Lung Data Analysis With Deep Learning

Bachelor Thesis

to obtain the academic degree of

Bachelor of Science

in the Bachelors's Program

Bioinformatics

Submitted by

Hari Krishnan Kesavan Vijayakumar

Submitted at

Johannas Kepler University
Institute for
Machine Learning
&
University of South Bohemia
Faculty of Science

Supervisor

Univ.-Prof. Dr. Sepp Hochreiter

Co-Supervisors

Andreas Mitterecker, MSc

Guarantor:

Ing. Ph.D. Rudolf Vohnout

November 2021

JOHANNES KEPLER
UNIVERSITY LINZ
Altenbergerstraße 69
4040 Linz, Österreich
www.jku.at
DVR 0093696

Bibliographical Detail

Kesavan Vijayakumar, Hari Krishnan, 2021: Lung Data analysis using deep learning. Bachelor Thesis. Bachelor Thesis, in English. 38 p., Institute for Machine Learning Johannes Kepler University, Linz, Austria

Annotation

Medical images can have extremely high resolutions which cannot be handled properly by typical state-of-art machine learning models. In this thesis I compared the performance of two approaches of multiple instance learning models where the high resolution images are downscaled into smaller patches and low dimensional embedding are calculated using Resnet. Then low dimensional embedding are aggregated using multiple instance learning to attain class labels. The data set for this thesis consisted of high resolution histological slides of human lung which were classified to contain cancer or not.

Declaration

I hereby declare that I have worked on my bachelor's thesis independently and used only the sources listed in the bibliography.

I hereby declare that, in accordance with Article 47b of Act No. 111/1998 in the valid wording, I agree with the publication of my bachelor thesis, in full to be kept in the Faculty of Science archive, in electronic form in a publicly accessible part of the IS STAG database operated by the University of South Bohemia in České Budějovice accessible through its web pages. Further, I agree to the electronic publication of the comments of my supervisor and thesis opponents and the record of the proceedings and results of the thesis defence in accordance with aforementioned Act No. 111/1998. I also agree to the comparison of the text of my thesis with the Theses.cz thesis database operated by the National Registry of University Theses and a plagiarism detection system.

Linz, 16-11-2021

Place, Date

Hari Krishnan Kesavan

Contents

1 Abstract

Machine learning plays an essential role in medical science to help diagnose cancer using histological slides. The performance of deep learning architectures, particularly Convolutional Neural Networks (CNNs), have shown great advancements in image classification. Medical image datasets are difficult to collect since they require medical expertise to label. Furthermore the high resolution of these images can be probelematic to analyze. These high-resolution images do not fit on a GPU without breaking them into smaller patches. Two techniques are evaluated through experiments namely CLAM (CLustering-constrained Attention Multiple instance learning) and Attention-based Multiple Instance Learning. The preprocessing modes like data augmentation, color correction, and image exposure are applied for both techniques. The performance of the models are evaluated using 10-fold cross-validation which result in an average accuracy of 58% and 56% respectively.

2 Introduction

Bioinformatics helps researchers to solve computational biological problems by unifying with machine learning to solve problems like protein structure prediction, image analysis, and visualization (image interpretation and data fusion) etc. The use of machine learning techniques has been extended to a broad spectrum of bioinformatics applications. There are several biological domains where machine learning techniques are applied for extracting important information from the available data. There are several fields where machine learning techniques are applied and had a enormous success. However, traditional machine learning techniques could not handle complex biological problems [17]. The deep learning approach can help to solve these problems and help to create and extract a new features for disease diagnosis. The limitations of using deep learning method are that they require a large number of manually annotated images with proper supervised learning settings and the dimensions of these images. The size of the medical slide images is produced in the gigapixel for better examination, making the task even more complicated. There are several approaches available to encounter these limitations by down-scaling the high-resolution images, but there are many possibilities of losing the information [19].

2.1 Deep Learning in Medical Imaging

In recent years, deep learning has had a tremendous impact in the field of machine learning due to the rapid development of improved CNNs and modern GPUs. The latest deep learning methods empower us to extract high-level abstracting features to perform difficult tasks. Computer vision enables the computer system to see, identity and process the images like human which helps deep learning methods to create a model to understanding world [10]. These technologies are also highly relevant for medical imaging. Medical imaging and computer vision constitute a wide and rapidly evolving field. Medical imaging is a process used to create images of the human body for diagnosing and treatment purposes. The most common technique used in recent days are x-rays, computer tomography (CT), ultrasound, MRI, and positron emission tomography (PET).

In the last few decades, the development of the medical imaging, pattern recognition, and image processing techniques has grown immensely. These techniques provide the most valuable information for diagnosis and treatment of diseases. There are three main steps involved in processing medical images:

1. Preprocessing of the images: This is the initial step of the training phase, the lower level, which involves image augmentation like scaling, translations and color processing.

2. Feature Extraction: Features plays a vital role in the performance of the deep learning models. The middle level helps in extracting useful information from the lower level to determine an algorithm that would be able to extract distinctive and complete feature representation.

3. Machine learning: The final stage of the process involves creating models like an artificial neural network based on the features extracted from the lower level to perform the classification task [6].

3 Related work

3.1 Machine Learning

Machine learning methods provide computational abilities to learn from experience. Based on the input data, there are several types of machine learning categories: Supervised techniques, Unsupervised techniques, and Semi-supervised techniques.

Supervised learning: Supervised learning uses annotated dataset to train a model to yield the desired output. It models relationships and dependencies between the target prediction output and input features and produces a set of the correct label based on the previously learned rules.

Unsupervised Learning: Unsupervised learning does not rely on an annotated dataset. Instead, it enables a model to discover patterns and anomalies and discover features that could be used to categorize data. It is also used to label the unlabelled dataset [19].

Semi-Supervised Learning: Semi-supervised learning uses small annotated images to train the model. Then the trained model is used to generate the pseudo-label for a larger data-set of images with annotations and learn a final model by combining both sets of images [30].

When the neural network begin to outperform other high-profile image analysis methods, deep learning arises to be the computer vision prominence. Medical images have multiple modalities and have a dense pixel resolution. The acquisition and interpretation of images are critical for accurate disease diagnosis. Deep learning applied to medical imaging has the potential to be the most disruptive technology since the introduction of digital imaging. In comparison to shallow neural network, deep learning models have larger capacities and generalize better. Adding more layers to the Neural Network allows for an easier representation of the interactions within the input data, as well as allows us to learn more abstract features and used as an input to the next hidden layer. Multiple layers of neurons are hierarchically stashed in a deep learning neural network, forming a feature representation [20].

3.2 Convolutional Neural Network

An Artificial Neural Network (ANN) was first introduced by Warren McCulloch and Walter Pitts, who created neural network based on the threshold logic [7]. ANN is the core part of the Deep Learning. Because of it versatility and scalability, ANNs are the best solution for solving the large and complex machine learning tasks. It is basically inspired by the working of biological neural network in human brain [9]. The basic architecture of the ANN is shown in the Figure 1.

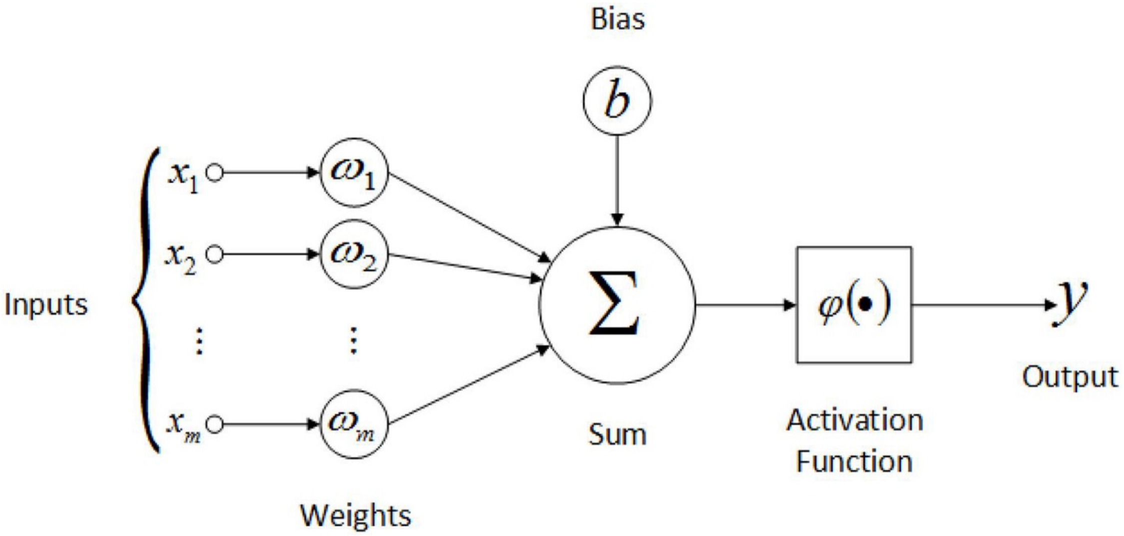


Figure 1: The basic structure of the perception consisting of input ($x_1, x_2, ...x_m$) which is connected to neuron with weight ($w_1, w_2, ...w_m$). The neuron sums up all the signal it receives and transfer it to a activation function which results in the non-linear output(y) [2].

