is a noncontrastive SSL algorithm that avoids collapse by an architectural twist on the JEA [25]. The main ingredients are the addition of a stop gradient operation on one of the branches, along with a predictor network.

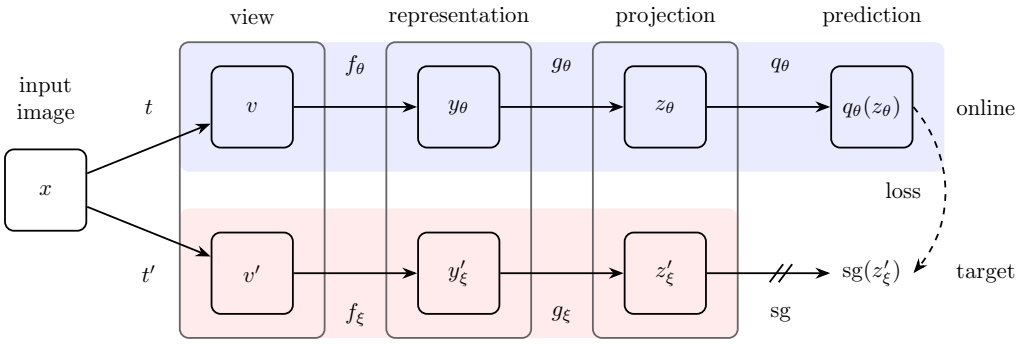

Figure 3.3: BYOL architecture

The architecture from the input image up to the projection layer is the same as the JEA 3.2 with one difference - the weights in the branches are not identical. The weights of the target network are a slow moving average of the online network. This is the original BYOL architecture, but more recent work has shown that this step is not essential [72]. That is, comparable performance is possible when the networks are siamese, with $\xi = \theta$ (up to but not including the predictor, which only exists on one branch). On the other hand, the stop gradient operation which freezes the weights on the target branch is essential to prevent collapse. The predictor network arises by adding a few extra layers on top of the projector of the online network but

not the target. This breaks the symmetry between the branches. The prediction vectors are normalised:

$$q_\theta(z_\theta) \leftarrow \frac{q_\theta(z_\theta)}{\|q_\theta(z_\theta)\|_2}, \tag{3.0.3}$$

and likewise the target vectors:

$$sg(z'_\xi) \leftarrow \frac{sg(z'_\xi)}{\|sg(z'_\xi)\|_2}. \tag{3.0.4}$$

The loss function is the mean square error of the prediction and the target:

$$\mathcal{L}_{byol} = \|q_\theta(z_\theta) - sg(z'_\xi)\|_2^2. \tag{3.0.5}$$

An interesting property of this algorithm is that it has a trivial solution, but the neural net training dynamics cause the weights to converge instead to useful representations. BYOL is state of the art on ImageNet in the linear evaluation domain. It is modestly outperformed by Barlow Twins in the semi-supervised low data domain.

3.0.4 Barlow twins

Barlow twins is a non-contrastive approach to self-supervised image learning that avoids collapse by making the representation non-redundant [83]. The main idea is that given two distorted view of the same image (see 3.1), the representations should be informative about each other *while being as uninformative as possible about the specific distortions applied*. The first property is called invariance and the second is called redundancy reduction. These properties appear naturally as two terms in the loss function. Following the JEA 3.2, it involves the following steps:

1. Sample a batch of unlabelled images X, and two data augmentations, τ_A and τ_B. Compute two distorted views of the batch:

$$X^A = \tau_A(X)$$
$$X^B = \tau_B(X)$$

   Note that $X^A_{i,:}$, $X^B_{i,:}$ will then be two distorted views of the i-th sample.
2. Compute the projection matrices, $Z^A = q(f(X^A))$ and $Z^B = q(f(X^B))$. Note that f is the encoder and q is a nonlinear projector head. If the output projector dimension is d, then each Z will be $n \times d$ where n is the batch size. Hence, the i-th row $Z^A_{i,:}$ is the projected representation of the i-th sample along the first branch, and likewise for $Z^B_{i,:}$ along the second branch. These matrices are normalised along the batch dimension.

3. Compute the barlow twins loss function on the normalised matrices:

$$\mathcal{L}_{\mathcal{BT}}(Z^A, Z^B),$$

and update the weights through backpropagation. Go back to the first step and repeat until convergence.

We now discuss the algorithm in more detail. The normalisation step is analogous to batch norm; specifically, for each projector dimension i compute the mean along the batch dimension:

$$\mu_i^A = \frac{1}{n}\sum_{k=1}^{n} Z_{ki},$$

and similarly the standard deviation:

$$\sigma_i^A = \sqrt{\frac{1}{n}\sum_{k=1}^{n}(Z_{ki} - \mu_i^A)^2}.$$

Then each term of Z^A is appropriately normalised:

$$Z_{ji} \leftarrow \frac{Z_{ji} - \mu_i^A}{\sigma_i^A}.$$

This is performed in the same way for Z^B. The motivation for this step is similar to the original motivations for batch norm. Henceforth we assume this normalisation step has been performed.

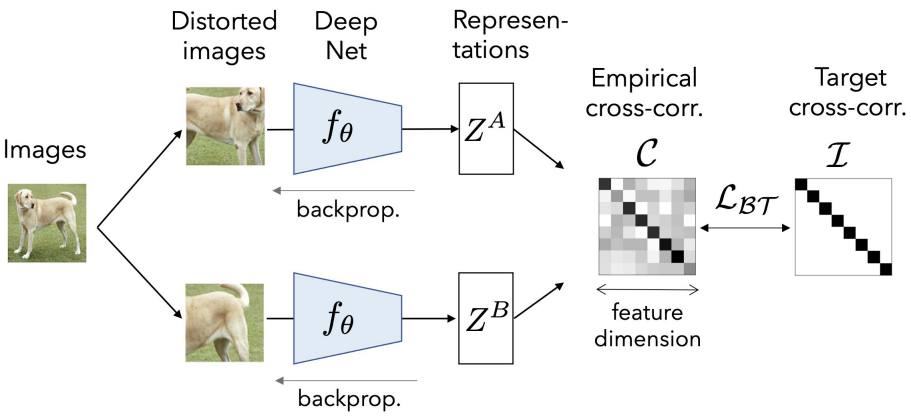

Figure 3.4: BT architecture

The loss function for Barlow Twins, roughly speaking, tries to make the cross correlation matrix between Z^A and Z^B equal to the identity matrix. More formally, we first compute the cross correlation matrix:

$$\mathcal{C}_{ij} = \frac{1}{n}\sum_{k=1}^{n} Z^A_{ki} Z^B_{kj},$$

which can be computed efficiently through a matrix multiplication:

$$\mathcal{C} = \frac{1}{n}(Z^A)^\top Z^B.$$

The Barlow Twins loss function then tries to equate the cross correlation matrix to the identity matrix:

$$\mathcal{L}_{\mathcal{BT}} = \sum_i (\mathcal{C}_{ii} - 1)^2 + \lambda \sum_{i \neq j} \mathcal{C}_{ij}^2 \tag{3.0.6}$$

where λ is a hyperparameter. Essentially, λ balances the fact that the sums have d and $d(d-1)$ terms respectively.

The first term in the loss function imposes an *invariance* constraint. One can view this with respect to our prior discussion as enforcing the intuitive condition that different views of the same image should have similar representations. The second term enforces a *redundancy reduction* condition, decorrelating the non-corresponding variables between the branches. Explicitly, given a single image x and distorted views $x^A = \tau_A(x)$ and $x^B = \tau_B(x)$, one can view the projected representations z^A and z^B as $\mathbb{R}^d$ valued random variables. Then the invariance condition imposes:

$$\text{Corr}(z_i^A, z_i^B) = 1, \quad \text{for all i,}$$

whereas the redundancy reduction term imposes:

$$\text{Corr}(z_i^A, z_j^B) = 0, \quad \text{for all } i \neq j.$$

Redundancy reduction has its origins in neuroscience, particularly through the work of Horace Barlow. He speculated that the goal of neural processing is to disentangle sensory inputs into a representation with statistically independent components [5]. Another perspective is that adding the redundancy reduction condition prevents a trivial solution to the Barlow Twins loss. Consider 3.0.6 without the second term. Then there exists a trivial solution mapping every component to any non-zero constant:

$$z_i \equiv c \neq 0.$$

In a similar way, if we remove the invariance condition then:

$$z_i \equiv 0,$$

is a collapsed solution. Hence, both terms are essential to prevent collapse.

Zbontar et al. [83] found that performance on downstream tasks increased monotonically with increasing projector dimension. This is in contrast to other self-supervised methods. For example, if the output dimension of the encoder is 2048, typically the projector output dimension will be 2048 or lower. This is similar to the bottleneck condition in autoencoders. Conversely, for Barlow Twins, performance on ImageNet semi-supervised learning and other downstream tasks, improved monotonically as the projector dimension increased from 2048 up to 8192. Some more recent work suggests that this might be because higher projector dimension enforces pairwise independence at the encoder level rather than just pairwise decorrelation (a weaker condition) [52]. Recall that redundancy reduction proposes that components should be statistically independent, but the Barlow Twins algorithm only has a correlation of zero condition.

