X-Ray Images Classification of Tuberculosis (TB), Pneumonia, and Normal Cases

Applied AI in Biomedicine report

Francesco Panebianco, francesco.panebianco@mail.polimi.it, 10632465 *Kiarash Rezaei, kiarash.rezaei@mail.polimi.it, 10809307*

I. Introduction

Chest X-ray is a useful test for the diagnosis of Pneumonia[1] but much less reliable for pulmonary tuberculosis[2]. Still, models that can lead to a timely diagnosis can prevent severe consequences, such as respiratory failure and death. But analyzing X-ray images can be difficult since it takes skill and knowledge to distinguish between normal and pathological cases. Automatic classification can be a scalable tool to ease the burden of the task and also provide more consistent results by reducing the variability of the human observer. In this report, we elaborate on the different preprocessing steps necessary for extracting meaningful features from the given dataset, together with different feature extraction methods and classification techniques. At the end, we evaluate the performance of models as well as the interpretability of the best models by using explainable AI techniques.

II. Material and Methods

A. Dataset Exploration

The given dataset consists of 15470 X-ray images of different sizes belonging to patients (not necessarily unique) in three classes: normal, pneumonia, and tuberculosis.

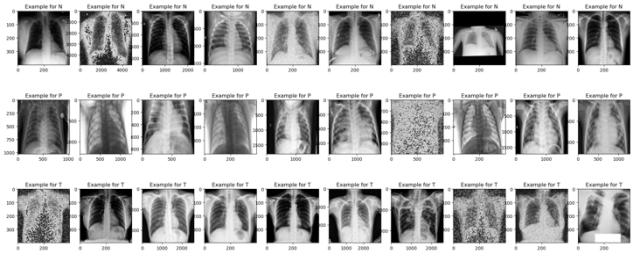


Figure 1. Examples of images belonging to the 3 classes

While exploring the dataset, we observed the following main issues: the dataset had an imbalanced class distribution, and the sizes of the images were different. Regarding the background, some of them were not compliant with the standard radiological format (had white backgrounds), and the dataset contained several noisy images with different levels of artificial random noise and duplicated images.

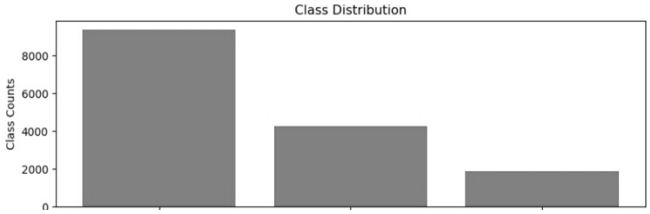


Figure 2. Class distribution

B. Preprocessing

To tackle the aforementioned issues, we used several preprocessing approaches together with some image processing techniques such as background inversion, mist reduction, gamma transformation, adaptive contrast equalization, and Gaussian smoothing. Firstly, we managed to detect the white background images by checking the median color along the spine, and then for white background cases, the color was inverted. A common problem associated with radiographic images is the appearance of a haze-like effect caused by environmental artifacts. Decreasing this mist effect and using the gamma transformation resulted in an improvement in contour definition and the contrast of all the images. To further enhance the contrast and detail visibility we applied Contrast-Limited Adaptive Histogram Equalization (CLAHE). Consequently, utilizing the gaussian smoothing filter provided us with cleaner and clearer images by decreasing the natural noise effect on the images. The final processed images have then been reshaped to a size of 512 by 512. It should be mentioned that we dealt with the imbalance in class distribution in different ways, as explained in the next sections.

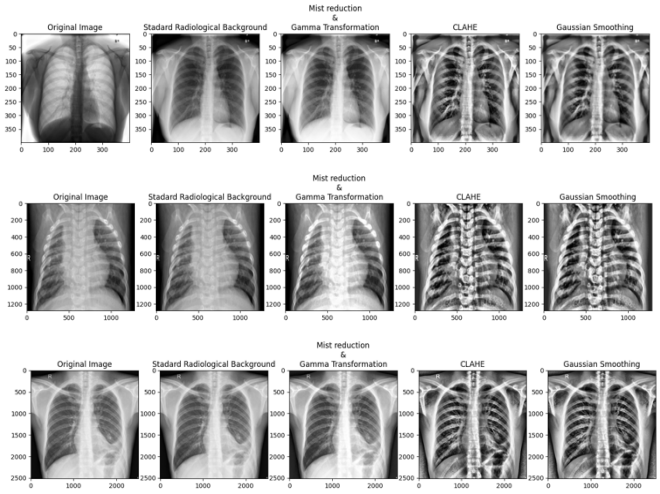


Figure 3. Image processing steps for normal, pneumonia, and tuberculosis classes

Afterwards, we tried to eliminate the noisy images by performing Laplacian variance analysis for each class. Since images containing noisy parts have more

variations in terms of the Laplacian operator (a mathematical tool used to detect changes in intensity in an image), we expected to have higher values for the noisy ones. After setting a proper threshold (=1000) on the Laplacian variance distribution corresponding to the dataset, we could eliminate a few noisy images.

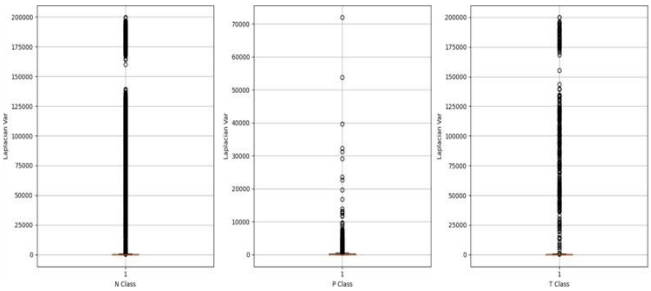


Figure 4. Laplacian variance box plot per class

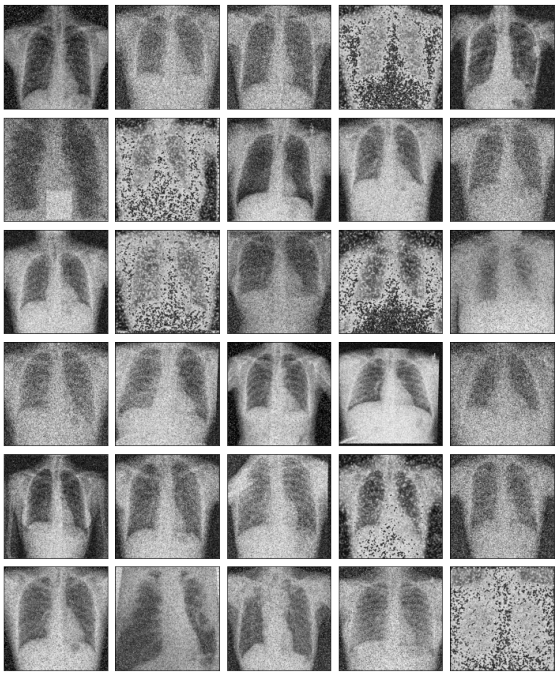


Figure 5. Examples of noisy images with Laplacian variance values greater than 150000

However, further inspection showed that such a filtering criterion could not eliminate all the noisy images, and the resulting dataset still contained some of them with different characteristics of noise. Thus, we manually labeled the dataset with 4 quality labels, then only kept the ones with the highest quality (labeled as 0), and among them removed the duplicated ones (if any). In this scenario, we reduced the size of the dataset from "15470" to "10792". To have a more unbiased estimate of the unseen test set, we divided the dataset into 3 parts, 20% of the data were considered the local test set, and the rest of the data were divided into 80% and 20% for training and validation respectively. The training data used for training the model. The hyperparameter tuning procedure was done using validation data, and finally, the best results were considered for testing on the local test set.

C. *Approaches*

We decided to approach the problem from three different perspectives. The first (and main) approach was to extract the features via deep convolutional neural networks (CNN) with a classification head at the end. We started to design a model from scratch and then utilized transfer learning and fine-tuning techniques on state-of-the-art pre-trained ImageNet models. The second approach was to use hand-crafted features in the spatial domain, and the third was to extract hand-crafted features from the Fourier domain. In the last two approaches, we used linear, kernel-based, and ensemble classifiers.

D. *Feature extraction with CNN models*

D.1. CNN from Scratch

As the starting point, we designed a CNN from scratch composed of a total of nine convolutional and Max pooling layers, as below. We performed the training process by choosing Adam with a learning rate of 1×10^{-3} and the Sigmoid Focal Cross Entropy as loss function.

The results were quite promising, as we obtained an accuracy

Layer (type)	Output Shape	Param #
conv2d_14 (Conv2D)	(None, 512, 512, 32)	320
max_pooling2d_10 (MaxPooling2D)	(None, 256, 256, 32)	0
conv2d_15 (Conv2D)	(None, 256, 256, 64)	18496
max_pooling2d_11 (MaxPooling2D)	(None, 128, 128, 64)	0
conv2d_16 (Conv2D)	(None, 128, 128, 128)	73856
max_pooling2d_12 (MaxPooling2D)	(None, 64, 64, 128)	0
conv2d_17 (Conv2D)	(None, 64, 64, 256)	295168
max_pooling2d_13 (MaxPooling2D)	(None, 32, 32, 256)	0
conv2d_18 (Conv2D)	(None, 32, 32, 512)	1180160

...

Total params: 3,929,347

Trainable params: 3,929,347

Non-trainable params: 0

Figure 6. The first CNN model architecture

of "92.8", an F1-score of "0.81" for the tuberculosis class, and "0.90" as the overall F1-score.

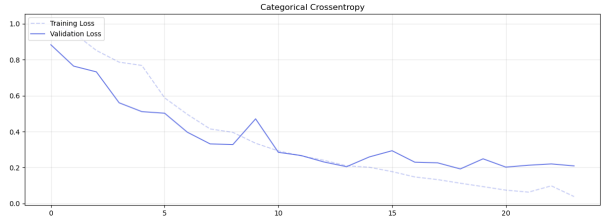


Figure 7. Training and validation loss plot

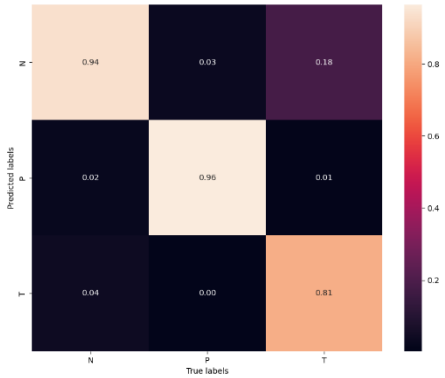


Figure 8. The first CNN model confusion matrix

We decided to proceed with deeper and more complex CNN models that could learn more discriminative features. Therefore, using state-of-the-art models was considered to be a viable option.