Convolutional Neural Networks (CNNs or Convnets) are the specific kind of neural network used to process the grid-like topology data and is widely used for analyzing visual images [8]. CNNs utilize a relevant filter to capture the non-linear and temporal dependencies of the images. The architecture employs only less parameter comparing to the other model and use shared weight for all the inputs [24]. The CNN architectures are built by stacking convolutional layers followed by pooling layers, allowing the CNN to learn a range of low-level features in the earliest layer and aggregate those features in the high-level layer. Finally, the set of features are fed into a fully-connected (FC) layer, which perform the classification [4]. The basic architecture of the CNN is shown in Figure 2.

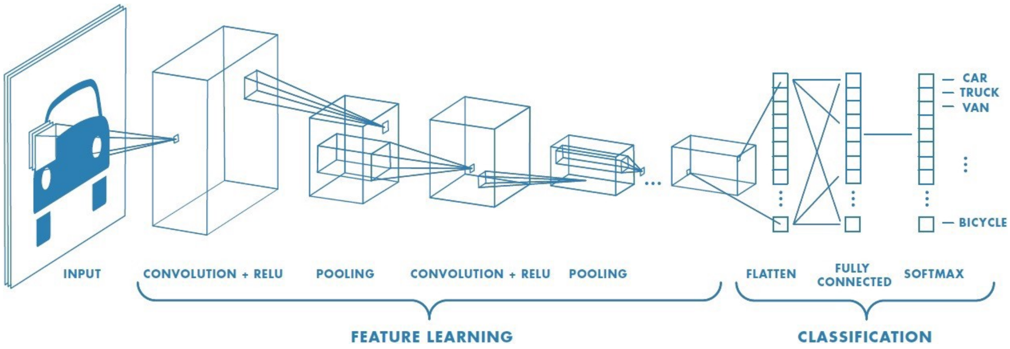
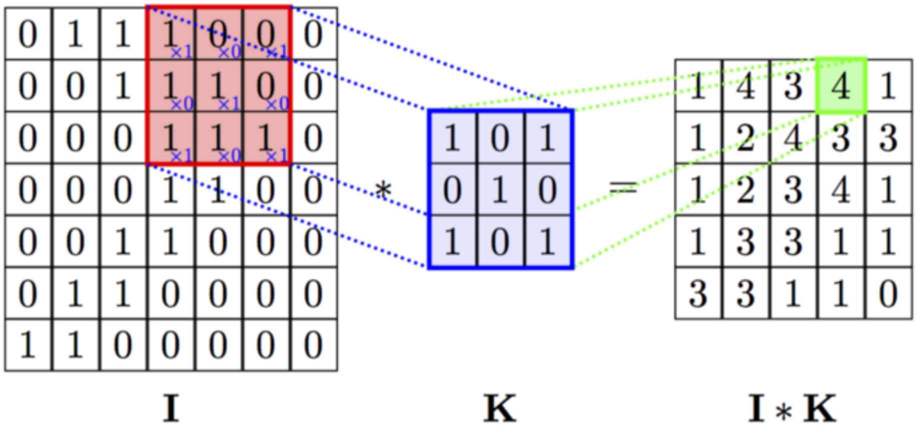


Figure 2: The basic CNN architecture for image classification. For example input image of car was used here for classification task. At first the convolutional layer with kernel size of 3x3 is used followed by pooling layer. These steps are repeated couple of times to extract features from the image. And the extracted feature is flattened and given as an input to the fully-connected layer which outputs the label for the given images [21].

3.2.1 Convolutional Layer

The first layer of a CNN architecture is a convolutional layer comprising several convolutional filters applied to the input images to extract features. A significant part of the convolutional layer are parameterized filters or kernels. The kernel size is usually a square matrix of 3x3, which has been frequently used. The filter slide over the whole input image is width and height. It calculates the square neighbor's dot product with its corresponding element of the filter kernel, which summed up to the single value as shown in Figure 3. Applying a convolutional filter in the input image results in a 2D array called a feature map. The obtained filter parameters are shared across the images, reduces the number of parameters used by the CNN. These feature maps help the CNN model helps to perform better classification [15].

The parameter that controls the convolutional steps are stride and padding. The amount of pixels the kernel slides over is represented by strides. The use of strides in the filter will result in the downsampling of the input image. For example, an input image of size 4x4 is convolved with a kernel size of 3x3, the size of the output image would be 2x2. This will reduce the input image volume if the network gets deeper and deeper, as there are possibilities of losing the information at the border of the images. One way of solving this problem is by adding a certain number of pixels around the input images' edge. Zero paddings add a padding of zero around the border, which is also a widely used approach because of the low complexity [29].



$$\begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} * \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 4 & 3 & 4 & 1 \\ 1 & 2 & 4 & 3 & 3 \\ 1 & 2 & 3 & 4 & 1 \\ 1 & 3 & 3 & 1 & 1 \\ 3 & 3 & 1 & 1 & 0 \end{pmatrix}$$

Figure 3: Visualization of a 2D convolution operation. I is the original input, The blue square labeled K is the kernel. The result on the right side is the dot product of the input and the kernel [14].

.

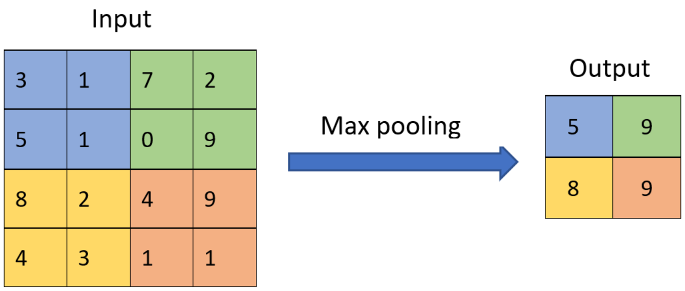
The size of the output image is calculated using the following formula:

$$\frac{V - R + 2Z}{S} + 1$$

Where V is the input image, R is the size of the receptive field, Z is the number of pixels added using zero-padding, and S is the stride size. As mentioned above, the results of the filter applied on the input image are 2D feature maps. The weight within the filters is tuned during the training phase and shared across the whole architecture [30].

3.2.2 Pooling Layer

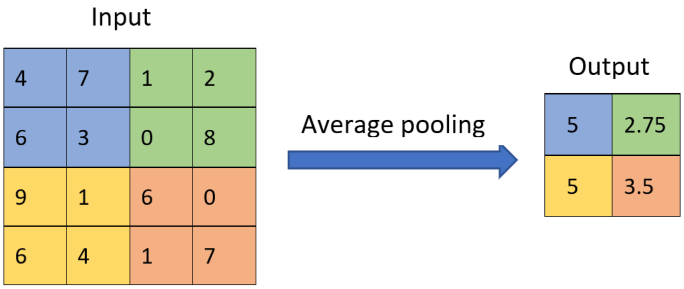
Pooling layers are based on the assumption that the brain performs some sub-sampling while processing an image. The main goal of the layer is to under sample the input image. Like the convolutional layer, the pooling layer consists of hyper-parameter stride S, which controls the image's sliding and helps the network from over-fitting. Figure 4 and Figure 5 illustrate max-pooling and average pooling.



3	1	7	2
5	1	0	9
8	2	4	9
4	3	1	1

5	9
8	9

Figure 4: The maximum values of each neighborhood of the input data(4x4) that is covered by the kernel(2x2) as an output.



4	7	1	2
6	3	0	8
9	1	6	0
6	4	1	7

5	2.75
5	3.5

Figure 5: The average values of each neighborhood of the input data(4x4) that is covered by the kernel(2x2) as an output.

3.2.3 Activation Function

The activation function is usually added to CNN to help the neural network learn complex patterns and decide which neuron should be fired next. The most commonly used activation function for the CNN is ReLU. It is easy to compute by not activating all the neurons simultaneously and solving vanishing grading. The advantage of the ReLU activation is that it converts all the negative values to zero, which avoids activating unwanted neurons.

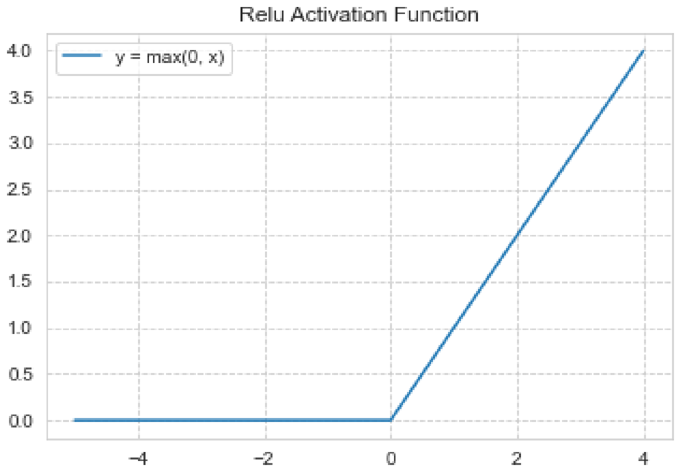


Figure 6: The rectified linear activation function is a monotonic linear that output zero if the input of the activation is in negative value and pass the same value if input is above zero. This help the classifier to fit and train in the dataset properly [1].

3.2.4 Fully-connected Layer

Like other traditional ANNs, the convolution layers' features are flattened and connected to the fully connected layer. The activation function SoftMax is used to get the probability of the class as the final score.

