UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Assessment of malignancy levels of lung nodules by utilizing transfer learning deep model

Diploma Thesis

Lefki-Ioanna Panagiotou

Supervisor: Panagiota Tsompanopoulou

Volos 2021

UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Assessment of malignancy levels of lung nodules by utilizing transfer learning deep model

Diploma Thesis

Lefki-Ioanna Panagiotou

Supervisor: Panagiota Tsompanopoulou

Volos 2021

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Αξιολόγηση επιπέδων κακοήθειας πνευμονικών όζων χρησιμοποιώντας μοντέλα βαθιάς μάθησης με μεταφορά γνώσης

Διπλωματική Εργασία

Λευκή-Ιωάννα Παναγιώτου

Επιβλέπων/πουσα: Παναγιώτα Τσομπανοπούλου

Βόλος 2021

Approved by the Examination Committee:

Supervisor **Panagiota Tsompanopoulou**
Associate Professor, Department of Electrical and Computer Engineering, University of Thessaly

Member **Konstantinos Marias**
Head of Computational BioMedicine Laboratory (CBML) at the Institute of Computer Science at FORTH (ICS-FORTH) and Associate Professor of Department of Electrical and Computer Engineering at Hellenic Mediterranean University of Crete,

Member **Georgios Stamoulis**
Professor, Department of Electrical and Computer Engineering, University of Thessaly

Acknowledgements

Firstly, I would like to thank Head of Computational BioMedicine Laboratory (CBML) at the Institute of Computer Science at FORTH (ICS-FORTH) and associate professor of Dept. of Electrical and Computer Engineering at Hellenic Mediterranean University of Crete, Dr. Konstantinos Marias and his PhD student, Eleftherios Trivizakis for giving me the opportunity to work on a project I always wanted and for their guidance throughout my journey of thesis writing.

Secondly I am thankful to Dr. Elias Houstis, emiretus professor of University of Thessaly, Department of Electrical and Computer Engineering, who believed in me and gave me the appropriate knowledge to complete my studies.

Moreover, I would like to thank associate professor Dr. Panagiota Tsompanopoulou and professor Dr. Georgios Stamoulis for being also my thesis supervisors as well as all of the department's professors and staff who helped me the last 5 years.

Last but not least, I want to express my gratitude to all the people who have supported me on the journey of my life so far. In particular, I want to thank my parents, my brother and my cousins who set right standards for me and offered me unconditional love and support.

DISCLAIMER ON ACADEMIC ETHICS AND INTELLECTUAL PROPERTY RIGHTS

«Being fully aware of the implications of copyright laws, I expressly state that this diploma thesis, as well as the electronic files and source codes developed or modified in the course of this thesis, are solely the product of my personal work and do not infringe any rights of intellectual property, personality and personal data of third parties, do not contain work / contributions of third parties for which the permission of the authors / beneficiaries is required and are not a product of partial or complete plagiarism, while the sources used are limited to the bibliographic references only and meet the rules of scientific citing. The points where I have used ideas, text, files and / or sources of other authors are clearly mentioned in the text with the appropriate citation and the relevant complete reference is included in the bibliographic references section. I also declare that the results of the work have not been used to obtain another degree. I fully, individually and personally undertake all legal and administrative consequences that may arise in the event that it is proven, in the course of time, that this thesis or part of it does not belong to me because it is a product of plagiarism».

The declarant

Lefki-Ioanna Panagiotou

Abstract

It has been proven that lung cancer is one of the most commonly and lethal diseases, diminishing life expectancy worldwide. In past years, professional scanners that can project the entire human body, have produced an increasing number of bio-medical images displaying lung cancer nodules. At the same time, the scientific community took advantage of ongoing advances in computer vision via Deep Learning (a sub-field of Artificial Intelligence) to deal with challenges like segmentation, detection, and classification of lung cancer nodules. All of them have a common aim; provide support and assistance to radiologists for punctual diagnoses.

In this thesis, we offer a range of Deep Learning approaches for analyzing images from the Lung Image Database Consortium-Image Database Resource Initiative [1] (LIDC-IDRI) database using Convolutional Neural Networks techniques. All these methods are applied to solve the 2-class, 3-class, 4-class and 5-class lung cancer nodule classification problem.

Περίληψη

Έχει αποδειχθεί ότι ο καρκίνος του πνεύμονα είναι μια από τις πιο κοινές και θανατηφόρες ασθένειες, μειώνοντας το προσδόκιμο ζωής παγκοσμίως. Τα τελευταία χρόνια, επαγγελματικοί σαρωτές που μπορούν να προβάλλουν ολόκληρο το ανθρώπινο σώμα, παρήγαγαν έναν αυξανόμενο αριθμό βιο-ιατρικών εικόνων που εμφανίζουν οζίδια καρκίνου του πνεύμονα. Ταυτόχρονα, η επιστημονική κοινότητα εκμεταλλεύτηκε τις συνεχείς προόδους στην υπολογιστική όραση μέσω της Βαθιάς Μάθησης (ένα υπο-πεδίο της Τεχνητής Νοημοσύνης) για να αντιμετωπίσει προκλήσεις όπως η κατάτμηση, η ανίχνευση και η ταξινόμηση των όζων του καρκίνου του πνεύμονα. Όλα τα παραπάνω έχουν ως κοινό στόχο να παρέχουν υποστήριξη και βοήθεια στους ακτινολόγους για ακριβείς διαγνώσεις.

Σε αυτήν την εργασία, προσφέρουμε μια σειρά από προσεγγίσεις Βαθίας Μάθησης για την ανάλυση εικόνων από τη βάση δεδομένων Lung Image Database Consortium-Image Database Resource Initiative [1] (LIDC-IDRI) χρησιμοποιώντας τεχνικές Συνελικτικών Νευρωνικών Δικτύων. Όλες αυτές οι μέθοδοι εφαρμόζονται για την επίλυση του προβλήματος ταξινόμησης όζων καρκίνου του πνεύμονα σε 2, 3, 4 και 5 κατηγορίες.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

Cancer is a term used to describe a group of diseases characterized by abnormal cell proliferation [2] [3]. Lung cancer is among the most frequent and fatal types of cancer. The last one is the second most commonly diagnosed cancer in men, after prostate cancer, and the third most prevalent cancer in women, after breast and colorectal cancer [4] [5]. According to GLOBOCAN Estimates of Incidence and Mortality Worldwide for 36 Cancers in 185 Countries [6], in the year 2021, 2,206,771 more new cases of lung cancer were presented, occupying a percentage of 11.4 against all new cancer cases. In total, there were an estimated 1,796,144 deaths as a cause of lung cancer, which represents approximately the 18.0% of all cancer deaths.

All the above emphasizes the importance of a prompt and accurate diagnosis of lung cancer cases. For this purpose, in the last decades, a variety of new medical imaging techniques [7] [8] has been developed, with Computed Tomography (CT) scan [9] to be he most functional one [10]. As a result, many datasets with bio-medical images have been constructed [11]. The ever-growing field of Artificial Intelligence (AI) [12] exploits these data positively ,through Deep Learning (DL) techniques [13], to examine them and learn more about tumors. More specifically, Convolutional Neural Networks (CNN) technique is well-established in bio-medical imaging analysis [14] and this is proved by many researches relative τo segmentation [15] [16], detection [17] [18] and classification [19] [20] of lung cancer nodules.

In this thesis, we adopt a Deep Learning (DL) approach to categorize lung cancer nodules from *Lung Image Database Consortium-Image Database Resource Initiative* [1] (LIDC-IDRI) database, according to their malignancy level. Our methods use pretrained Convolutional Neural Networks (pretrained CNN) [21] and Transfer Learning [22] and have as a result

the 2-class (binary), 3-class (trinary), 4-class and 5-class classification of nodules.

The structure of the thesis has the following format. In chapter 2, an analytical report of Medical Imaging domain is provided. Then, in chapter 3, we present an extended introduction to Deep Learning and some of the techniques that uses. Chapter 4 is referred to the LIDC-IDRI database that will be later used, in chapter 5, to apply our Deep Convolutional Neural Networks architectures. The results will be presented in chapter 6. In chapter 7, we discuss about the importance of our work for the healthcare system and we refer to some limitations that we faced. Finally, in chapter 8 we conclude our findings and make some thoughts for future work to improve our methodology.

Chapter 2

Medical Imaging

Medical Imaging [23] [24] is a part of *biological imaging* [25] and encompasses a variety of techniques and processes for viewing the human body for diagnostic, monitoring, treatment, and follow up purposes. The most commonly used imaging techniques in clinical medicine include X-ray radiography, computed tomography (CT), magnetic resonance imaging (MRI), ultrasound, and digital pathology (Fig.2.1 and Fig.2.2). Each one of these technologies exploits physical phenomena such as light, electromagnetic radiation, radioactivity, nuclear magnetic resonance, and sound to provide invaluable information for better understanding and treating the part of the body being investigated.

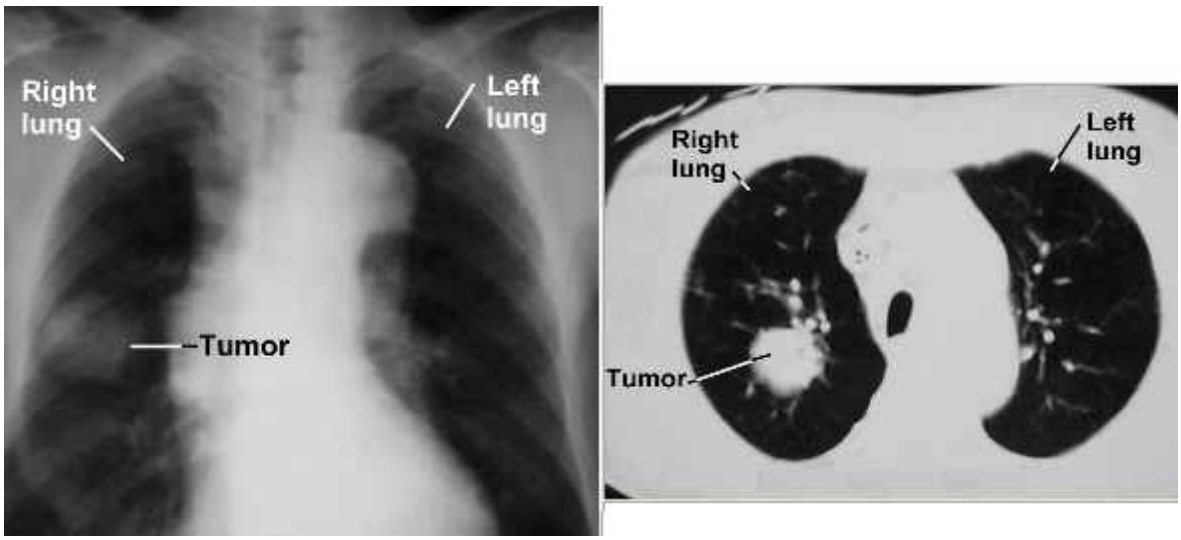


Figure 2.1: X-ray and CT scan of lung cancer. (Figure from [26])

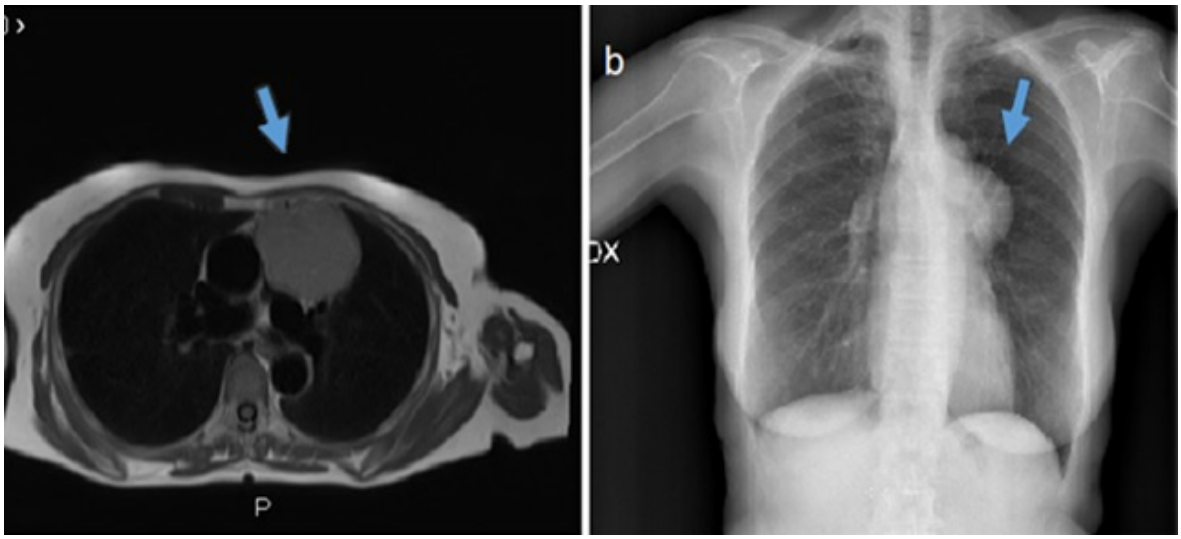


Figure 2.2: A 73-year-old woman with a visible pulmonary nodule at MRI (left) and chest x-ray (right). All images courtesy of Marco Ali. (Figure from [27])

Medical imaging is crucial in a variety of medical settings and at all major levels of health care. In the medical community, it is firmly believed, that the only way who can lead to effective decisions in public health and preventive medicine, as well as in curative and palliative care, is the existence of accurate diagnoses. In general, diagnostic imaging services are critical for verifying, correctly diagnosing, and documenting the progress of various diseases, as well as analyzing treatment responses.