The Barlow Twins algorithm has several additional properties that make it potentially more applicable in the low data regime. For example, it does not require large batches, and can be trained with batches as low as 128 [83]. This is in contrast to contrastive methods, which require very large batch sizes to work well [12]. Additionally, Barlow Twins achieves state-of-the-art performance in semi-supervised learning with limited data. Fine tuning a ResNet50 model with 1% of labelled ImageNet data, after SSL-pretraining,[2] resulted in 55% accuracy for Barlow Twins, compared to 48.3% for SimCLR and 53.2% for BYOL [83].

3.0.5 VICReg

Variance-Invariance-Covariance Regularization (VICReg) is a non-contrastive method that is similar to Barlow Twins [4]. In fact, it was motivated by a desire to generalise Barlow Twins to a JEA when the networks need not be siamese, and when the data on each branch can vary. This can be seen in 3.5. Due to how the loss function is defined, X and X' can differ distributionally, and in fact can be multimodal - for example X may consist of image data and X' audio data. While Barlow Twins regularises through a cross correlation matrix between the branches, VICReg regularises each branch separately, through an autocorrelation matrix and a variance

[2] i.e. perform transfer learning: i) pretrain the network using the given SSL algorithm on all of unlabelled ImageNet; ii) use 1% of labelled ImageNet data to fine tune the network.

term. Formally, the VICReg loss has 3 terms:

$$l_1(Z, Z') = \|Z - Z'\|_2^2,$$

the MSE between the branches. The second term is:

$$l_2(Z) = rr(Z, Z) + rr(Z', Z'),$$

where rr is the redundancy reduction operator from Barlow Twins. In other words, $rr(Z, Z)$ is the sum of the off diagonal entries of the autocorrelation matrix of Z, and likewise for Z'. Note that in Barlow Twins, the redundancy reduction term comes from the cross-correlation matrix between the branches: $rr(Z, Z')$. If the loss function was just composed of these two terms, there is a trivial zero solution along each branch. To this end, a third term is required. This term maintains the variance of each component of Z and Z' above a threshold. First, define $S(x, e) = \sqrt{var(x) + \epsilon}$ the regularised standard deviation, where ϵ is a small constant. Then, $S(Z^j, \epsilon)$ denotes the (regularised) sample standard deviation of Z along the batch dimension. The loss then enforces the condition that all these SDs should be larger than some threshold γ:

$$l_3 = \sum_{j=1}^{d} max(0, \gamma - S(Z^j, \epsilon)) + max(0, \gamma - S(Z'^j, \epsilon)), \tag{3.0.7}$$

where d is the projector dimension and usually $\gamma = 1$. The VICReg loss is then a linear combination of these three terms:

$$\mathcal{L}_{vicreg} = \lambda_1 l_1 + \lambda_2 l_2 + \lambda_3 l_3. \tag{3.0.8}$$

The first term is an invariance term, and terms 2 and 3 are designed to prevent collapse. One difference to Barlow Twins is that Z and Z' are not normalised. VICReg performs similarly to Barlow Twins on ImageNet transfer tasks. Its main advantage is applicability to multi-modal domains.

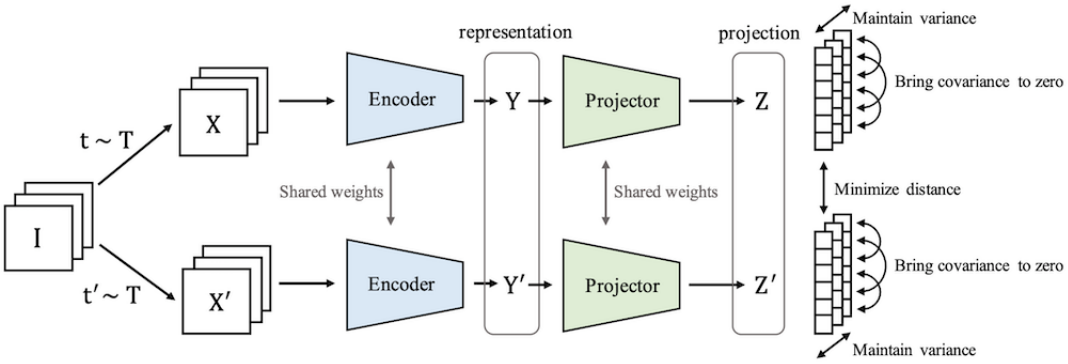

Figure 3.5: VICReg architecture. This is a joint embedding architecture. Note that the weights do not have to be shared.

3.0.6 Other work on self-supervised learning

The JEA is just one form of SSL that has particular success of late, regarding transfer performance on ImageNet and similar datasets. We briefly mention a few other approaches. These approaches historically preceded the JEA.

Image rotation

Gidaris et al. [22] proposed pretraining a model by predicting image rotation. Given an image dataset $\mathcal{D}$ they produce four copies of the dataset $\mathcal{D}_0, \mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3$ which involve 2d rotation of each element in $\mathcal{D}$ by $90 \times k$ degrees, for $k = 0, 1, 2, 3$. Hence $\mathcal{D}_1$ consists of images in $\mathcal{D}$ rotated by 90 degrees. The model is then trained in a supervised fashion to predict $k \in \{0, 1, 2, 3\}$. They demonstrate that pretraining in this manner yields effective representations. In order for the model to predict the amount of rotation it must have a good internal representation of the objects involved. Of course this only works for objects that have a typical orientation in images (dogs, cars etc). In particular, skin lesions do not have a natural (typical) orientation.

90° rotation

Colourisation approaches

This work is analagous to the 'blank out the text' approach that is common in NLP. Colourisation approaches try and predict plausible colour from an image which has had colour removed or distorted in some way. Zhang et al. [85] converted images to grayscale and then trained a network to infer colourised versions. Hence, the network will learn that apples are red or green, (but not blue) and to do so will have to represent the geometric border defining an apple and so on.

Zhang et al. [86] generalised this approach in their work on split brain autoencoders. They partition the channels $\mathcal{C}$ into disjoint subsets $\mathcal{C}_1, \mathcal{C}_2$ and likewise the autoencoder is composed of two parallel functions $\mathcal{F}_1$ and $\mathcal{F}_2$. Then $\mathcal{F}_2$ receives the $\mathcal{C}_2$ channels and attempts to predict $\mathcal{C}_1$. Similarly, $\mathcal{F}_1$ predicts $\mathcal{C}_2$ from $\mathcal{C}_1$. In particular, this allows for joint prediction of colour from grayscale, but also prediction of grayscale from colour. In other words, no channels are masked in the training objective, conversely to the pure colourisation scheme.

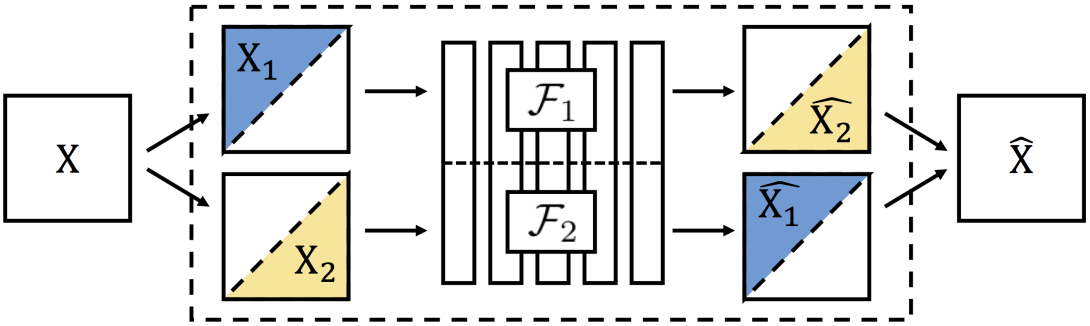

Figure 3.6: Split brain autoencoder

Generative models

Generative models are a form of self-supervised learning, which model the distribution down to the *pixel* level. This is in contrast to discriminative self-supervised models (which are all the other models discussed), which learn more abstract representations. Generative models have had huge success in the last decade, with techniques like *Generative Adversarial Networks*[24] and *Diffusion Models* [30] able to generate realistic images from complicated distributions. However, the goal of image generation is somewhat different to the goal of representation learning, and these techniques have not had the same success in the latter domain [47]. Indeed, producing photo realistic art is a rare human ability, and seems not to be a necessary skill for visual representation learning.

Figure 3.7: Image generated by DALL.E from the prompt 'DALL.E generating images'. Putting a magnifying glass on itself?