D.2. Transfer learning and augmentation techniques

In this section, we focus our analysis on the first two pre-trained models that demonstrated the best performance and a shallower network as the baseline model. While we provide quantitative measures for these three models, it is important to note that we have experimented with several other pre-trained models as well. However, for the sake of brevity and clarity, we will only list the other models we have explored without providing detailed quantitative evaluations.

The following table corresponds to all networks we tried to use and the reasons we did/did not proceed further with them.

no	name	Selected	Reason
1	ResNet50	No	Low performance
2	ResNet152	No	Low performance
3	EfficientNet B5	No	Serialization and library compatibility issues with Google Colab and Kaggle
4	MobileNet	Yes	As a baseline model
5	MobileNetV2 and V3	No	Low performance
6	DenseNet121	Yes	good performance
7	DenseNet169	Yes	good performance
8	DenseNet201	No	Memory limitation

Table 1. Model selection

Our first aim was to find the best type of architecture among the pre-trained models able to extract as many informative features as possible without facing common deep learning problems such as gradient vanishing, dying neurons, etc., as well as considering GPU and memory constraints. Therefore, we started with MobileNet having one of the most lightweight architectures as a baseline. Another reason was its decent performance based on a recent research paper [5] that was similar to this task. As the next step, we decided to examine the effect of having skip connections on the overall performance by benefiting from the residual structures of ResNet models. However, due to poor results, we stopped exploring them. Lastly, we chose DenseNet to examine how dense connectivity patterns can affect the classification abilities of the classifier. At this point, our choices in models to train were not limited to the peculiarities of their architecture, but also incorporated data augmentation techniques. Generally, augmentation techniques, such as rotation, scaling, and zooming, can effectively mitigate the class imbalance and increase the diversity of the training dataset, thereby improving the model's ability to generalize to the unseen test set. Another important factor is selecting a proper loss function. In this regard, we chose Sigmoid Focal Cross Entropy, as it seemed to effectively handle imbalanced datasets by assigning higher weights to misclassified examples from the minority classes, thus improving model performance in the underrepresented labels[3]. To further avoid overfitting, we ultimately opted for the early stopping technique. We initialized the learning rate value at 1×10^{-4} and chose Yogi as the optimizer algorithm[4]. Moreover, the classifier head was composed of global average pooling (GAP), dropout, and dense layers.

For each model, we performed the training process using a combination of freezing and fine-tuning of the weights of certain layers and gathered their best performances. The results are summarized in the table below.

Model	Accuracy	F1-score	Tuberculosis F1-score	augmentation
Scratch	0.93	0.90	0.80	No
MobileNet	0.88	0.81	0.62	Yes
DenseNet 121	**0.97**	**0.95**	**0.92**	**No**
DenseNet 121	0.94	0.92	0.85	Yes
DenseNet 169	0.97	0.95	0.90	No
DenseNet 169	0.95	0.93	0.86	Yes

Table 2. Deep learning model Performance

As Table 2 shows, DenseNet models outperform the other structures. It has been observed that dense connectivities, as implemented in DenseNet models, tend to be more effective in extracting informative features from these images. The following plots show that, for DenseNet models, augmentation does not improve performance.

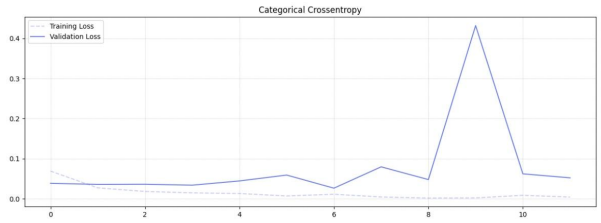


Figure 9. Training and validation loss plot of DenseNet121 without augmentation

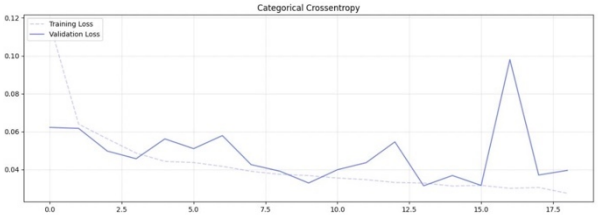


Figure 10. Training and validation loss plot of DenseNet121 with augmentation

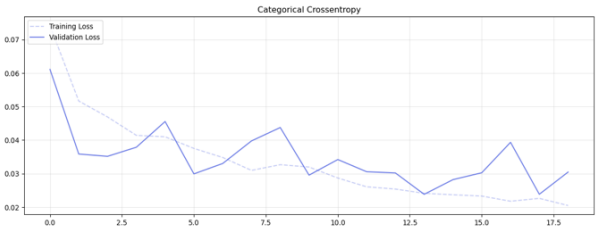


Figure 11. Training and validation loss plot of DenseNet169 without augmentation

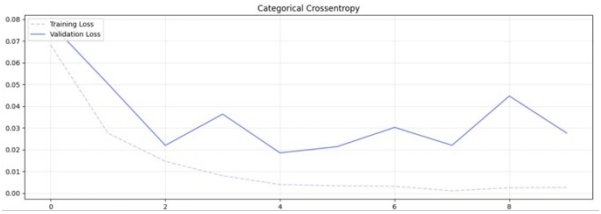


Figure 12. Training and validation loss plot of DenseNet169 with augmentation

However, it is noteworthy that augmentation techniques increased the localization ability of models because they were less prone to capturing noise. As a result, their performance was better while performing GradCam as an explainable AI technique, which will be explained in detail in its corresponding section.

The following confusion matrices belong to the best DenseNet121 and DenseNet169 models, respectively.

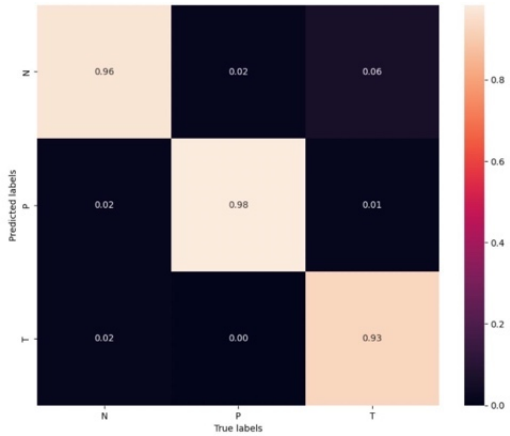


Figure 5. DenseNet121 (without augmentation) confusion matrix

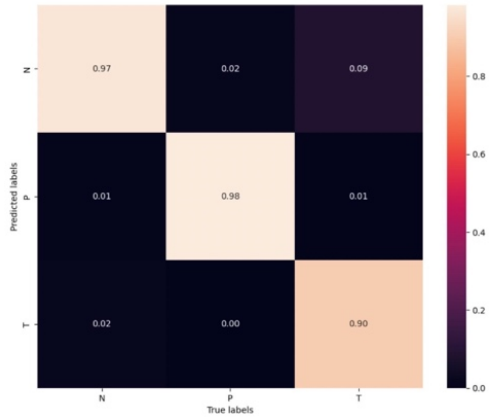


Figure 14. DenseNet169 (without augmentation) confusion matrix

To further improve the performance of the DenseNet121 model, which has the best performance, we decided to use an attention mechanism. We used a multi-head attention layer after the pre-trained model. However, the performance dropped, as its confusion matrix shows below.

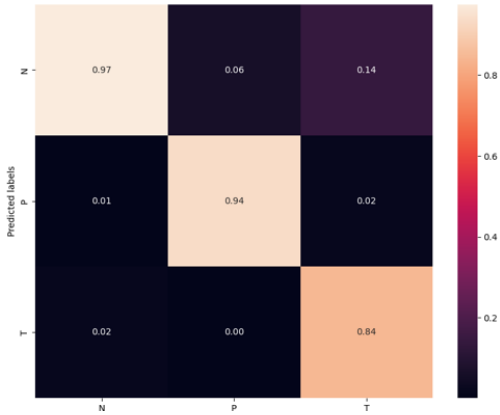


Figure 15. DenseNet121 (without augmentation) with multi-head attention layer

E. Hand-crafted features

In this section, we focus on traditional computer vision techniques for extracting features from images from a statistical point of view.

E.1. spatial features

We tried to use the most relevant features that can represent the texture and spatial dependencies within the images of our dataset. Then their statistical properties were calculated. We used four categories of features namely intensity, local binary pattern (LBP), gray level co-occurrence matrix (GLCM), and wavelet based. In the following, we briefly explain the rationale behind choosing these features. With Local Binary Pattern (LBP), we can capture patterns such as edges, corners, and texture variations by comparing the intensity of each pixel with its neighboring pixels. In terms of obtaining information about the spatial relationships

between pixels, the Gray Level Co-occurrence Matrix (GLCM) was chosen since it calculates the co-occurrence of gray-level values in an image and reveals statistical aspects such as contrast, energy, and entropy. Finally, wavelet-based features can help analyze the image in terms of texture, edges, and smoothness by decomposing it into different frequency components. However, since handling them all at once was not ideal, in light of the curse of dimensionality and potentially confusing explanation of predictions, we used each group separately. As the first group, we extracted the statistical characteristics of intensity.

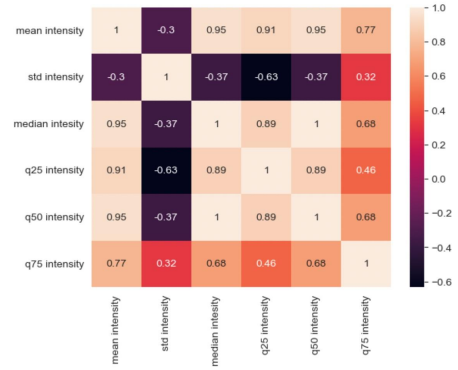


Figure 16. intensity features correlation matrix

As can be seen, only a few features are not correlated with each other. In the next step, we analyzed the feature space in terms of separability among the classes to choose the best classifier.

