Master Thesis

Comparison of Deep Learning and radiomics for the detection of immune-induced pneumonitis in lung cancer patients

Alessio Romita

Master Thesis DKE-21-25

Thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science of Artificial Intelligence
at the Department of Data Science and Knowledge Engineering
of the Maastricht University

Thesis Committee:

Dr E. Hortal Quesada
Dr. A. Briassouli

Maastricht University
Faculty of Science and Engineering
Department of Data Science and Knowledge Engineering

July, 2021

Abstract

Immunotherapy-induced pneumonitis is one of the most dangerous side effects of immunotherapy. If not treated early, it can lead to risk for patients or, in the worst scenario, to a deadly outcome. Detecting this side effect in time is a top priority for doctors, who need to make proper decisions to safeguard patients' lives. AI models applied to medical imaging can support doctors in a better early and non-invasive diagnosis. In this work, we explored different solutions to detect immunotherapy-induced pneumonitis in patients with lung cancer. The approaches developed are based on the use of radiomic features and Deep Learning. In addition, a combination of these approaches has been explored. Our dataset comprises 459 patients' CT scans, of which 29 patients were diagnosed with Immunotherapy-induced pneumonitis. The best radiomic model achieved a ROC-AUC score of 0.91, and the Deep Learning model returned a ROC-AUC score of 0.81. The combined model using the average of the probabilities output by the radiomic and Deep Learning model improved the previous result slightly, achieving a ROC-AUC score of 0.92. Our work has shown that AI can be a valuable tool to aid doctors in detecting Immunotherapy-induced pneumonitis.

Contents

1 Introduction 3

1.1 Immunotherapy-indued pneumonitis 3

1.2 Medical imaging for IIP diagnosis 4

1.3 Related Work 5

1.3.1 Deep Learning Medical Image Analysis 5

1.3.2 Radiomic features 5

1.4 Research Questions 7

2 Methods 8

2.1 Dataset Description 8

2.2 Class Imbalance 11

2.3 Image Pre-processing 12

2.4 Evaluation 12

2.5 Deep Learning Approach 13

2.5.1 Deep Learning Pre-processing 13

2.5.2 Data Augmentation 14

2.6 Radiomic Approach 15

2.6.1 Radiomic pre-processing 15

2.7 Combined Model 16

3 Experiments 17

3.1 Deep Learning Model Experiments 18

3.1.1 DenseNet-121 experiment 18

3.1.2 Customized DenseNet experiments 18

3.1.3 Extended Customized DenseNet experiments 18

3.2 Radiomic Model Experiments 18

3.2.1 Radiomic model entire lung mask 19

3.2.2 Radiomic model split lung mask 19

3.3 Combined Model Experiment 20

3.3.1 Features level model 20

3.3.2 Late Fusion model 20

3.3.3 Weighted Late Fusion model 20

4 Result **21**
4.1 Deep Learning Model 21
4.1.1 DenseNet-121 model 21
4.1.2 Customized DenseNet model 23
4.1.3 Extended Customized DenseNet 24
4.2 Radiomic Models Result 25
4.2.1 Radiomic model entire lung mask experiment 25
4.2.2 Radiomic model split lung mask experiment 27
4.3 Combined models 29

5 Discussion **32**

6 Conclusion **35**

7 Supplementary Material **38**
7.1 Deep Learning Metadata Analysis 38
7.2 Deep Learning Model Confusion Matrices 40

Chapter 1

Introduction

1.1 Immunotherapy-indued pneumonitis

Lung cancer still remains the leading cause of cancer-related deaths both in Europe and USA [1]. In the past decades, newer strategies for lung cancer treatment have been developed. Among these, immunotherapy is radically changing the treatment paradigm. Immunotherapy agents, the so-called immune-checkpoint inhibitors (ICI), are administered to patients as intravenous infusions. Compared to radiotherapy and chemotherapy, immunotherapy induces a reaction in the immune system to fight against tumors [2]. At the time of writing, there are six approved immune-checkpoint inhibitors as standard treatment for lung cancer [3]. For metastatic (stage IV) lung cancer patients, clinical trials have proven an increased survival with respect to the use of chemotherapy only. Nevertheless, an artificially “weaponized” immune system can also produce auto-immune reactions, which cause damage to healthy tissues. Immune-related adverse events (irAEs) are frequent during ICI treatment. Even though these events are often mild, an accurate and early discovery is crucial for the optimal clinical management as well as to guarantee a good prognosis while keeping a high quality of life[4]. In stage IV non-small cell lung cancer (NSCLC) treated with immune-checkpoint inhibitors, one of the most dangerous irAEs is pneumonitis [5]. The percentage of patients with NSCLC that developed immunotherapy-induced pneumonitis (IIP) ranges between 4-10%. Grades III-IV represent a life-threatening condition, often requiring immediate ICU (Intensive Care Unit) intervention. Milder conditions (grade I-II) still determine the interruption of immunotherapy treatment. Medications such as glucocorticoids can be administered to reduce the effects of IIP. However, if wrongly administered to other-cause pneumonitis, they may lead to devastating effects such as internal bleeding. The optimal clinical management relies on the differential diagnosis of IIP from other-cause pneumonitis, such as because of bacterial infections or prior radiation treatment (radiation pneumonitis). The gold standard for diagnosis is bronchoscopy with alveolar lavage. However precise, this examination suffers from three major drawbacks: delayed diagnosis because of the waiting time of specimens' results, troublesome procedure for the patients, and in-applicability to most patients with respiratory commodities. Therefore, medical imaging is often used to support the diagnosis.

1.2 Medical imaging for IIP diagnosis

For lung cancers, Computed Tomography (CT) and X-ray imaging are two of the most common image modalities for the diagnosis of both tumors or pneumonitis. Although both of these modalities rely on X-rays to measure the electron density of body tissues, there are some notable differences. The most important difference between the two modalities is the output images. For instance, the CT scanner collects 360-degree projection data and solves an inverse 3D image reconstruction problem, whereas the X-ray scanner produces one integral 2D projection plane. CT scans are 3D images formed by stacking together 2D slices, and the X-ray scans are single 2D images. As mentioned in the previous section, the gold standard to detect IIP is bronchoscopy with alveolar lavage. However, this invasive diagnostic technique can not be executed on patients with particular respiratory comorbidities. In this context, medical images are used as non-invasive tools to support the diagnosis. For IIP diagnosis, thorax CT scans of the patients are acquired and visually inspected by doctors. The visual interpretation of these scans is time-consuming and challenging and, potentially leading to a wrong diagnosis. Immunotherapy-induced pneumonitis patterns share common image characteristics with pneumonitis caused by other factors. For this reason, a diagnosis solely based on visual inspection is difficult to be performed by doctors. State-of-the-art knowledge on IIP imaging patterns is based on few clinical case series, which reported a tendency of IIP patterns to appear near tumors or metastatic tissues [6]. In addition, for some patients, patterns of IIP tend to appear in both lungs and/or in asymmetric positions, compared to bacterial pneumonitis. An example of patients with pneumonitis from different causes is shown in Figure 1.1. Nevertheless, there is a need to boost doctors' capability of inspecting medical images for a better diagnosis. Artificial Intelligence (AI) gained interest in the recent decade as a valuable tool for reducing the burden on clinicians by assisting them in data analysis. From Computer-aided diagnosis systems to clinical decision aids, AI is improving clinical decision-making. This work has explored how AI, and more specifically computer vision algorithms, can be applied to improve the detection of IIP in lung cancer patients treated with immune-checkpoint inhibitors.

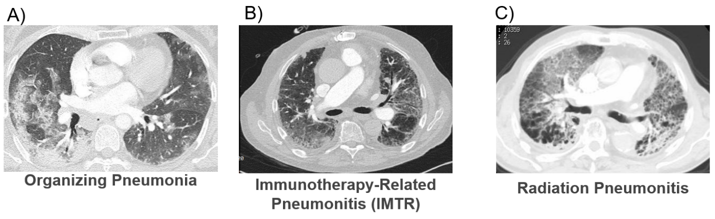


Figure 1.1: Patients who have developed A) Organizing Pneumonia, B) Immunotherapy-induced pneumonitis, and C)Radiation Pneumonitis. In all the scans, common imaging patterns, such as the presence of opacities, are present, making the diagnosis based on visual inspection only challenging.

1.3 Related Work

This section introduces related work, with a particular focus on deep learning and radiomic applications.

1.3.1 Deep Learning Medical Image Analysis

Artificial intelligence and, in particular, Deep Learning (DL) raised a massive interest in the scientific community in the past decades. The scientific community has shown remarkable results for computer vision classification, object recognition, and segmentation tasks. Recently, DL applications for medical image analysis have been developed, and they have been showing promising results for both classification and segmentation tasks [7] [8]. Besides using DL applications for cancer prognosis, many studies have investigated the possible application that can be developed to detect and localize respiratory diseases such as pneumonitis. Rajpurkar et al. [7] in their work built a Convolutional Neural Network based on a DenseNete-121 architecture to detect cases of pneumonitis. They compared the performance of their model with the performance of trained radiologists. Their results show that the model built outperforms doctors' performance. Hasmi et al. [9] proposed the use of a weighted classifier to detect pneumonitis. In their approach, the model combined the prediction of various state-of-the-art classifiers to make better predictions. Their model achieved an AUC (Area Under the Curve) score of 99.76 on their test set. Finally, DL models have been developed to detect pneumonitis caused by the SARS-COV2 from X-ray and CT scans of symptomatic patients [10] [11].