3.2.5 Residual Nets

Residual Nets (ResNets) were introduced in 2015 by Kaiming He et al. [11] and won the ILSVRC challenge by delivering the lowest error rate of under 3.6%. The critical goal of ResNet was to solve the vanishing gradient problem of the deep network where the loss of model get to zero after few runs and performs no learning. The residual network theory makes an identity relation between the layer, as seen in the Figure 8. A standard neural network model aims to get the *f(x)* learning to input the activation function, whereas ResNet discovers the residual mapping of *f(x)+x*. By adding the residual block to architecture help the network initially to models the identity function. Adding more residual block helps the network increase the performance even before several layers have not started to learn.

The architecture of ResNet is based on a deep residual framework, which is primarily influenced by the VGG network theory [11]. The full architecture of the ResNet36 is shown in the

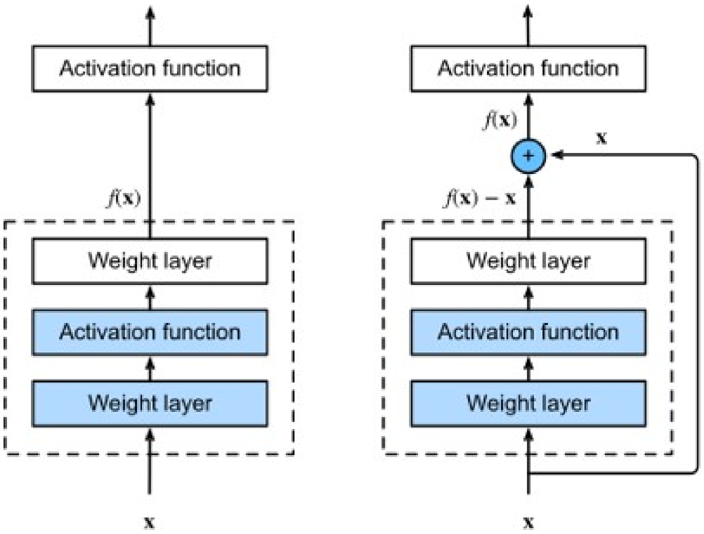


Figure 7: The figure illustrate the difference between classical neural block and residual building block. Every layer in classic neural networks feeds into the next. Every layer feeds into the next layer and straight into the layers 2–3 steps away in a network with leftover blocks.

Figure 8. ResNet34 takes 224x224 pixel images as an input followed by a convolutional layer and a max-pooling layer. Subsequently, a four separate multiple residual units consisting of two convolutional layers using the 3x3 kernel with Batch Normalization and ReLU activation function. Finally, the layer is followed by the FC layer and SoftMax with average pooling.

3.3 Weakly supervised object detection

Weakly supervised learning is a branch of machine learning often used when the available data is not appropriately annotated and insufficient to obtain a good performing model. Weakly supervised object detection is used widely because of the rapid growth of the image level annotated dataset than the slide level dataset. Weakly supervised object detection(WSOD) use the knowledge of the pre-trained network to obtain the slide level annotation. WSOD combined with modern Convolutional Neural Network helps us to increase performance [5]. The latest approached proposed by Tang et al. [25] trains end-to-end multi-class classification networks by obtaining classification scores using weighted sum-max pooling, which proves to have an increase in performance of the model.

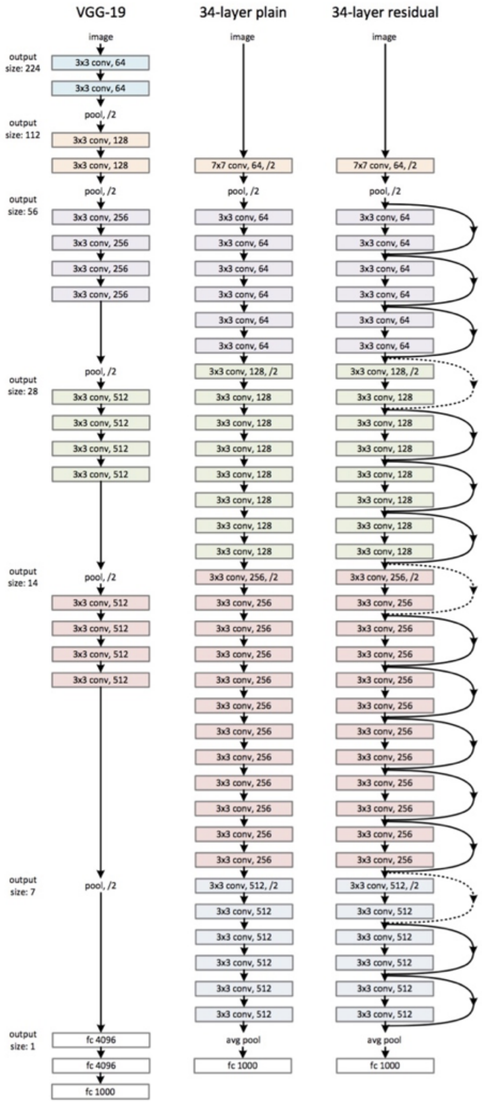


Figure 8: The Figure describes the full architecture of Resnet34 comparing with VGG network and plain 34-layer network. The dotted line the 34-layer residual indicate the change in dimensional using strides[11].

3.3.1 Attention

The human brain can visualize the information parallel by focusing on the essential part of the whole scene. For example, let us consider Figure 9 below as a scene pictured by a human brain. The human brain will instantly focus on the important feature images like black nose, the shape of the ear, and ignoring sweater and blanket to identify a dog in the scene. When it comes to computer vision, algorithms will treat all the parts of the image equally instead of focusing on the important features of the images. The main aim of the attention model is to integrate attention mechanisms in deep learning neural network using modern CNN [27].

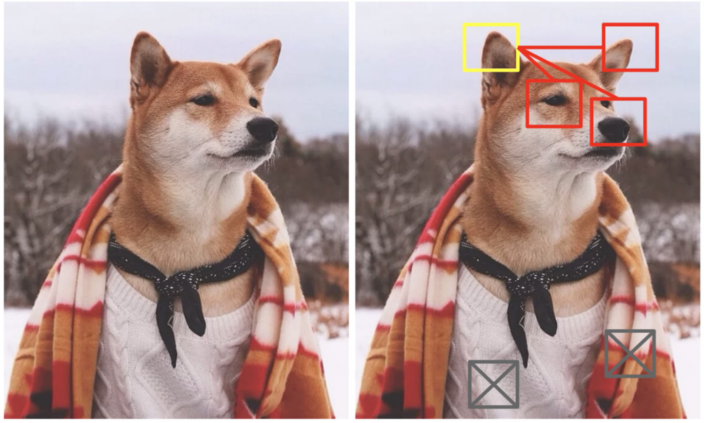


Figure 9: Figure shows gives the detailed explanation how attention mechanism identify the object based on the features.

In the early stages, the attention mechanism was used in the natural language processing (NLP). Attention mechanisms have been widely applied to recurrent neural networks and long to tackle sequential decision tasks. Top information is gathered sequentially and decides where to attend for next feature learning step [28]. The idea behind the attention mechanism is motivated by Neural Machine Translation (NMT). These models are based on bidirectional recurrent neural network (RNN) composed of encoder-decoder architecture. The encoder tries to form input information into a context vector with a fixed length. The decoder takes the vector as input from the encoder hidden states and outputs the relevant information. With this framework, the model can selectively focus on valuable parts of the input sequence and learn to associate between them. The context vector produced by the encoder should be the good representation of the input sequence [28, 22].

The main drawback of this method is that all the necessary information must be compressed to fixed length. To overcome this issue Bahdanau et al. [3] introduced an attention mechanism. The context vector of the model of Bahdanau et al. is a alignment score of annotation mapped to input sentence using encoder. The context vector is concatenated with the previous decoder output and fed into the decoder RNN for that time step along with the previous decoder hidden state to produce a new output [23].

3.3.2 Hard Attention

Hard attention is a stochastic process which uses hardattention mechanism to focus on only specific subset of the encoder for a last layer of the decoder. The last layer of the decoder is initialized with a Reinforcement learning agent where the learning policy is trained for each step to attain hard attention score. In hard attention mechanism, weight of an important part of the images is computed instead of the whole image, which will help to reduce the computational cost [13].

3.3.3 Soft Attention

Soft attention, developed recently, can trained end-to-end for the convolutional network. A deep network module capturing top information is used to generate affine transformation. The affine transformation is applied to the input images to get the attention region and then feed to another deep network module. The whole process can be trained end to end by using the differential network layer, which performs spatial transformation [26]. The major problem in the hard attention is using Reinforcement Learning, as it is subjected to have high variance, which scales linearly with a number of hidden units. Soft Attention mechanism computes weight for whole images, which helps the model to pay attention on the each and every part of the image. But computing weights for the whole input image will have high computational cost. The soft attention mechanism is a widely used method in the field of computer vision [16, 12] and also used in this thesis.

3.4 Multiple Instance Learning

In a classical machine learning image classification task, the whole images are used to classification based class label, whereas in multiple instance learning (MIL) images are broke down into bunch of instances to extract feature from it. Then main goal of the MIL is to predict the label for each instance in the given bag using the bag label to find out the key instance that define the class for the classification. Each instance in the bag x_i is given with the label y_i, where y_i is the label of the bag. The size of bag varies depending upon the size of image. The main approaches of the MIL are instance level approach and embedding level approach. In the instance level approach score for each instance are calculated, on the other hand in the embedding level approach lower dimensional embedding are calculated independent of the class instance. The workflow of embedding level multiple instance learning is show in the figure. The embedding level approach is used in this thesis [12].