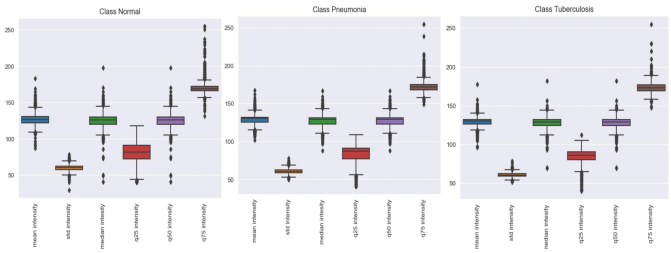


Figure 17. intensity feature space boxplot for each class

The box plot shows that the distribution of features with respect to intensity is approximately the same for all three classes. Therefore, the classes cannot be easily separated in this feature space. Thus, we chose a ensemble classifier such as random forest (RF). As was expected, the performance was poor with an accuracy of 0.62 and an overall F1-score of 0.46, especially for tuberculosis detection, which has an F1-score of 0.20, and the model tends to capture the noise.

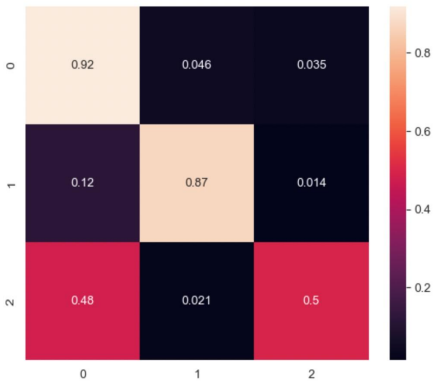


Figure 18. RF confusion matrix - intensity features

Considering local binary patterns (LBP), we construct the feature space by computing the LBP histogram for 12 different points in each image using a neighborhood radius of 5 pixels. The histogram results in 14 bins.

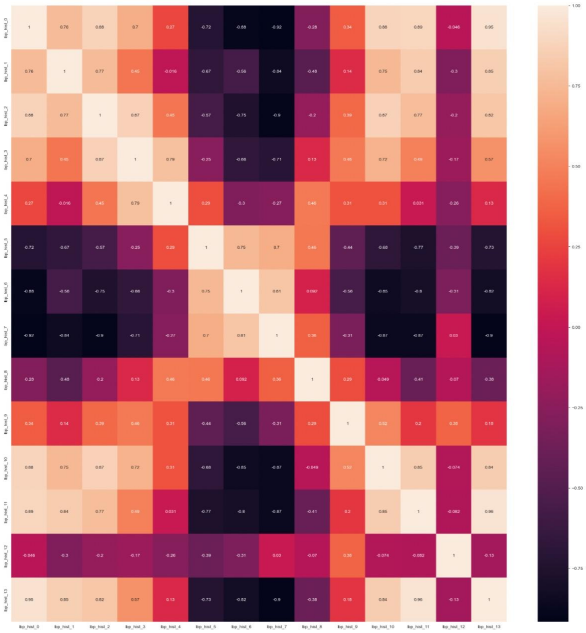


Figure 19. LBP features correlation matrix

To deal with the curse of dimensionality and redundant features, we tried to reduce the size of the feature space by either using principal component analysis (PCA) or the Chi-square statistical test to score each feature with respect to the categorical target and select the most informative features. PCA could not provide us with good components, but with Chi-square scoring, we have chosen the first three highest scoring features, which were the 6th, 12th, and 13th ones. Performing classification with RF resulted in the confusion matrix below.

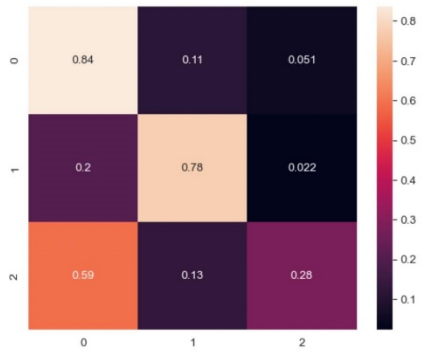


Figure 20. RF confusion matrix -LBP features

Again, RF could not perform well, as we obtained 0.75 as the accuracy, F1-score of 0.64, and the tuberculosis F1-score equals to 0.35.

Hence, we proceeded with the next group of features, namely the Gray Level Co-occurrence Matrix (GLCM). Firstly, the GLCM is calculated for the pair of pixels within each image with a distance of 1 and 2 pixels both vertically and horizontally. Then, several properties were extracted from each matrix. Specifically, the contrast value shows the local intensity variation, dissimilarity depicts the difference in pixel intensity between neighboring pixels, homogeneity measures how the distribution of elements is close to the diagonal distribution, and angular second moment (ASM) provides us with a uniformity measurement of pixel distribution, energy, and correlation between neighboring pixels. Overall, a total of 24 features have been extracted.

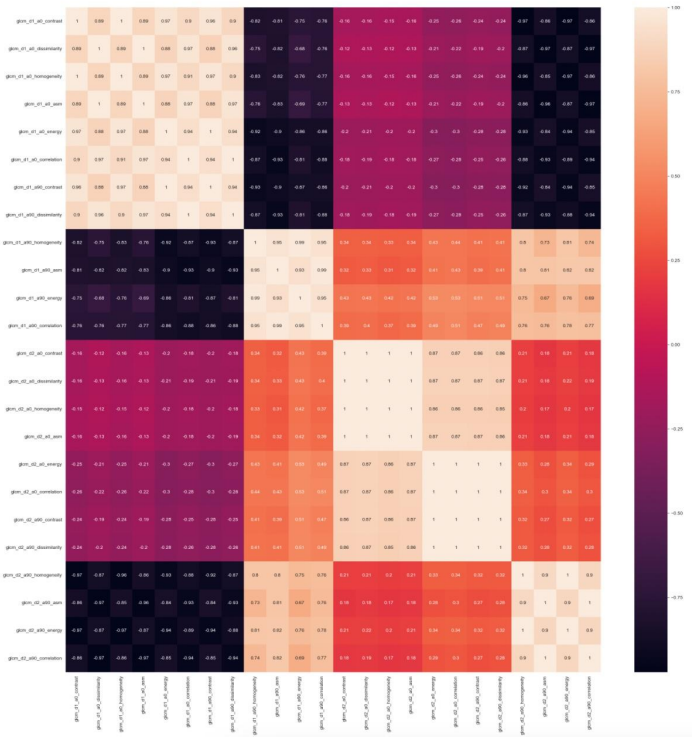


Figure 21. GLCM features correlation matrix

As the definition of features and their correlation matrix show, a lot of them are highly correlated. Therefore, we selected the 5 most informative features based on the Chi-square statistical test and performed the classification task using a RF as the classifier.

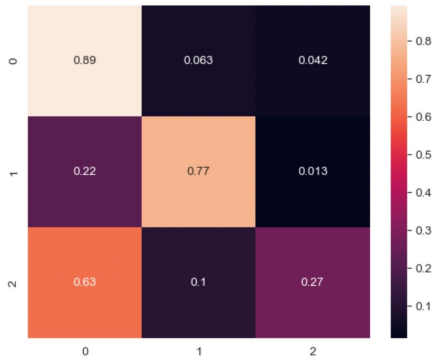


Figure 22. RF confusion matrix -GLCM features

With this model, accuracy, overall F1-score slightly increased to 0.79 and 0.66, respectively, while the F1-score for the tuberculosis class remained the same.

Lastly, we explored wavelet-based features. For this purpose. We decomposed each image into different frequency sub-bands using the discrete wavelet transform (DWT) function. The process consists of applying a set of high-pass and low-pass filters to the image to extract high-frequency details such as edges and low-frequency components of the image. The result is a set of coefficients representing the original image at multiple resolutions. More specifically, the DWT coefficients in our case include the approximation coefficients (cA) and the detailed coefficients (cH, cV, and cD) that represent the coarse-scale (low-frequency) and fine-scale (high-frequency) details in horizontal, vertical, and diagonal orientations. To practically use the coefficients as features, the statistical properties such as mean, variance, kurtosis, skewness, energy, and entropy belonging to each coefficient matrix have been extracted. After dropping features with irrelevant values (NaN and infinity), we obtained 21 features.

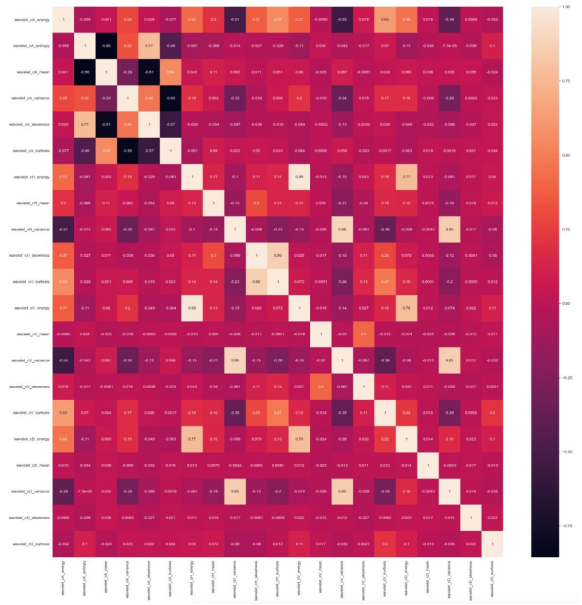


Figure 23. Wavelet-based feature correlation matrix

Correlation analysis shows these features can be more informative since they are less correlated with each other in comparison to the previous cases. To analyze the class separability in the current feature space, we decided to visualize it in a reduced three-dimensional space using t-distributed stochastic neighbor embedding (t-SNE).

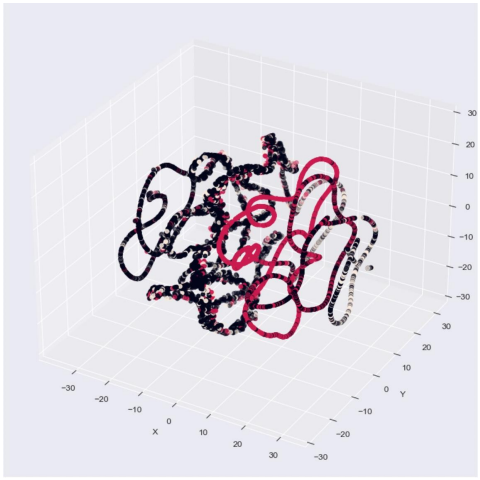


Figure 24. t-SNE reduced feature space plot

The snake-shape of the t-SNE plot implies that the classes share the same values for most of the features; therefore, they cannot be easily distinguished. Similar to the previous cases, we tried to reduce the number of features. As PCA analysis of the first 10 had a low cumulative fraction of explained variance, we opted for statistical test feature selection again. We selected the best 5 features using mutual information as the metric. The reason we used a different criterion was that there were some negative values, thus Chi-square could not be used. Ultimately, we performed the classification with an RF.