2.1 Computed Tomography

In *Computed tomography* (CT), the term tomography comes from the Greek words tomos (a cut, a slice, or a section) and graphein (to write or record) [28]. It is a structural imaging process that uses sophisticated X-ray technology to help detect a variety of diseases and conditions. It combines several X-ray projections obtained from various angles to make cross-sectional images of locations inside the body by rotating the X-ray tube around the body. Simultaneously, at the opposite end of the X-ray tube, a detector array captures transmission projection data, which is based on the idea that the density of the tissue passed through by the X-ray beam may be determined by calculating the attenuation coefficient [29]. The grey levels in a CT slice that are related to X-ray attenuation, represent the proportion of X-rays absorbed as they passing through each voxel [30].

All the information from a CT scan is stored in DICOM files. *Digital Imaging and Communications in Medicine* (DICOM) is the international standard to transmit, store, retrieve, print, process, and display medical imaging information, enabling the integration of medical imaging devices such as scanners[31]. Each DICOM file has a, so called, header part, which contains significant metadata, as well as a part for the pixel data of the image. Patient health information (PHI) such as patient name, medical record number, and date of birth, as well as image acquisition characteristics such as picture dimensions, voxel size, repetition time (TR), and voxel data type, are commonly found in the header component of a DICOM file (Fig.2.3).

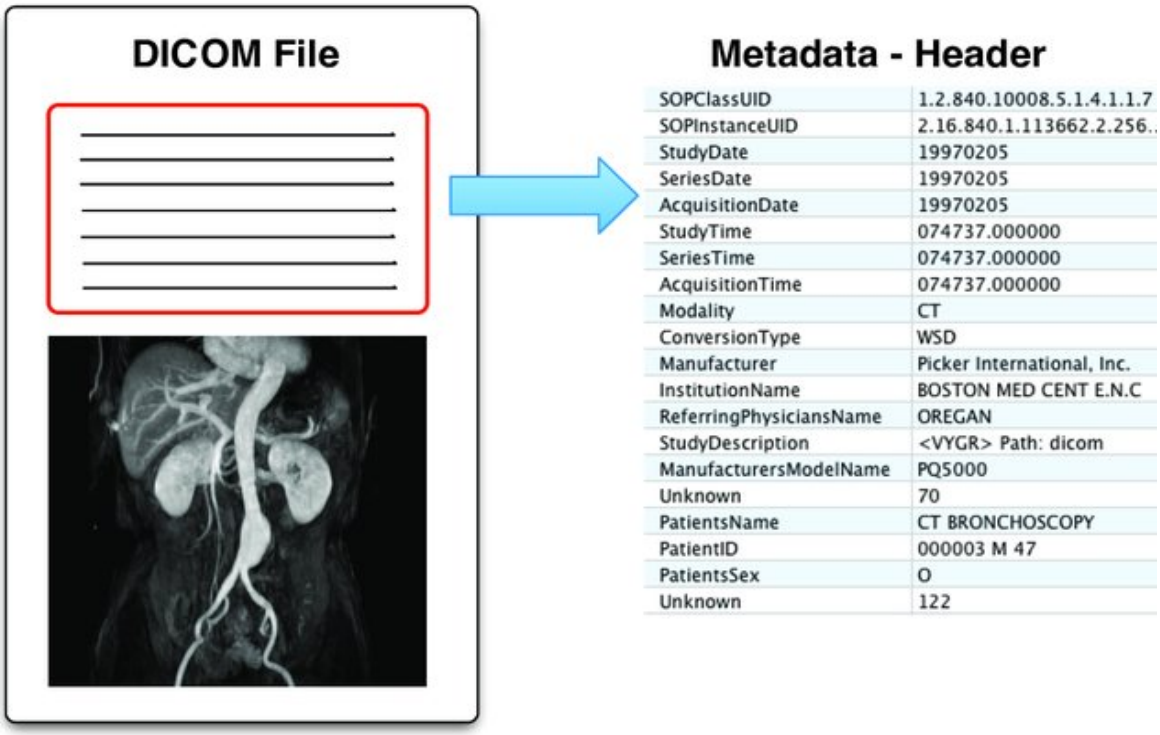


Figure 2.3: DICOM object with metadata and image. The header is zoom in (right hand side) and it illustrates a few of the attributes that the object contains. (Figure from [32])

Chapter 3

Deep Learning

Deep Learning is a subset of *Machine Learning* (ML), which is a subset of *Artificial Intelligence* (AI) Fig.3.1. Artificial Intelligence is a domain of computer science, which is concerned with training computer systems to think and learn by imitating the way that the human brain functions. In this way, machines will be capable of performing tasks that typically require human intelligence. [33] Machine Learning and Deep Learning are the two main techniques that help Artificial Intelligence work.

Machine Learning, as well as Deep Learning, learns from data coming from the past and makes predictions or decisions based on them. The main difference between those two applications of Artificial Intelligence is that the second one is also very helpful in the wider field of Data Science because it is a way to automate predictive analytics.[34] It can accept as input unlabeled data in its raw format and after many complex computations end up defining the hierarchy of features and how useful each one could be in the research, without human intervention. [35]

3.1 Neural Networks

The term Deep Learning is closely related to *Artificial Neural Networks* (ANN). Artificial Neural Networks was reported for the first time by McCulloch and Pitts in the 1940s [37]) as mathematical models made to work and make decisions similar to the way that humans' brain does. The most important thing to construct functional ANNs is data. Fortunately, nowadays in the Big Data era, we can train them with plenty of data, making them more robust and trustworthy for humans.

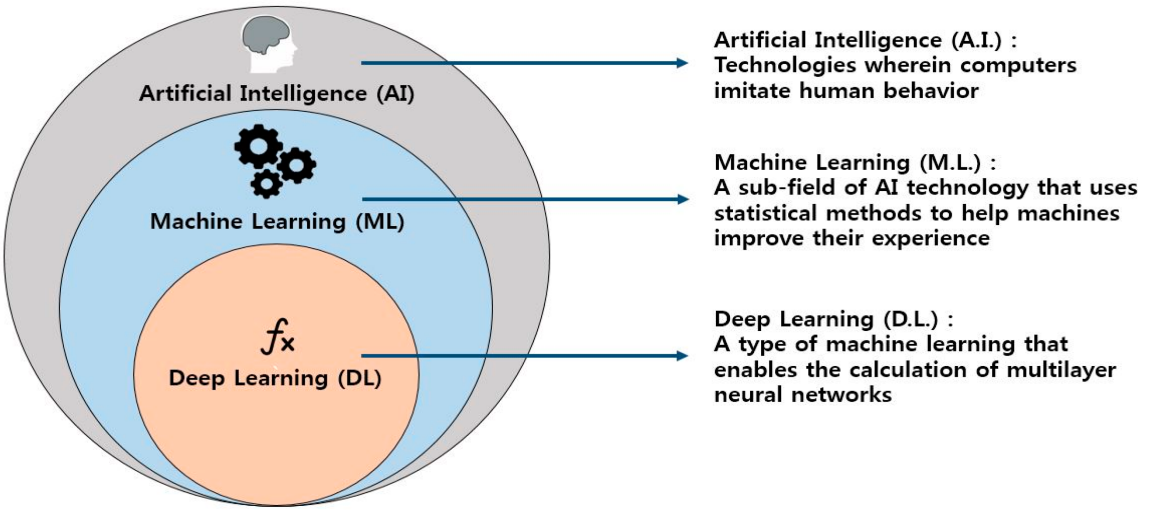


Figure 3.1: AI, Machine Learning, Deep Learning and the relationship between them (Figure from [36])

The basic form of *Neural Networks* (NNs) consists of three layers, which are *Input layer*, *Hidden layer* and *Output layer*. In the Input layer, Neural Network accepts the independent variables or inputs of the model (x). Then, each hidden layer is composed of a large number of simple elements, called neurons, and is responsible for processing the inputs. A Neural Network is characterized as Artificial if only one Hidden layer exists. Deep neural networks are Artificial Neural Networks that have more than one hidden layer between input and output. This can be shown in Fig.3.2. Finally, the Output layer generates the desired predictions.

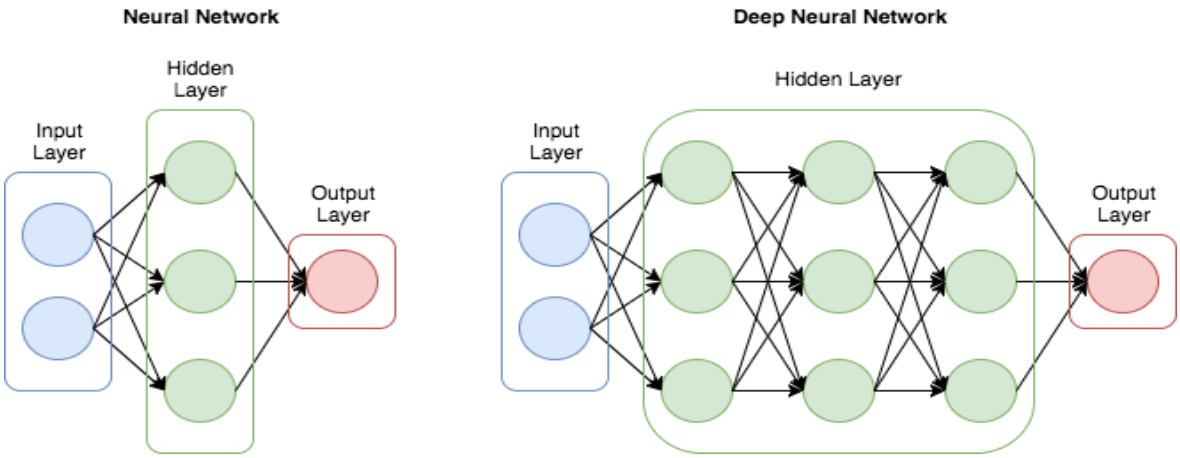


Figure 3.2: Artificial vs Deep Neural Networks (Figure from [38])

In Fig.3.3, the learning process of a neuron (or so called perceptron) is given schematically, just for a better understanding of the function of Neural Networks. As it seems, every input $\mathbf{x_i}$, is associated with the corresponding weight $\mathbf{w_i}$. Then, the sum of these terms becomes the argument of an activation function ϕ, resulting in the output y. This can be described mathematically with Eq.3.1.

$$y = \phi(\mathbf{w}^T\mathbf{x}) \tag{3.1}$$

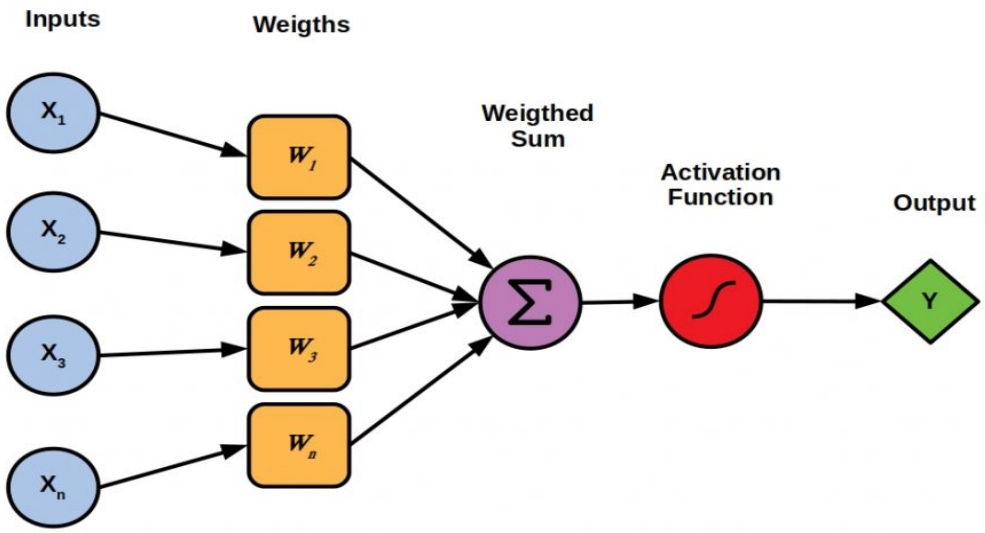


Figure 3.3: Schema of a perceptron (Figure from [39])