1.3.2 Radiomic features

Radiomic features are quantitative features extracted from medical images [12]. They have been widely used with machine learning models to perform survival analysis, classification, and regression tasks. The extraction of radiomic features is done on a specific region of interest (ROI) of the images, previously segmented, such as tumor regions or healthy tissues. Pyradiomics is one of the most common open-source radiomic software[1]. The process used for the extraction is described in Figure 1.2. Radiomic features are divided into different categories. Three groups of radiomic features are generally defined:

- Statistical features: They give information about the distribution of the intensity values of the voxels/pixels in the images.
- Shape features (Morphological features): They describe the geometrical aspect of the ROI from which they are extracted.
- Texture features: They are calculated over different matrices that represent the textures in the images. This category of features can be further split into several groups based on the matrices on which the features are calculated:
  - Gray Level Co-occurrence matrix (GLCM) describes the second-order joint probability function of an image region constrained by the mask.
  - Gray Level Run Length Matrix (GLRLM) quantifies gray level runs, which are defined as the length in the number of pixels, of consecutive pixels that have the same gray level value.
  - Gray Level Size Zone Matrix (GLSZM) quantifies gray level zones in an image. A gray-level zone is defined as the number of connected voxels that share the same gray level intensity.
  - Neighbouring Gray Tone Difference Matrix (NGTDM) quantifies the difference between a gray value and the average gray value of its neighbors within a specific distance.
  - Gray Level Dependence Matrix(GLDM) quantifies gray level dependencies in an image. A gray level dependency is defined as the number of connected voxels within a specific distance that are dependent on the center voxel.

Radiomics have shown in past research promising results for survival analysis in lung and head, and neck cancer [13]. However, what radiomics capture and represent as information is still under discussion. In addition, other researchers have shown how they suffer from different sources of bias like: scanner setting, volume, and delineations of ROI [14][15]. Applications of radiomics have currently been limited only to predict cancer prognosis or treatment response. At the time of writing, there are no studies but one that investigated radiomics for the differential diagnosis of IIP. Colen et al.[16] extracted radiomics from the segmented lung lobes and used them to classify patients with IIP. This work suggested that radiomic features could be used to build a model able to predict IIP. It is essential to mention that the dataset used by the authors contains just two patients who developed IIP, opening the debate whether these results hold on a more extensive and diverse dataset.

[1] https://pyradiomics.readthedocs.io/

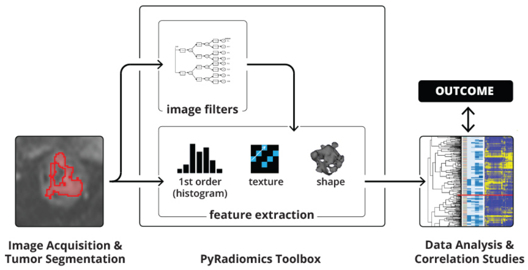


Figure 1.2: Image representation of pyradiomics pipeline. Starting from the left, the medical images and their mask are the input to the pyradiomics library. Then, the features are extracted from the masks and are used for modeling and analysis.

1.4 Research Questions

The research questions that our work aims to answer are presented in this section.

- Is it possible to extract CT-derived biomarkers able to predict pneumonitis caused by immunotherapy?
- Are radiomic features able to give similar performances than the Deep Learning approaches in predicting immuno-therapy induced pneumonitis, but intrinsically improving the explainability of the model?
- Could a combination of these two approaches lead to an improvement in the prediction?

Chapter 2

Methods

This section describes the methodologies applied to answer the research questions. It starts with a description of the dataset and related issues. The preprocessing steps applied to standardize the images in the dataset are described, followed by the evaluation metrics used and the developed approach.

2.1 Dataset Description

The dataset used contains 459 CT scans of stage IV NSCLC patients treated with immunotherapy and acquired among six different medical centers within the Netherlands and Belgium as part of a clinical trial. Only one CT scan per patient acquired three months after the start of the treatment was included in the study. Prior or follow-up CT scans were discarded. The dataset includes 29 patients who have developed IIP and 35 patients with pneumonitis caused by the other factors. The rest of the patients did not present any form of pneumonitis. Examples of CT scans from a patient without any pneumonitis, a patient affected by IIP, and a patient with pneumonitis from other causes are shown in Figure 2.1. Given the nature of the trial, CT scans have been acquired with different scanners and settings among the centers. The CT images have been pre-processed as explained in Section 2.3. For modeling purposes, the dataset has been split into training and validation sets. Due to the amount of data available, we split our data based on the center of acquisition. All the centers except(Zuyderland, NKI, Amsterdam, Erasmus and Leuven) the Maastricht MUMC center have been considered for the training set. Instead, in the validation set, we used the images acquired just in the Maastricht medical center. The CT images are stored in the Digital Imaging and Communications in Medicine (DICOM)[1] format, which is the standard for medical images. The DICOM files contain the related image and information about the patients, including information about the generation of the images. However, the previously mentioned DICOM format contains both the metadata about the extraction setting and

[1] https://www.dicomstandard.org/

the images with its metadata. We analyzed three settings for the extraction, namely Kilovoltage peak (KVP), the Slice Thickness, and the scanner manufacturer. The KVP is the peak potential applied to the x-ray tube; it influences the quality of the images. The slice thickness represents the acquisition thickness of the image (that differs from the distance between each slice in the final CT scan), and the manufacturer data indicate the brand of the scanner used. The distribution of this metadata among the dataset is shown in Figure 2.2 and the Table 2.1 distribution of the outcome among the center of acquisition.

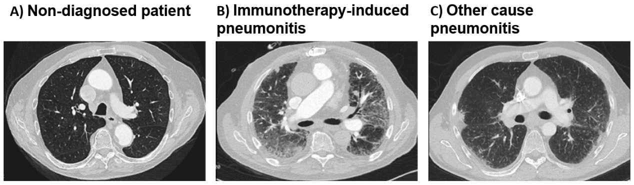

A) Non-diagnosed patient
B) Immunotherapy-induced pneumonitis
C) Other cause pneumonitis

Figure 2.1: From left to right: CT scan of 1) a Non-diagnosed patient. 2) a patient who has developed Immunotherapy-induced pneumonitis. 3) a patient who has pneumonitis from other factors.

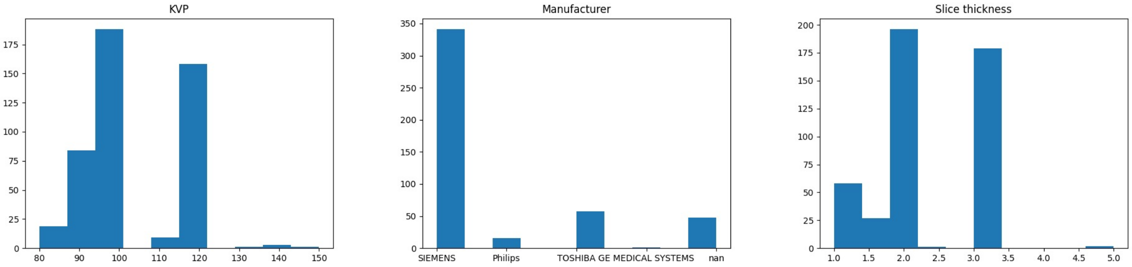


Figure 2.2: Distribution of the metadata in the dataset. From left to right, Kilovoltage peak, Manufacturer and Slice Thickness.

Center	No. IIP patients	No. No IIP patients
Center 1	7	189
Center 2	2	42
Center 3	6	78
Center 4	1	50
Center 5	3	71
Center 6	10	0

Table 2.1: The number of patient's scans for each center based on the outcome.

2.2 Class Imbalance

As said in the previous section, IIP affects around 4-10 percent of the patients treated. Consequently, the dataset at hand is highly imbalanced. The ratio between patients with IIP and patients with no IIP is around one over fifteen in our dataset. Figure 2.3 shows the distribution of the outcome among the patients. In our work, we merged the case of pneumonitis from other causes with the non-diagnosed patients to solve the problem as a binary classification problem.

Machine Learning and Deep Learning models' performances suffer from skewed distribution of the data. The problem is amplified if the number of samples is limited. However, different strategies have been developed to overcome this problem and improve performances. Over-sampling and under-sampling techniques, which modify training distributions, can be used to overcome this issue. In under-sampling, we down-sample the majority class matching the number of samples of the minority class. Conversely, in oversample, we oversample the minority class [17]. Additional techniques use a specialized loss function or a weight for the class in the Loss function. An example is the Focal loss [18], adapted from the Cross-Entropy loss to consider classes that are more difficult to recognize.

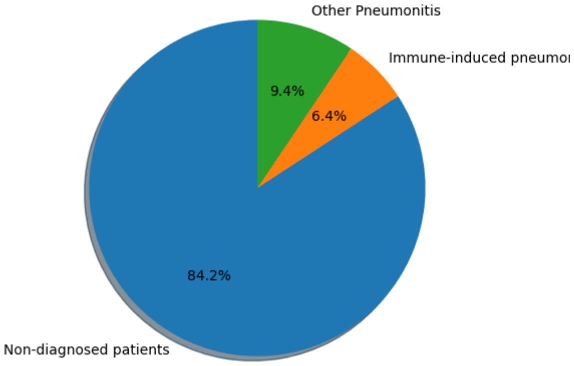


Figure 2.3: Percentages of patients for each outcome . In blue the non-diagnosed patients; In orange the IIP patients and in green the patients with pneumonitis from other cause.