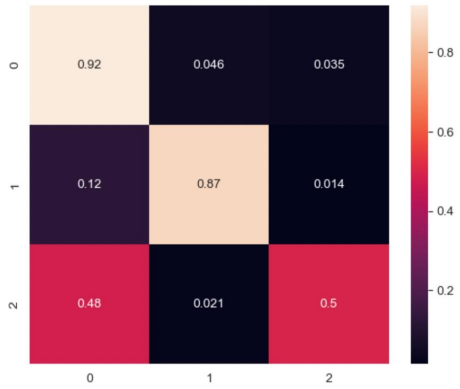


Figure 25. RF confusion matrix -wavelet-based features

Although this model is still not comparable with deep learning models, especially in tuberculosis detection, the performance has been significantly improved compared to the previous models on hand-crafted features. Having an accuracy of 0.85, an overall F1-score of 0.78, and a Tuberculosis F1-score equal to 0.59 suggests that wavelet-based features are the most informative ones so far. Hence, we continued processing and analyzing them to check for better results. After normalizing features separately, we checked their distributions for each class through the following boxplots.

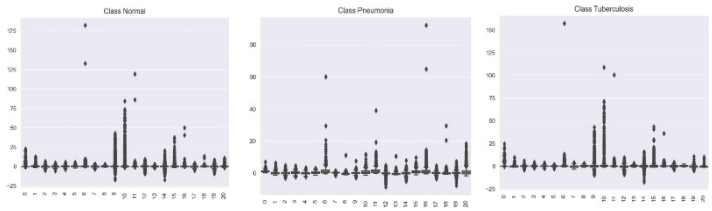


Figure 26. wavelet-based feature space boxplot for each class

It is observed that for each class, we have different variations of outliers. To avoid their effects on the classifier and to have a more robust and reliable representation of the data, we performed Windsorization as a preprocessing step in the current feature space. In other words, we replaced 10% of the high and low extreme values with less extreme ones, which resulted in modifying each class distribution as below.

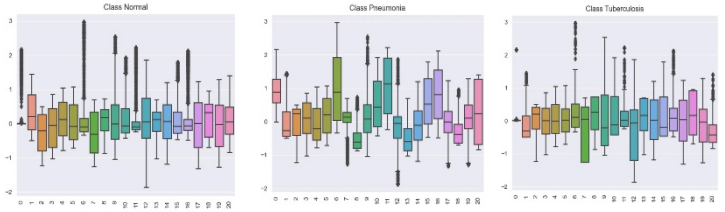


Figure 27. wavelet-based feature space boxplot for each class after Windsorization

With regards to the classification task, we ended up choosing a support vector machine classifier (SVC) with a radial basis kernel function. Moreover, classes were weighted at 0.8, 1, and 3.2, respectively.

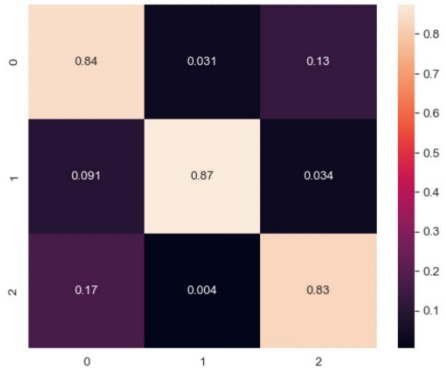


Figure 28. SVC confusion matrix - Preprocessed wavelet-based features

With this configuration, we could increase the accuracy up to 0.85, the overall F1-score to 0.81, and the tuberculosis F1-score to 0.64.

For the sake of explainability, which will be discussed in detail in its devoted section, we also decided to use a natively explainable model called explainable boosting machine (EBM) and analyze its performance in the current feature space.

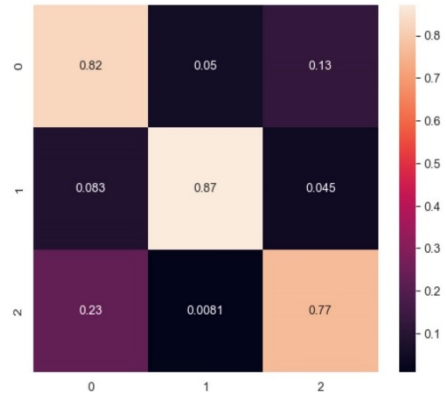


Figure 29. EBM confusion matrix - Preprocessed wavelet-based features

With EBM, the accuracy, overall F1-score, and tuberculosis F1-score slightly decreased to 0.83, 0.78, and 0.60, respectively. However, considering all aspects, especially its explainability, EBM might be a more reasonable choice.

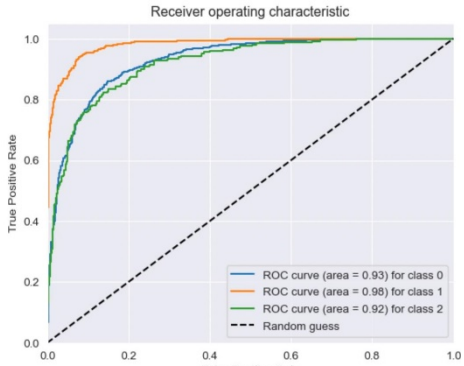


Figure 30. EBM ROC curve

The area under the curve (AUC) of the receiver operating characteristic (ROC) plot shows that for the pneumonia class, we have a bigger area than the other two classes. It means that the EBM model performs better in this class. The ROC curve natively works on binary classifiers, however, since multiclass EBM model classifies in a one-to-rest manner, the plot shows one curve for each one-to-rest classifier.

E.2. Fourier features

Images as 2D signals can be analyzed in the Fourier domain. Fourier features can reveal a number of information such as frequency information, texture and, etc. After searching for different features to be extracted, we ended up choosing the following features as the most relevant ones to this task: mean, variance, skewness, kurtosis, spectral moments in the range of 2 to 5, entropy, and the Gabor feature. We will elaborate more on the Gabor feature later. However, it should be noted that due to the characteristics of the images, they comprise very high and very low frequency components in the spectral domain (plus infinite and minus infinite). To avoid problems in the feature extraction process, we estimated the highest and lowest frequency components with the maximum and minimum non-infinite Fourier coefficients within each image.

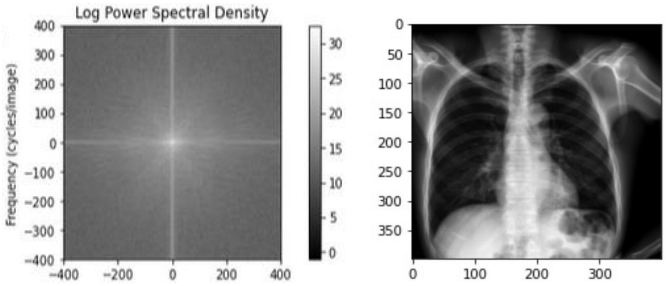


Figure 31. An example of image in Frequency and spatial domain

In general, the Gabor filter is a linear filter that is widely used in image processing for object recognition and texture analysis. By convolving a Gabor filter with an image in the Fourier domain, we can extract relevant frequency information. Here, we first defined a set of Gabor filters with the same frequency (=0.2) and different bandwidths and orientations in the range of 0 to Pi and 0 to log (8), respectively. By convolving each image with Gabor filters and calculating mean values for each filter response, we obtained a vector of Gabor features (size = 43). Lastly, we constructed a feature vector of size 49 for each image and normalized it.

With regards to the classification task, we examined random forest (RF), support vector machine classifier (SVC), and extreme gradient boosting (XGBoost). In this phase of the project, we aimed to assess the impact of the classifiers' hyperparameters on the performance because the number of extracted features on the Fourier domain are much less than the spatial domain, and we used all of them together. For RF, we have used 250 estimators with a maximum depth of 50, as

we did not witness any improvements by increasing the value of parameters. With an RF classifier and the mentioned configuration, we could get an accuracy of 0.84, an overall F1-score of 0.72, and the tuberculosis F1-score of 0.42.

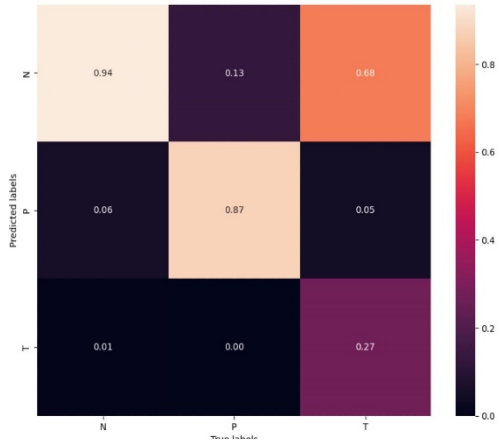


Figure 32. RF confusion matrix - Fourier features

With regards to SVC, we obtained the best performance by setting the regularization term to 30 as well as using a radial basis function (RBF) as the kernel.

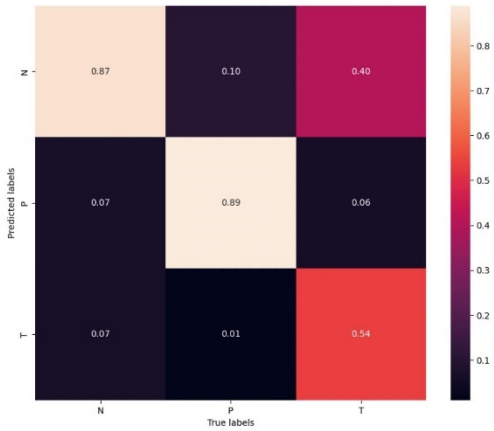


Figure 33. SVC confusion matrix - Fourier features

Using SVC as the classifier resulted in improvements in both the overall F1-score (=0.77) and the tuberculosis F1-score (=0.57). However, the accuracy slightly decreased to 0.83 due to mis-classification in the normal class.