Activation functions and weights are responsible for transferring the input from one layer to another in the correct form. In most cases non-linear activation functions are used, because they are differentiable and can solve the *Vanishing Gradient problem* [40] by transforming the output values to be within an acceptable range, but there are also linear. Some common activation functions are presented below mathematically and schematically in Fig.3.4:

1. Binary step function, which is a linear function

$$y = \begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases} \tag{3.2}$$

2. Linear or Identity function, where the activation is proportional to the input

$$y = x \tag{3.3}$$

3. Sigmoid function, which is a non-linear function

$$y = \frac{1}{1 + e^{-x}} \tag{3.4}$$

4. TanH function, which is very similar to the sigmoid

$$y = \frac{2}{1 + e^{-2x}} - 1 \tag{3.5}$$

5. ReLU, which is the most common function used for hidden layers

$$y = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases} \tag{3.6}$$

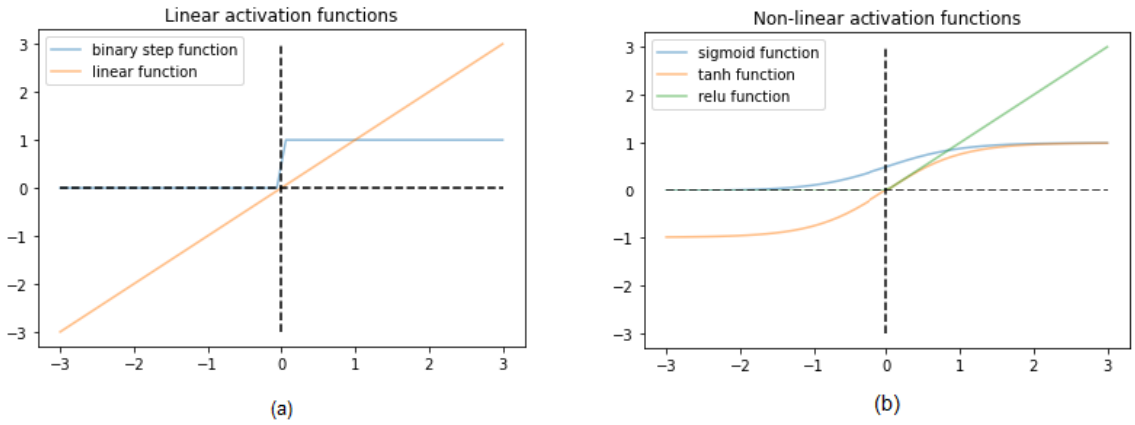


Figure 3.4: Linear(a) and Non-linear(b) activation functions

3.2 Convolutional Neural Network

Convolutional Neural Networks (CNN) is a category of *Deep Neural Networks* (DNN), which is inspired by the primary visual cortex. They are suitable for many research domains associated with processing data that has a grid pattern, such as images and videos. Furthermore, they are designed based on the architectural principle of *shared weights*. The purpose of this principle is to reduce the number of training parameters by setting many weights to the same value.

As shown in Fig. 3.5, every Convolutional Neural Network consists of two main parts. The first one, known as *convolutional base*, is used to extract the features from the input array. *Convolutional Layers* and *Pooling Layers* belong to the convolutional base. The second part of a Convolutional Neural Network, known as *head*, is used to determine the final output, i.e. the class of an image if the task is a classification problem. *Fully-Connected Layers* and *Loss Layer* belong to head.

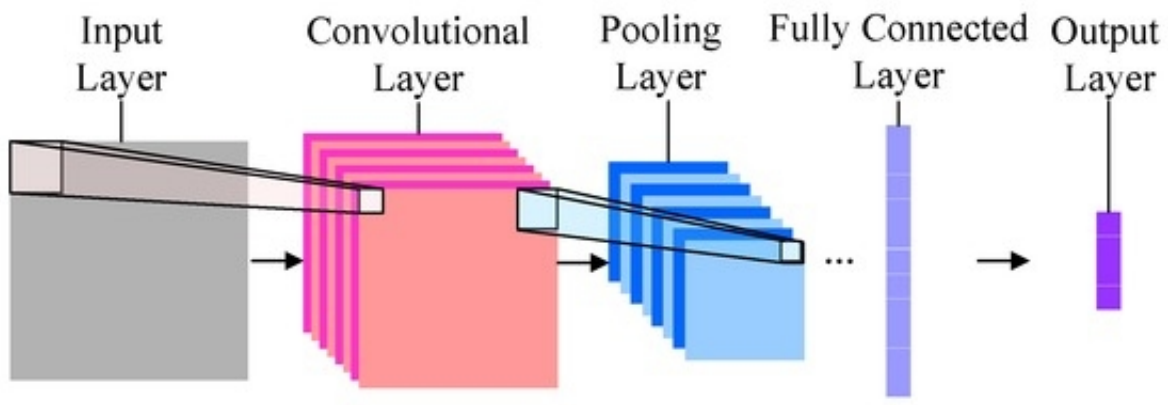


Figure 3.5: Convolutional Neural Network architecture (Figure from [41])

The parameters in the Convolutional Layer are arrays with small dimensions called filters or kernels. They operate by scanning over an image and calculating the dot product between filters and this image, ending up in maps that illustrate the extracted features. This operation is shown in Fig. 3.6. Then, the Pooling Layer is following to reduce the overall computational complexity of image analysis. This is done by reducing the spatial dimension of the input image (without loss of information) after convolution and the number of parameters in the net. There are many different pooling operations such as average and max pooling (Fig. 3.7). Finally, the operation ends by flattening the output of the convolutional base to be ready for the Fully-Connected Layer, which is a FeedForward Neural Network, and computing the final

output.

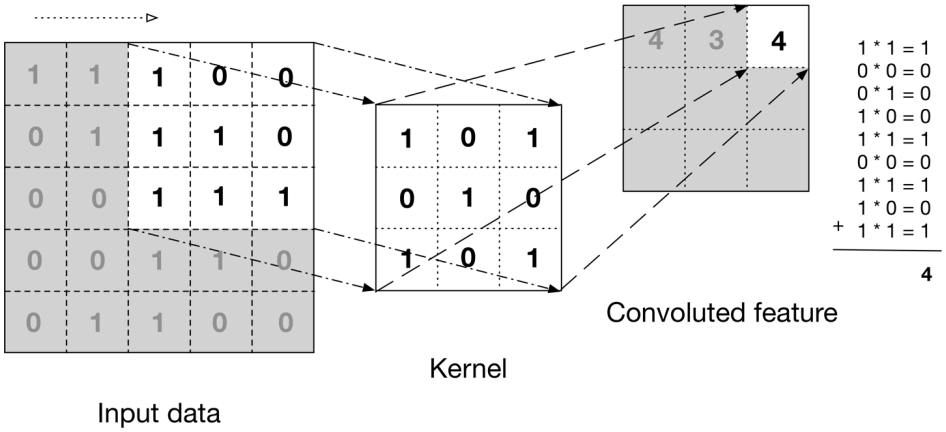


Figure 3.6: Convolution procedure (Figure from [42])

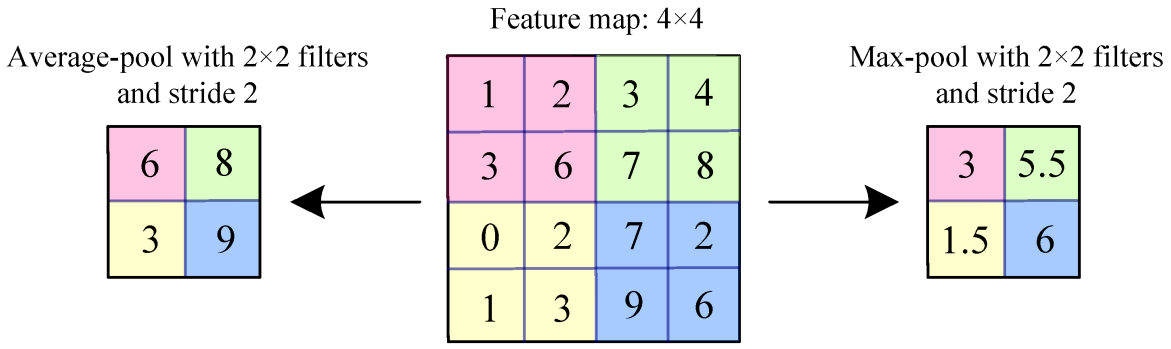


Figure 3.7: Pooling procedure (Figure from [43])

3.2.1 Transfer Learning

A very common problem in the field of Deep Learning is the lack of labeled data, that can be used for training deep learning models. Specifically, in the domain of medical imaging, labeled data collection encounters many obstacles, such as the high cost and the increased required effort of radiology experts. So, it was necessary to make deep Convolutional Neural Network models functional even in small datasets. *Transfer Learning* is one of the techniques used for this purpose. It is a process in which a Convolutional Neural Network is trained on small datasets by exploiting a network that was previously trained (*pretrained models*) on a large dataset. For example, many models used the ImageNet challenge dataset to train their filters and weights. The most used pretrained models are AlexNet [44], VGG [45], ResNet [46], Inception [47], and DenseNet [48].

As shown in Fig. 3.8, there are two ways to adapt a pretrained model in a Convolution Neural Network. The first method is called fixed feature extraction. According to the procedure that it follows, all the fully connected layers, from the Convolutional Neural Network pretrained model, are removed and only the convolutional base (convolution and pooling layers), along with the filters and weights, are remaining. Then, we add new fully connected layers to the convolutional base and we update only the weights of these layers in the training. In the second method, which is called fine-tuning, we again replace the fully connected layers with new ones, but also we fine-tune all or part of the filters of the convolusional base. More often, we select the deeper layers to fine-tune, because these are responsible for extracting more complex features related to the given data.

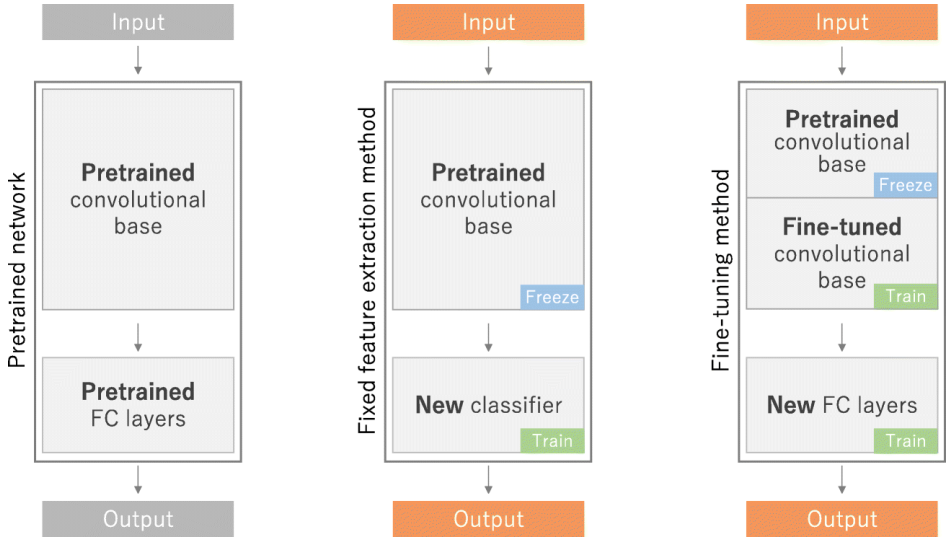


Figure 3.8: Fixed feature extraction and fine-tuning techniques for the use of Transfer Learning (Figure from [49])

Chapter 4

Analysis of LIDC-IDRI database

The *Lung Image Database Consortium-Image Database Resource Initiative* [1] database, also known as LIDC-IDRI, is the world's largest and most widely used pulmonary lung nodule database. The National Cancer Institute [50] (NIH) established the first component of LIDC-IDRI under the name LIDC. Then, the IDRI database was generated with the help of the Foundation for the National Institutes of Health [51] (FNIH) and eight medical imaging companies. Finally, the CT scans from the two databases above are all combined, and we led in the advanced LIDC-IDRI database with the extra help of the Food and Drug Administration [52] (FDA).