2.3 Image Pre-processing

The original images have been converted to the NRRD format, which allows using the SimpleITK library for pre-processing [2]. In addition, conversion to this format discards all the metadata present in the DICOMs unrelated to the images. A re-sampling is needed to standardize images and avoid that different pixel spacing affects the robustness of texture features. A spacing of (0.98, 0.98, 3.0) has been chosen, with the values closer to the other spacing. If the image spacing was less than the chosen spacing, it was re-sampled; otherwise, they were discarded. We discarded a total of three images. A list of the usual step to pre-process the images is summarized below:

- Convert from DICOM format to NRRD format.
- Extract the spacing of the images from the dataset.
- Select the most common bigger spacing.
- Re-sample the images with smaller spacing.
- Discard the images with a bigger spacing.

2.4 Evaluation

The evaluation of the binary classification has been done using the Area Under the ROC (Receiver Operating Characteristics) Curve score. The curve indicates the extent to which the model is able to discriminate between the two classes. The higher the score is, the greater the discriminative ability power of the model. In addition to the AUC ROC curve and score, we reported the score related to the Precision, Recall, and F-1 values (picking the best threshold from the AUC ROC curve) and the related Confusion Matrix. The Confidence Interval (CI) of the ROC AUC score for each radiomic model has been calculated, bootstrapping the related test set hundred times. The Calibration plot for each model has been generated. It represents the agreement between the predicted probability and the observed outcomes.

[2] https://simpleitk.readthedocs.io/

2.5 Deep Learning Approach

We built a model using a deep convolution neural network to perform a binary classification task using the Pytorch library[3] as the first approach. The model took as input the image to be classified and outputs the corresponding label. The first choice made was about the architecture of the network. Different architectures were available, and researchers usually pick an architecture and customize it for their task. To avoid de-bugging issues, we decided to make a conservative choice by using a predefined architecture. MONAI is a framework for medical image analysis [4] that offers different predefined architectures to perform classification and segmentation tasks. The architecture picked from this framework is a 3D DenseNet [19]. The main difference with previous architectures such as AlexNet or ResNet is the presence of the so-called Dense Block in which each layer is linked to the other layers inside the block. Each layer passes its feature map to the next layer, which concatenates the features. This architecture has shown robustness against the vanishing gradient problem and excellent performances in classification tasks. The next choice was about the loss function to be used for the binary classification problem. A good and standard choice is the Binary Cross-Entropy loss (BCE loss). Pytorch offers a predefined binary cross-entropy loss called BCEWITHLOGITSLOSS. This loss combines the sigmoid function with the binary cross-entropy formula given more numerical stability. In addition, this loss offers the possibility to use a specific parameter to balance the minority class importance. The pre-processing steps applied are explained in the following subsection.

2.5.1 Deep Learning Pre-processing

Two relevant constraints for training DL models are the memory requirement and the speed of the process. To overcome these issues, a Graphics Processor Unit (GPU) is required. However, 3D CT scans are used in our work, which further increases the constrain on memory usage. A smaller region has to be selected to overcome this issue. In addition, the background of the images did not contain any valuable information for our task. Therefore, we decided to segment the lung region of the CT images to reduce the dimension of the images. The segmentation has been done, thresholding the Hounsfield Unit (HU) values. The algorithm can return a binary mask where the lungs are specified or the original image with just the lung region visible. In addition, we generated a bounding box around the lung using the lung binary mask. Using only the segmented lung, some information can be lost. We generated a bounding box to overcome this potential issue, so damaged tissues close to the heart were visible.

We applied cropping and padding on the x, y, and z axes to standardize the image's dimension and remove the patient's table. Then, the HU image values have been clipped to a range of (-1000, 0), which is the typical range of the lung's HU. Then, the image value has been normalized between 0 and 1.

[3] https://pytorch.org/

[4] https://monai.io/

2.5.2 Data Augmentation

One of the possible solutions to overcome class imbalance and the limited sample size was the use of augmentation. Using augmentation, we generated new images applying specific transformations. However, when augmentation is applied to medical images, we should be careful to keep the image as natural as possible without creating non-real images.

The torchIO [5] library has been used to apply such transformation. The techniques used for the augmentation are listed below, and an example of an augmented sample is shown in Figure 2.4.

- Random Translation: The images have been translated with respect to the x and y axes of a factor between -10 and 10 mm.
- Random Rotation: The images have been rotated of a degree value between -10 and 10 with respect to the origin.
- Random Cropping: The images are cropped of a factor equal to the 5% of each dimension.
- Jittering: Add a value to the images equal to the 0.1% of the maximum pixel value.

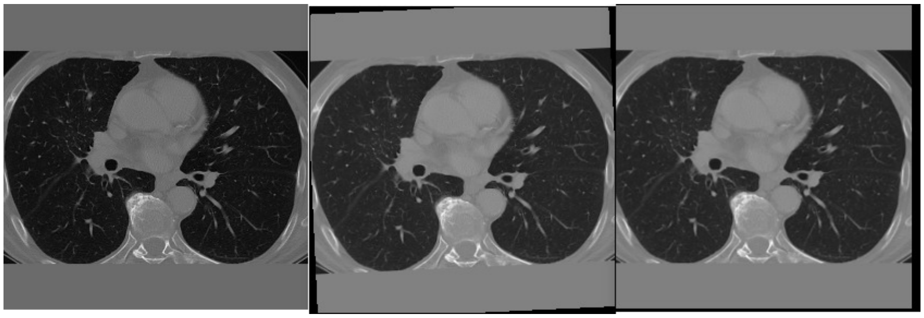


Figure 2.4: From left to right: 1)Original slice of a CT scan, 2)Rotated slice of the CT scan and 3) Translated slice of a CT scan.

[5] https://torchio.readthedocs.io/

2.6 Radiomic Approach

As second approach, radiomic features have been extracted from the lungs of each patient. For the extraction, the python library pyradiomics has been used. We experimented with two configurations:

- In the first setting, radiomic features have been extracted from the whole lung mask.
- In the second, radiomic features have been extracted from the right and left lung independently.

The extraction returned the same type of features for each of the two settings, but their number is doubled in the second configuration. Different feature selection and dimensionality reduction techniques have been tested and evaluated for each of the settings. Recursive features Elimination (RFE) and Principle component analysis (PCA) were the main techniques applied. The resulting data have been split and fitted to various machine learning models based on the acquisition center, and the result has been compared. Correlation analysis has been applied to prevent problems of multi-collinearity between predictors [20]. The scikit-learn library[6] has been used to apply feature selection/reduction and build different machine learning models. The group of features extracted are GLCM, GLSZM, GLRLM, and first-order features. In addition, the same features are extracted with a wavelet filter applied. All the features have been normalized between 0 and 1 using the MinMaxScaler function of the scikit-learn library in python. Finally, the stability of the models built with respect to the data split and with other factors as normalization of the data and model hyper-parameters have been tested. A detailed description of the experiment is presented in Section 3.2.

2.6.1 Radiomic pre-processing

In contrast to the DL approach, radiomics does not require a GPU to extract the features and build the model. This makes radiomic extraction easier and faster than DL approaches. However, radiomic extraction requires inputting the 3D images an the masks for the extraction with pyradiomics. The CT images and the binary mask mentioned in the DL pre-processing have been used for the extraction. The second setting required to have a binary mask for each of the patient's lungs. The generation of these binary masks has been done by splitting the images by their x-axis centers. However, the segmentation algorithm could fail to segment one of the lungs; this issue was caused by mass or other pathological conditions. To overcome this issue, we manually deleted all the patients for which the mask shows just a lung or a small portion of it for this approach. We discarded a total of 26 masks from the total.

[6] https://scikit-learn.org/

2.7 Combined Model

When we train different models to perform a task, single performances can differ. An ensemble of the models can lead to an improvement in the final results. For this reason, the last approach developed is based on a combination of the DL and the radiomic models. We decided to use three strategies: Features Level combined model, Late fusion model (equal importance), and Weighted Late Fusion (trained on the probabilities). A summary of the experimented combined model is represented in Figure 2.5.

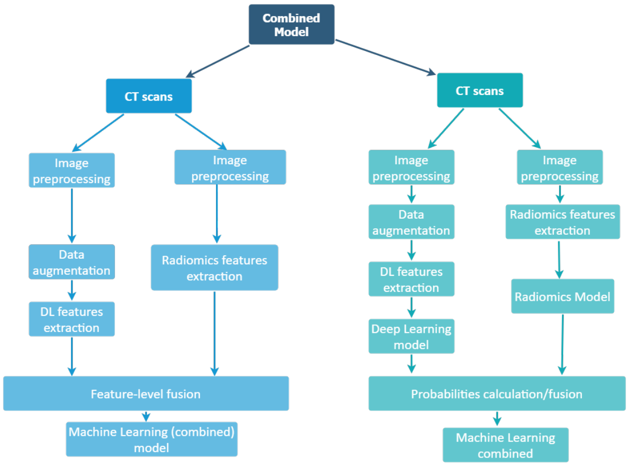


Figure 2.5: On the left: Representation of the combined model using the features from the Deep Learning and radiomic model. On the right :Representation of the combined model using the probabilities of the two models.