For XGBoost, we used 400 decision trees with a maximum depth of 10 on 80% of the training set at each boosting round. Moreover, 80% of features have been chosen randomly for each tree, and the learning rate has been set to 0.1.

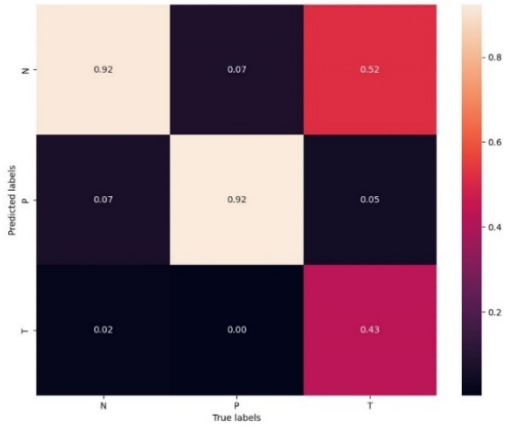


Figure 34. XGBoost confusion matrix- Fourier features

With the XGBoost classifier, we could increase the overall F1-score to 0.78 and accuracy to 0.86 as the tuberculosis F1-score stayed the same (=0.57).

As can be seen, XGBoost has the best performance among other models in the Fourier domain. However, its performance in detecting tuberculosis is still not comparable with that of deep learning models.

The following table is a summary of the performance of all models using hand-crafted features (both spatial and Fourier domains).

Model	Feature Domain	Feature Space	Accuracy	F1-score	Tuberculosis F1-score	class weighting
RF	Spatial	Intensity	0.62	0.46	0.20	No
RF	Spatial	LBP	0.75	0.64	0.35	No
RF	Spatial	GLCM	0.79	0.66	0.35	No
RF	Spatial	Wavelet	0.85	0.78	0.59	No
SVC	**Spatial**	**Wavelet - Processed**	0.85	**0.81**	**0.64**	**Yes**
EBM	Spatial	Wavelet - Processed	0.83	0.78	0.60	Yes
RF	Fourier	-	0.84	0.72	0.42	No
SVC	Fourier	-	0.83	0.77	0.57	Yes
XG Boost	Fourier	-	**0.86**	0.78	0.57	Yes

Table 3. Performance of models on handcrafted features

F. Explainable AI

In this part, we perform various explainable AI (XAI) techniques to validate the results obtained by the best models, using both handcrafted and deep learning approaches.

F.1 Gradient Weighted Class Activation Mapping (Grad-CAM)

Grad-Cam[6] is one of the most common explainable AI (XAI) techniques when dealing with images and CNN models. We have performed Grad-Cam on DenseNet models, as they outperformed the other ones. In Grad-CAM visualization, the heatmap is generated by assigning higher values to the pixels that exhibit the strongest activation in the model's final convolutional layer. These activations indicate the areas of the input image that contribute most significantly to the model's prediction. In the following figures, we have chosen three random samples from each class that have been predicted correctly by the model. Then we examined the effect of augmentation techniques on each model by comparing their heatmap results.

F.1.1 Grad-CAM for DenseNet121

In figures 34 to 37, we presented Grad-CAM visualization for the DenseNet121 model without and with augmentation, for the normal, pneumonia and tuberculosis classes, respectively.

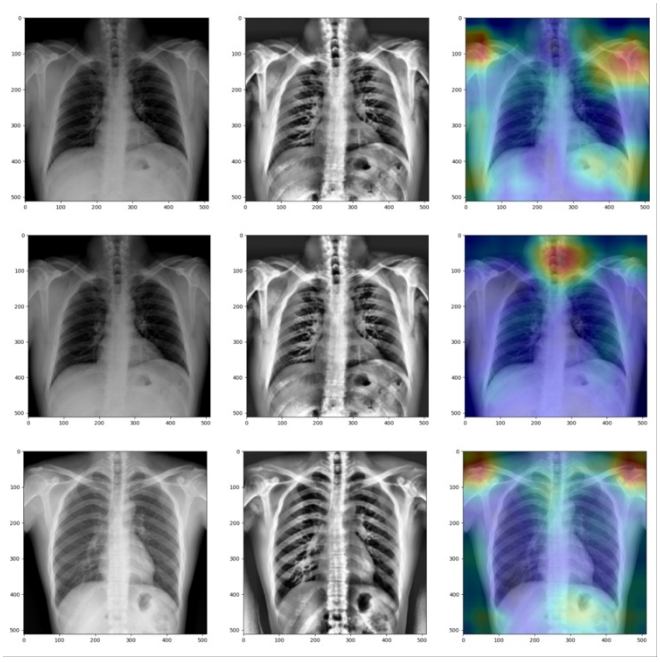


Figure 34. Images of normal class with their corresponding processed and heatmaps - DenseNet121 without augmentation

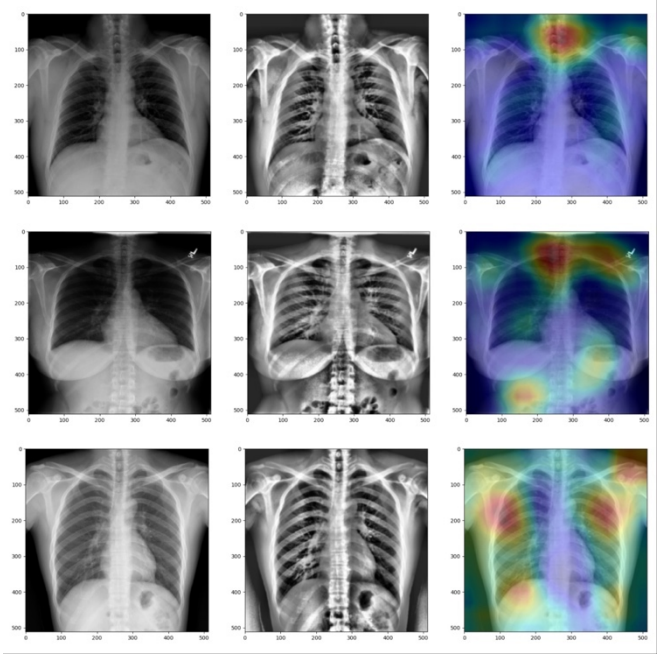


Figure 35. Images of normal class with their corresponding processed and heatmaps - DenseNet121 with augmentation

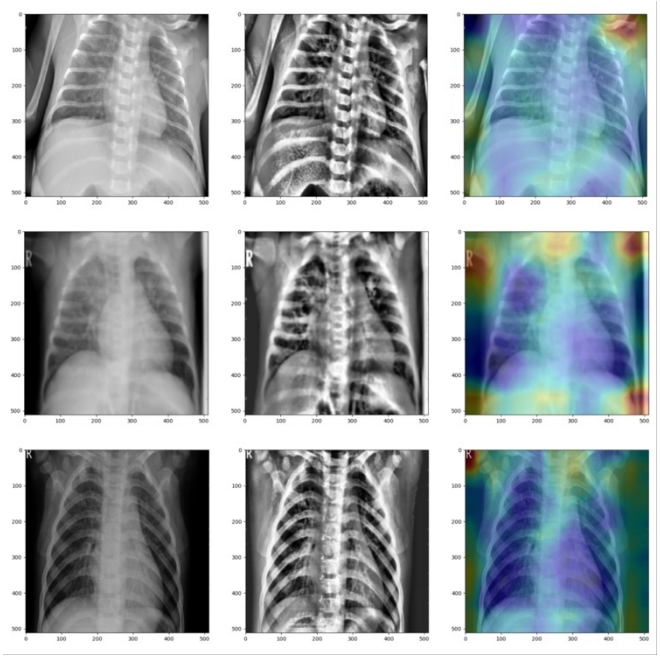


Figure 36. Images of pneumonia class with their corresponding processed and heatmaps - DenseNet121 without augmentation

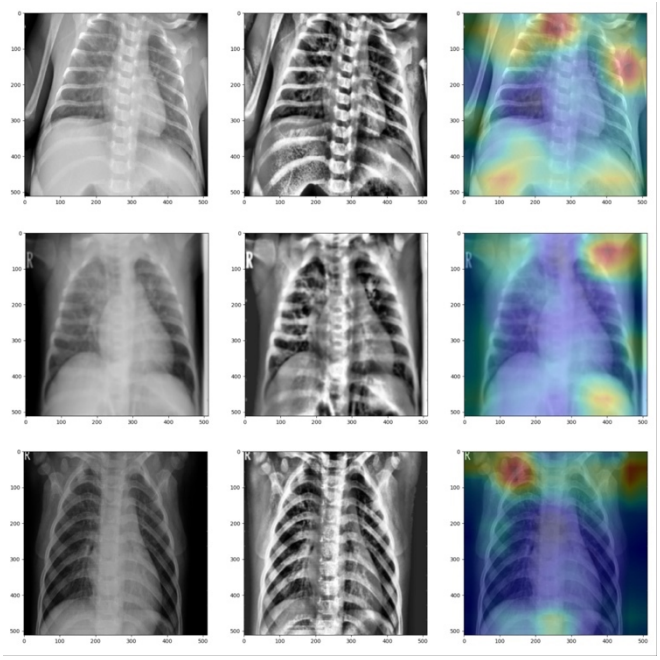


Figure 37. Images of pneumonia class with their corresponding processed and heatmaps - DenseNet121 with augmentation

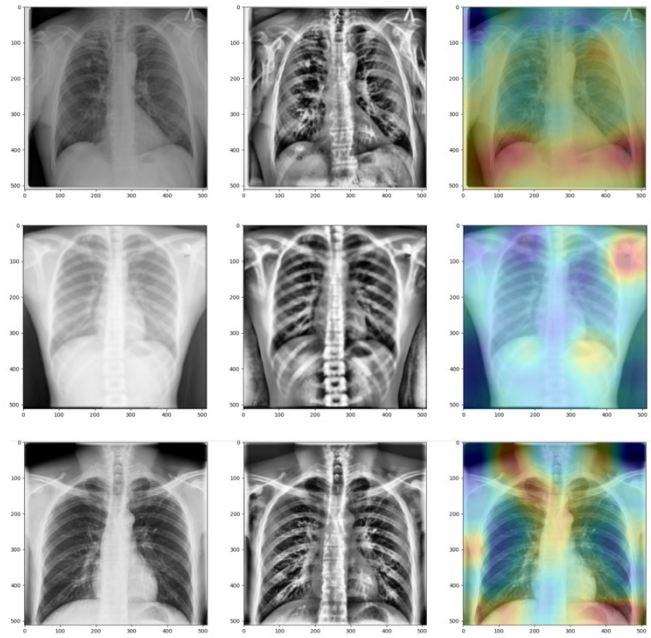


Figure 38. Images of tuberculosis class with their corresponding processed and heatmaps - DenseNet121 without augmentation