LIDC-IDRI dataset contains in total 1018 CT scans coming from 1010 different. Nodules is divided in three categories depending on their size; nodules with a diameter bigger than 3 mm (total 2648), nodules with a diameter of less than 3 mm (total 4702) and non-nodules with a diameter bigger than 3 mm [53] [54](Fig. 4.1). According to the privacy rules of the Health Insurance Portability and Accountability Act [55] (HIPAA) , anonymization software had to be used to erase all protected health information (PHI) contained inside the DICOM headers of the images[1]. For the creation of all these images were used many various types of scanners from manufacturers, such as GE Medical Systems LightSpeed, Phillips and Siemens.

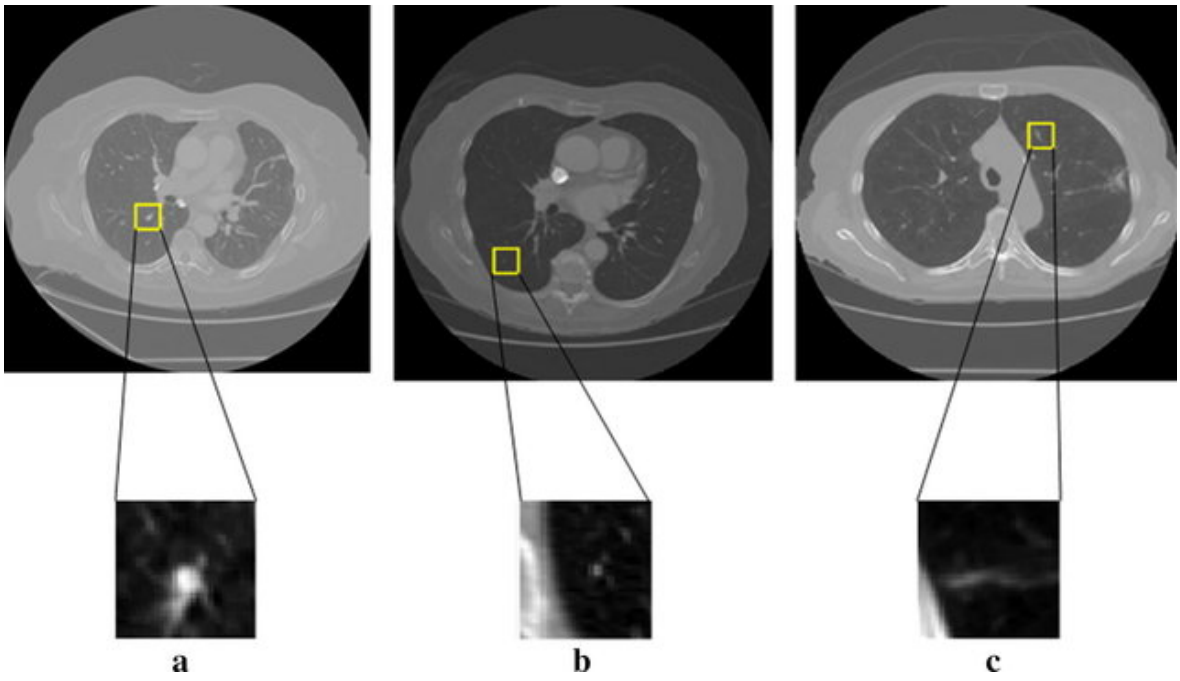


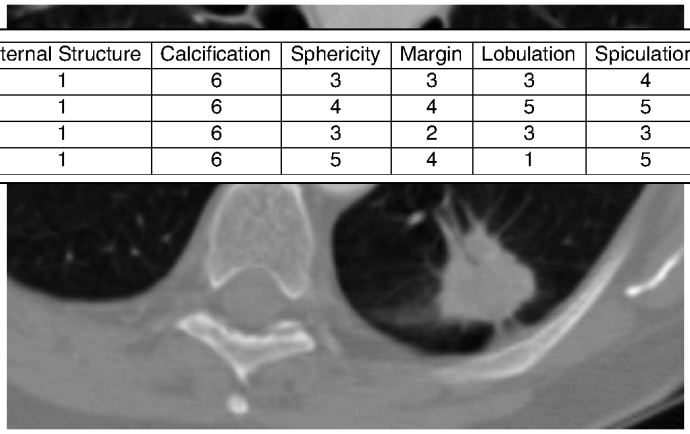
Figure 4.1: Division of nodules depending on their size. (a) Nodule with a diameter bigger than 3 mm, (b) Nodule with a diameter of less than 3 mm and (c) Non-nodule with a diameter bigger than 3 mm (Figure from [53])

Each of the 1018 scans can be either a standard-dose diagnostic CT scan or a lower-dose CT scan and has numerous 2D slices as well as an XML file [56]. This file contains information about the nodule's location, contour, and some other features. Following a four-radiologist two-phase image annotation approach, the following features emerged:

- *subtlety*, with values in range = (1,2,3,4,5). It describes how difficult is the detection of a nodule, where '5' denotes the easier detection.
- *internal structure*, with values in range = (1,2,3,4) or respectively ('Soft Tissue','Fluid','Fat','Air'), which denotes the internal composition of the nodule.
- *calcification*, with values in range = (1,2,3,4,6) or respectively ('Popcorn', 'Laminated', 'Solid', 'Non-central', 'Central', 'Absent'). It describes the pattern of calcification, if present.
- *sphericity*, with values in range = (1,2,3,4,5) or respectively ('Linear', 'Ovoid/Linear', 'Ovoid', 'Ovoid/Round', 'Round'), which describes the three-dimensional shape of the nodule in terms of its roundness.

- *margin*, with values in range = (1,2,3,4,5) or respectively ('Poorly Defined', 'Near Poorly Defined', 'Medium Margin', 'Near Sharp', 'Sharp'). It denotes how well-defined the nodule margin is
- *lobulation*, with values in range = (1,2,3,4,5), which is the degree of lobulation ranging from none to marked.
- *spiculation*, with values in range = (1,2,3,4,5), which refers to the extent of spiculation present, ranging from none to marked
- *texture*, with values in range = (1,2,3,4,5) or respectively ('Non-Solid/GGO', 'Non-Solid/Mixed', 'Part Solid/Mixed', 'Solid/Mixed', 'Solid')
- *malignancy*, with values in range = (1,2,3,4,5) or respectively ('Highly Unlikely', 'Moderately Unlikely', 'Indeterminate', 'Moderately Suspicious', 'Highly Suspicious'). It describes the assessement of the likelihood of malignancy.

In Fig. 4.2 is performed a lung nodule image and its annotation for all the features presented above.



Physician	Subtlety	Internal Structure	Calcification	Sphericity	Margin	Lobulation	Spiculation	Texture	Malignancy
1	5	1	6	3	3	3	4	5	5
2	5	1	6	4	4	5	5	5	5
3	5	1	6	3	2	3	3	5	5
4	5	1	6	5	4	1	5	4	4

Figure 4.2: Example nodule from the LIDC-IDRI dataset with diagnostic feature values from four radiologists. (Figure from [57])

Chapter 5

Our architectures for lung cancer nodules classification

5.1 Data preprocessing

For the purposes of our research, we used a part of LIDC-IDRI database. More specifically, we processed only bio-medical images with nodules greater than 3mm. The total number of them was 2648 different nodules in a DICOM format.

From the DICOM file, we isolated every 3D scan along with information related to it. This information concerns the intercept and slope, which are necessary for converting pixels from Hounsfield units to intensity. Furthermore, we kept information about the pixel-based Region of Interest (RoI), which is given for each nodule separately and is also a 3D image with the same dimensions as the nodule. The final, and more important, clue was the annotation, of at least one to at most four radiologists, for the malignancy level of each nodule.

5.1.1 First phase of preprocessing

In the first phase of preprocessing, we used the RoI mask of each nodule to help us separate the slices with the most information on them. In our case, the given RoI mask was a binary mask, which means that the value of each pixel is ’0’ if it belongs to the background and ’1’ if it belongs to the nodules’ area. The dimensions are the same as the corresponding nodule image [58].

So the first step was to split each 3D nodule image and its corresponding mask, in the 2D slices that included. Coming up next, we set a threshold in the number of pixels that

belong to the nodules' area. Following the histogram in Fig. 5.1 and the mean value of the number of aces in RoI masks, we decided to set the threshold in the value of 10. It means, practically, that every slice of a nodule image that has less than 10 aces (or less than 10 pixels that correspond to nodule area) in the mask, was deleted. In this way, only the rest slices with important information for each nodule formed our database. In total, we have 23754 2D slices.

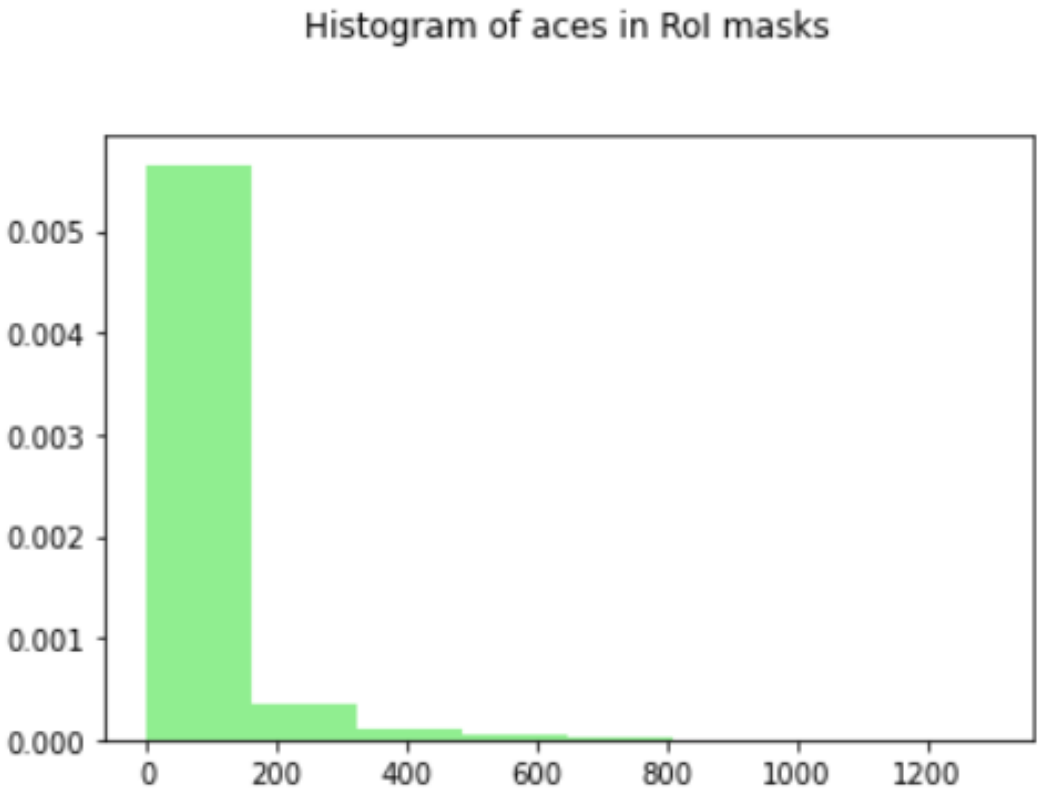


Figure 5.1: Histogram of the number of aces in RoI masks. Mean value is 15 aces.

5.1.2 Second phase of preprocessing

In the second phase, we normalized the raw CT 2D slices from above, in the range $(0, 1)$. Then, we constructed 3 databases by processing the previous 2D slices. The three cases are presented below:

1. **Raw CT 2D slices**, which contains the normalized raw CT 2D slices.

2. **Lung window 2D slices**, which contains only the part of the 2D slice that illustrates the lung area. It is done by converting the pixels from Hounsfield units to intensity

with the formula :

$$pixelValue = slope \cdot 2DImageArray + intercept \tag{5.1}$$

where slope and intercept is given from metadata of the DICOM file. Then, we scale pixel intensity data using specified window level, width, and intensity range $(0, 1)$.

3. **Concatenation of the two previous datasets**, which contains 3D images. Here, we stack the 2D raw CT slice with the corresponding 2D lung window slice and a 2D array with pixel values equal to zero and dimensions the same as the other slices.