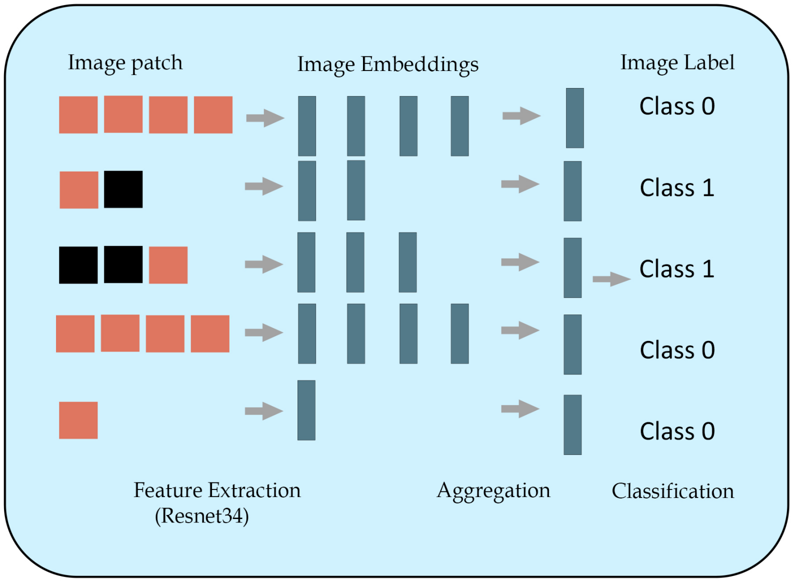


Figure 10: Figure gives detailed explanation about the workflow embedding level approach. 1. The high resolution images are downscaled into smaller patches. 2. The image patches are then used to create embedding using Resnet34 and pass through permutation-invariant aggregation function. 3. Finally transform into class probability.

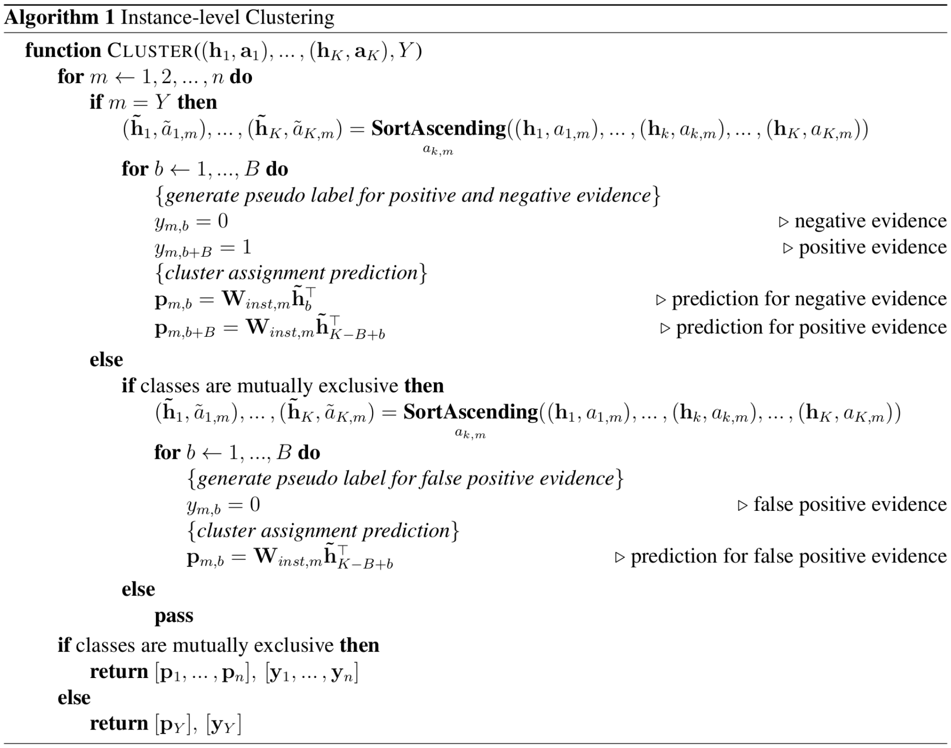
3.5 Clustering-constrained attention Multiple Instance learning (CLAM)

In CLAM, Slide level aggregation is aggregated using an attention-based pooling function to attain the class's patch-level representation. Using the attention-based pooling function, the model will learn structural features evidence for both the positive and negative classes [18].

The parallel classifiers are built to attain the attention score a_{km}, where *k* is the patch and *m* is the class. The attention scores are calculated by using the following formula.

$$a_{k,m} = \frac{exp\left\{W_{a,m}(tanh(V_a h_k^{\tau}) \bigodot sigm(U_a h_k^{\tau}))\right\}}{\sum_{j=1}^{N} exp\left\{W_{a,m}(tanh(V_a h_j^{\tau}) \bigodot sigm(U_a h_j^{\tau}))\right\}}$$

$$z = \sum_{k=1}^{N} a_{k,m}\mathbf{h}_k$$



Algorithm 1 Instance-level Clustering

```
function CLUSTER($(\mathbf{h}_1, \mathbf{a}_1), ..., (\mathbf{h}_K, \mathbf{a}_K), Y$)
    for $m \leftarrow 1, 2, ..., n$ do
        if $m = Y$ then
            $(\tilde{\mathbf{h}}_1, \tilde{a}_{1,m}), ..., (\tilde{\mathbf{h}}_K, \tilde{a}_{K,m}) = \underset{a_{k,m}}{\textbf{SortAscending}}((\mathbf{h}_1, a_{1,m}), ..., (\mathbf{h}_k, a_{k,m}), ..., (\mathbf{h}_K, a_{K,m}))$
            for $b \leftarrow 1, ..., B$ do
                {generate pseudo label for positive and negative evidence}
                $y_{m,b} = 0$                                        ▷ negative evidence
                $y_{m,b+B} = 1$                                      ▷ positive evidence
                {cluster assignment prediction}
                $\mathbf{p}_{m,b} = \mathbf{W}_{inst,m}\tilde{\mathbf{h}}_b^{\top}$                  ▷ prediction for negative evidence
                $\mathbf{p}_{m,b+B} = \mathbf{W}_{inst,m}\tilde{\mathbf{h}}_{K-B+b}^{\top}$          ▷ prediction for positive evidence
        else
            if classes are mutually exclusive then
                $(\tilde{\mathbf{h}}_1, \tilde{a}_{1,m}), ..., (\tilde{\mathbf{h}}_K, \tilde{a}_{K,m}) = \underset{a_{k,m}}{\textbf{SortAscending}}((\mathbf{h}_1, a_{1,m}), ..., (\mathbf{h}_k, a_{k,m}), ..., (\mathbf{h}_K, a_{K,m}))$
                for $b \leftarrow 1, ..., B$ do
                    {generate pseudo label for false positive evidence}
                    $y_{m,b} = 0$                                    ▷ false positive evidence
                    {cluster assignment prediction}
                    $\mathbf{p}_{m,b} = \mathbf{W}_{inst,m}\tilde{\mathbf{h}}_{K-B+b}^{\top}$    ▷ prediction for false positive evidence
            else
                pass
    if classes are mutually exclusive then
        return $[\mathbf{p}_1, ..., \mathbf{p}_n], [\mathbf{y}_1, ..., \mathbf{y}_n]$
    else
        return $[\mathbf{p}_Y], [\mathbf{y}_Y]$
```

Figure 11: $p_{m,k}$ is cluster assignment scores predicted for k^{th} and h is bag of k instance [18].

As mentioned above, the CLAM using instance-level clustering to boost up the training by add a clustering layer after the first fully hidden layer [12]. The instance level cluster algorithms are shown in Figure 11.

3.6 Lung Diseases

Lung cancer was a rare disease in the earlier 20th century with less than 400 recorded cases, but it later became the leading cause of cancer death in the mid-twentieth century. Lung cancer is the most common cause of cancer death among men and women around the world, more than twice as many people as bread cancer died of lung cancer. Around 2.7 million people in the European Union's member states are predicted to be diagnosed with cancer in 2020, with almost 1.3 million dying from it.

Lung cancer is expected to be diagnosed in more men than women. It can be prevented in more than 40% of cases if caught early enough. Tobacco consumption is the primary cause of lung cancer, 85% of death from lung cancer were attributable to it. But tobacco consumption is not the only the mean cause of lung cancer, even though it is responsible for most other lung diseases. These often do not primarily lead to death but severe hospitalizations and decreased life quality.

The symptoms of lung cancer are often hidden until it is too late to do something against the tumor. Therefore, it is one of the deadliest cancer types we now know. If a patient shows symptoms, the most common ones are fever, coughing, thorax pain, and dyspnea. In a few cases, a paralysis of the respiratory muscles can occur, which can leads to death within a few days.

3.6.1 Chronic Obstructive Pulmonary Disease (COPD)

COPD is a result of tobacco consumption in 9 out of 10 cases. Other causes can be infections, heredity, pollution of the environment, and asthmatic diseases. COPD is a constriction of the small airways which is caused by inflammation. This inflammation leads to a greater production of mucous and destruction of the lung tissue until the last stadium of destruction, the lung emphysema.

Normally the exhalation is the most affected mechanism in COPD which leads to a massive over-concentration of CO2 inside the lungs. Therefore, most patients feel like they suffocate due to the unbalanced ratio of O2 and CO2. Other symptoms are cough, chronical bronchitis, and secretion.