3.0.7 Why self-supervised learning?

The most basic application of SSL is when there is a large amount of unlabelled data and a small amount of labelled data. In this setting, SSL significantly outperforms training from random initialisation. For example, training a ResNet-50 on 1% of ImageNet labels yields 25.4% accuracy. If the network is first pretrained with Barlow Twins[3], the accuracy is instead 55%. Moreover, SSL now outperforms supervised transfer learning on several out of distribution downstream tasks. These include object localisation and segmentation problems on several datasets [28, 83]. This is true even while keeping the pretraining distribution fixed as ImageNet, for both supervised and self-supervised transfer. In other words, SSL algorithms can

[3]on all of *unlabelled* ImageNet

already exceed supervised transfer when pretrained on the same data. Since it is much easier to obtain more unlabelled compared to labelled data, SSL is more scalable, so this is an important point.

On the other hand, if the downstream task very closely matches ImageNet classification, supervised transfer is likely to be superior. In early experiments we found that supervised transfer was superior to SSL transfer, with respect to CIFAR10 classification on a fraction of the labels. This is perhaps unsurprising given the class overlap of CIFAR10 with ImageNet.

Chapter 4

Methodology

4.0.1 Data

Skin lesion classification is a well studied problem, and as discussed it is possible to train models that outperform human experts [8] which is primarily due to the existence of large labelled datasets. The existence of large datasets is not available for all types of problems and hence our work is motivated by cases where such datasets do not exist, such as oral cancer. Our model training set only has 2554 samples which makes less than 12% percent of total data, more closely mirroring the oral cancer study in [67] which had 3851 training samples among 4 categories. Note that over 90% of oral cancer cases are squamous cell carcinomas [78].

Lesion type	**Train**	**Test**
Actinic Keratosis	306	498
Basal Cell Carcinoma	500	2549
Benign Keratosis	467	1663
Dermatofibroma	55	173
Melanoma	500	3339
Nevus	500	10601
Squamous Cell Carcinoma	171	414
Vascular Lesion	55	186
Total	2554	19423

Table 4.1: Number of samples in the training and test set, per lesion type.

The data comes from an open source dataset, ISIC2019 [69] which includes ISIC2018 as a subset. This is a labelled skin lesion dataset. There are 8 lesion categories, 4 of which are benign, that is not cancerous (benign keratosis, dermatofibroma, nevus, vascular lesion) with the remainder being malign, that is cancerous or pre-cancerous. The category actinic keratosis is precancer, whereas basal cell carcinoma, melanoma, squamous cell carcinoma are all cancerous. There are only 171 squamous cell carcinoma samples, making this a very challenging dataset.

4.0.2 Framework

The general transfer learning procedure can be seen in the 'Fine Tune' component of Figure 4.1 and in Algorithm 1. Also see the earlier section on transfer learning 2.0.5 for background. First, the pretrained encoder is frozen and the linear head is fit for 1 epoch against the frozen representation. This is essentially multinomial logistic regression on $x' = f_\theta(x)$ with θ frozen. Next, the encoder is unfrozen and we run a learning rate finder to find a good maximum learning rate lr to use.[1] Lastly, we train the whole network for 40 epochs using the 1cycle policy and maximum learning rate lr.[2] The Adam optimiser is used at all stages [43]. We explain all steps of the algorithm in detail in the following sections. Step 4. is essentially a hyperparameter search and step 5. is the main training loop, both discussed in detail in the implementation section, including background information.

Algorithm 1 Transfer Learning

1: Freeze the encoder f_θ.
2: Train the final linear layer L for one epoch.
3: Unfreeze f_θ.
4: Run a learning rate finder to find a good maximal learning rate lr.
5: Train network for specified number of epochs using 1cycle policy and lr: `fit_one_cycle(epochs,` lr`)`.

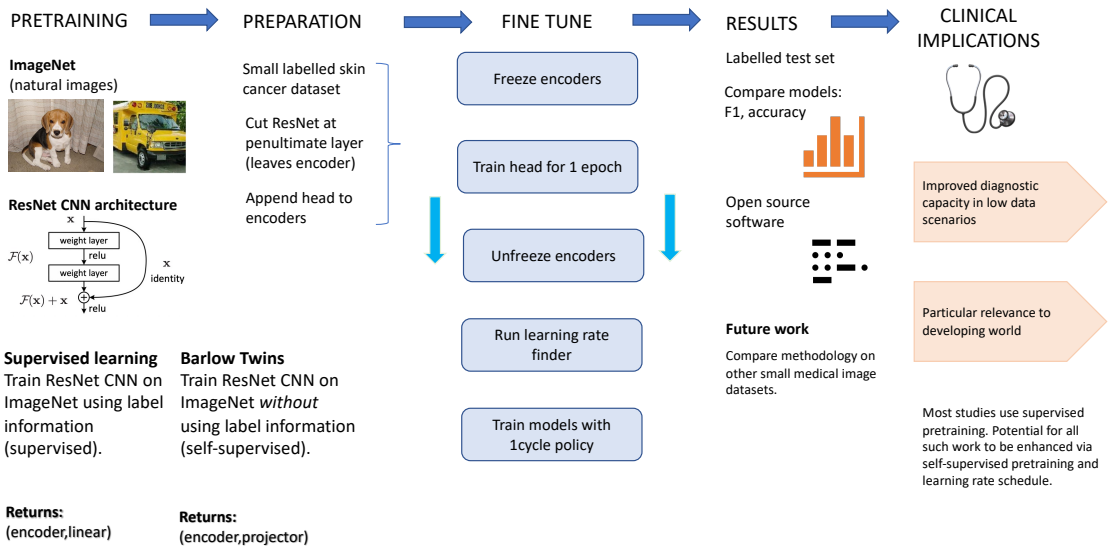

Figure 4.1: Framework showing SSL transfer vs. Supervised transfer

[1] This is done using `learn.lr_find()` in FastAI.
[2] This is done using `learn.fit_one_cycle(40)` in FastAI.

In the second step of Algorithm 1, it does not matter much how the linear layer is trained as the point is just to align the head with the encoder, so the pretrained representations are not harmed in subsequent learning. Moreover, learning rate schedules are more essential for deep networks, and when training for multiple epochs. We use the Adam optimiser [43] with a fixed learning rate of 0.001.

We apply the transfer learning methodology in 1 across two kinds of initial weights: supervised and self-supervised. The supervised initial weights have been pretrained on ImageNet in a supervised fashion; i.e. the network has been trained with cross entropy to predict class labels out of 1000 total classes [56]. The self-supervised initial weights have similarly been pretrained on ImageNet but using the Barlow Twins algorithm [58]. Note that no labels are used in this algorithm. The initial weights both come from the same backbone architecture, a ResNet-50. This is a convolutional neural network with 50 layers, with residual connections between layers. The penultimate layer of a ResNet-50 has dimensionality 2048, and we call the network up to this penultimate layer the 'encoder', denoted f_θ. For the supervised network, the final layer is a linear layer with input dimension 2048 and output dimension 1000. This last linear layer is removed, and a newly initialised linear layer is appended to the network, with the correct shape (input dimension 2048, output dimension 8 for the number of lesion categories). The Barlow Twins network involves a ResNet-50 encoder, followed by several projector layers: $(P \circ f_\theta)$. The projector network is removed, and a linear layer is similarly appended. Hence, the initialised models have the form: $Linear_8 \circ f_\theta$.

Step 5 of 1 is the main training loop. In this step the networks are trained using cross entropy loss in a supervised fashion. This is done via a modern learning rate and momentum scheduler called the '1cycle' policy (described in the next subsection), and using the FastAI software library - an extensible wrapper on top of PyTorch [31, 54]

4.0.3 Implementation

Data augmentation

The ISIC images are of varying dimensions, but neural networks require input of homogenous dimensionality. Therefore, we resize the data to 256×256. A batch size of 64 is used to suit our deep learning models. During training, each time a mini-batch is sampled we apply random data augmentation before passing it through the network. Each element of the batch is randomly cropped, rotated and flipped. The rotation is by a random angle in $[0, 45]$ degrees, and the resize scale and resize ratio for cropping are $(0.7, 1.0)$ and $(0.75, 1.33)$, respectively.

	Crop	Flip	Rotate
Probability	1.0	0.25	0.25

Table 4.2: Probabilities for augmentations.

Data augmentation done in this way during supervised learning is a standard strategy to prevent overfitting. The models sees several slightly different views of each image during training. This procedure can be seen in Figure 4.2. Each column represents a minibatch of size 2, with data augmentation applied. This is an example of different views of the same data presented to the model during training.