Since the deep Convolutional Neural Networks, that we will use later, need as input 3D images with same dimensions, we applied zero-padding [59] [60] in all 2D slices of the first two dataset cases and in all 2D slices of each 3D image in the third case. So, we resize all 2D slices in dimension $(98, 98)$, because, as shown in Fig. 5.2, the bigger dimension was 98 in x axis. After zero-padding, we converted each 2D slice of the first two cases in 3D image by stacking it 3 times.

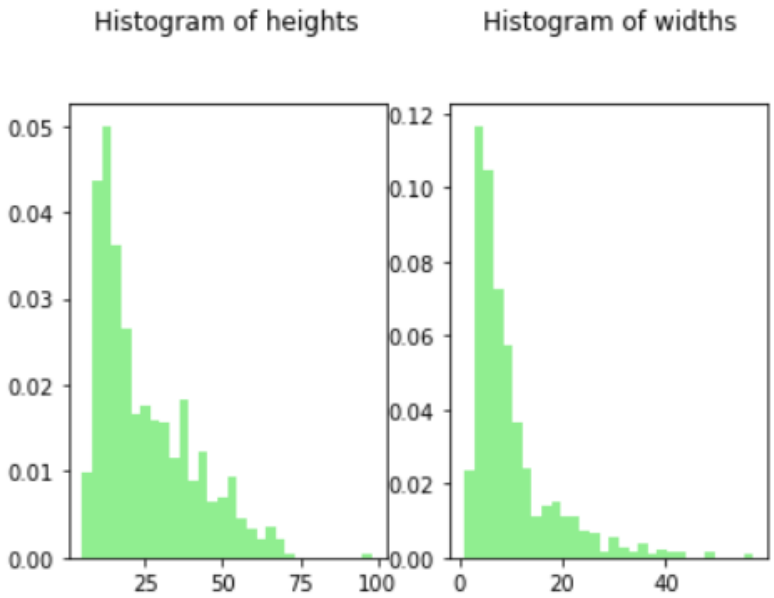


Figure 5.2: Histogram of values of width and height of all images. Max value is 98.

5.1.3 Third phase of preprocessing

As mentioned in Chapter 4 the original labels contain 5 different classes that indicate the malignancy level of each nodule :

- class ’0’: ‘Highly Unlikely’
- class ’1’: ‘Moderately Unlikely’
- class ’2’: ‘Indeterminate’
- class ’3’: ‘Moderately Suspicious’
- class ’4’: ‘Highly Suspicious’

Scince there are annotations of at least one to at most four radiologists for each nodule, we had to make the final label one-dimensional. This is done by taking the mode of the classes appeared in the annotation of a nodule. If there was more than one mode we kept the greater value as we want to avoid assuming a nodule benign although it was malignant. Then, we create three subcases for each case of Subsection 5.1.2:

- 5-class classification dataset with all images and labels as they emerged from the previous preprocess. The number of instances per class is shown in Fig. 5.3.

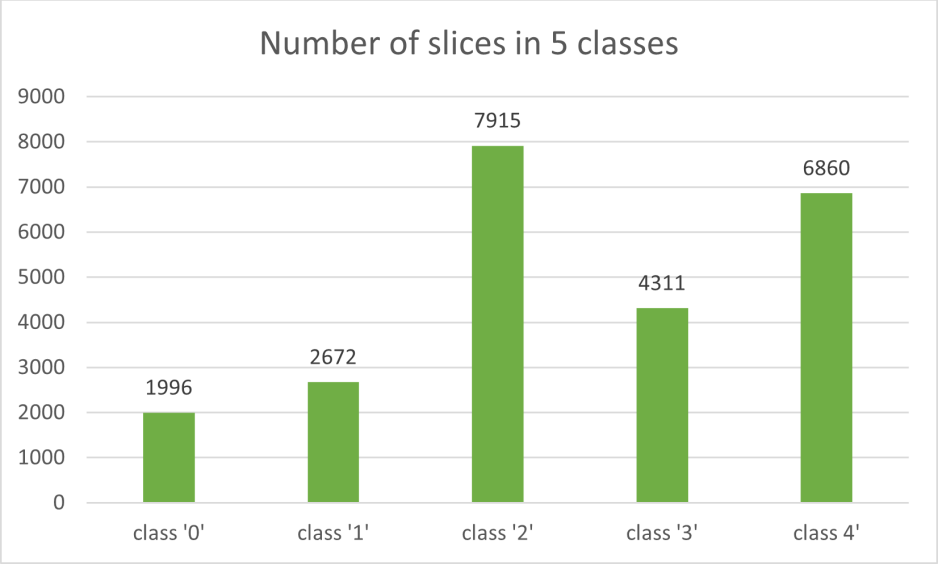


Figure 5.3: Barplot of number of instances per class in 5-class classification dataset.

- 4-class classification dataset with all images that do not belong to 'Intermediate' class and with labels:
  - class '0': 'Highly Unlikely'
  - class '1': 'Moderately Unlikely'
  - class '2': 'Moderately Suspicious'
  - class '3': 'Highly Suspicious'

  The number of instances per class is shown in Fig. 5.4.

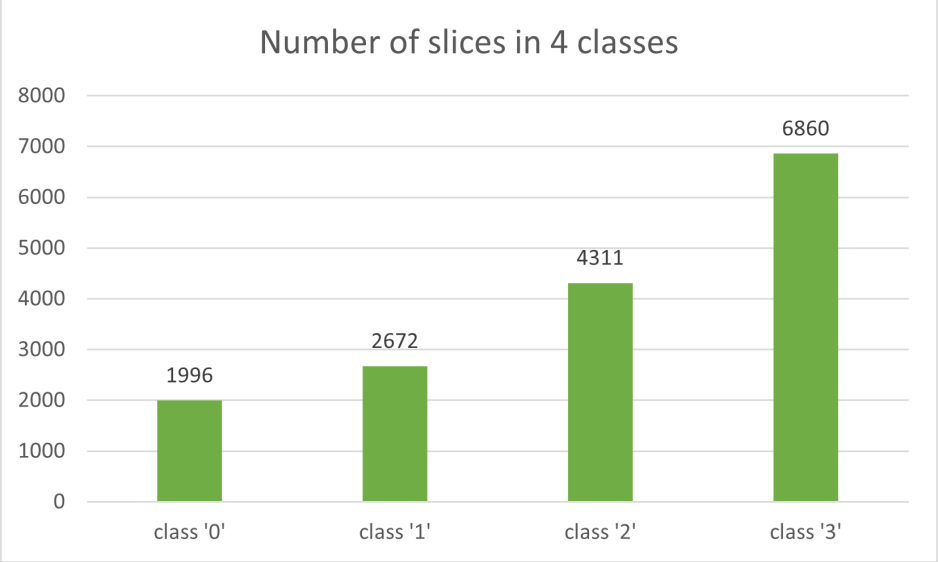


Figure 5.4: Barplot of number of instances per class in 4-class classification dataset.

- 3-class classification dataset with all images from the previous preprocess, but to prepare the labels we merged the previous class '0' with class '1' and class '3' with class '4' and kept class '2' as it was. So, now we have:
  - class '0': 'Highly Unlikely and Moderately Unlikely'
  - class '1': 'Indeterminate'
  - class '2': 'Moderately Suspicious and Highly Suspicious'

  The number of instances per class is shown in Fig. 5.5.

Figure 5.5: Barplot of number of instances per class in 3-class classification dataset.

- 2-class classification dataset with all images that do not belong to 'Intermediate' class and with labels:
  - class '0': 'Highly Unlikely and Moderately Unlikely'
  - class '1': 'Moderately Suspicious and Highly Suspicious'

  The number of instances per class is shown in Fig. 5.6.

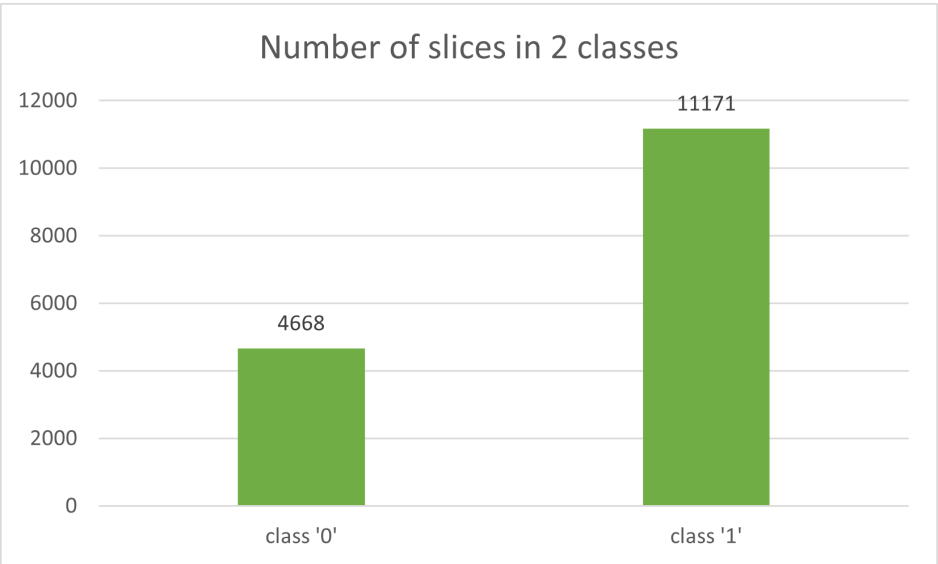


Figure 5.6: Barplot of number of instances per class in 2-class classification dataset.

5.2 Preparation of deep model's inputs

Before we construct deep models for the three classification problems (binary, trinary and 5-way) described previously, it was necessary to ensure their functionality and robustness. So, in Subsection 5.2.1 we talk about Stratified K-fold Cross Validation [61] and in Subsection 5.2.2 about Data Generator technique, which are responsible for the way that the data will be loaded in the models.

5.2.1 Split datasets into train, validation and test sets

First of all, we used Stratified K-fold Cross Validation method to split the dataset into train and test sets. Through this method, the dataset is separated in K folds, where $(K-1)$ folds is used for training and 1 fold for testing. In our case, the seperation is applied on the nodules and not on the 3D slices that our datasets contain,and K is equal to 5. Stratified K-Fold algorithm ensures that every fold will contain same percentage of every class. This is very important, because the data in our problem is imbalanced and there is the danger that all instances of a class will be only in the training or in the testing set and the classifier will fail to make correct predictions.

The function of Stratified K-fold Cross Validation method is described in Fig. 5.7. We make K iterations and each time one different fold will be the testing set. The procedure reaches to end when each fold has been used as the testing set. The final performance of the classifier is the average of the performances of all iterations. It is worth mentioning, that 10 percent of training set is used as a validation set in the fitting process of the classifier. It contains same amount of instances of all classes just like train and test set do.

5.2.2 Data Generator

In our project, we had to handle images that consume a lot of memory space. In this circumstance, the creation of a custom *Data Generator* [63] is highly recommended. It is a class of the Python's high-level package, Keras [64], that can be modified based on our needs. The use of a Data Generator allows us to transmit, in your architecture, in batches the data that you want. The following is a description of how our custom Data Generator works:

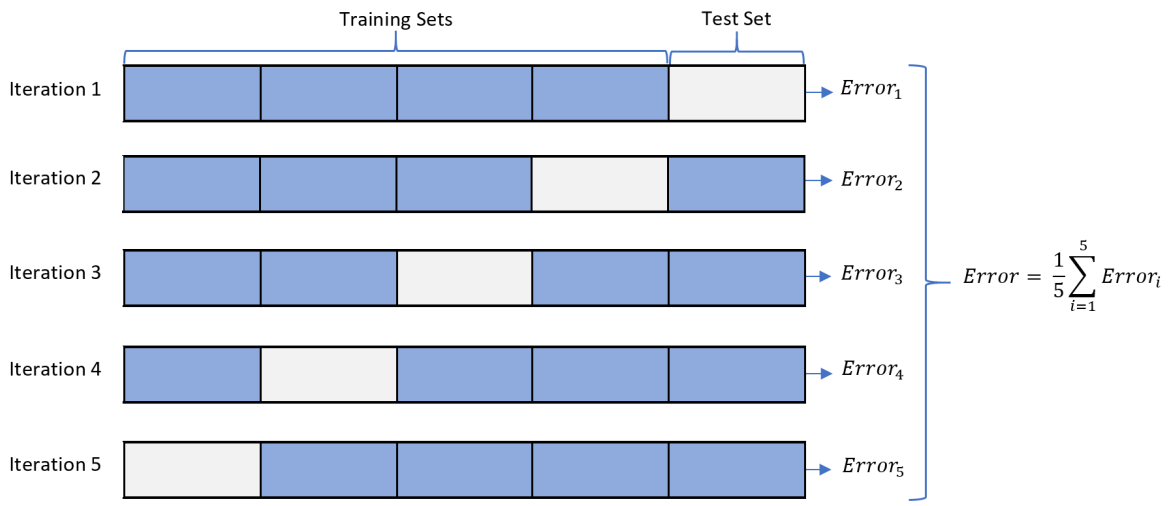


Figure 5.7: K-fold representation, where K=5. (Figure from [62])

1. takes a list that contains IDs of nodules
2. loads a given number of IDs (batch size)
3. loads all preprocessed 3D slices $(3 \times 98 \times 98)$ that belong to those nodules and concatenate them in a 4D array with shape $(4 \times 3 \times 98 \times 98)$
4. pass this 4D array through our architecture either for training or validation or evaluation of the model
5. repeat steps 2, 3 and 4 as many times as needed to load all instances.