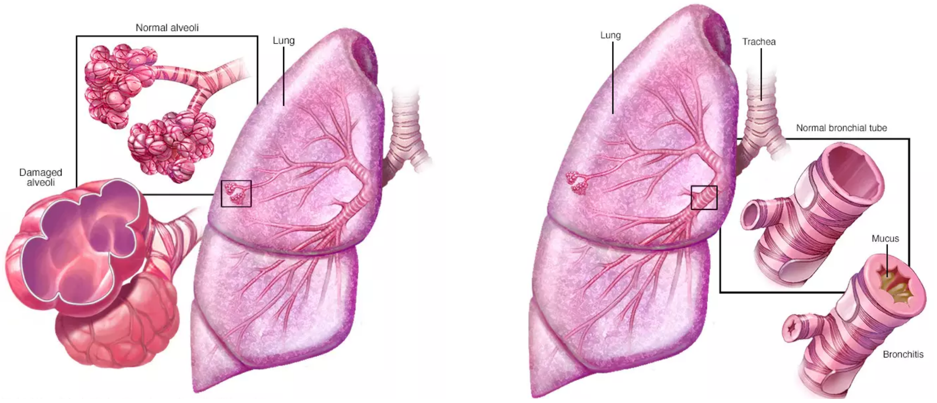


Figure 12: Example of COPD affected lung

Left: Slide from fibrosis **Right**: Slide from COPD

3.6.2 Idiopathic Pulmonary Arterial Hypertension (IPAH)

IPAH, short for idiopathic pulmonary arterial hypertension, which causes a higher blood pressure in the lung circulation. Due to the higher pressure in the right circulation pathway the right heart is more stressed which can cause a right heart insufficiency. The mean symptoms are dyspnea, edemas, fatigue, thorax pain, and a growing belly range.

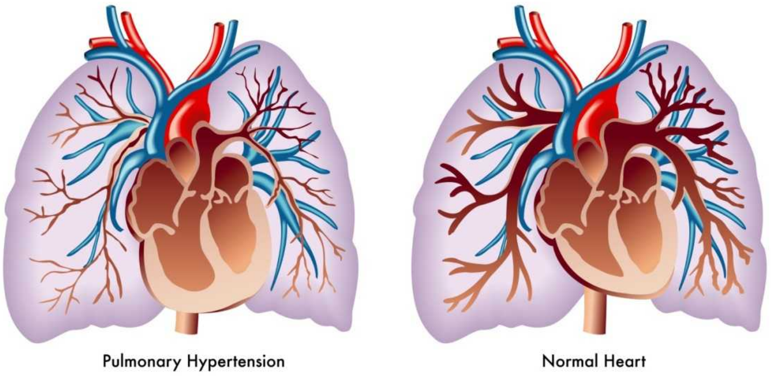


Figure 13: Illustration o the difference between the healthy normal lung vs IPAH affected lung.

3.6.3 Fibrosis

Fibrosis is the state of the lung when the alveoli are surrounded with more and more connective tissue. So, the lungs become hard and cannot expand anymore, which means that breathing is going to be really tough due to limited lung volume. Lung Fibrosis can have multiple causes like medical intake (Bleomycin, Busulfan, Amiodaron), idiopathic, allergies, lung inflammation, and systemic diseases. The symptoms are, like the other diseases, dyspnea, cough, fatigue, and a higher breathing frequency.

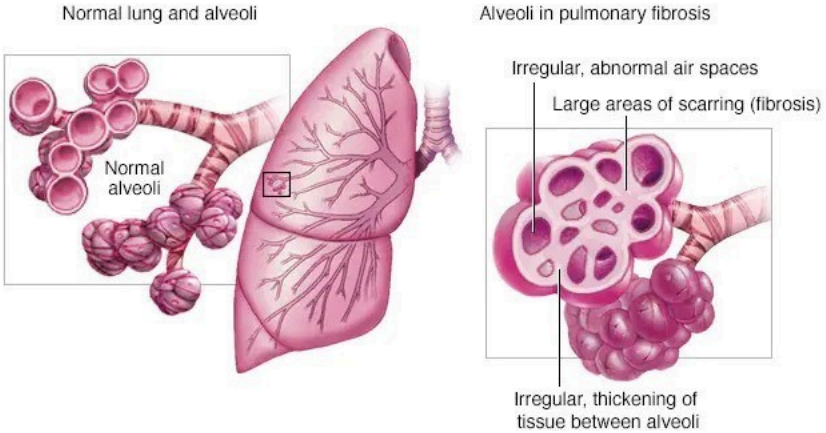

Normal lung and alveoli
Normal alveoli
Alveoli in pulmonary fibrosis
Irregular, abnormal air spaces
Large areas of scarring (fibrosis)
Irregular, thickening of tissue between alveoli

Figure 14: This figure shows the difference in the alveoli of normal healthy lung vs Fibrosis affected lung

Lung diseases are difficult to diagnose because the patients show symptoms really late. Often too late to heal them and most of the symptoms are unspecific, so no one could say that this exact patient would have lung cancer. Therefore, the biggest problem in diagnosing lung cancer is that no one would ever think of it without shown symptoms. If a patient does show any symptoms the first diagnostic feature is a radiography followed by a CT and histological probes. Lung cancers often show tumor markers, which can help finding and defining the tumor to get the best therapy for each patient. But these markers often make it more difficult to find the tumor because nowadays, there are plenty of them so, one could hardly find the right one for the right tumor. So, it is quite challenging to diagnose lung cancer even though one does not search for a specific one.

4 Experimental Setup

4.1 Dataset

A deep learning model requires a large number of labeled data points for training to obtain useful results. Getting annotated medical images requires special expertise, which is not only expensive but also must have to overcome human error. Moreover, It is difficult to get data from rare diseases and this causes dataset imbalance. The dataset consists of 247 images of the lung's histological slides with multiple different classes, namely Chronic Obstructive Pulmonary Disease(COPD), donor, fibrosis, and Idiopathic Pulmonary Arterial Hypertension (IPAH) where each class represents a disease. The slides were collected and processed by Ludwig Boltzmann Institute in Graz. The distribution of the whole dataset is shown in Figure 15.

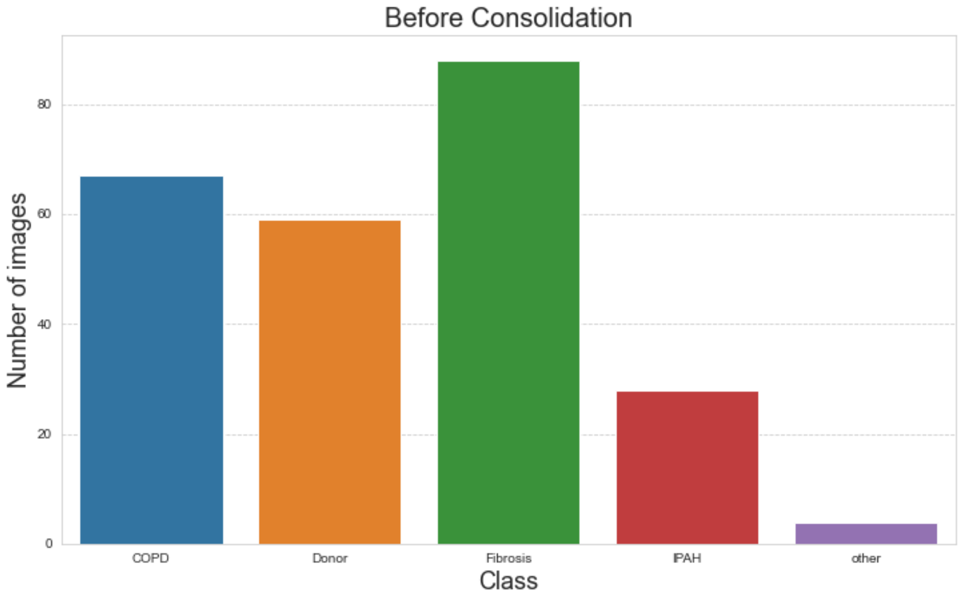


Figure 15: The figure shows the distribution of the data-set after merging

Due to the small sample sizes it would be difficult to perform multi-class classification. So instead of performing multi-class classification, I decided to consolidate all classes into two classes: healthy and non-healthy classes to perform a binary classification as shown in the figure 16. Non-healthy class consists of images from COPD, Fibrosis and IPAH classes, and the healthy class consists of images from the donor class. After consolidation was performed, the whole dataset was randomly split into a training set, validation set and test set with a relative size of 79%, 11% and 10 %.