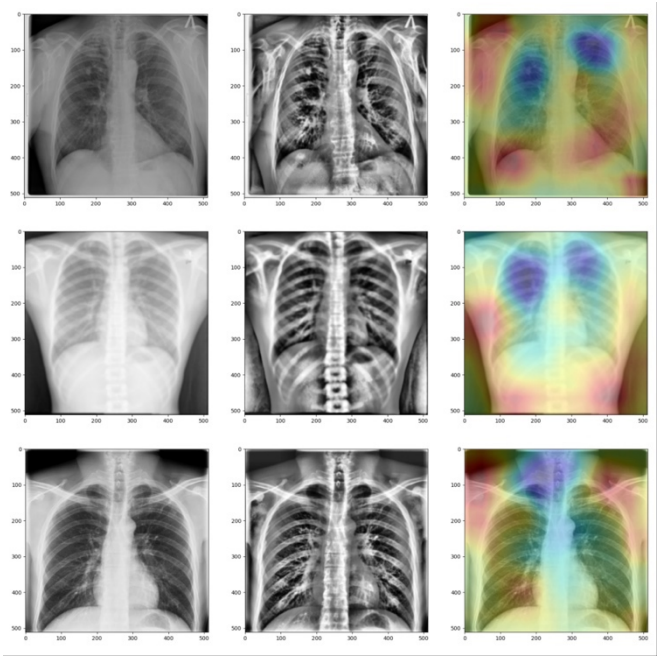


Figure 39. Images of tuberculosis class with their corresponding processed and heatmaps - DenseNet121 with augmentation

Figure 34 shows that despite the great performance of DenseNet121 with an accuracy of 0.97 and an F1-score of 0.97 for normal class, it did not use meaningful areas (mostly background) to extract features and performing classification, as the most activated neurons referred to the upper right and left pixels rather than the ones close to lungs. On the other hand, Figure 35 shows with augmentation techniques, we could slightly improve the region of interest; even though, considering the performance, we witnessed a drop of 3% and 2% in accuracy and F1-score, respectively.

Based on visualization of the figure 36, same to the normal class, for the pneumonia class, the model did not use relevant regions of interest. Moreover, in figure 37 we can see that even augmentation techniques did not improve it considerably.

As the figures 38 and 39 show, for the tuberculosis class the region of interest is closer to the lungs and it has become better using augmentation techniques. However, considering only the classification performance metrics, augmentation caused a drop from 0.92 to 0.85 in the F1-score of this class that claims that better performance has been obtained by capturing the noise rather than extracting more informative features.

F.1.2 Grad-CAM for DenseNet169

We repeated our analysis for DenseNet169.

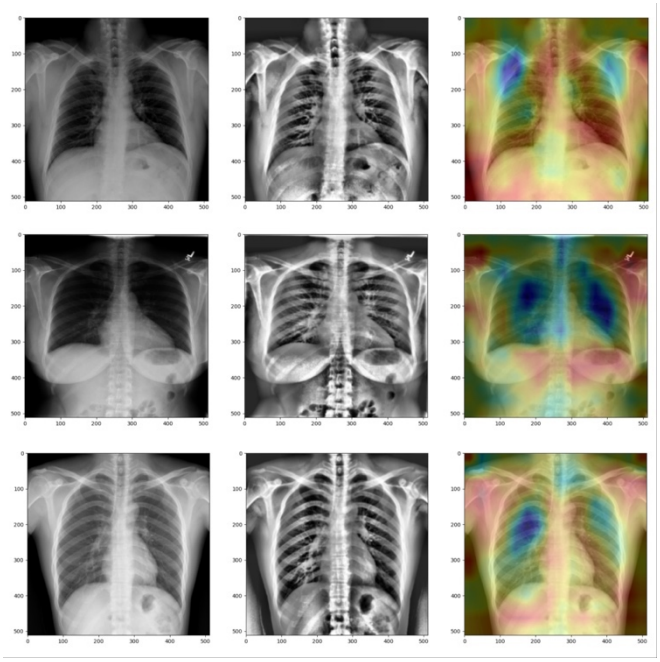


Figure 40. Images of normal class with their corresponding processed and heatmaps - DenseNet169 without augmentation

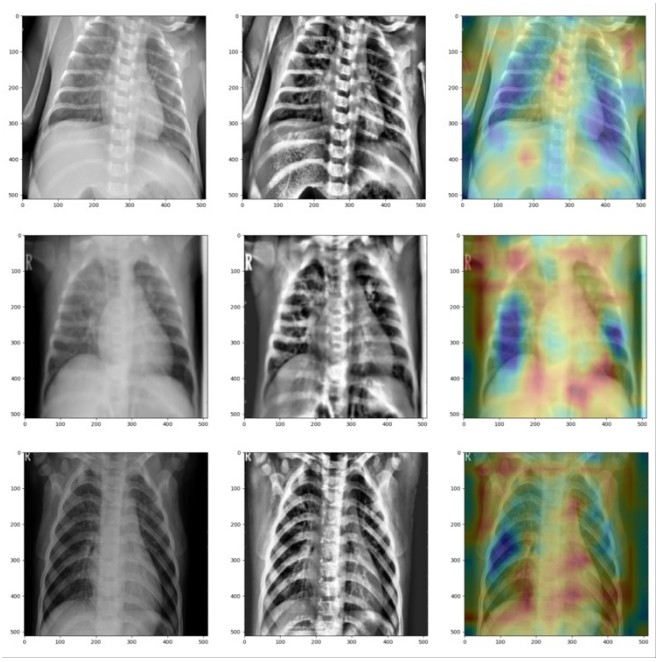


Figure 42. Images of pneumonia class with their corresponding processed and heatmaps - DenseNet169 without augmentation

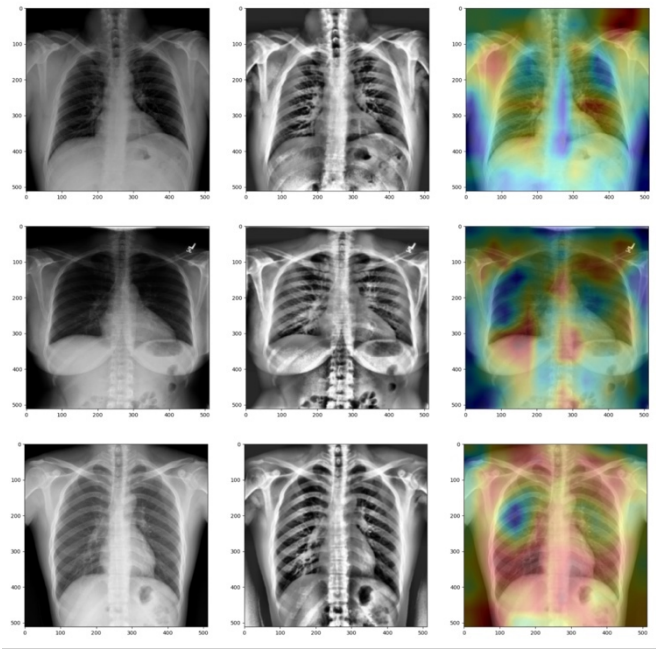


Figure 41. Images of normal class with their corresponding processed and heatmaps - DenseNet169 with augmentation

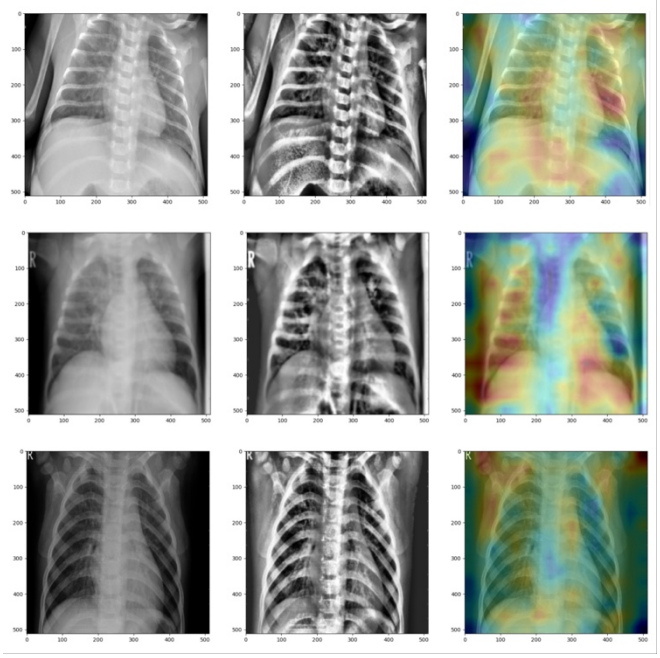


Figure 43. Images of pneumonia class with their corresponding processed and heatmaps - DenseNet169 with augmentation

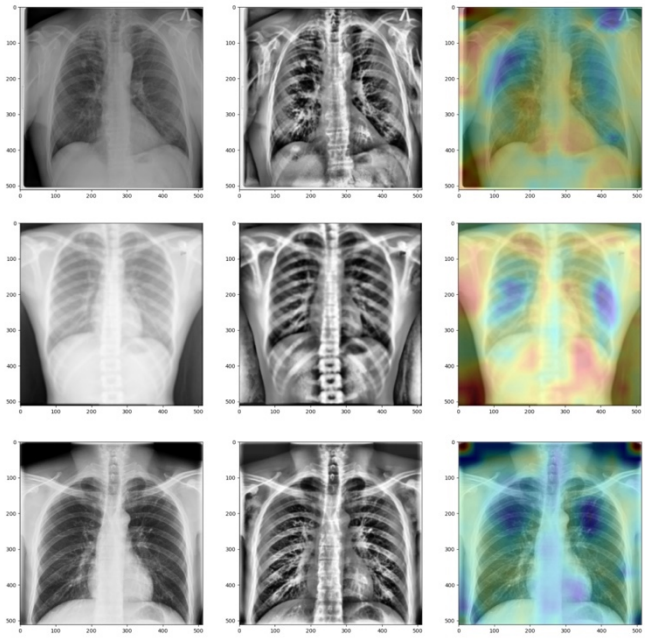


Figure 44. Images of tuberculosis class with their corresponding processed and heatmaps - DenseNet169 without augmentation