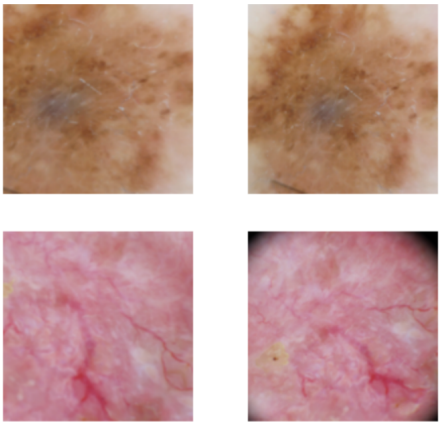

Figure 4.2: Train and test augmentation

Test time augmentation is another strategy to improve generalisation, closely related to the above technique [64]. The standard way of computing probabilities at test time is to take the test data, pass it through the neural network, and return an output probability through a softmax layer. Test time augmentation simply repeats this process several times, for augmented views of the test data. This will return several probabilities for each test input, which are then averaged to get the final probability. For example, in Figure 4.2, test time augmentation would compute the probabilities for each column the standard way, and then average the probabilities. This is because the columns represent the same data, but under different augmentation. We use exactly the same augmentations as during training, i.e. as in Table 4.2, and compute the average across three probabilities for each test input. Predictions are then made by taking the argmax of these probabilities, as usual.

1cycle policy and learning rate finder

The 1cycle training policy is a learning rate and momentum scheduler [66, 65]. It involves starting training with a very low learning rate l_1 which is then increased up to a maximal learning value l_2. Next, the learning rate is slowly decreased to l_1, and for the last several epochs, to a much lower final learning rate $\frac{l_1}{K}$, where $K >> 1$. In other words, the learning rate has an increasing period, from l_1 up to l_2, followed by a decreasing period down to $\frac{l_1}{K}$. The momentum is also scheduled, but inversely to the learning rate. Momentum decreases at the start of training, to a minimum, then is increased to a maximum. The scheme is most easily understood pictorially and can be seen in 4.4.

The 1cycle schedule diverges from standard approaches, which predominantly involve learning rate decay - starting at the maximal learning rate, and decreasing to a minimum [82]. Smith [65] showed that increasing the learning rate rapidly at the start of training has a regularising effect. It has also been found that when using this policy other forms of regularisation must be reduced - hence we do not consider dropout and similar techniques in this work. Moreover, the policy has a particular advantage when the amount of training data is limited, making it especially suited to our purposes.

An important hyperparameter in the 1cycle policy is the maximal learning rate (i.e. the peak in 4.4). In fact, this hyperparameter can be automatically inferred. A learning rate finder launches a mock training session and trains the model for several iterations, increasing the learning rate each time and recording the loss. The learning rate starts at a very low value and increases to a high value. A representative plot of this procedure can be seen in 4.3. The loss will decrease at the start, before eventually increasing (or perhaps oscillating). A maximal lr is chosen that is somewhere in between a sharp valley and the minimum.

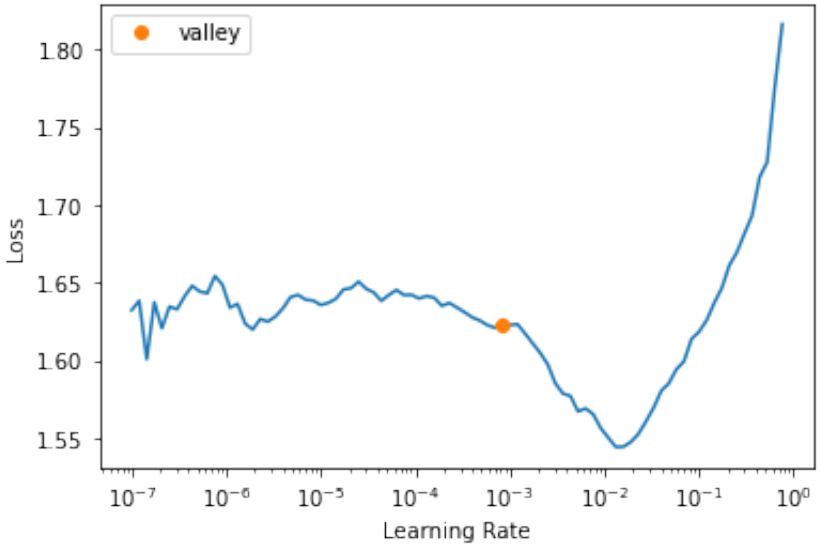

Figure 4.3: Loss v.s. learning rate. Plot comes from one of our learning rate finder runs.

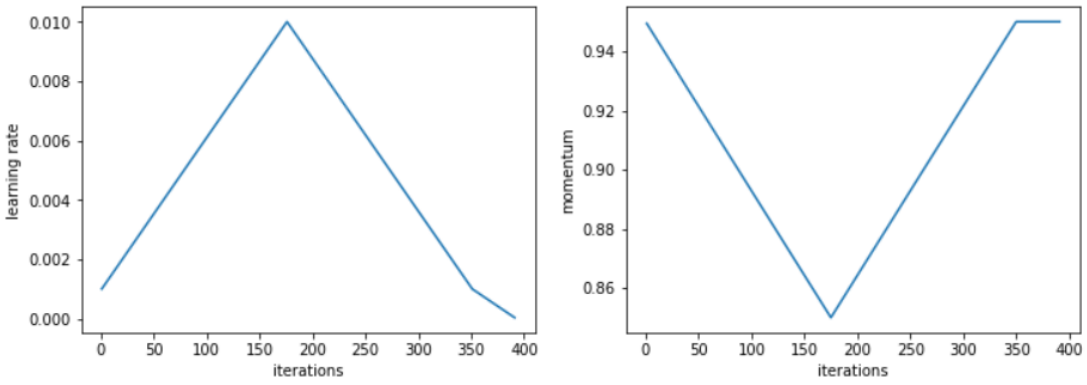

Figure 4.4: 1cycle policy. Learning rate and momentum follow inverse schedules [19].

Chapter 5

Results

Explanation of results section

The results chapter has three sections. In the first section we discuss the main fine tuning results, which compares supervised and self-supervised transfer learning using the framework from the prior chapter. Following this, we consider two possible ways to improve these main results - use a bigger model, or pretrain on the target data. Pretraining on the target data is done with Barlow Twins, and our implementation is explained in that section.

5.0.1 Comparison of supervised transfer and self-supervised transfer

We ran multiple experiments in model training (i.e trained 35 models with different weight and bias head initialisation), against two different backbone models. The backbone models have the *same* architecture, but different initial weights. The framework section provides extensive detail about this model fine tuning procedure 4.0.2. The fine tuning is done through algorithm 1.

Therefore, the initial weights of the backbone models were either from a ResNet-50 pretrained in a *supervised* manner on ImageNet; or from a ResNet-50 pretrained in a *self-supervised* manner (with Barlow Twins) on ImageNet [56, 58]. We call these 'supervised models' and 'self-supervised models' respectively, which simply refers to how the initial weights were generated. Table 5.1 shows 0.663 mean test accuracy of the supervised models and a 0.703 mean accuracy of the self-supervised models. This demonstrates the strength of using self-supervised transfer learning in the domain of cancer image classification.

	Supervised	**Self-supervised**
Mean Accuracy	0.663	0.703
Standard Deviation	0.0198	0.014
Number of Models	35	35

Table 5.1: Model fine tuning results for two kinds of initial weights.

Also displayed is the mean F1 score, for each lesion type. This can be seen in 5.2. We calculate the F1 score for each of the 35 runs, and then display the average value. The supervised models had a mean weighted average of 0.68, v.s. 0.71 for self-supervised. Moreover, self-supervised models had a larger F1 score for every lesion category, although the difference was not always large. Of particular note, the largest F1 differential among the malign categories was for squamous cell carcinoma, with self-supervised 10.8% higher than supervised. Squamous cell carcinoma also had the lowest training support, with only 171 samples. Displayed in two separate tables are the precision and recall values for each category, across the initial weights. These are also the mean values across 35 runs.

Lesion Type	f1-score (sup)	f1-score (self-sup)	Support
Actinic keratosis	0.44	0.45	498
Basal cell carcinoma	0.72	0.73	2549
Benign keratosis	0.48	0.49	1663
Dermatofibroma	0.33	0.37	173
Melanoma	0.56	0.57	3339
Melanocytic nevus	0.77	0.82	10601
Squamous cell carcinoma	0.35	0.39	414
Vascular lesion	0.31	0.51	186
Weighted avg.	0.68	0.71	19423
Accuracy	0.66	0.70	19423

Table 5.2: Combined classification report for skin lesion classification model. Each value is a mean across 35 runs.