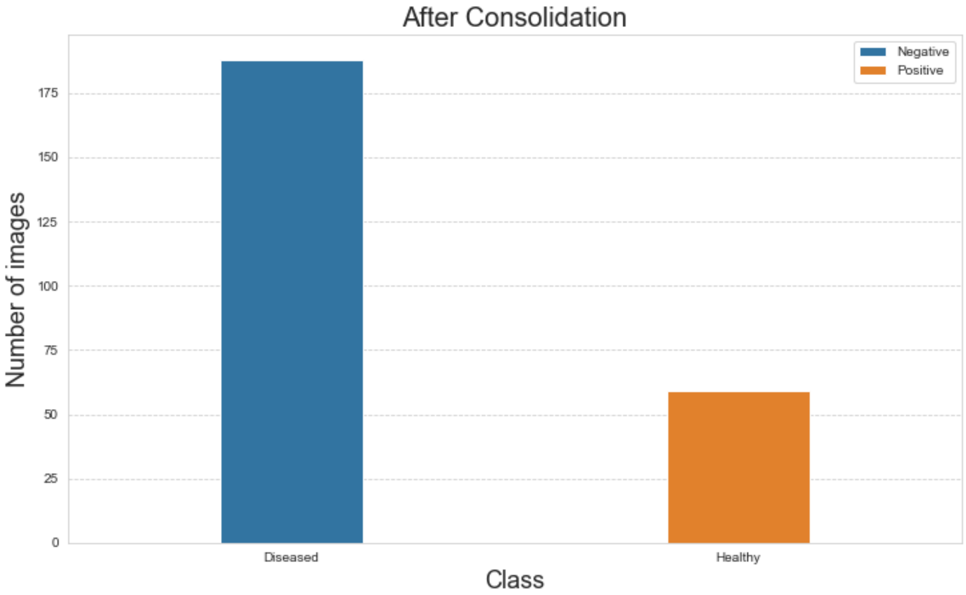


Figure 16: The figure shows the distribution of the data-set after merging

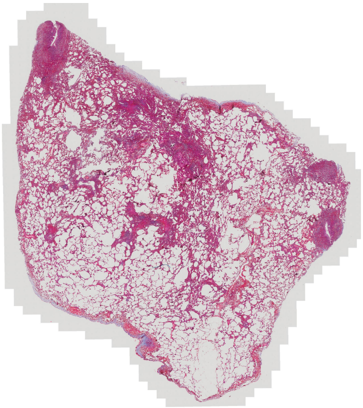


Figure 17: Example of Histological slides

Slide from fibrosis

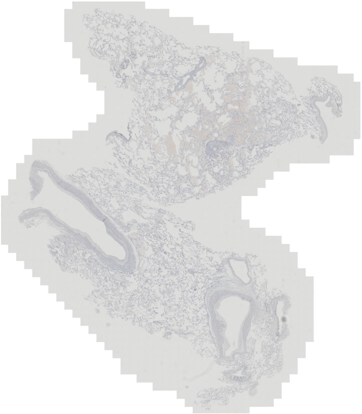


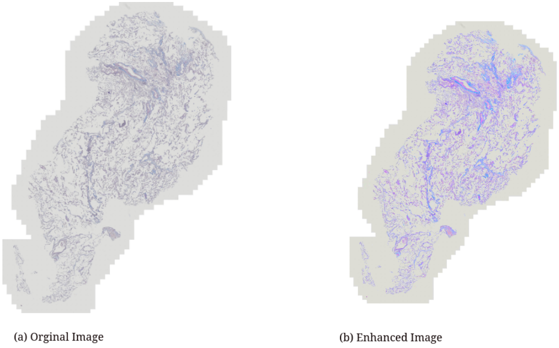
Figure 18: Example of Histological slides

Slide from COPD

4.2 Methods

4.2.1 Preprocessing

Preprocessing of the dataset involves image augmentation and removal of damaged images. Image enhancement aims to provide a better input for a machine learning model. Image enhancement is performed by the standard python library called PIL, which increases the images' color content. The 40% color of image overlaps with the white background, which gives a hard time for the algorithm to identify the critical region of the image. By enhancing the colors of the images, helps to increase the efficiency of the model. The result of the image enhancement is shown in Figure 19.



(a) Orginal Image

(b) Enhanced Image

Figure 19: Preprocessing of the Slide images for segmentation.

Left: Slide before images enhancement. **Right**: Slide after image enhancement

Whole slide images consist of billions of pixels, and it is also essential to remove the background noise and only keep the tissue regions. This is done by thresholding the saturation channel of the image followed by additional morphological closing to fill small gaps and holes. the contours of the detected foreground objects are then filtered based on an area threshold and stored for downstream processing while the segmentation mask for each slide is saved for visualization. The extracted images are used to create image patches with the size of 256x256 pixels. Depending upon the size of the images, the number of patches for each image can vary. Then, the image patches are used to compute low dimensional features using pre-trained Resnet34, which helps to reduce the training time and computational cost.

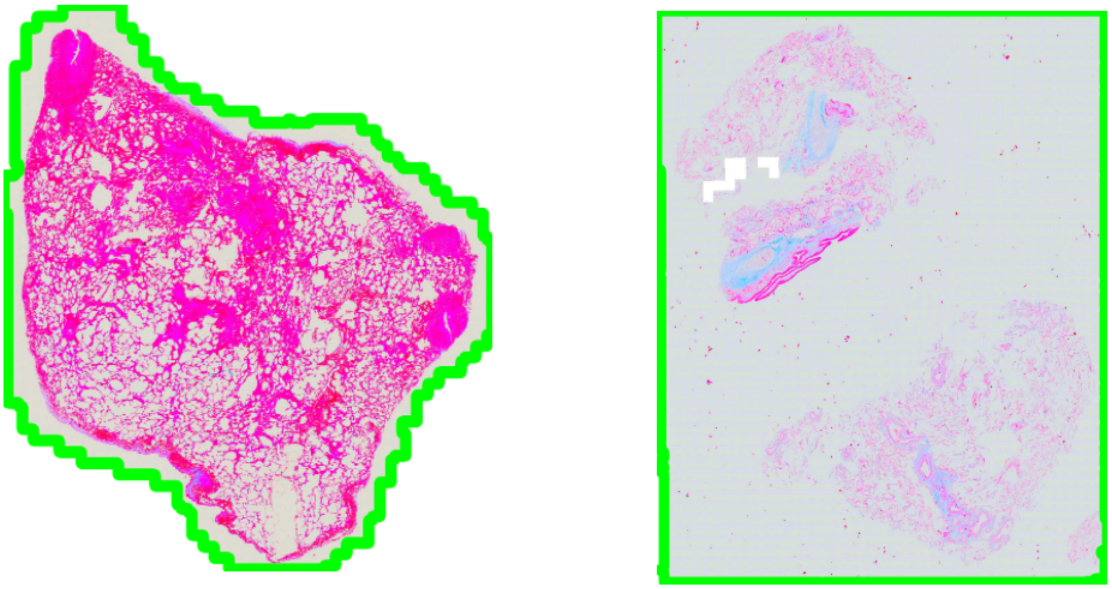


Figure 20: Example of segmented slides.

Right: Segmentation processing could not able find the key part of the images, this will make attention model hard to find the high diagnostic part since the quality of image is poor.

4.2.2 Training

The model is trained by the randomly sampled dataset by the batch size of 1. The loss function of cross-entropy is used for calculating the instance level and slide level against the true label. The total loss is calculated by the summing up the both loss. The Adam optimizer with a learning rate of 1e-3 are used. The model is trained over 50 epoch using early stop with patience of 10 if the validation score is not decreasing over the time. The metrics and loss are monitored using Tensorboard.

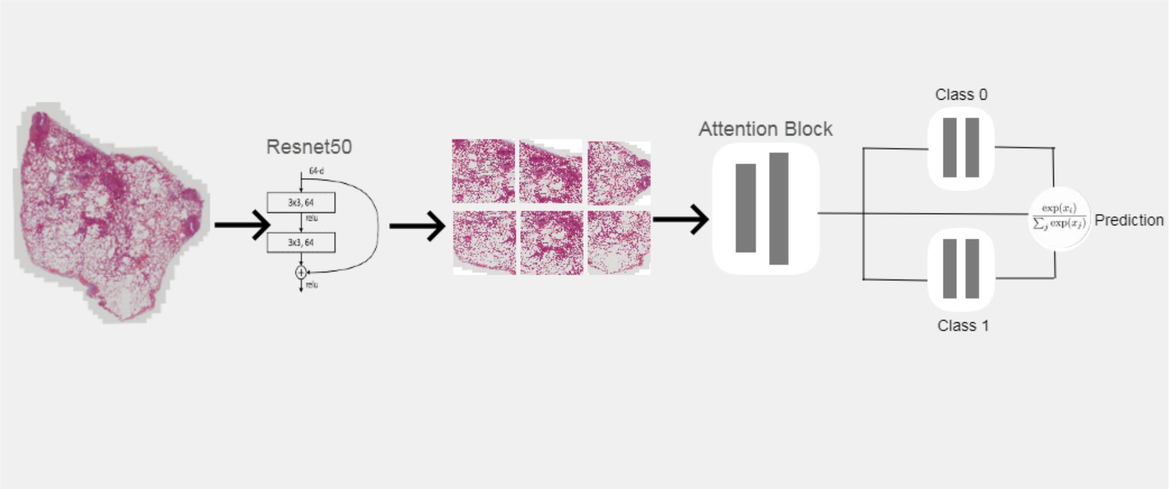


Figure 21: The figure shows the complete architecture of CLAM model.

4.2.3 Evaluation

The main metric used to evaluate the performance of the model is the area under ROC curve (AUC) and accuracy. Other metrics like F_1 score, Precision and Recall are also computed for experimental purpose.