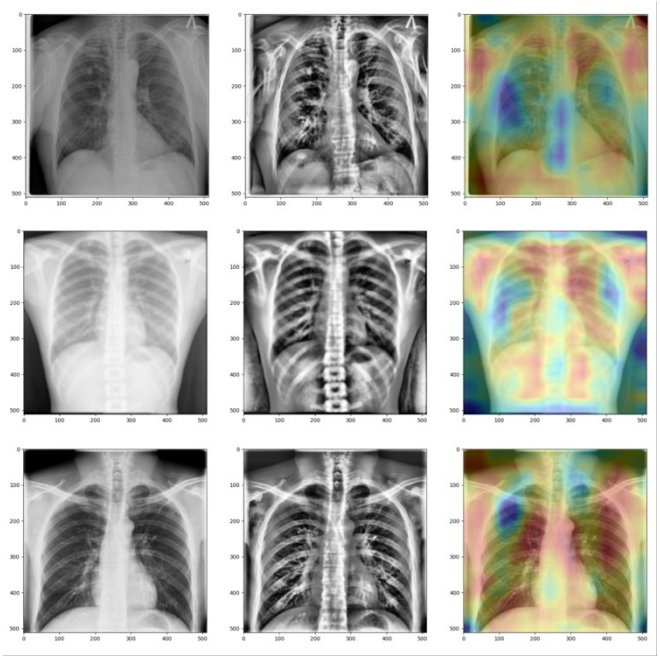


Figure 45. Images of tuberculosis class with their corresponding processed and heatmaps - DenseNet169 with augmentation

Figures 40, 42 and 44 show the better performance for DenseNet169 in finding informative regions for all classes than DenseNet 121, even without using augmentation. Furthermore,

Comparing the figures of the same classes, we can easily conclude that augmentation techniques have greater positive impact on the feature extraction phase in DenseNet169 than

In fact, not only a deeper network performs better in choosing meaningful regions of interest also, these regions can be improved more (compared to a shallower network) with proper augmentation techniques.

F.2 SHapley Additive exPlanation (SHAP)

As the second XAI technique we decided to use SHAP which generally assigns importance values to features based on their impact on the model's output. By increasing the number of features the computation time of SHAP values increases exponentially and in the context of deep learning it can be a serious challenge. Thus, we dealt with this challenge by using 1024 samples for computing SHAP values. In this way, we could control the number of times the model needs to be evaluated with different combinations of features to estimate the SHAP values as accurately as possible.

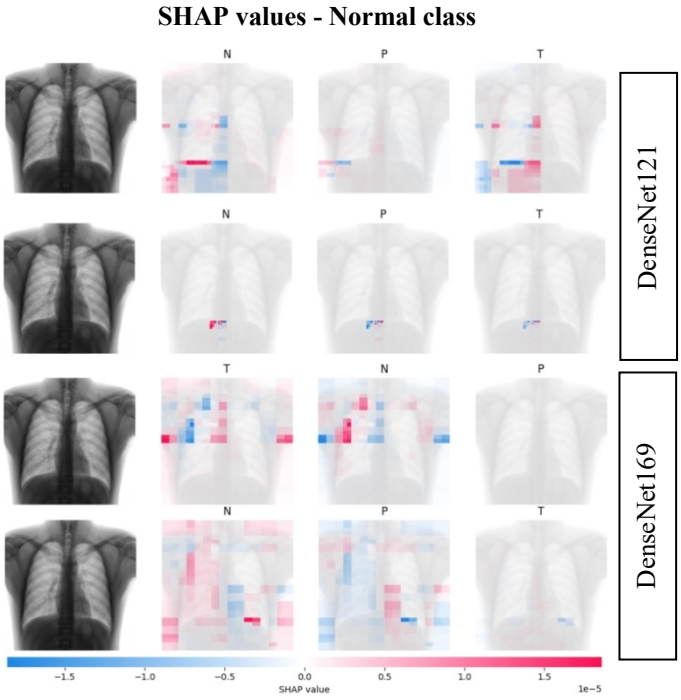


Figure 46. (From up to down) SHAP value plots of DenseNet121, DenseNet121 with augmentation, DenseNet169, DenseNet169 with augmentation

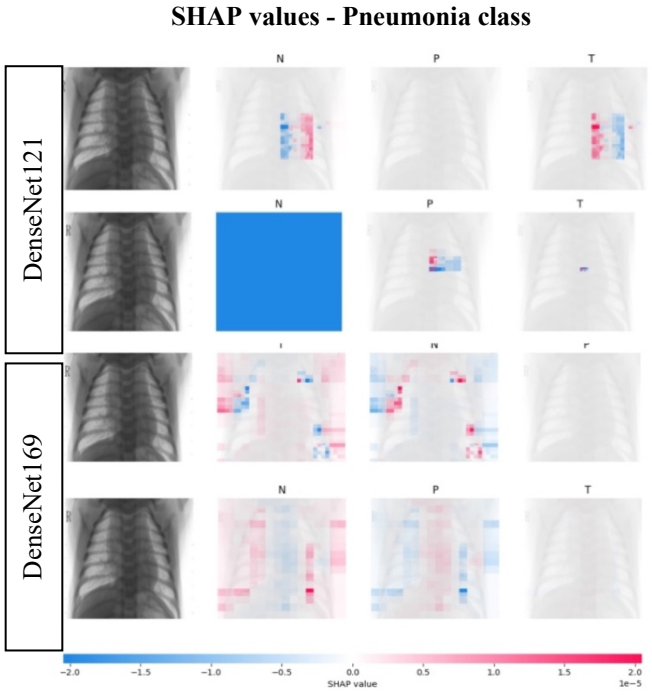


Figure 47. (From up to down) SHAP value plots of DenseNet121, DenseNet121 with augmentation, DenseNet169, DenseNet169 with augmentation

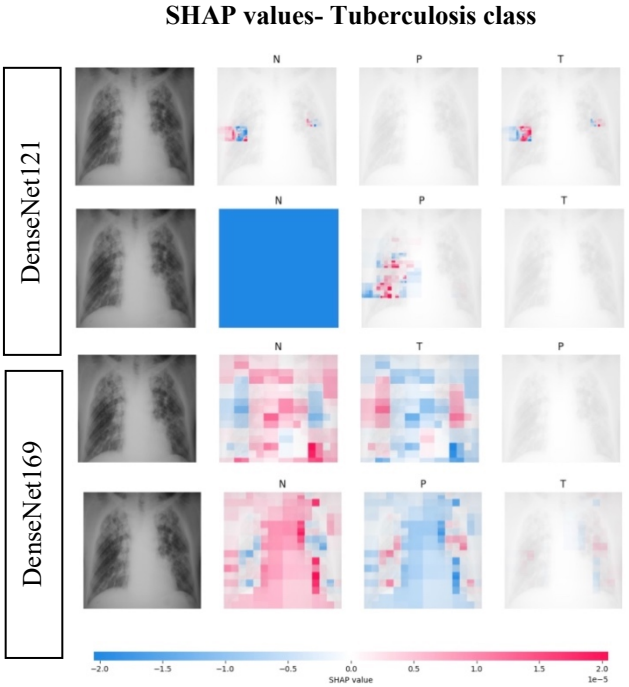


Figure 48. (From up to down) SHAP value plots of DenseNet121, DenseNet121 with augmentation, DenseNet169, DenseNet169 with augmentation

Even with estimating SHAP values, the computational time was high, and we could not perform it on more images of the dataset. Thus, it is hard to have a general overview based on the results. However, from the plots we can verify that with deeper networks (DensNet169), more (reliable) features could contribute to the prediction of the model since, for each class, the number of high SHAP values increased significantly. On the other hand, having full blue images in shallower networks (DenseNet121) shows that the model's prediction for the class of that specific image was based on more global characteristics of it, rather than specific localized features, and the model could not extract reliable features for predicting the probability for that class.

F.3 Local Interpretable Model Agnostic Explanation (LIME)

LIME[7] is a local, model-agnostic, black-box explainability technique that determines the importance of features (or pixels of the image in the case of convolutional neural networks) by performing inference on perturbated versions of the input sample to estimate how the output changes as a result of the variation using an explainable model. We decided to use it on our CNN from scratch and our DenseNet 169 model with augmentation as an alternative to GradCAM to localize the most relevant areas of the x-ray scans that contributed to the prediction of the class.

It can be seen from the side-by-side comparison in figure 49, that the model from scratch is much more likely to base its prediction on the background area outside the lungs, whereas the interpretation of the same images with the DenseNet169 model mostly highlights areas of the x-ray scan inside the chest.