Lesion Type	precision	recall	f1-score	support
Actinic keratosis	0.36	0.57	0.44	498
Basal cell carcinoma	0.70	0.76	0.72	2549
Benign keratosis	0.47	0.49	0.48	1663
Dermatofibroma	0.37	0.30	0.33	173
Melanoma	0.50	0.63	0.56	3339
Melanocytic nevus	0.85	0.70	0.77	10601
Squamous cell carcinoma	0.36	0.34	0.35	414
Vascular lesion	0.25	0.67	0.35	186
Weighted avg.	0.70	0.66	0.68	19423
Accuracy				0.66

Table 5.3: Supervised weights. Mean classification report

Lesion Type	precision	recall	f1-score	support
Actinic keratosis	0.37	0.57	0.45	498
Basal cell carcinoma	0.70	0.75	0.73	2549
Benign keratosis	0.49	0.50	0.49	1663
Dermatofibroma	0.43	0.32	0.37	173
Melanoma	0.52	0.63	0.57	3339
Melanocytic nevus	0.86	0.78	0.82	10601
Squamous cell carcinoma	0.48	0.34	0.39	414
Vascular lesion	0.53	0.51	0.51	186
Weighted avg.	0.72	0.70	0.71	19423
Accuracy				0.70

Table 5.4: Self-supervised weights. Mean classification report.

Also displayed are two representative ROC curves. Note that the test set has 19423 samples so is only balanced with respect to the nevus category which has 10601 test examples. For all other categories it is imbalanced. The ROC is generally not as meaningful in such a setting, (see for example [61], ROC tends to overestimate performance in the imbalanced data setting) which is why we emphasise the earlier results, but is included here for completeness.

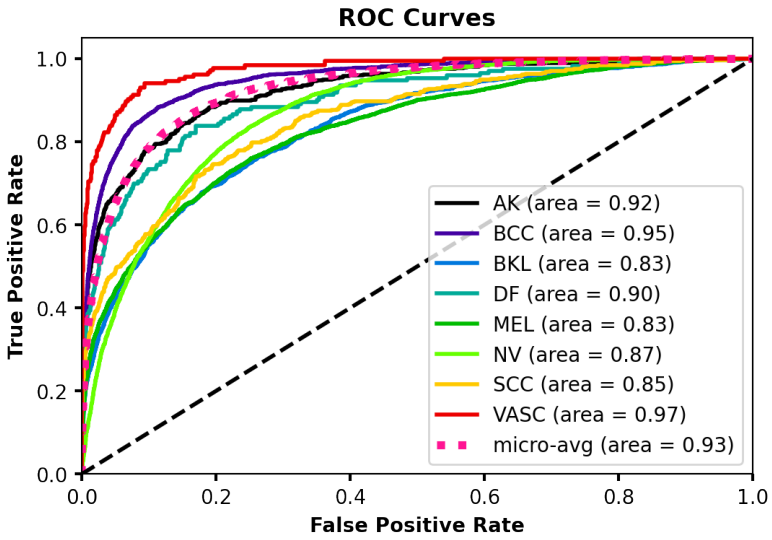

Figure 5.1: A representative ROC curve for supervised initial weights. The model displayed had a test accuracy equal to the overall mean of 0.66.

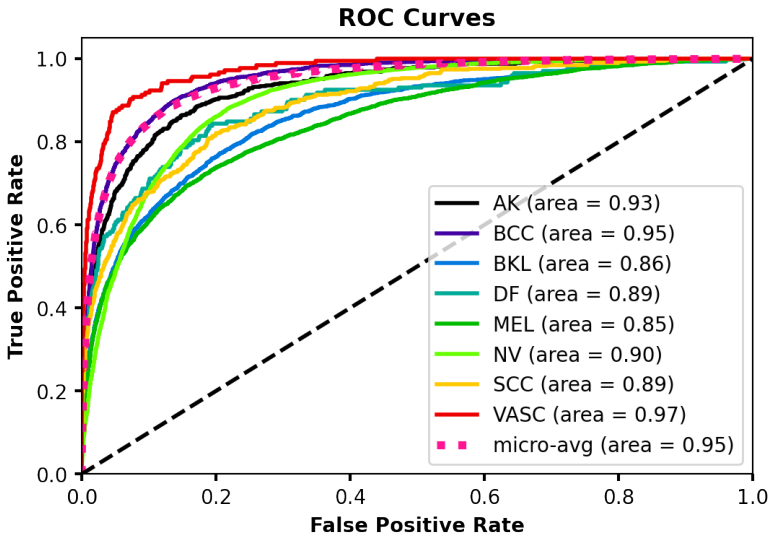

Figure 5.2: A representative ROC curve for Barlow Twins initial weights. The model displayed had a test accuracy equal to the overall mean of 0.7.

Despite demonstrating the superiority of SSL transfer over supervised transfer, the performance is still not acceptable for clinical applications. In the SSL setting [1] there are several ways to potentially improve performance:

1. use a larger network
2. pretrain on more unlabelled data beyond ImageNet

5.0.2 Larger model

Unfortunately there are no open source pretrained Barlow Twins models larger than a ResNet-50. However, there exist larger models pretrained with VICReg.[2] VICReg performs very similarly to Barlow Twins on ImageNet transfer (54.8% semi-supervised accuracy on 1% labels vs. 55% for BT). To establish a baseline, we first fine tuned a VICReg pretrained ResNet-50 [10]. Test accuracy was 0.7, exactly in line with the mean accuracy of Barlow Twins. This shows that VICReg performs similarly to Barlow Twins on the present data, just as on ImageNet. Next, we fine tuned a ResNet-200 (x2) [10] that had also been pretrained with VICReg on ImageNet. This network has encoder dimension 4096 and 250 million parameters, v.s. 23 million parameters for a ResNet-50. It is therefore a much larger model. Somewhat suprisingly, test accuracy was the same at 0.7. It therefore seems that the performance of these algorithms (Barlow Twins and VICReg) are saturated with respect to ImageNet pretraining with the ResNet architecture class.

5.0.3 Pretraining on target data

Next we consider whether pretraining on more unlabelled data beyond ImageNet can provide better results. We combined the train and test sets for a total of 21977 unlabelled samples, and pretrained a network using Barlow Twins. Similarly to before, Pytorch and FastAI were used along with self_supervised library [73]. The initial weights were those from the original Barlow Twins network and the augmentations used were similar to previous papers [83, 25]. The full details are on GitHub. [3] We applied all augmentations with the same probability as [83, 25] and in the same order. Generally the level of augmentations applied (amount of blur, jitter etc) was the same.[4] A difference was several of the augmentations had a 'same_on_batch' parameter, which is set to 'False'. This means, for example, the amount of blur applied to a batch varies across the batch, rather than being constant.

[1] i.e. keeping the amount of labelled data fixed

[2] See earlier section for more discussion of VICReg.

[3] `https://github.com/hamish-haggerty/cancer-proj`

[4] An exception was solarisation. While [83] used the default solarisation from PIL.Imageops library we used the defaults from the Kornia library. This is a minor difference.

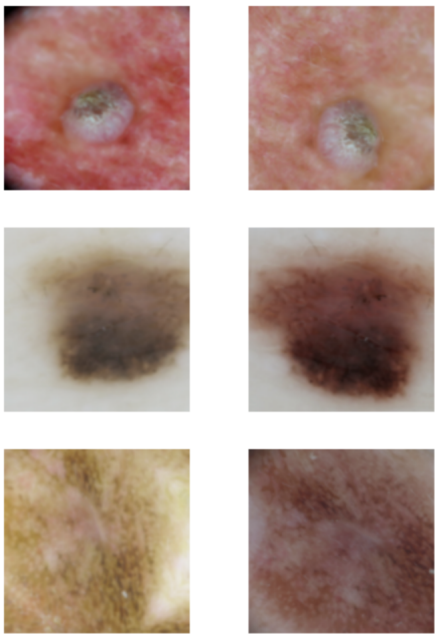

Figure 5.3: Barlow Twins augmentations. Each row has two distorted views of the same sample.

The batch size was lowered from 1024 to 128 due to memory limitations [83]. We reinitialised a new projector head with 3 layers, all of size 8192, and set $\lambda = \frac{1}{8192}$ (see 3.0.6). The encoder was frozen for one epoch while updating the projector, analogous to training the head only in transfer learning. We then unfroze the encoder and trained the whole network for 100 epochs using the 1cycle policy. Hence we use the same scheme as in algorithm 1 with the head replaced by the projector. The resulting network is then fine tuned as usual. This whole process is repeated 6 times. We perform this procedure 6 times rather than 35 for two reasons. Firstly, the pretraining step takes far longer training time compared to just fine tuning as there are 21977 unlabelled samples. Secondly, the results are anyway extremely stable across the 6 runs.

	Supervised	**Self-supervised**	**Pre-pretrained**
Mean Accuracy	0.663	0.703	0.723
Standard Deviation	0.0198	0.014	0.0037
Number of Models	35	35	6

Table 5.5: The Pre-pretrained models have been pretrained on all unlabelled data, starting from the self-supervised (Barlow Twins) initial weights. This results in higher accuracy and more stability. We call these pre-pretrained models because the weights have been pretrained twice: once on ImageNet, then on the current unlabelled data.

The resulting mean accuracy is 0.72 ± 0.004. Hence there is a modest increase in accuracy, and a larger increase in stability.