5 Results

The evaluation of the classification performance of the model were done using 10-fold cross-validation. The stratified cross-validation creates a k-fold partition of the entire dataset. Then, for each of the k experiments, it uses (k-1) folds for training and the remaining fold for testing. Stratified cross-validation rearranges the whole dataset in the way that the each fold has the representation of the each class to avoid the disproportion of the one class. The classification error is estimated as the average of the separate errors obtained from k experiments. Each fold is randomly divided into training and test dataset of 80% and 20%, respectively. In each fold, the performance of the model has been evaluated on the validation set for model selection. And finally, the best-performing model has been evaluated in the test set at the end of the training. The classification performance of both CLAM and MIL models are compared using AUC, F_1, and accuracy on all the folds.

The classification scores in Table 1 and Table 2 clearly show that both models have difficulties in classifying the healthy images (labeled: “healthy”). Table 3 shows that there is no significant difference in performance between both models.

Folds	AUC	Accuracy	F_1 Score
1	0.56	0.69	0.71
2	0.52	0.72	0.72
3	0.53	0.50	0.68
4	0.47	0.72	0.76
5	0.52	0.68	0.71
6	0.66	0.72	0.74
7	0.71	0.62	0.69
8	0.39	0.50	0.78
9	0.46	0.62	0.71
10	0.79	0.81	0.69
Avg	0.56	0.66	0.72

Table 1: Scoring metrics of the CLAM model on the test data of all 10-folds.

Folds	AUC	Accuracy	F_1 Score
1	0.67	0.70	0.71
2	0.63	0.72	0.72
3	0.62	0.68	0.68
4	0.55	0.72	0.72
5	0.47	0.68	0.71
6	0.60.	0.68	0.74
7	0.62	0.73	0.56
8	0.51	0.59	0.74
9	0.50	0.62	0.66
10	0.61	0.67	0.69
Avg	0.58	0.67	0.69

Table 2: Scoring metrics of the MIL model on the test data of all 10-folds.

Fold	AUC MIL	AUC CLAM	ABS	Rank
1	0.67	0.56	0.108	7
2	0.63	0.52	0.111	8
3	0.62	0.53	0.086	6
4	0.55	0.47	0.072	5
5	0.47	0.52	0.048	3
6	0.60	0.65	0.057	4
7	0.66	0.71	0.047	2
8	0.51	0.39	0.124	9
9	0.50	0.46	0.042	1
10	0.61	0.79	0.184	10

Table 3: Comparison between AUC of the MIL and CLAM model. A paired Wilcoxon test using the AUC values of both models shows no significant difference (p-value: 0.3843).

6 Discussion

The presented result show comparison of the performance of the two models, the Attention-based Multiple Instances learning and the Clustering-constrained attention Multiple Instance learning (CLAM).The results from Table 3 clearly shows that there is no significant difference between the performance of the CLAM and Attention-based MIL. Both models have similar AUC, Accuracy, and F_1 scores. For the further discussion the CLAM model will be considered.

The actual size of the raw histological slides is over 30.000x30.000 pixels per image. These high-resolution images are incredibly difficult to input in the GPU. To tackle this problem the whole histological slide images are broke down into smaller patches. The smaller patches are packed together in a bag and used for feature extraction to create a embedding for each patches independently inside the bag.

In general, all the attention-based model seems to have auspicious results comparing to any other typical baseline models due to its state-of-the-art architecture. However, the performance of deep learning model mainly depending upon the size and complexity of the dataset. The size of the dataset combined with high complexity of images played a vital role in the outcome of the CLAM model in this experiment. Figure 22 and Figure 23 shows the AUC score in the validation set for 1 - 10 folds during the training. The early stop was initialized to avoid over-fitting the training data. 75% of the folds AUC score lies between 0.5 - 0.6 which clearly show that the model has hard time to separate the positive class from the negative. The imbalance dataset is also one the reason behind this problem because ratio of the negative class is really higher than the ratio of the positive class. So the each folds contain more negative instance than positive negative.

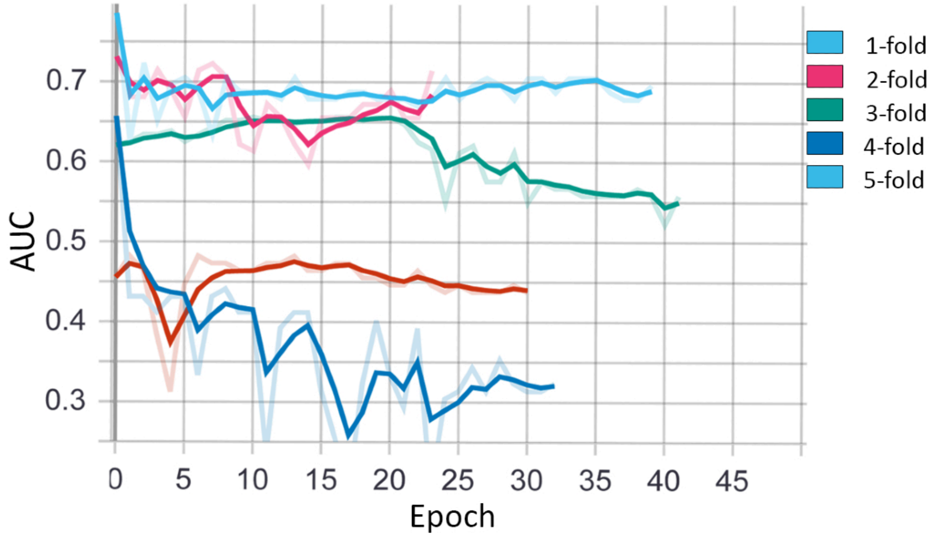


Figure 22: The figure shows the AUC scores of the validation set for the fold 1 to 5 during the training.

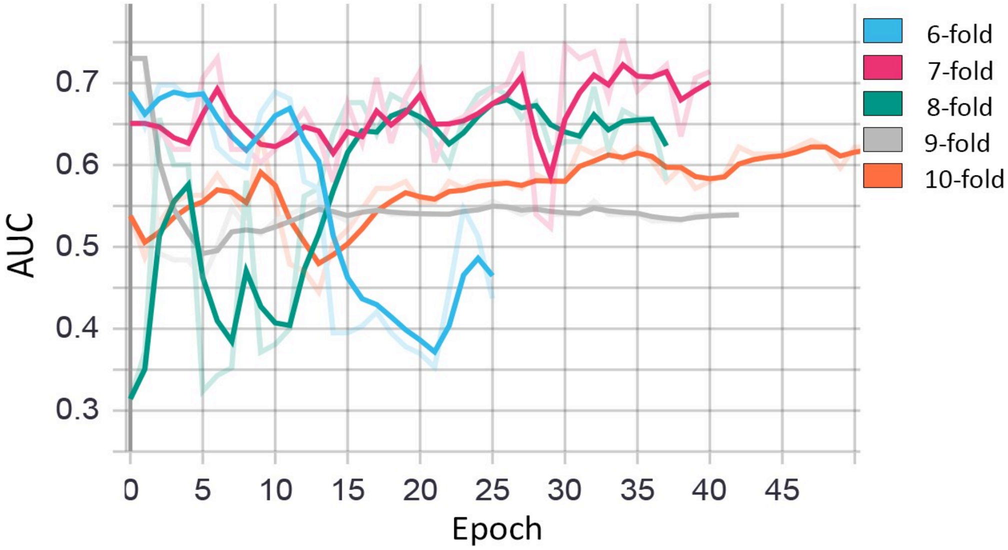


Figure 23: The figure shows the AUC scores of the validation set for fold 6 to 10 during the training.

Figure 24 and Figure 25 shows the validation loss of fold 1 to 10 during the training and the figures clearly shows that half of the fold's validation loss are not approaching to zero. The reason behind the problem could be the properties of the dataset or the hyperparameter of the model. As mention in the section dataset, we did not have enough data-points for a specific class to implement multi-class classification. So, I had to combine all the non-healthy classes into one class to perform the binary class classification. And the properties of the images vary in accordance with their with respective classes. The different learning rate were used to train multiple models, but the change in learning rate did not improve the performance of the model. The other reason could be the images are too complex for the model to learn.

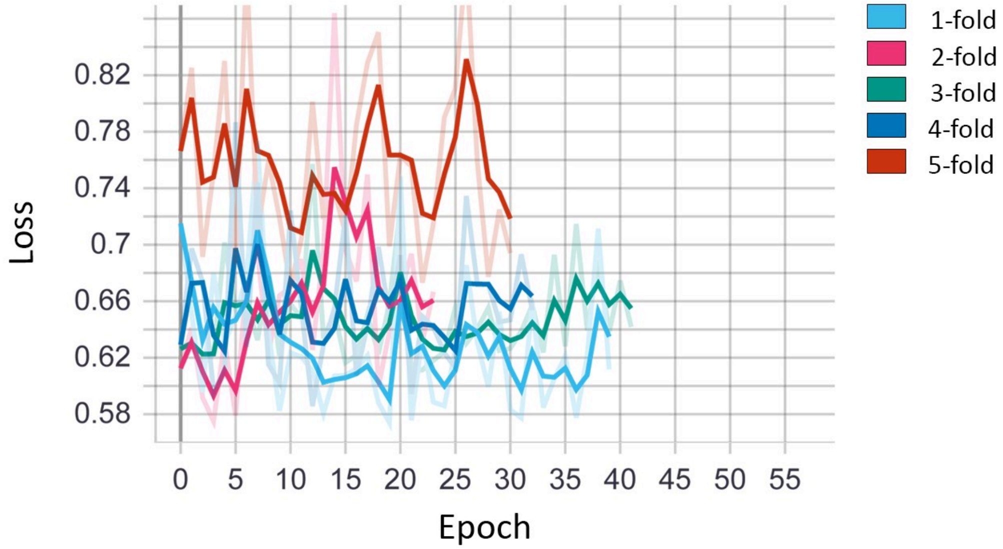


Figure 24: The figure shows the loss score of the validation set for the 1-5 folds during the training.