Chapter 3

Experiments

In this section, the experiment done in this work have been described. The first section describes the experiments related to the DL model, the second section the radiomic model, and the third and last section the combined model experiments. Figure 3.1 shown a summary of all the experiments done.

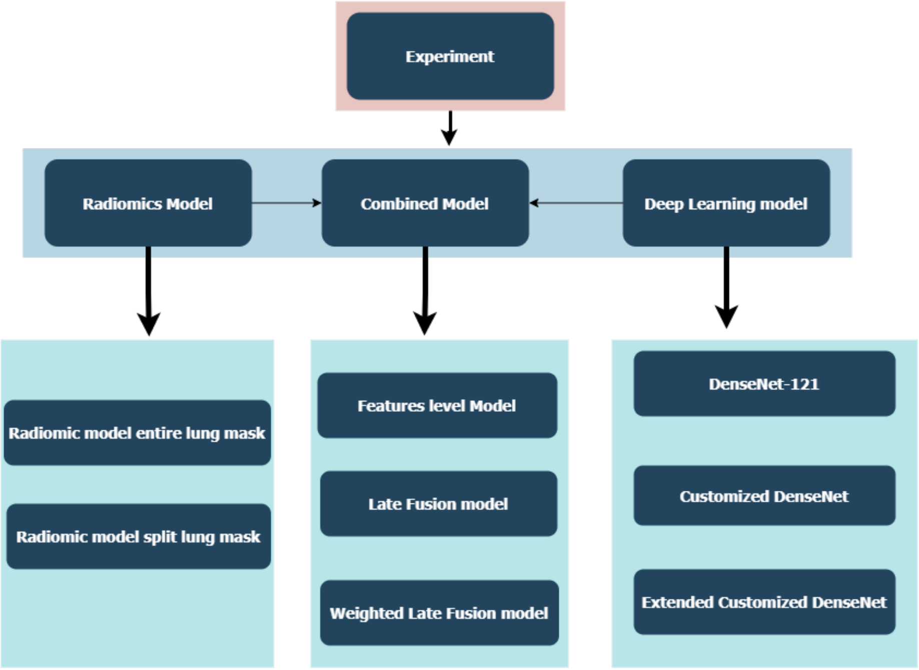


Figure 3.1: Summary of the experiment for each approach.From left to right: 1) radiomic model experiments, 2) Combined model experiments and 3) Deep Learning model experiments.

3.1 Deep Learning Model Experiments

As mentioned in the previous sections, our model used a DenseNet architecture. We experimented with different configurations from the standard DenseNet-121 to custom DenseNet with different layers in each Dense block and hyperparameter settings. To contrast the class imbalance problem, we experimented with two possible solutions: oversampling and weighted loss. In addition, different image dimensions have been tested as input to the network. All the models trained in the experiments have been trained on the dataset split by center as explained in Section 2.1.

3.1.1 DenseNet-121 experiment

In the first experiment, we trained a DenseNet-121 architecture. The loss function used was the weighted Binary Cross-Entropy loss. Oversampling has been used in combination with augmentation to overcome the class imbalance problem. We trained the network twice using as input the segmented lung and the bounding box around them. The dimension set was (x=224, y=224, z=32). The number of slices used represented the minimum number that the network can handle without returning an error caused by the dimension of the images.

3.1.2 Customized DenseNet experiments

In the second experiment, a custom DenseNet has been used. This architecture has a block configuration of (6, 6, 6), a growth rate of 8 and 32 initial filters. Using this setting for the architecture, we significantly reduced the number of parameters of the network. We trained the model with the weighted Binary Cross-Entropy loss and no oversampling and augmentation in this experiment. We choose to use only the Binary Cross-Entropy loss to check how the model performed without augmentation. The model has been trained twice using the segmented lungs and the bounding box around them. The image dimension used was (x=224, y=224, z= 16).

3.1.3 Extended Customized DenseNet experiments

In the last experiment, we retrained the model described in Section 3.1.2 using oversampling and applying augmentation to make the model more robust and prevent overfitting.

3.2 Radiomic Model Experiments

Two models were built for the radiomic approach; the first used radiomic features from the lungs region (the binary mask representing the segmented lung). The second one used radiomic features extracted from each of the lungs independently (the split binary mask of the lung). The Random Forest model hyperparameters estimate has been done using a Randomized search on a predefined parameter dictionary. The best configuration is selected based on a score assign to each configuration. Correlation analysis has been done using the Spearman coefficient to capture non-linear relations between the variables; in contrast with the Pearson coefficient. The dataset has been split into train and validation sets based on acquisition centers as described in Section 2.1.

3.2.1 Radiomic model entire lung mask

We used the features extracted from the whole lung mask in the first experiment to build the model. To reduce the number of features, we applied RFE without specifying the number of features to be selected. Then, correlation analysis has been done to avoid multi-collinearity problems and overfitting. Finally, the remained features have been fitted to Logistic regression, Random forest, and Support Vector Machine model. We have tested all the models using different combinations of each feature's type.

3.2.2 Radiomic model split lung mask

In the second experiment, we used the second group of extracted features from each lung independently. RFE and correlation analysis have been performed as in the previous experiment. To further reduce the number of features, we applied PCA. This step has been done because the number of features in these models is double that of the previous. The number of components generated has been decided by plotting the explained variance versus the number of them. As for the previous experiment, we tested different feature classes and fit them in the three models.

3.3 Combined Model Experiment

In this section, the experiments done using three different strategies are explained.

3.3.1 Features level model

This model has been built using the radiomic features presented in Table 4.2 and the features of the last feed-forward layer of the last experimented DL model. We applied RFE to select the features ranked first by the algorithm. Then, we performed correlation analysis with the Spearman coefficient to eliminate the highly correlated features with a threshold of 0.85. Finally, we have fitted the features into the Random forest, Logistic regression, and Support Vector Machine classifiers.

3.3.2 Late Fusion model

Instead of using the features extracted from the two models, this model used the output of the two models to make predictions, called the Late Fusion model. In this experiment, we average the two probabilities returned by the two models and generated a new probability for each sample. In this experiment, we assumed that both models had the same importance.

3.3.3 Weighted Late Fusion model

In the last experiment, we used the probabilities output by the two models. The main difference with the experiment described in Section 3.3.2 is that we did not assume that the two models have the same importance. Then, we trained a Logistic regression model, giving the two models' probabilities as input and learning the parameters to assign to each input. The model has been trained on the same data used for the input model. Then, the analysis of these probabilities was used to weight each modality based on their expected importance.

Chapter 4

Result

In this section, the results from the experiments described in the previous chapter are shown. First, the results of the DL experiments, then the results of the radiomic experiments, and finally, the results of the combined model experiments are described. The AUC-ROC curve and score are shown for each result and the related confusion matrix. In the DL result section, the confusion matrix is shown for the last experiment only. The confusion matrices of the other experiments can be found in the Supplementary material (chapter 7).

4.1 Deep Learning Model

4.1.1 DenseNet-121 model

The AUC-ROC scores for the two models were respectively 0.75 for the model that had as input the segmented lungs and 0.75 for the model of the experiment DenseNet-121 experiment, respectively. The two AUC-ROC curves and the calibration plot for these two models are shown in Figure 4.1 and 4.2.

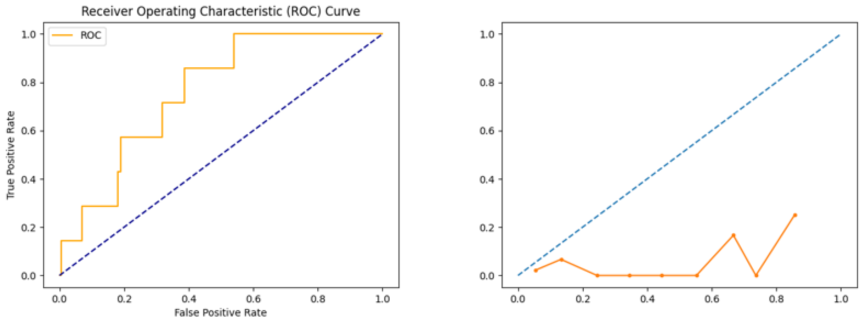


Figure 4.1: AUC-ROC curve and Calibration plot for the DenseNet-121 experiment using the bounding box.

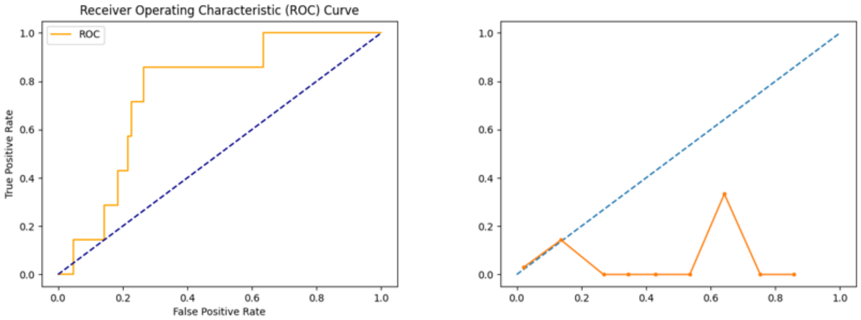


Figure 4.2: AUC-ROC curve and Calibration plot for the DenseNet-121 experiment using the segmented lung.