Chapter 6

Discussion

The literature review revealed that the most common approach in cancer image classification is to use supervised transfer learning 2.0.5. This is true across many studies and types of cancer including skin, oral, lung [15, 67, 80] as well as other medical image classification problems like diabetic retinopathy [35]. Therefore the potential improvement found to this machine learning pipeline has huge clinical implications. Our results show that SSL-transfer via Barlow Twins, was superior to supervised transfer, on a small cancer image dataset, across several metrics including accuracy and F1 score.

To put further emphasis on these results, recall that the model architectures were the same across experimental conditions. That is, both SSL-transfer and supervised transfer had a ResNet-50 base architecture (see earlier sections for discussion 4.0.2, 5). Similarly, both base models had been pretrained on the same data, i.e ImageNet. [1]

ImageNet features 1.2 million samples which is tiny relative to the amount of image data on the web. Obtaining more unlabelled image data is much easier than more labelled data, and SSL methods are much more scalable *with respect to data* than supervised methods. Keeping this in mind, the fact that SSL-transfer is *already* superior to supervised transfer while keeping the architecture (i.e. ResNet-50) and data (i.e. ImageNet) fixed is significant.

In the literature, it was reported that SSL methods can benefit from larger architectures [25] compared to supervised methods. Taking this into account, we fine tuned a model pretrained with a similar SSL algorithm (VICReg) where the model had over 10x more parameters than a ResNet-50. The transfer performance was unchanged 5.0.2. Therefore, we found that larger models alone do not appear to help SSL transfer, when pretraining on ImageNet.

We also found that pretraining the Barlow Twins initial weights a second time on the target data led to improved performance 5.5. This suggests that performance can be improved by collecting more unlabelled data from the target distribution,

[1] Where SSL-transfer ignored the ImageNet labels.

which is a cheaper alternative to collecting labelled data. Oral cancer diagnosis is of particular interest, since there is a lack of available labelled datasets [63]. Moreover, research in this area [67] (indeed across all medical image classification 2.0.5) tends to focus on supervised transfer learning. As our results show, this may be inferior to SSL transfer in the low data setting. Since most oral cancers are squamous cell carcinomas, we expect our results to generalise to that setting. Therefore for similar problems we recommend: a) considering SSL-transfer instead of supervised transfer; b) collecting more unlabelled target data and pretraining a second time before fine tuning.

A limitation of this work is that we only performed the comparison on ISIC skin lesion images. Future work could explore whether the results extend to other medical classification problems that involve low amounts of labelled data. We also need expert analysis, i.e qualitative analysis by skin specialists, especially regarding the false positive results produced. Furthermore, we note that medical diagnosis systems require more information about how decisions have been made, which is not available since we are using deep learning, which is a black box approach to machine learning. We can add a layer to our framework that uses ensemble learning methods [84] such as XGBoost [11] and Random Forests [7] that can provide if-then rules; however, they would still face challenges as we are dealing with images rather than tabular data with features. If certain features are extracted, then ensemble learning and tree-based methods can be used determining which features are most contributing (important) in discriminating the different types of skin lesions.

Our results suggest that additional SSL pretraining on the target data boosts performance (Table 5.5). Hence, future work could explore pretraining Barlow Twins on unlabelled data beyond ImageNet, i.e. general (non-target) data. This requires investment of more compute, rather than expensive labelling. This could be explored in tandem with using larger architectures. Although we found pretraining a larger architecture on ImageNet alone does not help, this may not be true when scaling up to larger datasets.

Another limitation of our work is that we performed the supervised vs. SSL-transfer comparison only on the ResNet architecture class (which was due to availability of pretrained models). Future work could explore the comparison for other architectures, such as vision transformers. This is of particular interest since recent work has found that on some medical classification problems vision transformers can outperform CNNs, when both are pretrained on ImageNet [1]. Moreover, vision transformers appear to benefit particularly from *scale*. Dosovitskiy et al. [17] found that vision transformers can match or exceed ResNet transfer performance when pretrained on datasets larger than ImageNet. Therefore future work could

explore the supervised vs SSL transfer comparison using vision transformers pre-trained on larger datasets. We expect SSL superiority in medical image transfer to rise in this case.

Furthermore, modest changes to the SSL framework may lead to additional gains. For example, the augmentations used to train Barlow Twins were chosen due to knowledge of which augmentations tend to lead to good transfer performance on ImageNet when training other SSL algorithms. It is unclear if these are the best augmentations for medical image transfer, which can be evaluated in future work.

Our framework is general and can hence be applied to other medical diagnosis problems that have class imbalance and limited training data. Furthermore, our model is based on image data, and extensions can be done so that it can work with tabular data with more insights to the decision making process using explainable artificial intelligence [75].

Chapter 7

Conclusion

In this paper, we presented a framework that applied SSL to cancer image diagnosis given limited and class imbalanced data. We demonstrated that pretraining with Barlow Twins can outperform standard supervised pretraining. Supervised pretraining yielded 0.66 accuracy, compared to 0.7 for Barlow Twins pretraining, a large difference. Since most of the literature on cancer image diagnosis uses supervised pretraining, our framework opens the door to improved performance of such models given data challenges. Furthermore, our work suggests that additional gains are possible by SSL pretraining on target data. On the other hand, pretraining larger models on ImageNet alone does not appear to help. We envision that this work will be applicable in training more accurate medical image classification models in the low labelled data regime.

Code and Data

All experiments were run in Google Colab, generally on a single A100 or V100 GPU. Our code is open source on GitHub. [1]

[1] `https://github.com/hamish-haggerty/cancer-proj`

References

[1] Ayana, G., Dese, K., Dereje, Y., Kebede, Y., Barki, H., Amdissa, D., Husen, N., Mulugeta, F., Habtamu, B., Choe, S.W., 2023. Vision-transformer-based transfer learning for mammogram classification. Diagnostics 13. URL: https://www.mdpi.com/2075-4418/13/2/178, doi:10.3390/diagnostics13020178.

[2] Bakker, S.H., Jacobs, W.C.H., Pondaag, W., Gelderblom, H., Nout, R.A., Dijkstra, P.D.S., Peul, W.C., Vleggeert-Lankamp, C.L.A., 2018. Chordoma: a systematic review of the epidemiology and clinical prognostic factors predicting progression-free and overall survival. European Spine Journal 27, 3043–3058. URL: https://doi.org/10.1007/s00586-018-5764-0, doi:10.1007/s00586-018-5764-0.

[3] Baldota, S., Sharma, S., Malathy, C., 2021. Deep transfer learning for pancreatic cancer detection, in: 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), Kharagpur, India. pp. 1–7. doi:10.1109/ICCCNT51525.2021.9580000.

[4] Bardes, A., Ponce, J., LeCun, Y., 2021. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. arXiv:2105.04906.

[5] Barlow, H., 2001. Redundancy reduction revisited. Network: Computation in Neural Systems 12, 241–253. doi:10.1080/net.12.3.241.253.

[6] Baykul, T., Yilmaz, H.H., Aydin, U., Aydin, M.A., Aksoy, M., Yildirim, D., 2010. Early diagnosis of oral cancer. The Journal of International Medical Research 38, 737–749. doi:10.1177/147323001003800302.

[7] Breiman, L., 2001. Random forests. Machine Learning 45, 5–32. doi:10.1023/A:1010933404324.

[8] Brinker, T.J., Hekler, A., Enk, A.H., Berking, C., Haferkamp, S., Hauschild, A., Weichenthal, M., Klode, J., Schadendorf, D., Holland-Letz, T., von Kalle, C., Fröhling, S., Schilling, B., Utikal, J.S., 2019. Deep neural networks are superior to dermatologists in melanoma image classification. European Journal of Cancer 119, 11–17.

[9] Cancer.net, 2022. Oral and oropharyngeal cancer: Statistics. URL: https://www.cancer.net/cancer-types/oral-and-oropharyngeal-cancer/

statistics. adapted from the American Cancer Society's (ACS) publication, Cancer Facts & Figures 2022, the ACS website, the International Agency for Research on Cancer website, and the National Cancer Institute's Surveillance, Epidemiology, and End Results (SEER) Program. (All sources accessed February 2022.).

[10] Caron, M., Larochelle, H., 2021. VicReg: Variance-invariance-covariance regularization. GitHub repository. Https://github.com/facebookresearch/vicreg.

[11] Chen, T., Guestrin, C., 2016. Xgboost: A scalable tree boosting system, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM. pp. 785–794. doi:10.1145/2939672.2939785.

[12] Chen, T., Kornblith, S., Norouzi, M., Hinton, G., 2020. A simple framework for contrastive learning of visual representations, in: International Conference on Learning Representations (ICLR).