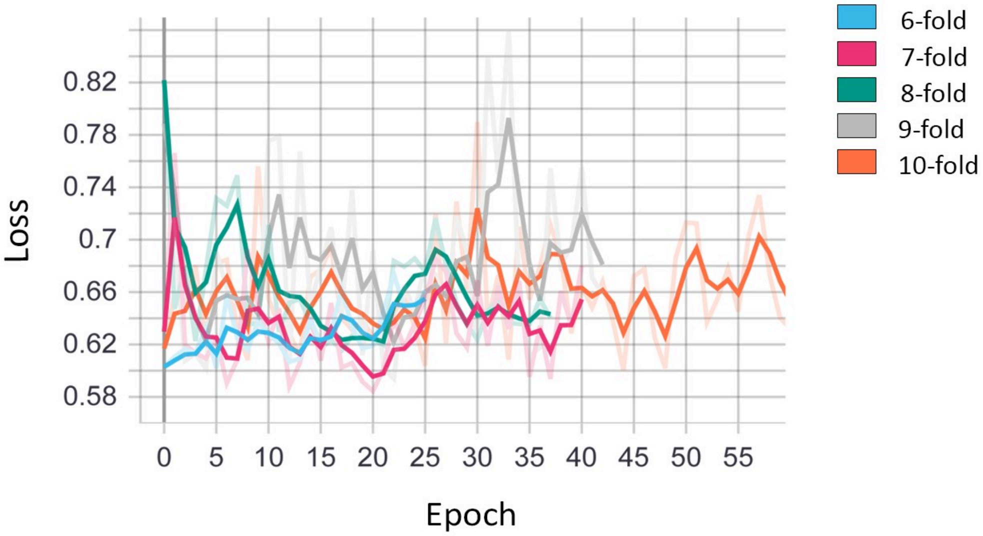


Figure 25: The figure shows the loss score of the validation set for the 6-10 folds during the training.

7 Conclusion

In this thesis, I compared the performance of Attention-based MIL pooling and CLAM for the classification of given high-resolution medical images. The primary purpose of the analysis is to find the best performing model with the limited data available. The results clearly illustrate that there is no significant difference in the performance between the two models. I observed that the most common failures of the model are the size of the dataset and image complexity. For further research, a more extensive dataset and proper preprocessing of the images must be performed to improve the model’s performance because the result did not show a promising outcome.

List of Figures

List of Tables

References

[1] Abien Fred Agarap. *Deep Learning using Rectified Linear Units (ReLU)*. 2019. arXiv: 1803.08375 [cs.NE].

[2] Jayesh Bapu Ahire. *The Artificial Neural Networks Handbook*. URL: https://medium.com/@jayeshbahire/the-artificial-neural-networks-handbook-part-4-d2087d1f583e. (accessed: 17.03.2021).

[3] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. *Neural Machine Translation by Jointly Learning to Align and Translate*. cite arxiv:1409.0473Comment: Accepted at ICLR 2015 as oral presentation. 2014. URL: http://arxiv.org/abs/1409.0473.

[4] Belhassen Bayar and Matthew C. Stamm. "A Deep Learning Approach to Universal Image Manipulation Detection Using a New Convolutional Layer". In: (2016). URL: https://doi.org/10.1145/2909827.2930786.

[5] Hakan Bilen and Andrea Vedaldi. *Weakly Supervised Deep Detection Networks*. 2016. arXiv: 1511.02853 [cs.CV].

[6] C H Chen. *Computer Vision in Medical Imaging*. WORLD SCIENTIFIC, 2014. eprint: https://www.worldscientific.com/doi/pdf/10.1142/8766. URL: https://www.worldscientific.com/doi/abs/10.1142/8766.

[7] *Convolutional neural network*. URL: https://en.wikipedia.org/wiki/Convolutional_neural_network (visited on May 16, 2021).

[8] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. The MIT Press, 2016.

[9] Aurlien Gron. *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. 2nd. O'Reilly Media, Inc., 2019.

[10] Aurlien Gron. *Deep Learning for Vision Systems*. Manning Publications, 2020.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. *Deep Residual Learning for Image Recognition*. 2015. arXiv: 1512.03385 [cs.CV].

[12] Maximilian Ilse, Jakub M. Tomczak, and Max Welling. “Attention-based Deep Multiple Instance Learning”. In: (2018). arXiv: 1802.04712 [cs.LG].

[13] Sathish Reddy Indurthi, Insoo Chung, and Sangha Kim. “Look Harder: A Neural Machine Translation Model with Hard Attention”. In: (July 2019). URL: https://www.aclweb.org/anthology/P19-1290.

[14] Aseem Kashyap. *How Convolution Neural Networks interpret images*. URL: https://towardsdatascience.com/how-convolution-neural-networks-interpret-images-1f99913070b2. (accessed: 23.03.2021).

[15] Qiuhong Ke, Jun Liu, Mohammed Bennamoun, Senjian An, Ferdous Sohel, and Farid Boussaid. *Computer Vision for Humanâ□□Machine Interaction*. Ed. by Marco Leo and Giovanni Maria Farinella. Computer Vision and Pattern Recognition. Academic Press, 2018, pp. 127–145. URL: https://www.sciencedirect.com/science/article/pii/B9780128134450000058.

[16] Susanne Kimeswenger, Elisabeth Rumetshofer, Markus Hofmarcher, Philipp Tschandl, Harald Kittler, Sepp Hochreiter, Wolfram HÃ¶tzenecker, and GÃ¼nter Klambauer. “Detecting cutaneous basal cell carcinomas in ultra-high resolution and weakly labelled histopathological images”. In: (2019). arXiv: 1911.06616 [eess.IV].

[17] Pedro Larranaga, Borja Calvo, Roberto Santana, Concha Bielza, Josu Galdiano, IÃ±aki Inza, JosÃ© A. Lozano, RubÃ©n ArmaÃ±anzas, GuzmÃ¡n SantafÃ©, Aritz PÃ©rez, and Victor Robles. “Machine learning in bioinformatics”. In: *Briefings in Bioinformatics* 7.1 (Mar. 2006), pp. 86–112. eprint: https://academic.oup.com/bib/article-pdf/7/1/86/23992771/bbk007.pdf. URL: https://doi.org/10.1093/bib/bbk007.

[18] Ming Y. Lu, Drew F. K. Williamson, Tiffany Y. Chen, Richard J. Chen, Matteo Barbieri, and Faisal Mahmood. “Data Efficient and Weakly Supervised Computational Pathology on Whole Slide Images”. In: (2020). arXiv: 2004.09666 [eess.IV].

[19] Alexander SelvikvÃ¥g Lundervold and Arvid Lundervold. “An overview of deep learning in medical imaging focusing on MRI”. In: *Zeitschrift fÃ¼r Medizinische Physik* 29.2 (2019). Special Issue: Deep Learning in Medical Physics, pp. 102–127.

[20] Muhammad Razzak, Saeeda Naz, and Ahmad Zaib. “Deep Learning for Medical Image Processing: Overview, Challenges and Future”. In: (Apr. 2017).

[21] Sumit Saha. *Convolutional neural network*. URL: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53 (visited on June 11, 2021).

[22] Prodip Hore Sayan Chatterjee. *A Comprehensive Guide to Attention Mechanism in Deep Learning for Everyone*. URL: https://www.analyticsvidhya.com/blog/2019/11/comprehensive-guide-attention-mechanism-deep-learning/ (visited on Apr. 12, 2021).

[23] Abhishek Singh. *Brief Introduction to Attention Models*. URL: https://towardsdatascience.com/attention-networks-c735befb5e9f (visited on Apr. 12, 2021).

[24] Vincent Tatan. *Understanding CNN (Convolutional Neural Network)*. URL: https://towardsdatascience.com/understanding-cnn-convolutional-neural-network-69fd626ee7d4 (visited on Mar. 17, 2021).

[25] Chong Wang, Weiqiang Ren, Junge Zhang, Kaiqi Huang, and Steve Maybank. *Large-Scale Weakly Supervised Object Localization via Latent Category Learning*. Jan. 2015.

[26] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. *Residual Attention Network for Image Classification*. 2017. arXiv: 1704.06904 [cs.CV].

[27] Lilian Weng. *Attention? Attention!* URL: http://lilianweng.github.io/lil-log/2018/06/24/attention-attention.html (visited on Apr. 12, 2021).

[28] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. “Show, Attend and Tell: Neural Image Caption Generation with Visual Attention”. In: (2016). arXiv: 1502.03044 [cs.LG].

[29] *Zero Padding in Convolutional Neural Networks explained*. URL: https://deeplizard.com/learn/video/qSTv_m-KFk0. (accessed: 23.03.2021).

[30] S. K. Zhou, H. Greenspan, C. Davatzikos, J. S. Duncan, B. van Ginneken, A. Madabhushi, J. L. Prince, D. Rueckert, and R. M. Summers. “A Review of Deep Learning in Medical Imaging: Imaging Traits, Technology Trends, Case Studies With Progress Highlights, and Future Promises”. In: *Proceedings of the IEEE* (2021), pp. 1–19.