4.1.2 Customized DenseNet model

The model that used the custom architecture using the segmented lung return an AUC-ROC score of 0.70. Instead, the model that used the bounding box around the lung as input return a score of 0.84. The AUC-ROC curves with the calibration plots are shown in Figure 4.3 and 4.4.

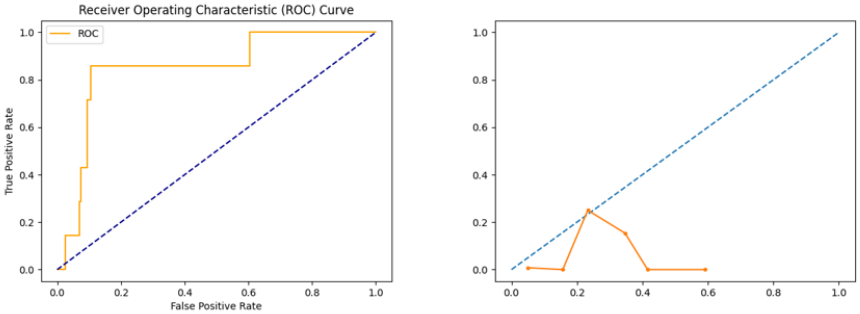


Figure 4.3: AUC-ROC curve and Calibration plot for the Customized DenseNet experiments with the bounding box.

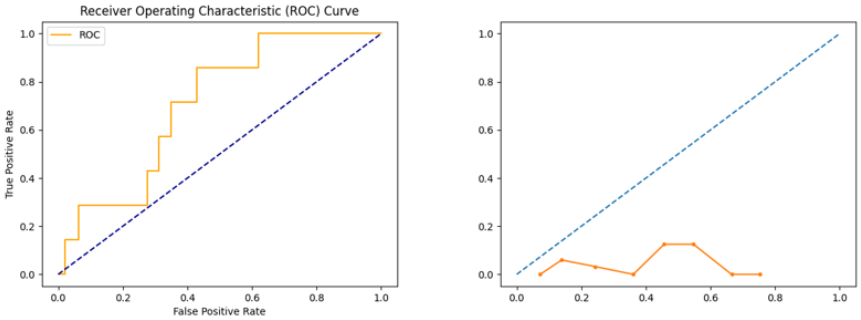


Figure 4.4: AUC-ROC curve and Calibration plot for Customized DenseNet experiments with the segmented lung.

4.1.3 Extended Customized DenseNet

The model trained in the Extended Customized DenseNet experiment achieved an AUC-ROC score of 0.81. The input of the model was the bounding box around the lungs. The AUC-ROC curves and calibration plot for the model are shown in Figure 4.5; the confusion matrix generated using the best threshold is shown in Figure 4.6.

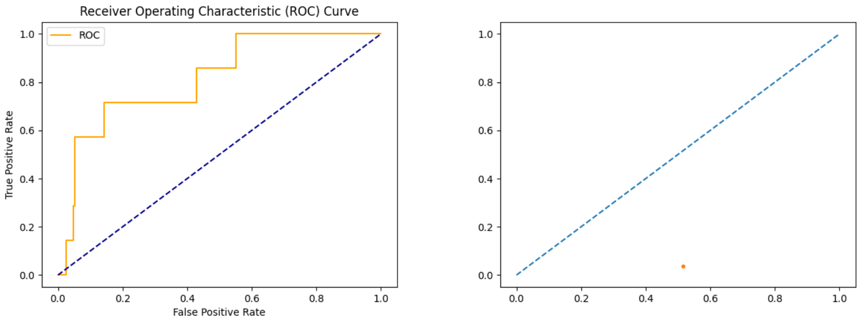


Figure 4.5: AUC-ROC curve and Calibration plot for the Extended Customized DenseNet experiments.

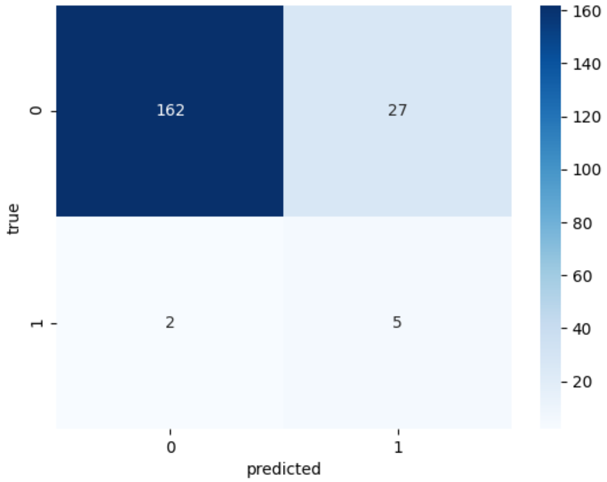


Figure 4.6: Confusion Matrix for Extended Customized DenseNet experiments.

4.2 Radiomic Models Result

4.2.1 Radiomic model entire lung mask experiment

The features extracted from the whole mask of the lung are 609, considering all the features class mentioned in the experiment setting. The classes associated with the best model are the GLCM and GLSZM features. The features ranked as the most important from the RFE algorithm were 180, and these features have been further decreased, dropping the highly correlated features. The threshold used to drop the features has been set to 0.6 and the number of features got is 8. The features are reported in Table 4.2. These features have been fitted to three different classifiers: a Support Vector Machine, a Random Forest, and a Logistic regression. The AUC ROC score and Confidence Interval (CI) for each classifier are shown in Table 4.1. The ROC curves and calibration plot for the Logistic model are shown in Figure 4.7 and, the related confusion matrix generated selecting the best threshold is shown in Figure 4.8. The Confidence Interval generated over 100 bootstraps for the validation set was [0.83-0.95] for the Logistic regression model.

Metric	Logistic regression	Random Forest	Support Vector
AUC ROC score	0.91	0.89	0.84
Confidence interval	0.82-0.95	0.77-0.96	0.75-0.92

Table 4.1: AUC ROC score and CI result for each classifier for radiomic model entire lung mask.

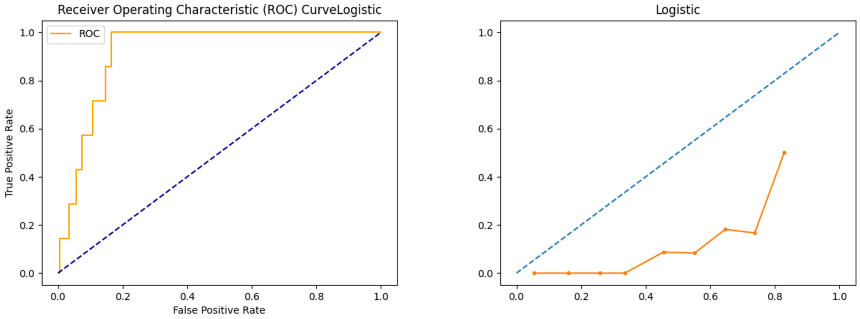


Figure 4.7: ROC curve and Calibration plot for the Logistic regression of radiomic model entire lung mask.

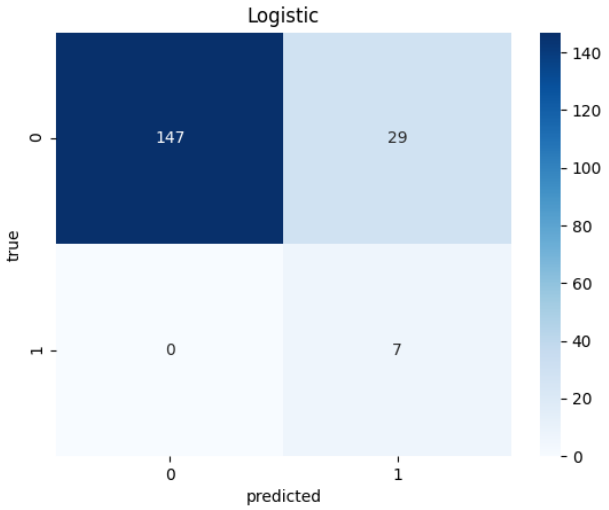


Figure 4.8: Confusion matrix for the radiomic model entire lung mask.

Features
originalGlcmDifferenceVariance
originalGlcmIdmn
originalGlszmGrayLevelNonUniformityNormalized
originalGlszmLowGrayLevelZoneEmphasis
LHLGlszmZoneEntropy
HLHGlcmClusterShade
HHHGlszmSizeZoneNonUniformityNormalized
waveletLLLGlszmLowGrayLevelZoneEmphasis

Table 4.2: Table of the features selected for the radiomic model entire lung mask.

4.2.2 Radiomic model split lung mask experiment

The total number of features extracted from each lung independently was 1335. This model includes all the radiomic feature classes. The selected features after the features selection process was 17. The features are reported in Table 4.4. To further reduce the number of features we applied PCA, we selected ten components explaining more than 90 percent of the variance. These features have been fitted to three different models: Support Vector Machine, Random Forest, and Logistic regression.

The AUC ROC score for each classifier is presented in Table 4.3. The ROC curves and calibration plot for the Logistic model are shown in Figure 4.9 and, the related confusion matrix generated selecting the best threshold is shown in Figure 4.10. The Confidence Interval generates on 100 bootstraps for the validation set was [0.72-0.94] for the Logistic regression model.

Metric	Logistic regression	Random Forest	Support Vector
AUC ROC score	0.86	0.56	0.81
Confidence interval	0.72-0.92	0.29-0.79	0.14-0.92

Table 4.3: AUC ROC score and CI results for each classifier for radiomic model split lung mask experiment.