[13] Clark, K., Vendt, B., Smith, K., Freymann, J., Kirby, J., Koppel, P., Moore, S., Phillips, S., Maffitt, D., Pringle, M., Tarbox, L., Prior, F., 2013. The cancer imaging archive (tcia): maintaining and operating a public information repository. Journal of Digital Imaging 26, 1045–1057. URL: https://link.springer.com/article/10.1007/s10278-013-9622-7, doi:10.1007/s10278-013-9622-7.

[14] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L., 2009. ImageNet: A Large-Scale Hierarchical Image Database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255. URL: https://ieeexplore.ieee.org/document/5206848, doi:10.1109/CVPR.2009.5206848.

[15] Dildar, M., Akram, S., Irfan, M., Khan, H.U., Ramzan, M., Mahmood, A.R., Alsaiari, S.A., Saeed, A.H.M., Alraddadi, M.O., Mahnashi, M.H., 2021. Skin cancer detection: A review using deep learning techniques. International Journal of Environmental Research and Public Health 18, 5479.

[16] Dilsizian, S.E., Siegel, E.L., 2014. Artificial intelligence in medicine and cardiac imaging: harnessing big data and advanced computing to provide personalized medical diagnosis and treatment. Current cardiology reports 16, 1–8.

[17] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N., 2020. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 .

[18] Esteva, A., Kuprel, B., Novoa, R.A., Ko, J., Swetter, S.M., Blau, H.M., Thrun, S., 2017. Dermatologist-level classification of skin cancer with deep neural networks. Nature 542, 115–118. URL: https://doi-org.wwwproxy1.library.unsw.edu.au/10.1038/nature21056, doi:10.1038/nature21056.

[19] fast.ai, 2021. One cycle policy. URL: https://fastai1.fast.ai/callbacks.one_cycle.html. accessed: 2023-03-29.

[20] Ferlay, J., Colombet, M., Soerjomataram, I., Parkin, D.M., Piñeros, M., Znaor, A., Bray, F., 2021. Cancer statistics for the year 2020: An overview. Int J Cancer Online ahead of print.

[21] Filella, X., Foj, L., 2016. Prostate cancer detection and prognosis: from prostate specific antigen (psa) to exosomal biomarkers. International journal of molecular sciences 17, 1784.

[22] Gidaris, S., Singh, P., Komodakis, N., 2018. Unsupervised representation learning by predicting image rotations. arXiv preprint arXiv:1803.07728 URL: https://arxiv.org/abs/1803.07728.

[23] Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press. URL: http://www.deeplearningbook.org.

[24] Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial networks, in: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (Eds.), Advances in Neural Information Processing Systems 27, Curran Associates, Inc.. pp. 2672–2680. URL: http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf.

[25] Grill, J.B., Strub, F., Altché, F., Tallec, C., Richemond, P.H., Buchatskaya, E., Doersch, C., Pires, B.A., Guo, Z.D., Azar, M.G., Piot, B., Kavukcuoglu, K., Munos, R., Valko, M., 2020. Bootstrap your own latent: A new approach to self-supervised learning. Advances in Neural Information Processing Systems 33.

[26] Guo, J., Li, B., 2018. The application of medical artificial intelligence technology in rural areas of developing countries. Health equity 2, 174–181.

[27] Guo, Y., Ashour, A.S., 2018. Multiple convolutional neural network for skin dermoscopic image classification. arXiv preprint arXiv:1807.08114.

[28] He, K., Fan, H., Wu, Y., Xie, S., Girshick, R., 2019. Momentum contrast for unsupervised visual representation learning. arXiv preprint arXiv:1911.05722 .

[29] He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.

[30] Ho, J., Chen, X., Srinivas, A., Duan, Y., Abbeel, P., 2020. Denoising diffusion probabilistic models. arXiv preprint arXiv:2006.11239 URL: https://arxiv.org/abs/2006.11239.

[31] Howard, J., Gugger, S., 2020. Deep Learning for Coders with Fastai and Pytorch: AI Applications Without a PhD. O'Reilly Media, Incorporated. URL: https://books.google.no/books?id=xd6LxgEACAAJ.

[32] Hubel, D.H., Wiesel, T.N., 1962. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. The Journal of Physiology 160, 106–154.

[33] III, S.G.A., McLennan, G., Bidaut, L., McNitt-Gray, M.F., Meyer, C.R., Reeves, A.P., Gamsu, G., Henschke, C., Hoffman, E.A., Kazerooni, E.A., MacMahon, H., Beek, E.J.R.V., Aberle, D.R., Yankaskas, B., Austin, P.J., Goldin, J., Prokop, A.F., Cody, D.D., Lynch, D.A., Mazzone, J.C., Fenton, L.E., van Ginneken, B., Lambin, P., Brown, M.S., Barnhart, R.S., Kalpathy-Cramer, Freymann, J.E., Kirby, J.S., Gavrielides, M.A., Kiciak, P.B., Bakis, C.E., 2011. The lung image database consortium (lidc) and image database resource initiative (idri): a completed reference database of lung nodules on ct scans. Medical Physics 38, 915–931. URL: https://doi.org/10.1118/1.3528204, doi:10.1118/1.3528204.

[34] International Skin Imaging Collaboration, 2023. ISIC archive: A comprehensive resource for skin imaging data. URL: https://www.isic-archive.com. accessed: 2023-04-20.

[35] Jabbar, M.K., Yan, J., Xu, H., Ur Rehman, Z., Jabbar, A., 2022. Transfer learning-based model for diabetic retinopathy diagnosis using retinal images. Brain Sciences 12, 535. doi:10.3390/brainsci12050535.

[36] Jaiswal, A., Babu, A.R., Zadeh, M.Z., Banerjee, D., Makedon, F., 2020. A survey on contrastive self-supervised learning. arXiv preprint arXiv:2011.00362 .

[37] Jeschke, J., Van Neste, L., Glöckner, S.C., Dhir, M., Calmon, M.F., Deregowski, V., et al., 2012. Biomarkers for detection and prognosis of breast cancer identified by a functional hypermethylome screen. Epigenetics 7, 701–709.

[38] Jussupow, E., Spohrer, K., Heinzl, A., Gawlitza, J., 2021. Augmenting medical diagnosis decisions? an investigation into physicians' decision-making process with artificial intelligence. Information Systems Research 32, 713–735.

[39] Kamisawa, T., Wood, L.D., Itoi, T., Takaori, K., 2016. Pancreatic cancer. The Lancet 388, 73–85.

[40] Kazarian, A., Blyuss, O., Metodieva, G., Gentry-Maharaj, A., Ryan, A., Kiseleva, E.M., et al., 2017. Testing breast cancer serum biomarkers for early

detection and prognosis in pre-diagnosis samples. British journal of cancer 116, 501–508.

[41] Khan, R., Akbar, S., Mehmood, A., Shahid, F., Munir, K., Ilyas, N., Asif, M., Zheng, Z., 2023. A transfer learning approach for multiclass classification of Alzheimer's disease using MRI images. Frontiers in Neuroscience 16, 1050777. doi:10.3389/fnins.2022.1050777.

[42] Kim, H.E., Cosa-Linan, A., Santhanam, N., Jannesari, M., Maros, M.E., Ganslandt, T., 2022. Transfer learning for medical image classification: a literature review. BMC Medical Imaging 22, 69.

[43] Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 .

[44] Kumar, A., Raghunathan, A., Jones, R., Ma, T., Liang, P., 2022. Fine-tuning can distort pretrained features and underperform out-of-distribution. arXiv preprint arXiv:2202.10054 .

[45] Kwong, T., Mazaheri, S., 2021. A survey on deep learning approaches for breast cancer diagnosis. arXiv preprint arXiv:2109.08853.

[46] LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proceedings of the IEEE 86, 2278–2324.

[47] LeCun, Y., Misra, I., 2021. Self-supervised learning: The dark matter of intelligence. Facebook AI Blog.

[48] Li, Y., Lu, H., Wang, W., Zhao, W., 2020. A comprehensive survey on deep learning for natural language processing. arXiv preprint arXiv:2003.01200 .

[49] Litjens, G., Kooi, T., Bejnordi, B.E., Setio, A.A.A., Ciompi, F., Ghafoorian, M., van der Laak, J.A., van Ginneken, B., Sánchez, C.I., 2017. A survey on deep learning in medical image analysis. Medical Image Analysis 42, 60–88. URL: https://www.sciencedirect.com/science/article/pii/S1361841517301135, doi:10.1016/j.media.2017.07.005.

[50] Majtner, T., Bajić, B., Yildirim, S., Hardeberg, J.Y., Lindblad, J., Sladoje, N., 2018. Ensemble of convolutional neural networks for dermoscopic images classification. arXiv preprint arXiv:1808.05071.