5.3 Convolutional Neural Network Classifiers

As our datasets contain only a few images, it is not possible to successfully extract features with a custom Convolutional Neural Network. This led us to adopt the transfer learning method, keeping the filters and weights from pretrained models. In our project, we finally chose the Inception-V3 model [65], because, after trials with many pretrained models, it was the one with the best performance. It is trained in over than one million images from ImageNet database [66], that belong to one thousand different classes and thus it is more stable and accurate than a new custom made CNN. In Fig. 5.8 is presented the architecture of Inception-V3 model, as described in the original paper. In the Subsections 5.3.1, 5.3.2 and 5.3.4, we convert the last three layers to make the model more suitable for our experiment.

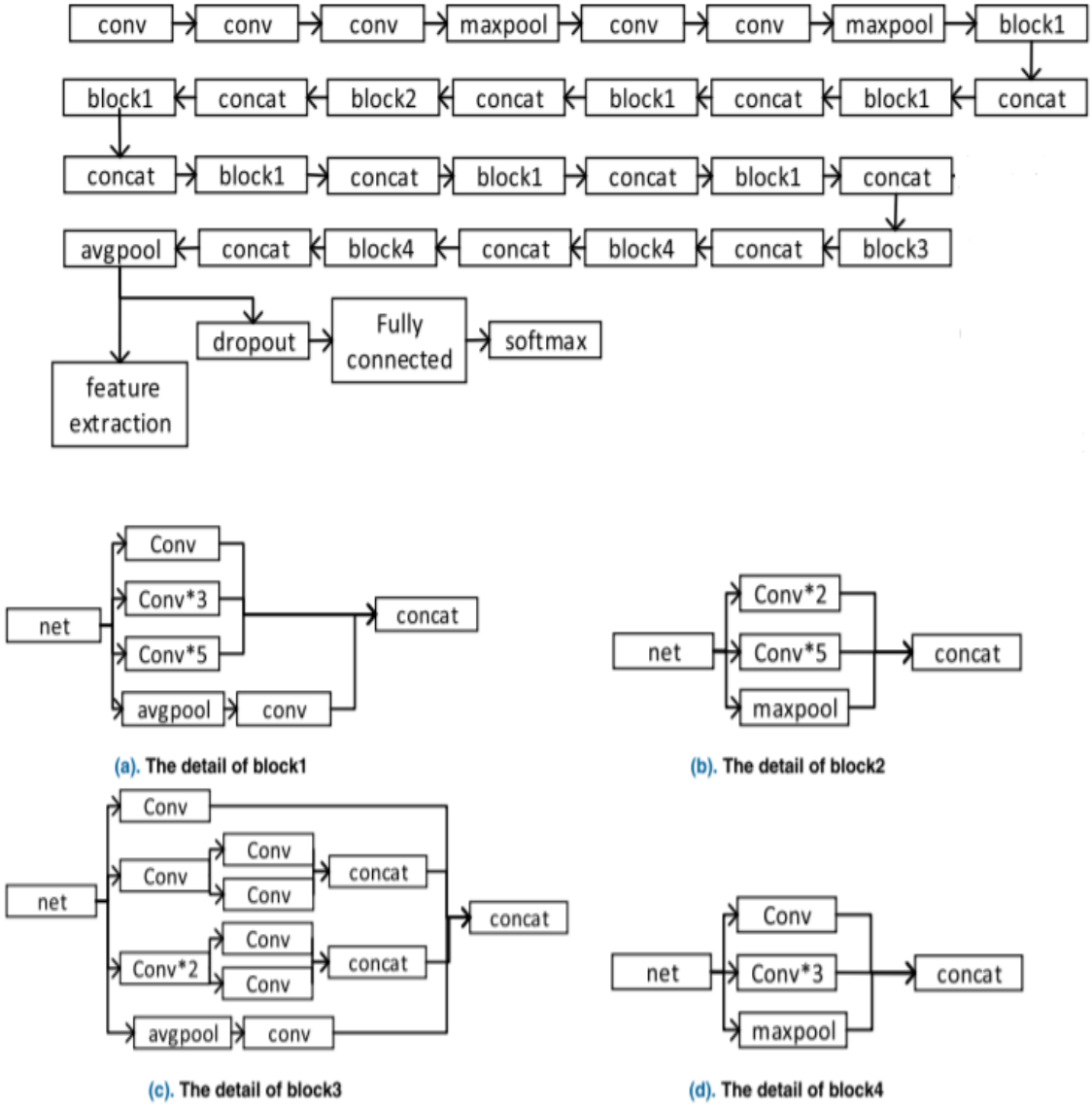


Figure 5.8: Diagram of the Inception-V3 model. (Figure from [67])

5.3.1 Binary case

For each of the three cases of dataset (raw CT, lung window and concatenated) we applied the architecture in Table 5.1. We followed the fine-tuning method only in the Neural Network layers and we ended up with Adam [68] as optimizer and the value 0.001 as learning rate.

Input tensor shape $(3 \times 98 \times 98)$
Inception-V3 convolution layers with no pooling in last layer
Flatten
150 neurons, Dense, ReLU Dropout 0.4
50 neurons, Dense, ReLU Dropout 0.3
1 neuron, Dense, sigmoid

Table 5.1: Binary model architecture for raw CT, lung window and concatenated dataset.

5.3.2 Trinary case

For the raw CT dataset we applied the architecture in Table 5.2. After the fine-tuning procedure, we selected Adam as optimizer and the value 0.001 as learning rate. This time, for lung window and concatenated dataset we differentiate the architecture as shown in Table 5.3. After the fine-tuning procedure, we selected Adam as optimizer and the value 0.005 as learning rate.

Input tensor shape $(3 \times 98 \times 98)$
Inception-V3 with no pooling in last layer
Flatten
80 neurons, Dense, ReLU Dropout 0.4
3 neuron, Dense, softmax

Table 5.2: Trinary model architecture for raw CT dataset.

Input tensor shape $(3 \times 98 \times 98)$
Inception-V3 with no pooling in last layer
Flatten
150 neurons, Dense, ReLU Dropout 0.4
3 neuron, Dense, softmax

Table 5.3: Trinary model architecture for lung window and concatenated dataset.

5.3.3 4-class case

For the raw CT dataset we applied the architecture in Table 5.4. After the fine-tuning procedure, we selected Adam as optimizer and the value 0.005 as learning rate. This time, for lung window we differentiate the architecture as shown in Table 5.5 and we selected Adam as optimizer and the value 0.001 as learning rate. For concatenated dataset we chose the architecture in Table 5.6. After the fine-tuning procedure, we selected Adam as optimizer

and the value 0.005 as learning rate.

Input tensor shape $(3 \times 98 \times 98)$
Inception-V3 with no pooling in last layer
Flatten
150 neurons, Dense, ReLU Dropout 0.4
50 neurons, Dense, ReLU Dropout 0.3
4 neuron, Dense, softmax

Table 5.4: 4-class model architecture for raw CT dataset.

Input tensor shape $(3 \times 98 \times 98)$
Inception-V3 with no pooling in last layer
Flatten
80 neurons, Dense, ReLU Dropout 0.4
40 neurons, Dense, ReLU Dropout 0.3
4 neuron, Dense, softmax

Table 5.5: 4-class model architecture for lung window dataset.

Input tensor shape $(3 \times 98 \times 98)$
Inception-V3 with no pooling in last layer
Flatten
150 neurons, Dense, ReLU Dropout 0.4
4 neuron, Dense, softmax

Table 5.6: 4-class model architecture for concatenated dataset.

5.3.4 5-class case

For each of the three cases of dataset (raw CT, lung window and concatenated) we applied the architecture in Table 5.7. We followed the fine-tuning method only in the Neural Network layers and we ended up with Adam [68] as optimizer and the value 0.0005 as learning rate.

Input tensor shape $(3 \times 98 \times 98)$
Inception-V3 convolution layers with no pooling in last layer
Flatten
150 neurons, Dense, ReLU Dropout 0.4
75 neurons, Dense, ReLU Dropout 0.4
5 neuron, Dense, softmax

Table 5.7: 5-class model architecture for raw CT, lung window and concatenated dataset.

Chapter 6

Results

6.1 Evaluation Metrics

After the training of a deep learning model, comes the testing process. On this stage, we have to evaluate our model in new unseen data, which tough are related to the training data. This is done by using different *evaluation metrics* [69] that are suitable for each type of problems. For the purpose of this classification problem, we first need to define the *Confusion Matrix* [70] (Fig. 6.1). In this, four variables are calculated:

- *True Positives* (TP) number describes the number of instances that was actually belong to positive class and was correctly predicted.

- *False Negatives* (FN) number describes the number of instances that was actually belong to positive class and was wrongly predicted.

- *False Positives* (FP) number describes the number of instances that was actually belong to negative class and was wrongly predicted.

- *True Negatives* (TN) number describes the number of instances that was actually belong to negative class and was correctly predicted.

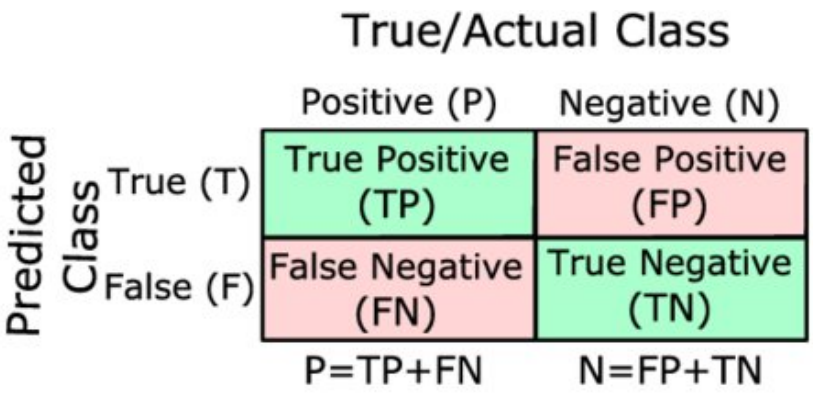


Figure 6.1: Confusion Matrix overview for binary classification (Figure from [69]).

Confusion matrix can, also, be created for multi-class classification with dimensions $N \times N$, where N is the number of classes. In this case, it is like we construct a confusion matrix of each class against all the other remaining classes. For example, for class A in Fig. 6.2 the following apply:

$$FN = E_{AB} + E_{AC} \tag{6.1}$$

$$FP = E_{BA} + E_{CA} \tag{6.2}$$

$$TN = TP_B + E_{CB} + E_{BC} + TP_C \tag{6.3}$$

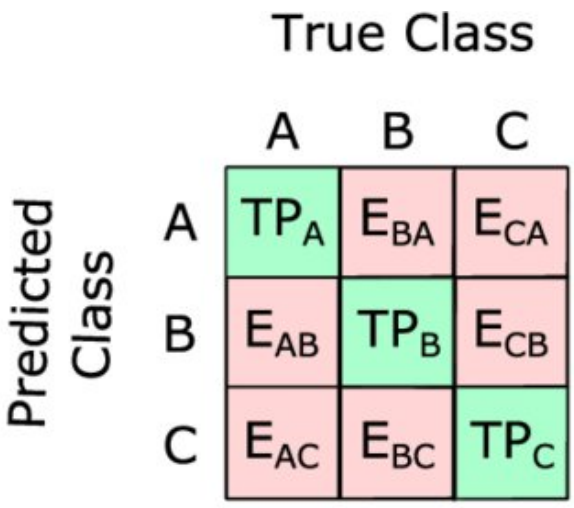


Figure 6.2: Confusion Matrix overview for multi-class classification (Figure from [69]).