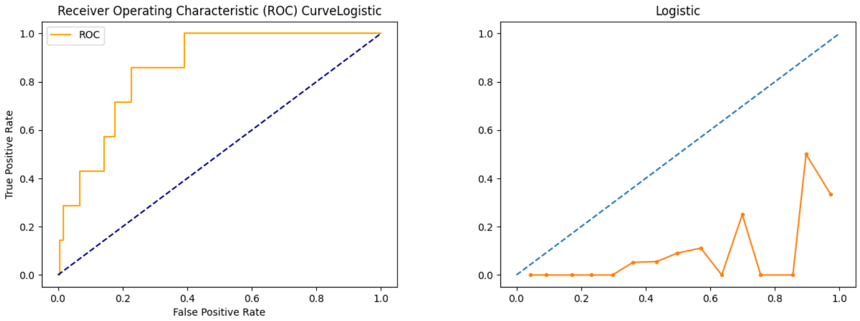


Figure 4.9: ROC curve and Calibration plot for the Logistic regression of radiomic model split lung mask experiment.

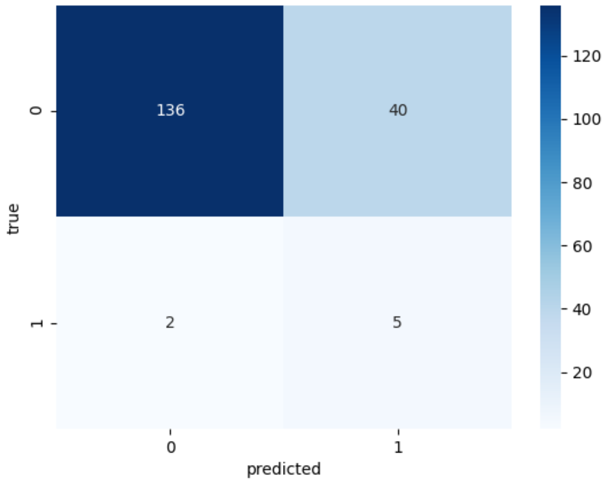


Figure 4.10: Confusion matrix for the radiomic model split lung mask experiment.

Features	Features
originalFirstorder90PercentileRight	LHLFirstorderMeanRight
originalFirstorderEnergyRight	LHLGlszmZoneEntropyRight
originalFirstorderInterquartileRangeRight	HLLFirstorderSkewnessRight
originalFirstorderMedianRight	HLHFirstorderMeanRight
originalFirstorderMinimumRight	LHHFirstorderMean
originalGlcmContrastRight	waveletLHHFirstorderSkewness
LLHFirstorderRangeRight	waveletHLHFirstorderSkewness
LLHGlcmIdmnRight	waveletHHLFirstorderSkewness
originalFirstorder90Percentile	

Table 4.4: Table of the features selected in the second radiomic model. If the term ”Rigth” is included at the end of the features name it means that it was extracted from the right lung. Otherwise, the features are from the left lung.

4.3 Combined models

The total number of features available of the radiomic model entire lung mask and Extended Customized DenseNet experiments for the Feature Level model was 100. The RFE algorithm has been applied to these features, and the number of features ranked first was 50. Then, these features were fitted on Support Vector Machine, Random Forest models, and Logistic regression. The AUC-ROC scores for these models are shown in Table 4.5 and the AUC-ROC curves and calibration plot for the Logistic regression model are shown in Figure 4.11. The Late Fusion model that averages with equal importance the probabilities from the Deep Learning and radiomic model has an AUC-ROC score of 0.92. The AUC-ROC score curve and calibration plot are shown in Figure 4.12. The confusion matrices for both models picking the best threshold from the respective AUC-ROC curves are shown in Figure 4.13. The Logistic regression model fitted with the probability of the Deep Learning and Radiomic models returns an AUC-ROC score of 0.92 and a CI of 0.83-95. The AUC-ROC curve and the calibration plot are shown in Figure 4.14.

AUC-ROC score			
metric	Logistic regression	Random Forest	Support Vector
AUC-ROC score	0.84	0.80	0.80
Confidence Interval	0.64-0.92	0.65-0.95	0.65-0.93

Table 4.5: AUC-ROC score result of the Feature Level model.

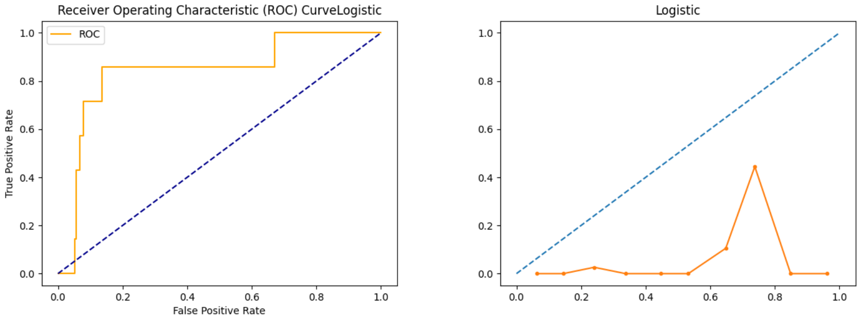


Figure 4.11: AUC-ROC curve and Calibration plot for the Weighted Late Fusion model.

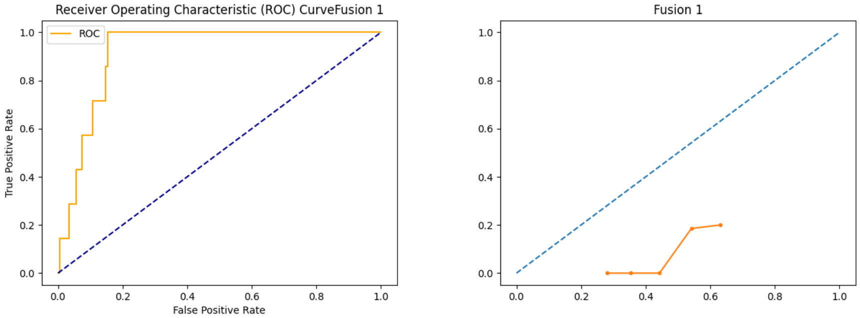


Figure 4.12: AUC-ROC curve and Calibration plot for the Late Fusion model.

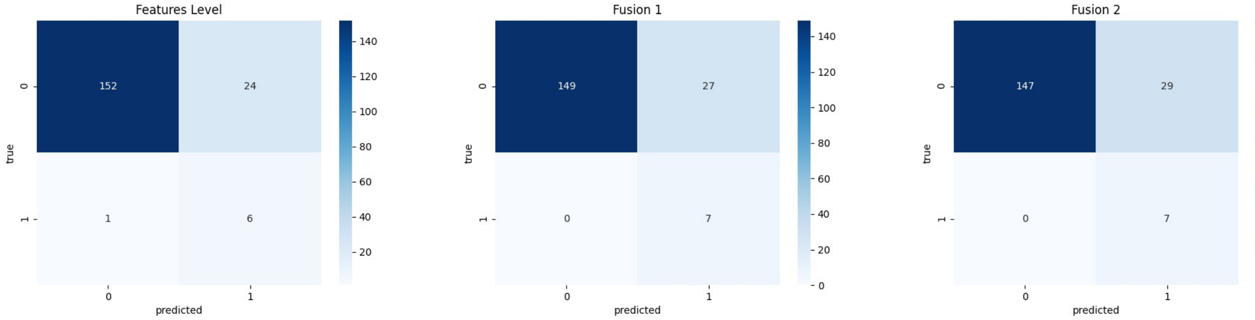


Figure 4.13: From left to right: 1) Confusion matrix of the Feature level model, 2) The confusion matrix of the Late Fusion model with equal importance,3) The confusion matrix of Weighted Late Fusion model.

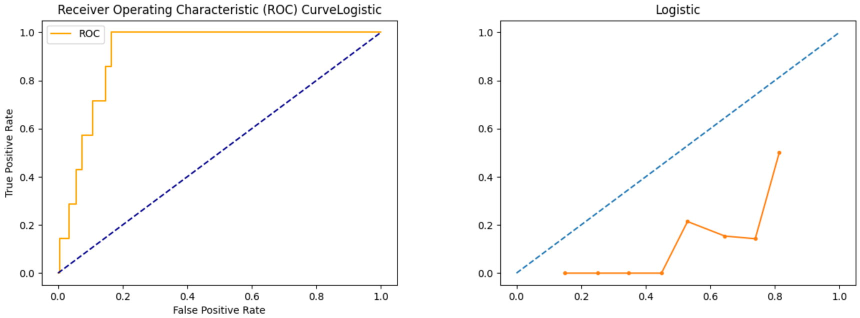


Figure 4.14: AUC-ROC curve and calibration plot of the Weighted Late Fusion model.

Model	Recall(class 1)	F1(class 1)	Recall(class 0)	F1(class 0)
Radiomic	1	0.33	0.84	0.91
Deep Learning	0.71	0.26	0.86	0.92
Late Fusion	1	0.34	0.86	0.92

Table 4.6: Recall and F1 values for the best model of each approach.