[51] Maron, R.C., Weichenthal, M., Utikal, J.S., Hekler, A., Berking, C., Hauschild, A., Enk, A.H., Haferkamp, S., Klode, J., Schadendorf, D., Jansen, P., Holland-Letz, T., Schilling, B., von Kalle, C., Fröhling, S., Gaiser, M.R., Hartmann, D., Gesierich, A., Kähler, K.C., Wehkamp, U., Karoglan, A., Bär, C., Brinker, T.J., Schmitt, L., Peitsch, W.K., Hoffmann, F., Becker, J.C., Drusio, C., Jansen, P., Klode, J., Lodde, G., Sammet, S., Schadendorf, D., Sondermann, W., Ugurel, S., Zader, J., Enk, A., Salzmann, M., Schäfer, S., Schäkel, K.,

Winkler, J., Wölbing, P., Asper, H., Bohne, A.S., Brown, V., Burba, B., Deffaa, S., Dietrich, C., Dietrich, M., Drerup, K.A., Egberts, F., Erkens, A.S., Greven, S., Harde, V., Jost, M., Kaeding, M., Kosova, K., Lischner, S., Maagk, M., Messinger, A.L., Metzner, M., Motamedi, R., Rosenthal, A.C., Seidl, U., Stemmermann, J., Torz, K., Velez, J.G., Haiduk, J., Alter, M., Bär, C., Bergenthal, P., Gerlach, A., Holtorf, C., Karoglan, A., Kindermann, S., Kraas, L., Felcht, M., Gaiser, M.R., Klemke, C.D., Kurzen, H., Leibing, T., Müller, V., Reinhard, R.R., Utikal, J., Winter, F., Berking, C., Eicher, L., Hartmann, D., Heppt, M., Kilian, K., Krammer, S., Lill, D., Niesert, A.C., Oppel, E., Sattler, E., Senner, S., Wallmichrath, J., Wolff, H., Giner, T., Glutsch, V., Kerstan, A., Presser, D., Schrüfer, P., Schummer, P., Stolze, I., Weber, J., Drexler, K., Haferkamp, S., Mickler, M., Stauner, C.T., Thiem, A., 2019. Systematic outperformance of 112 dermatologists in multiclass skin cancer image classification by convolutional neural networks. European Journal of Cancer 119, 57–65. URL: https://www.sciencedirect.com/science/article/pii/S0959804919303818, doi:https://doi.org/10.1016/j.ejca.2019.06.013.

[52] Mialon, G., Balestriero, R., LeCun, Y., 2022. Variance covariance regularization enforces pairwise independence in self-supervised representations. arXiv preprint arXiv:2209.14905 arXiv:2209.14905.

[53] Pan, S.J., Yang, Q., 2010. A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering 22, 1345–1359. doi:10.1109/TKDE.2009.191.

[54] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., . Pytorch: An imperative style, high-performance deep learning library.

[55] Pearson, K., 1901. On lines and planes of closest fit to systems of points in space. Philosophical Magazine 6, 559–572.

[56] PyTorch, 2021. torchvision.models.resnet50. URL: https://pytorch.org/vision/main/models/generated/torchvision.models.resnet50.html. accessed: 2023-03-29.

[57] Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., 2018. Improving language understanding by generative pre-training .

[58] Research, F., 2021. Barlow twins: Self-Supervised Learning via Redundancy Reduction. URL: https://github.com/facebookresearch/barlowtwins. accessed: 2023-03-29.

[59] Rigel, D.S., Russak, J., Friedman, R., 2010. The evolution of melanoma diagnosis: 25 years beyond the abcds. CA: A Cancer Journal for Clinicians 60, 301–316.

[60] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L., 2015. Imagenet large scale visual recognition challenge. International Journal of Computer Vision 115, 211–252. doi:10.1007/s11263-015-0816-y.

[61] Saito, T., Rehmsmeier, M., 2015. The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. PLoS ONE 10, e0118432. URL: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0118432, doi:10.1371/journal.pone.0118432.

[62] Seemendra, A., Singh, R., Singh, S., 2021. Breast cancer classification using transfer learning, in: Singh, P., Noor, A., Kolekar, M., Tanwar, S., Bhatnagar, R., Khanna, S. (Eds.), Evolving Technologies for Computing, Communication and Smart World, Springer, Singapore. doi:10.1007/978-981-15-7804-5_32.

[63] Sengupta, N., Sarode, S.C., Sarode, G.S., Ghone, U., 2022. Scarcity of publicly available oral cancer image datasets for machine learning research. Oral Oncology 126, 105737.

[64] Shanmugam, D., Blalock, D., Balakrishnan, G., Guttag, J., 2021. Better aggregation in test-time augmentation, in: 2021 IEEE/CVF International Conference on Computer Vision (ICCV), IEEE Computer Society. pp. 1194–1203. URL: https://doi.ieeecomputersociety.org/10.1109/ICCV48922.2021.00125, doi:10.1109/ICCV48922.2021.00125.

[65] Smith, L.N., 2018. A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay. arXiv preprint arXiv:1803.09820 arXiv:1803.09820.

[66] Smith, L.N., Topin, N., 2017. Super-convergence: Very fast training of neural networks using large learning rates. arXiv preprint arXiv:1708.07120 arXiv:1708.07120.

[67] Song, B., Li, S., Sunny, S., Gurushanth, K., Mendonca, P., Mukhia, N., Patrick, S., Gurudath, S., Raghavan, S., Tsusennaro, I., Leivon, S.T., Kolur, T., Shetty, V., Bushan, V., Ramesh, R., Peterson, T., Pillai, V., Wilder-Smith, P., Sigamani, A., Suresh, A., Kuriakose, M.A., Birur, P., Liang, R., 2021. Classification of imbalanced oral cancer image data from high-risk population. J Biomed Opt 26, 105001.

[68] Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., Liu, C., 2018. A survey on deep transfer learning. Artificial Intelligence Review 52, 1–40.

[69] Tasinkevych, A., 2019. ISIC 2019: Skin lesion analysis towards melanoma detection. https://www.kaggle.com/datasets/andrewmvd/isic-2019. Accessed: 2023-03-01.

[70] Team, I., 2023. Imagenet. URL: https://www.image-net.org/. online; accessed 12-April-2023.

[71] Thapa, C., Camtepe, S., 2021. Precision health data: Requirements, challenges and existing techniques for data security and privacy. Computers in biology and medicine 129, 104130.

[72] Tian, Y., Chen, X., Ganguli, S., 2021. Understanding self-supervised learning dynamics without contrastive pairs. arXiv preprint arXiv:2102.06810 .

[73] Turgutlu, K., 2022. Self-supervised learning library. URL: https://github.com/KeremTurgutlu/self_supervised. available at: https://github.com/KeremTurgutlu/self_supervised.

[74] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I., 2017. Attention is all you need, in: Advances in Neural Information Processing Systems, pp. 5998–6008.

[75] Vilone, G., Longo, L., 2020. Explainable artificial intelligence: a systematic review. arXiv preprint arXiv:2006.00093 .

[76] Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A., 2008. Extracting and composing robust features with denoising autoencoders, in: Proceedings of the 25th international conference on Machine learning, ACM. pp. 1096–1103.

[77] Wang, S., Dong, L., Wang, X., Wang, X., 2020. Classification of pathological types of lung cancer from ct images by deep residual neural networks with transfer learning strategy. Open Medicine (Warsaw) 15, 190–197. doi:10.1515/med-2020-0028.

[78] Warnakulasuriya, S., 2009. Global epidemiology of oral and oropharyngeal cancer. Oral Oncology 45, 309–316.

[79] Wu, Y., Schuster, M., Chen, Z., Le, Q.V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al., 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144 .

[80] Yadav, S., Jain, A.K., Shinde, P., 2019. A survey on deep learning techniques for lung cancer detection. International Journal of Innovative Technology and Exploring Engineering (IJITEE) 8, 1216–1220. URL: https://www.ijitee.org/wp-content/uploads/papers/v8i10s/J10430881019.pdf.

[81] Yosinski, J., Clune, J., Bengio, Y., Lipson, H., 2014. How transferable are features in deep neural networks?, in: Advances in Neural Information Processing Systems, pp. 3320–3328.

[82] You, K., Long, M., Wang, J., Jordan, M.I., 2019. How does learning rate decay help modern neural networks? arXiv preprint arXiv:1908.01878 .

[83] Zbontar, J., Jing, L., Misra, I., LeCun, Y., Deny, S., 2021. Barlow twins: Self-supervised learning via redundancy reduction, in: International Conference on Learning Representations (ICLR).

[84] Zhang, C., Ma, Y., 2012. Ensemble Machine Learning: Methods and Applications. Springer.

[85] Zhang, R., Isola, P., Efros, A.A., 2016a. Colorful image colorization. arXiv preprint arXiv:1603.08511 URL: https://arxiv.org/abs/1603.08511.

[86] Zhang, R., Isola, P., Efros, A.A., 2016b. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. arXiv preprint arXiv:1611.09842 URL: https://arxiv.org/abs/1611.09842.

[87] Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., He, Q., 2021. A comprehensive survey on transfer learning. Proceedings of the IEEE 109, 43–76.