So, for our problem, we utilized the following metrics based on all of the above information:

1. *Accuracy*, which expresses the proportion of correct predictions to the total number of instances considered. This metric is the most commonly used but it is not very good in the case of unbalanced data. It is calculated by the formula:

$$Accuracy = \frac{TP_{class\_name} + TN_{class\_name}}{TP_{class\_name} + TN_{class\_name} + FP_{class\_name} + FN_{class\_name}} \tag{6.4}$$

2. *Precision*, which expresses the proportion of the correct predicted instances of a class to the total number of instances that predicted to belong in this class. It is calculated by the formula:

$$Precision = \frac{TP_{class\_name}}{TP_{class\_name} + FP_{class\_name}} \tag{6.5}$$

3. *Recall*, which expresses the proportion of the correct predicted instances of a class to the total number of the instances that actually belong to this class. It is calculated by the formula:

$$Recall = \frac{TP_{class\_name}}{TP_{class\_name} + FN_{class\_name}} \tag{6.6}$$

4. *F1-score*, is a good choice when the data is imbalanced and combines the recall and precision. It is calculated by the formula:

$$F_1 - score = \frac{2 * Precision_{class\_name} * Recall_{class\_name}}{Precision_{class\_name} + Recall_{class\_name}} \tag{6.7}$$

5. *Area Under the Curve* (AUC) (Fig. 6.3) considered to be the best evaluation metric. It can be applied either in binary or in multi-class classification, but in the second case calculation of AUC follows the one-vs-all technique. Curve is reffered to *Receiver Operating Characteristic)* (ROC) curve and is defined as the False Positive Rate (FPR) versus the True Positive Rate (TPR) for a variaty of candidate threshold values between 0.0 and 1.0.

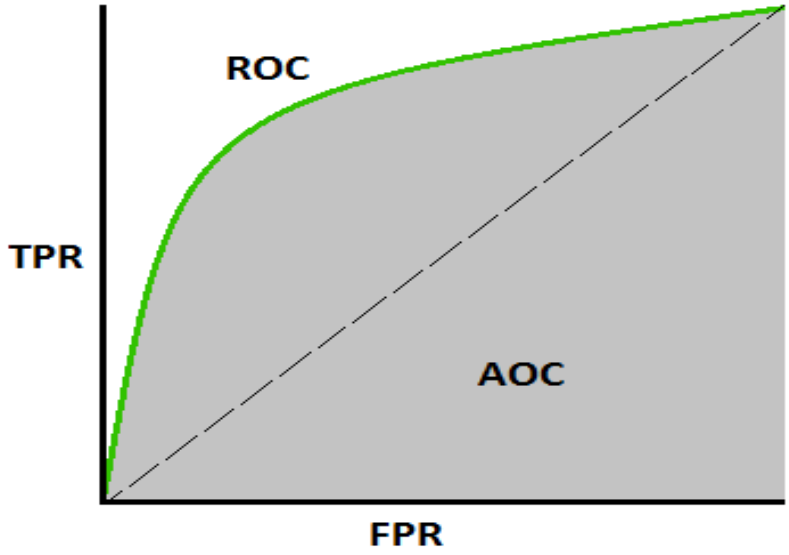


Figure 6.3: ROC-AUC curve. (Figure from [71]).

6.2 Results

6.2.1 Overview

In this section, we present the results obtained by the transfer learning deep models proposed in section 5.3. For each of the binary, trinary, 4-class and 5-class cases, we compare the results of training and testing them in Raw CT, Lung Window and Concatenated dataset. From the results in Tables 6.1, 6.2, 6.3, 6.4, we can easily understand that every classifier was negatively affected by the fact that our data was highly imbalanced.

6.2.2 Binary case

In the 2-class (binary) experiment the performance is satisfactory for each of the three dataset cases (Table 6.1). However, we see that the proposed method has the best outcome when it uses the Lung Window dataset for training and testing. Furthermore, We can observe that it is more accurate when a nodule is malignant (belongs to classs ’1’), than when a nodule is benign (belongs to classs ’0’). This is, mainly, because the classifier is trained in highly imbalanced data.

	Raw CT	Lung Window	Concatenated
Accuracy	0.80	**0.81**	0.76
AUC	0.79	**0.80**	0.78
Precision(class $'0'$)	0.63	**0.67**	0.57
Precision(class $'1'$)	0.89	0.89	**0.92**
Recall(class $'0'$)	0.76	0.75	**0.83**
Recall(class $'1'$)	0.81	**0.84**	0.74
$F_1 score$(class $'0'$)	0.69	**0.70**	0.67
$F_1 score$(class $'1'$)	0.85	**0.86**	0.82

Table 6.1: Results for binary case.

6.2.3 Trinary case

In the 3-class (trinary) experiment the performance is less satisfactory for each of the three dataset cases (Table 6.2). However, we see that the proposed method has the best outcome when it uses either the Raw CT or the Concatenated dataset for training and testing. We can observe that it is very accurate when a nodule is malignant (belongs to classs '2'), but it is more difficult to recognize the remaining classes '0' and '1'.

6.2.4 4-class case

In the 4-class experiment the performance is moderate for each of the three dataset cases (Table 6.2). However, we see that the proposed method has the best outcome when it uses the Raw CT dataset and the Concatenated dataset for training and testing. We can observe that it is very accurate when a nodule is malignant (belongs to classs '3') or is moderately unlikely to be malignant (belongs to class '1') but it is more difficult to recognize the remaining classes '0' and '2'. This experiment took place because we wanted to avoid the 'Intermediate' class as it has had much more instances from the other classes and further confuses the classifier.

	Raw CT	Lung Window	Concatenated
Accuracy	**0.61**	0.55	0.59
AUC(class $'0'$)	**0.60**	**0.60**	0.57
AUC(class $'1'$)	**0.66**	0.64	**0.66**
AUC(class $'2'$)	**0.78**	0.72	0.75
Precision(class $'0'$)	**0.52**	0.47	0.50
Precision(class $'1'$)	**0.52**	0.47	0.50
Precision(class $'2'$)	**0.78**	**0.78**	0.76
Recall(class $'0'$)	0.32	**0.35**	0.27
Recall(class $'1'$)	0.58	**0.62**	**0.62**
Recall(class $'2'$)	**0.76**	0.59	0.70
$F_1 score$(class $'0'$)	0.33	**0.34**	0.29
$F_1 score$(class $'1'$)	**0.54**	0.53	**0.54**
$F_1 score$(class $'2'$)	**0.77**	0.66	0.73

Table 6.2: Results for trinary case.

6.2.5 5-class case

In the 5-class experiment the performance is the least satisfactory for each of the three dataset cases (Table 6.2). However, we see that the proposed method has the best outcome when it uses the Raw CT dataset and the Lung Window dataset for training and testing. We can observe that it is very accurate when a nodule is malignant (belongs to classs '4'), but it is more difficult to recognize the remaining classes '0', '1', '2' and '3'. Specifically, we can see that the performance for the class '2' (Intermediate) is very bad and it led us to construct a model only for the other four classes as shown in subsection 6.2.4, which ultimately shows really good performance.

	Raw CT	Lung Window	Concatenated
Accuracy	**0.57**	0.43	0.53
AUC(class $'0'$)	0.64	**0.70**	**0.70**
AUC(class $'1'$)	0.72	**0.73**	0.72
AUC(class $'2'$)	**0.55**	0.51	0.53
AUC(class $'3'$)	**0.78**	0.75	0.77
Precision(class $'0'$)	**0.43**	0.42	0.36
Precision(class $'1'$)	**0.44**	0.36	0.41
Precision(class $'2'$)	**0.46**	0.30	0.36
Precision(class $'3'$)	0.70	0.72	**0.74**
Recall(class $'0'$)	0.36	0.49	**0.57**
Recall(class $'1'$)	0.60	**0.63**	0.61
Recall(class $'2'$)	**0.20**	0.10	0.15
Recall(class $'3'$)	**0.84**	0.76	0.73
$F_1 score$(class $'0'$)	0.36	**0.44**	0.43
$F_1 score$(class $'1'$)	**0.50**	0.39	0.49
$F_1 score$(class $'2'$)	**0.27**	0.12	0.21
$F_1 score$(class $'3'$)	**0.76**	0.72	0.74

Table 6.3: Results for 4-class case.

	Raw CT	Lung Window	Concatenated
Accuracy	**0.41**	0.36	0.37
AUC(class $'0'$)	0.70	**0.73**	0.70
AUC(class $'1'$)	**0.64**	**0.64**	0.63
AUC(class $'2'$)	**0.53**	0.51	0.51
AUC(class $'3'$)	**0.58**	0.52	0.55
AUC(class $'4'$)	**0.80**	0.78	0.80
Precision(class $'0'$)	0.35	0.25	**0.38**
Precision(class $'1'$)	**0.22**	0.19	0.21
Precision(class $'2'$)	**0.50**	0.26	0.39
Precision(class $'3'$)	**0.30**	0.28	0.29
Precision(class $'4'$)	**0.66**	0.63	0.62
Recall(class $'0'$)	0.49	**0.65**	0.53
Recall(class $'1'$)	0.52	**0.62**	0.52
Recall(class $'2'$)	**0.11**	0.02	0.03
Recall(class $'3'$)	**0.30**	0.07	0.23
Recall(class $'4'$)	0.76	0.72	**0.81**
$F_1 score$(class $'0'$)	0.32	0.35	**0.38**
$F_1 score$(class $'1'$)	0.31	0.30	**0.33**
$F_1 score$(class $'2'$)	**0.16**	0.03	0.04
$F_1 score$(class $'3'$)	**0.30**	0.02	0.24
$F_1 score$(class $'4'$)	**0.71**	0.67	0.70

Table 6.4: Results for 5-class case.

Chapter 7

Discussion

7.1 Classification

The classification procedure has been researched for many years through statistics and Machine and Deep Learning for solving many common problems and very good results have been achieved. Nevertheless, deep learning models have not been satisfactorily expanded and researched upon medical images. So far, the review procedure of a CT scan or other images coming from imaging techniques in clinical medicine (for example X-ray, MRI, ultrasound) is time-consuming and has many stages. It requires the participation of radiologists and physicians who are experts in reading and interpreting CT scans. Most of the time, there is a time delay, because of the difficult communication between them or because one or more of them do not work at this specific time.

So, the main goal of this thesis was to apply deep learning techniques based on Convolutional Neural Networks to solve the problem of categorization of CT scans with pulmonary nodules, and to create systems that doctors can consult, along with their knowledge. Since many pretrained models based on CNN have been successfully trained on large databases, for example ImageNet dataset, which contains more than one million images, we can use them to improve the performance of models trained on medical images. One significant feature of deep learning models is that, after the training procedure, the model is ready to accept new images for categorization and make a prediction in a few seconds, unlike doctors. Furthermore, these models have been trained on many different lung cancer cases, so they can help doctors in difficult and unknown cases.

The existing works try to classify the lung cancer nodules into 2 classes (benign or malignant), ignoring the separation of benign nodules in 'Highly Unlikely' and 'Moderately Unlikely' malignancy and the separation of malignant nodules in 'Highly Suspicious' and 'Moderately Suspicious' malignancy. Also, they ignore the existence of nodules that were annotated as indeterminate.

In this thesis, we propose architectures based on Convolutional Neural Networks. They try to classify the lung cancer nodules either on 2 classes (benign or malignant), either on 3 classes (benign, Indeterminate, malignant), either on 4 classes ('Highly Unlikely', 'Moderately Unlikely', 'Highly Suspicious', 'Moderately Suspicious'), or on 5 classes ('Highly Unlikely', 'Indeterminate', 'Moderately Unlikely', 'Highly Suspicious', 'Moderately Suspicious'). The last two tasks are suitable to help doctors in new and different from the already existing cases. Furthermore, we used not only the raw CT images but also the lung window of those images. So, we gave input in our models those two data types separately and in combination ending up with satisfactory results that can be helpful in the medical field.

7.2 Limitations

Through our research, we understood that the application of deep learning in the processing of medical images is of utmost importance for the immediate diagnosis of lung cancer, and consequently, the improvement of life expectancy. However, we faced some limitations related to the fact that deep learning techniques require plenty of data and the same sample availability per class, for the training procedure. Unfortunately, due to privacy rules, most of the medical data can not be used for researches. Although the LIDC-IDRI database, that we used for our experiments, is one of the largest datasets with medical images, it only contains 1010 CT scans with highly imbalanced data. The above forced us to use the transfer learning method (pretrained models), which is suitable for images with 2D volume, whereas all CT scans have 3D volume images. So, we had to work with 2D volume images and radiologists had to annotate each 2D volume image and not the initial 3D volume image. This is such a big deal because it can lead to information loss that may lead to false predictions for life issues.