Chapter 5

Discussion

In our work, we explored radiomic and DL models for the differential diagnosis of IIP. The radiomic-based models have been developed and compared to the DL models to check if they can improve the explainability level. In the third and last approach, the two previous models have been combined to analyze the presence of essential information from these two approaches. The Deep Learning models have shown good performance in discriminating the immuno-induced pneumonitis cases. The model that returned the best result based on the AUC-ROC score was the Customized DenseNet model, with a score of 0.84. However, the Extended Customized DenseNet, even if it has an AUC-ROC score of 0.81, shown more stability and generalization ability during the training. The underperformance of the model that used the segmented lungs can be caused by the poor performance of the segmentation algorithm; or because other tissues not visible when segmenting the lungs provide important information. In addition, the dimension of the 3D images to fit into the models influenced the final result. When we fitted a too big image to the original DenseNet-121, the model started to over-fit. The AUC-ROC score for the validation set could not increase more than 0.70 and showed a high false-positive rate. Instead, using a custom DenseNet, with fewer parameters and decreasing the image dimension, improved the final result in the validation set.

The radiomic models built in the experiments have shown the best performance using a Logistic regression classifier with good ROC-AUC scores and confidence intervals. However, the ROC-AUC score and the confusion matrices in Figure 4.8 showed better performance for the radiomic model entire lung mask with a ROC-AUC score of 0.91 (0.84 radiomic model split lung mask). The improvement was related to the precision of the model in classifying cases of IIP, reducing the number of false positives. The explainability of the radiomic model entire lung mask has been preserved in contrast with the radiomic model split lung mask. In the latter model, PCA has been applied. PCA is a powerful method for dimensionality reduction but at the cost of losing all the single feature information. We expected to have better or similar results from the model that used features extracted from each lung independently compared with the first one, which used features from the whole lung. The reason for the performance degradation of this model can be related to how we have generated the two lungs. Due to the unavailability of annotation for each lung, we split the segmented mask using the median point on the x-axis. That can lead to an inaccurate split of lungs. In addition, even if we took care of deleting patients with missing lungs or just a tiny portion visible, it is challenging to establish an appropriate threshold to discard samples based on the portion of the visible lung. The radiomic model that used the features extracted from the binary mask of the segmented lung achieved better results than the Deep Learning model that used the bounding box and segmented lung. This questions our previous statement that the bounding box can be more predictive than just the segmented lung. However, the best result of the radiomic model can be caused by overfitting. To assess this potential issue, we need additional data that are not available at the moment of writing. All the calibration plots have shown that the models were not perfectly calibrated on the validation set. The number of samples and data imbalance could be a reason for this issue.

The combined models developed achieved similar or better performance than the previous two approaches looking at the AUC-ROC score. However, the confusion matrices of the different classifiers do not show an improvement in discriminating the positive class but an improvement in the false positive rate. The model that used the averaged probabilities from the Deep Learning model and the radiomic model is the one that achieved the best AUC-ROC score among the other developed solution. In Table 4.6 a performance summary for each best model of each approach is shown.

The absence of annotations and the low sample numbers are the general limitation of our approaches. Both of these problems are caused by the nature of the IIP, as explained in Section 1.1. In addition, Deep Learning and radiomics have their drawbacks. Radiomic features can be extracted from vast regions, as we did with the whole lung area. However, this representation is not accurate when extracted from smaller regions such as tumors or nodules. On the other hand, Deep Learning is not interpretable from the beginning, and fitting the model with big 3D images can easily lead to overfitting problems. This highlights the necessity to introduce biological knowledge of the problem in the design of the models. By doing this, we could build more stable and even interpretable models. If annotations are not provided, a possible solution could be to use voxel-wise extraction to generate the feature maps of specific features and use them in the Deep Learning model to predict IIP. However, voxel-wise extraction is time-consuming and can require days to extract a single feature map from larger regions. Another possible solution is to use a patch-based implementation for the Deep Learning model and use smaller regions to make predictions. The main drawback of this approach is that we will not understand which region is the one driving the prediction. Still, a model able to segment or recognize the most promising regions to distinguish cases of IIP from others is a possible improvement to our work. And use this region to perform the classification. Our work proved that different AI system solutions could aid doctors in detecting cases of IIP in patients with lung cancer. If these models hold to further investigation, they could be integrated into the process and aid doctors in making decisions. Finally, the developed radiomic model achieved comparable results with the only available study in predicting IIP, but with more data available in this study. In addition, these results have been achieved even though radiomics have been extracted from the entire lung mask. The DL model built using a more extended area than the lung achieved slightly inferior results than the radiomic model. However, when it has been combined with the radiomic model, a performance improvement has been achieved.

Chapter 6

Conclusion

In this work, we have explored the possible AI solutions available to detect cases of immunotherapy-induced pneumonitis in patients with lung cancer. The Deep Learning model built in this work was capable of detecting the majority of the IIP patients in our validation set. The performance of the Deep Learning model highlights that useful features can be extracted from the CT scans to detect IIP, answering our first research question. Nevertheless, the radiomic model surpasses the Deep Learning model detecting all the cases of IIP in the validation set and reducing the false positive rate. In addition, this model has preserved its explainability, saving the meaning of the features. This answered our second research question and could be helpful in future work to explore and understand IIP patterns. The combined model presented in the experiment section slightly improved the performance compared to the radiomic and Deep Learning model, becoming the most performing approach developed and answering our third and last research question. It is worth noting that, even if our results showed promising performance, all the models need to be validated and improved using additional data. The data size and the class imbalance nature of the problem are the main limitations of our work. However, we have proved that the radiomic and DL model could be a helpful tool to aid doctors in detecting IIP.

Bibliography

[1] ME Rebecca L. Siegel MPH Mathieu Laversanne MSc Isabelle Soerjomataram MD MSc PhD Ahmedin Jemal DMV PhD Freddie Bray BSc MSc PhD Hyuna Sung PhD, Jacques Ferlay MSc. Global cancer statistics 2020: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA Cancer J Clin.*, 2021 May.

[2] Buhlaiga N Del Rincon SV Papneja N Miller WH Jr. Esfahani K, Roudaia L. A review of cancer immunotherapy: from the past, to the present, to the future. *Curr Oncol. 2020*, 2020.

[3] Zhang B. Twomey, J.D. Cancer immunotherapy update: Fda-approved checkpoint inhibitors and companion diagnostics. *The AAPS Journal*, 2021.

[4] Giovanni Cappello Michela Gabelloni Emanuele Neri Vanina Vani, Daniele Regge. Imaging of adverse events related to checkpoint inhibitor therapy. *Diagnostics 2020*, 2020.

[5] Shannon VR Sheshadri A. Zhong L, Altan M. Immune-related adverse events: Pneumonitis. *Adv Exp Med Biol. 2020*, 2020.

[6] Jumeau R Letovanec I Daccord C Bourhis J Prior JO Peters S Lazor R Beigelman-Aubry C. Pozzessere C, Bouchaab H. Relationship between pneumonitis induced by immune checkpoint inhibitors and the underlying parenchymal status: a retrospective study. *ERJ Open Res*, 2020.

[7] Zhao D. Cai L, Gao J. A review of the application of deep learning in medical image classification and segmentation. *Ann Transl Med. 2020*, 2020.

[8] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen A.W.M. van der Laak, Bram van Ginneken, and Clara I. Sánchez. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60–88, 2017.

[9] Mohammad Farukh Hashmi, Satyarth Katiyar, Avinash G Keskar, Neeraj Dhanraj Bokde, and Zong Woo Geem. Efficient pneumonia detec-

tion in chest xray images using deep transfer learning. *Diagnostics*, 10(6), 2020.

[10] Naik B. Dinesh P. et al. Nayak, J. Significance of deep learning for covid-19: state-of-the-art review. 2021.

[11] Li L Zhang X Zhang X Huang Z Chen J Wang R Zhao H Zha Y Shen J Chong Y Yang Y. Song Y, Zheng S. Deep learning enables accurate diagnosis of novel coronavirus (covid-19) with ct images. *IEEE/ACM transactions on computational biology and bioinformatics*, 2021.

[12] Alex Zwanenburg, Martin Vallières, Mahmoud A. Abdalah, Hugo J. W. L. Aerts, Vincent Andrearczyk, Aditya Apte, Saeed Ashrafinia, Spyridon Bakas, Roelof J. Beukinga, Ronald Boellaard, and et al. The image biomarker standardization initiative: Standardized quantitative radiomics for high-throughput image-based phenotyping. *Radiology*, 295(2):328–338, May 2020.

[13] Velazquez E. Leijenaar R. et al. Aerts, H. Decoding tumour phenotype by noninvasive imaging using a quantitative radiomics approach. *Nat Commun*, 2014.

[14] M.; Zhovannik I. ; Welch M.; Wee L.; Jaffray D.; Dekker A.; Hope A. Traverso, A.; Kazmierski. Machine learning helps identifying volume-confounding effects in radiomics. *Physica Medica-European Journal of Medical Physics*, 71:24–30, 2020.

[15] McIntosh C. Haibe-Kains B. Milosevic M. F. Wee L. Dekker A. Huang S. H. Purdie T. G. O'Sullivan B. Aerts H. Jaffray Welch, M. L. Vulnerabilities of radiomic signature development: The need for safeguards. *Radiotherapy and oncology : journal of the European Society for Therapeutic Radiology and Oncology*, 130:2–9, 2019.

[16] Fujii T. Bilen M.A. et al. Colen, R.R. Radiomics to predict immunotherapy-induced pneumonitis: proof of concept. 2018.

[17] Khoshgoftaar T.M. Johnson, J.M. Survey on deep learning with class imbalance. *Journal of Big Data*, 2019.