Chapter 8

Conclusion and future work

In this thesis, we dealt with an issue related to biomedical engineering. Our motivation was to help the medical community in the correct and timely diagnosis of lung cancer cases, as it is one of the most deadly forms of cancer. More specifically, we tried to assess the malignancy level of lung cancer nodules by using transfer learning deep models. We preprocessed the Lung Image Database Consortium-Image Database Resource Initiative (LIDC-IDRI) database in three different ways. Then, we used all these cases to train and test many different Convolutional Neural Network architectures, with the pretrained model, Inception-V3.

In the future, we want to improve this research by exploring 3D transfer learning, which will give us the opportunity to work with the initial 3D volume images from CT scans. Furthermore, we can use augmentation techniques to avoid the imbalance of the data. All the above will probably, let us create more robust models for the classification of lung cancer nodules.

Bibliography

[1] Samuel G. Armato III, Geoffrey McLennan, Luc Bidaut, Michael F. McNitt-Gray, Charles R. Meyer, Anthony P. Reeves, Binsheng Zhao, Denise R. Aberle, Claudia I. Henschke, Eric A. Hoffman, Ella A. Kazerooni, Sangeeta MacMahon, Maha Sallam, Michael D. Heath, Michael H. Kuhn, Ekta Dharaiya, Richard Burns, David S. Fryd, Marcos Salganicoff, Vikram Anand, Uri Shreter, Stephen Vastagh, Barbara Y. Croft, and Laurence P. Clarke. 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(2):915–931, 2011.

[2] `https://ecis.jrc.ec.europa.eu`. Accessed 5 Sep 2021.

[3] The ever-increasing importance of cancer as a leading cause of premature death worldwide. 2021.

[4] Global surveillance of trends in cancer survival 2000-14 (concord-3): analysis of individual records for 37 513 025 patients diagnosed with one of 18 cancers from 322 population-based registries in 71 countries. 2018.

[5] Lung cancer incidence in young women vs. young men: A systematic analysis in 40 countries. *International journal of cancer*, 2020. Accessed 5 Sep 2021.

[6] Global cancer statistics 2020: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: a cancer journal for clinicians*, 2021. Accessed 5 Sep 2021.

[7] H. Kasban, M. El-Bendary, and D. Salama. A comparative study of medical imaging techniques. 2015.

[8] Xiaofeng Zhang, Nadine Smith, and Andrew Webb. 1 - medical imaging. In David Dagan Feng, editor, *Biomedical Information Technology*, Biomedical Engineering, pages 3–27. Academic Press, Burlington, 2008. Accessed 5 Sep 2021.

[9] P. Suetens. *Fundamentals of Medical Imaging*. Cambridge medicine. Cambridge University Press, 2017.

[10] Godfrey N. Hounsfield. Computed medical imaging. *Science*, 210(4465):22–28, 1980.

[11] Imaging data sets (artificial intelligence). Accessed 5 Sep 2021.

[12] Norvig Peter Russell Stuart. Artificial intelligence: a modern approach. 2002.

[13] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis. Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience*, 2018, 2018.

[14] Convolutional neural networks: an overview and application in radiology. *Insights into imaging*, 9(4):611–629, 2018.

[15] Baris Kayalibay, Grady Jensen, and Patrick van der Smagt. Cnn-based segmentation of medical imaging data. *arXiv preprint arXiv:1701.03056*, 2017.

[16] Jose Dolz, Karthik Gopinath, Jing Yuan, Herve Lombaert, Christian Desrosiers, and Ismail Ben Ayed. Hyperdense-net: a hyper-densely connected cnn for multi-modal image segmentation. *IEEE transactions on medical imaging*, 38(5):1116–1126, 2018.

[17] Lung nodule detection in ct using 3d convolutional neural networks. In *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, pages 379–383. IEEE, 2017.

[18] Babu Kumar S and M Vinoth Kumar. Detection of lung nodules using convolution neural network: A review. In *2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA)*, pages 590–594, 2020.

[19] Dana Li, Bolette Mikela Vilmun, Jonathan Frederik Carlsen, Elisabeth Albrecht-Beste, Carsten Ammitzbøl Lauridsen, Michael Bachmann Nielsen, and Kristoffer Lindskov Hansen. The performance of deep learning algorithms on automatic pulmonary nodule

detection and classification tested on different datasets that are not derived from lidc-idri: A systematic review. *Diagnostics*, 9(4), 2019.

[20] Pranjal Sahu, Dantong Yu, Mallesham Dasari, Fei Hou, and Hong Qin. A lightweight multi-section cnn for lung nodule classification and malignancy estimation. *IEEE journal of biomedical and health informatics*, 23(3):960–968, 2018.

[21] Max Schwarz, Hannes Schulz, and Sven Behnke. Rgb-d object recognition and pose estimation based on pre-trained convolutional neural network features. In *2015 IEEE international conference on robotics and automation (ICRA)*, pages 1329–1335. IEEE, 2015.

[22] Mahbub Hussain, Jordan J Bird, and Diego R Faria. A study on cnn transfer learning for image classification. In *UK Workshop on computational Intelligence*, pages 191–202. Springer, 2018.

[23] Kunio Doi. Computer-aided diagnosis in medical imaging: historical review, current status and future potential. *Computerized medical imaging and graphics*, 31(4-5):198–211, 2007.

[24] Inês Domingues, Gisele Pereira, Pedro Martins, Hugo Duarte, Joao Santos, and Pedro Henriques Abreu. Using deep learning techniques in medical imaging: a systematic review of applications on ct and pet. *Artificial Intelligence Review*, 53(6):4093–4160, 2020.

[25] Mikhail Y Berezin and Samuel Achilefu. Fluorescence lifetime measurements and biological imaging. *Chemical reviews*, 110(5):2641–2684, 2010.

[26] CT and X-ray images. `http://www.aboutcancer.com/lung_xrays_abnormal.htm`. Accessed 21 Aug 2021.

[27] Rebekah Moan. Fast thoracic MRI emerges as alternative to chest x-ray. `https://www.auntminnieeurope.com/index.aspx?sec=ser&sub=def&pag=dis&ItemID=616238`. Accessed 28 Aug 2021.

[28] National Cancer Institute. Computed Tomography (CT) Scans and Cancer [Internet]. `https://www.cancer.gov/about-cancer/diagnosis-staging/ct-scans-fact-sheet.`, 2013. Accessed 17 Aug 2021.

[29] Boone JM Bushberg JT. The essential physics of medical imaging, third edition. 2011.

[30] Geochemical Instrumentation and Analysis. https://serc.carleton.edu/research_education/geochemsheets/techniques/CT.html. Accessed 29 Aug 2021.

[31] DICOM. https://www.dicomstandard.org/. Accessed 3 Sep 2021.

[32] Luís A. Bastião Silva. Medical imaging services supported on cloud. https://www.researchgate.net/publication/260374716_Medical_imaging_services_supported_on_cloud, 2014.

[33] https://builtin.com/artificial-intelligence. Accessed 13 Aug 2021.

[34] https://searchenterpriseai.techtarget.com/definition/deep-learning-deep-neural-network. Accessed 12 Aug 2021.

[35] https://www.ibm.com/cloud/learn/what-is-artificial-intelligence. Accessed 13 Aug 2021.

[36] Hyun Suk Lee and Junga Lee. Applying artificial intelligence in physical education and future perspectives. *Sustainability*, 13(1), 2021. Accessed 19 Aug 2021.

[37] W. S. McCulloch and W. Walter. A logical calculus of the ideas immanent in nervous. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943. Accessed 23 Aug 2021.

[38] Deep learning(ann, dnn, cnn, rnn, slp, mlp). https://dbrang.tistory.com/1537, 2014. Accessed 16 Aug 2021.

[39] Perceptrons – these artificial neurons are the fundamentals of neural networks. https://starship-knowledge.com/tag/what-are-perceptron, 2020. Accessed 17 Aug 2021.

[40] Sepp Hochreiter, Y Bengio, Paolo Frasconi, and Jfirgen Schmidhuber. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies. In J. F. Kolen and S. C. Kremer, editors, *A Field Guide to Dynamical Recurrent Neural Networks*, pages 237–243. IEEE, 2003.

[41] Min Peng, Chongyang Wang, Tong Chen, and Guangyuan Liu. Nirfacenet: A convolutional neural network for near-infrared face identification. *Information*, 7(4), 2016.

[42] Convolutional Neural Networks for Text Classification. `http://www.davidsbatista.net/blog/2018/03/31/SentenceClassificationConvNets/`. Accessed 17 Aug 2021.

[43] Wenfeng Gong, Hui Chen, Zehui Zhang, Meiling Zhang, Ruihan Wang, Cong Guan, and Qin Wang. A novel deep learning method for intelligent fault diagnosis of rotating machinery based on improved cnn-svm and multichannel data fusion. *Sensors*, 19(7), 2019.

[44] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, page 1097–1105, Red Hook, NY, USA, 2012. Curran Associates Inc.

[45] Zisserman A Simonyan K. Very deep convolutional networks for large-scale image recognition. `https://arxiv.org/pdf/1409.1556.pdf`, 2015.

[46] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[47] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, 2015.

[48] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, 2017.

[49] Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, 9, 20198.

[50] National cancer institute. Accessed 5 Sep 2021.

[51] Foundation for the national institutes of health. Accessed 5 Sep 2021.

[52] U.s. food and drug administration. Accessed 5 Sep 2021.

[53] Cnn models discriminating between pulmonary micro-nodules and non-nodules from ct images. *Biomedical engineering online*, 2018.

[54] Samuel G Armato III, Geoffrey McLennan, Luc Bidaut, Michael F McNitt-Gray, Charles R Meyer, Anthony P Reeves, Binsheng Zhao, Denise R Aberle, Claudia I Henschke, Eric A Hoffman, et al. 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(2):915–931, 2011.

[55] Health insurance portability and accountability act. Accessed 4 Sep 2021.

[56] R Jenkin Suji and Sarita Singh Bhadauria. Automatic lung segmentation and lung nodule type identification over lidc-idri dataset.

[57] Matthew C. Hancock and Jerry F. Magnan. Lung nodule malignancy classification using only radiologist-quantified image features as inputs to statistical learning algorithms: probing the lung image database consortium dataset with two statistical learning methods. *Journal of Medical Imaging*, 3, 2016.

[58] Marina Chane. Medical image annotations for deep learning. 2021. Accessed 3 Sep 2021.

[59] Saad Albawi, Tareq Abed Mohammed, and Saad Al-Zawi. Understanding of a convolutional neural network. In *2017 International Conference on Engineering and Technology (ICET)*, pages 1–6. Ieee, 2017.

[60] Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015.

[61] Sanjay Yadav and Sanyam Shukla. Analysis of k-fold cross-validation over hold-out validation on colossal datasets for quality classification. In *2016 IEEE 6th International conference on advanced computing (IACC)*, pages 78–83. IEEE, 2016.

[62] Rebecca Patro. Cross-validation: K fold vs monte carlo. 2021. Accessed 6 Sep 2021.

[63] Shervine Amidi Afshine Amidi. A detailed example of how to use data generators with keras.

[64] `https://keras.io/`. Accessed 5 Sep 2021.

[65] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[66] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[67] Cheng Wang, Delei Chen, Lin Hao, Xuebo Liu, Yu Zeng, Jianwei Chen, and Guokai Zhang. Pulmonary image classification based on inception-v3 transfer learning model. *IEEE Access*, 7:146533–146541, 2019.

[68] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[69] A. Tharwat. Classification assessment methods. *Applied Computing and Informatics*, 2020.

[70] Amalia Luque, Alejandro Carrasco, Alejandro Martín, and Ana de las Heras. The impact of class imbalance in classification performance metrics based on the binary confusion matrix. *Pattern Recognition*, 91:216–231, 2019.

[71] Sarang Narkhede. Understanding auc - roc curve. 2018. Accessed 3 Sep 2021.

Appendix

Code and System specifications

1 Code

Code written for this thesis, is given in many different jupyter notebooks. You can find it on Google Drive (https://drive.google.com/drive/folders/1TmNZABqgKEaDpEq9uhsdIAwTTBmBTHQp?usp=sharing).

2 System specifications

For the purposes of this thesis, we used two different systems:

- Windows x64, CPU AMD A10 Extreme Edition Radeon R8, 4 cores, 8 GB RAM, 2.00 GHz (for preprocess of images)
- Kaggle (online platform for running machine and deep learning architectures), which offers free GPU NVIDIA Tesla P100 with 2 CPU cores and 13 GB RAM (for runnig deep models)