[18] Girshick RB He K Dollár P. Lin T-Y, Goyal P. Focal loss for dense object detection. *IEEE international conference on computer vision (ICCV)*, 2017.

[19] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. 2017.

[20] Noora Shrestha. Detecting multicollinearity in regression analysis.t. *American Journal of Applied Mathematics and Statistics*, 2020.

Chapter 7

Supplementary Material

7.1 Deep Learning Metadata Analysis

As mentioned in the dataset section, the CT images have been acquired from different centers in the Nethelerand, thus with different scanner and settings. After we trained the final Deep Learning model we have analyzed the distribution of KVP and Slicer Thickness value in correct and wrong classified examples in the validation set. The distribution of this two metadata on the correct and wrong examples are shown in Figures 7.1 and 7.2

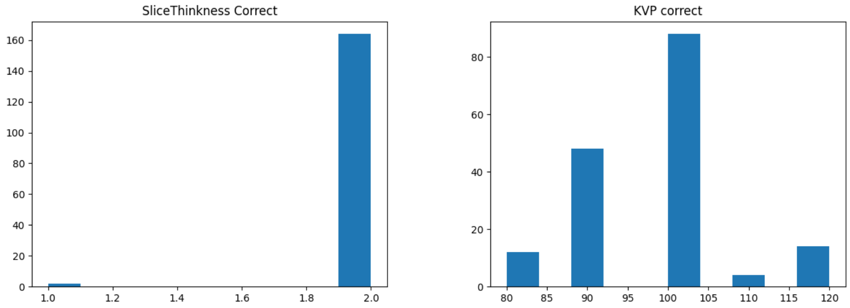
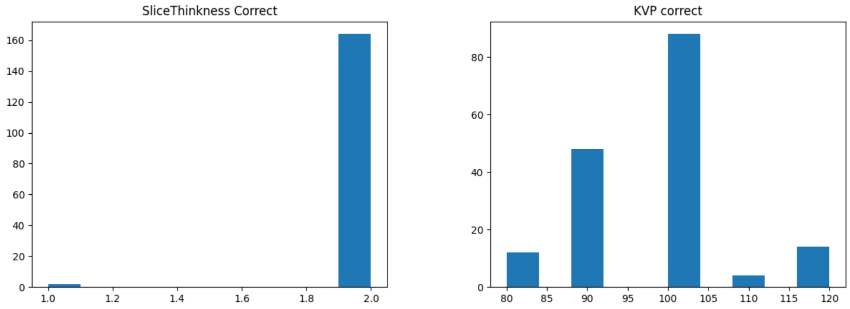


Figure 7.1: Distribution of Slice Thickness and KVP metadata within the correct classified examples.


SliceThinkness Correct

KVP correct

Figure 7.2: Distribution of Slice Thickness and KVP metadata within the wrong classified examples.

7.2 Deep Learning Model Confusion Matrices

The confusion matrices for the models of the DenseNet-121 experiment with segmented lung and bounding are shown in s 7.3 and 7.4. Instead the confusion matrices for the Customized DenseNet with the segmented lung and the bounding box are shown in Figures 7.5 and 7.6.

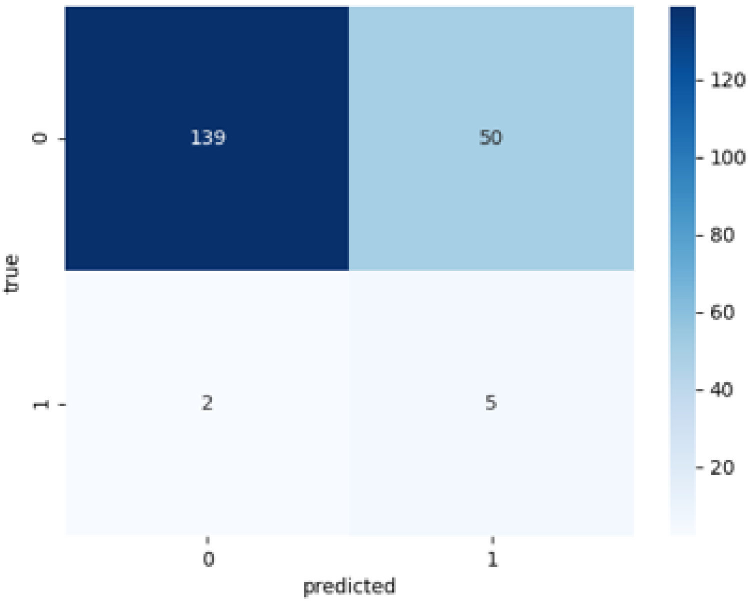


Figure 7.3: Confusion matrix for DenseNet-121 experiment with segmented lung.

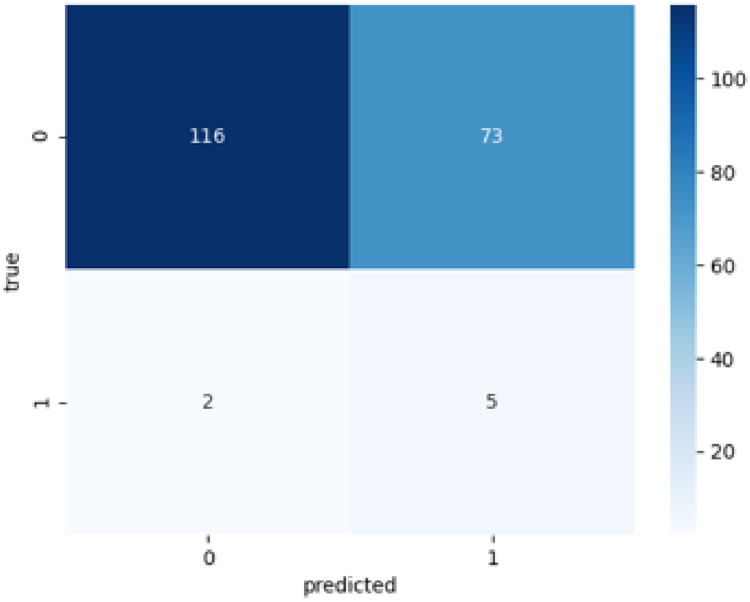


Figure 7.4: Confusion matrix for DenseNet-121 experiment with bounding box around the lung.

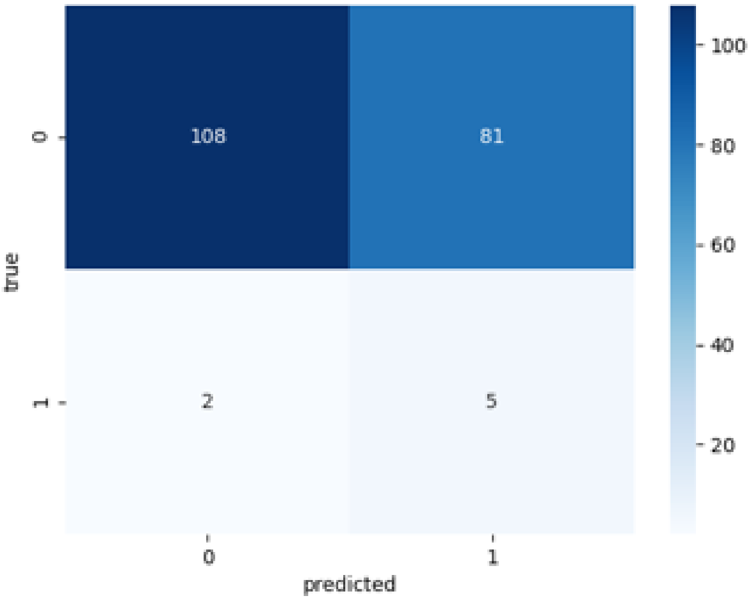


Figure 7.5: Confusion matrix for Customized DenseNet experiment with segmented lung.

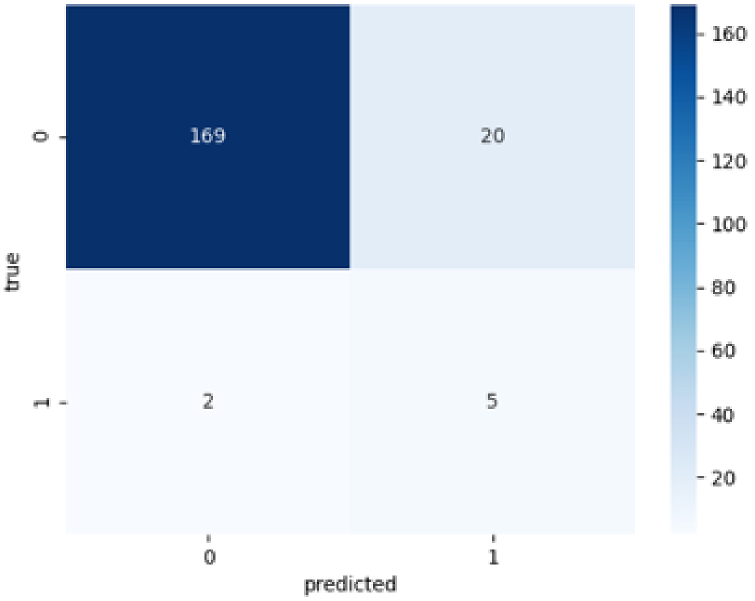


Figure 7.6: Confusion matrix for Customized DenseNet experiment with bounding box around the lung.