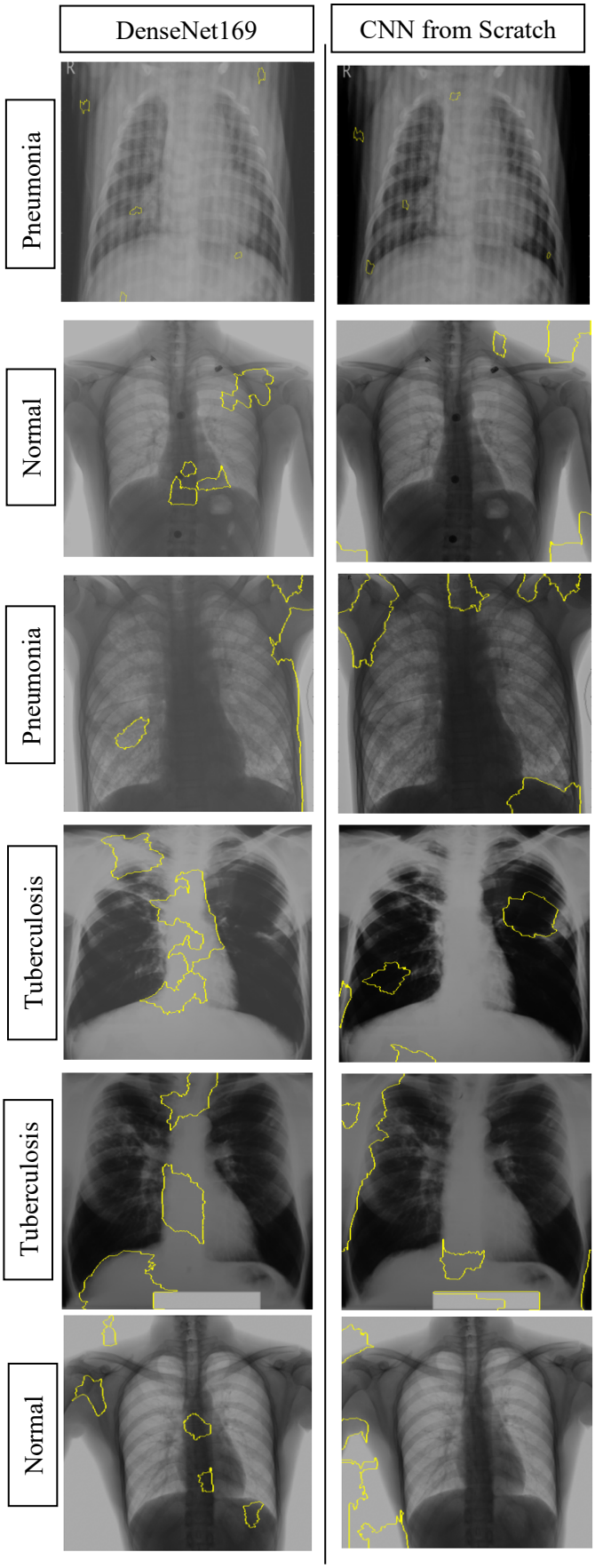


Figure 49: LIME predictions

F.4 Permutation importance for SVC model

In this part we applied the permutation feature importance techniques to the best model of handcrafted features, that is the Support Vector Machine classifier on processed wavelet feature space. In general, permutation importance evaluates the impact of each feature on the model's predictive performance by measuring the decrease in performance when the feature is randomized.

Weight	Feature
0.2198 ± 0.0105	wavelet_cV_energy
0.0925 ± 0.0049	wavelet_cH_energy
0.0822 ± 0.0036	wavelet_cH_variance
0.0586 ± 0.0131	wavelet_cD_energy
0.0576 ± 0.0104	wavelet_cA_skewness
0.0573 ± 0.0076	wavelet_cD_kurtosis
0.0496 ± 0.0117	wavelet_cA_energy
0.0445 ± 0.0089	wavelet_cA_variance
0.0426 ± 0.0074	wavelet_cH_kurtosis
0.0350 ± 0.0030	wavelet_cH_mean
0.0310 ± 0.0085	wavelet_cV_mean
0.0305 ± 0.0090	wavelet_cV_variance
0.0285 ± 0.0057	wavelet_cA_mean
0.0247 ± 0.0059	wavelet_cD_variance
0.0240 ± 0.0084	wavelet_cV_kurtosis
0.0213 ± 0.0050	wavelet_cA_kurtosis
0.0170 ± 0.0057	wavelet_cH_skewness
0.0145 ± 0.0085	wavelet_cA_entropy
0.0127 ± 0.0053	wavelet_cV_skewness
0.0008 ± 0.0024	wavelet_cD_skewness
... 1 more ...	

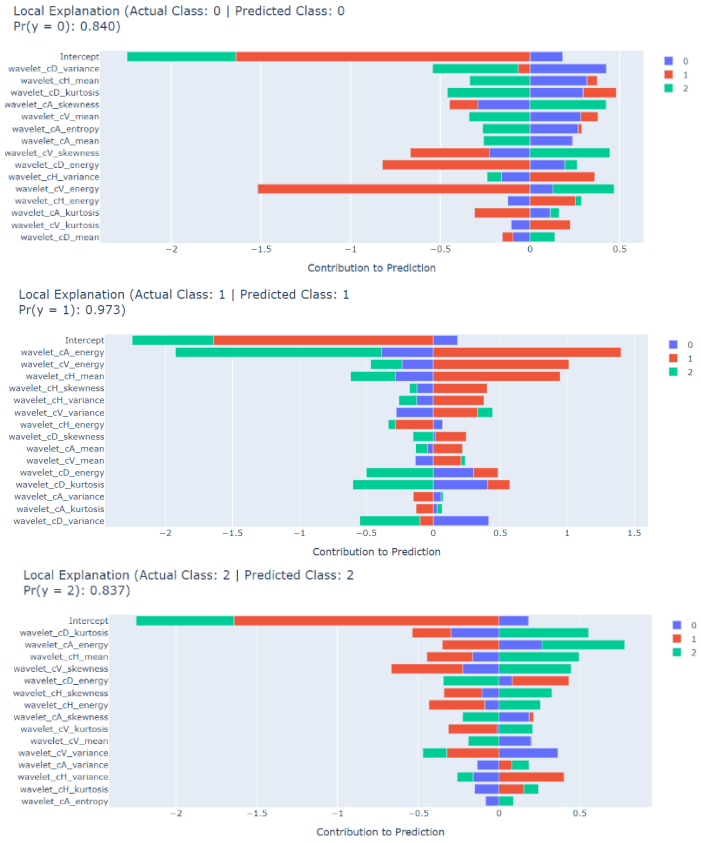
Figure 50. Feature importance plot for SVC model

Figure 50 shows the feature importance rankings based on their impact on the model's performance. As can be seen, the wavelet_cV_energy feature, which is the energy of the low frequency component (coarse-scale information), has the highest impact on the model's prediction. the next important features that influenced the output the most belonged to the high frequency components (cH and cD).

We also wanted to perform a shapely value explainability analysis on the same SVC model, but it ended up being too computationally expensive.

F.5 Explainable Boosting Machine

Explainable Boosting Machines[8] are glass-box models, which means they are natively explainable by design. The implementation we used is the one from the *interpret* library, which doesn't support global explanation for multiclass classifiers. Thus, we restricted our analysis to local interpretation of three predictions, one for each class.



The plot shows for each class (0 is Normal, 1 is Pneumonia, 2 is Tuberculosis) how much the feature contributes to change the prediction towards a certain class. We notice that there is no feature that highly discriminates the classes in the same way, but a combination of all variables in almost the same amount. The exception is the cV energy for the normal class and the cA energy for pneumonia, which seem to have a much more marked contribution on the final output.

F.6 Conclusions on explainability analysis of models on handcrafted features

While the usage of traditional machine learning models allows for efficient computation and a much broader set of explainability techniques compared to convolutional neural networks, it is our belief that in the specific context of image classification the latter is better suited to be used on the field. It's arguably more intuitive for a professional to understand how a highlighted patch of the image may have determined a certain prediction on the model part than receiving a justification based on the importance of obscure patterns like wavelet-based texture features.

III. RESULTS

The following table contains the results of all the model we examined.

Model	accuracy	F1-score	Tuberculosis F1-score	Augmentation or class weighting	XAI result
Scratch	0.83	0.90	0.80	No	bad
MobileNet	0.88	0.81	0.62	No	-
DenseNet 121	0.97	0.95	0.92	No	bad
DenseNet 121	0.94	0.92	0.85	Yes	bad
DenseNet 169	0.97	0.95	0.90	No	not bad
DenseNet 169	0.95	0.93	0.86	Yes	good
RF-spatial (Intensity)	0.62	0.46	0.20	No	-
RF-spatial (LBP)	0.75	0.64	0.35	No	-
RF-spatial (GLCM)	0.79	0.66	0.35	No	-
RF-spatial (wavelet)	0.85	0.78	0.59	No	-
SVC-spatial (Processed wavelet)	0.85	0.81	0.64	Yes	Vaguely understandable
EBM-spatial	0.83	0.78	0.60	Yes	Vaguely understandable
RF-Fourier	0.84	0.72	0.42	No	-
SVC-Fourier	0.83	0.77	0.57	Yes	-
XGB-Fourier	0.86	0.78	0.57	Yes	-

Table 4. Performance of all models

IV. DISCUSSION

In terms of final performance of CNNs, we realized deeper architectures can provide better results when proper augmentation techniques are used, even though their classification metrics might not introduce the best values. In other words, with the help of XAI and, more specifically, Grad-CAM and SHAP plots, we could understand if relevant features are being extracted from images or if the model suffers from capturing noise. As a result, we could choose a model with the maximum capability of generalization. Apart from deep learning models, although traditional ML models were not so effective, we could understand the characteristics of the images by manipulating different feature spaces in both Fourier and spatial domains. We believe that combining some of the hand-crafted feature extraction approaches with deep learning techniques can be a possibility to further improve the performance of CNNs, as a future work.

V. Conclusion

In this project, we addressed the problem of X-ray image classification for 3 classes, namely, normal, pneumonia, and tuberculosis, using deep learning models as well as ensemble and kernel-based classifiers by extracting features from spatial and Fourier domains. Due to the special characteristic of our dataset, we performed several preprocessing and image processing techniques to increase the quality of features to be extracted by the models. As expected, our results demonstrate the effectiveness of CNN models since even a simple one could outperform traditional classifiers using hand-crafted features. Further performance improvements have happened utilizing transfer learning and augmentation techniques. Moreover, the combination of different XAI techniques with deep learning approaches, and SVM classifiers allows us to overcome the inherent black-box nature of these models as well as analyzing their performance in a more detailed manner.

References

[1] Moberg, A., Taléus, U., Garvin, P., Fransson, S., & Falk, M. (2016). Community-acquired pneumonia in primary care: clinical assessment and the usability of chest radiography. Scandinavian Journal of Primary Health Care.

[2] Kumar, N., Bhargava, S., Agrawal, C., George, K., Karki, P., & Baral, D. (2005). Chest radiographs and their reliability in the diagnosis of tuberculosis.. JNMA; journal of the Nepal Medical Association.

[3] Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In Proceedings of the IEEE international conference on computer vision (pp. 2980-2988).

[4] Zaheer, M., Reddi, S., Sachan, D., Kale, S., & Kumar, S. (2018). Adaptive methods for nonconvex optimization. *Advances in neural information processing systems*, *31*.

[5] *Ayaz M, Shaukat F, Raja G. Ensemble learning based automatic detection of tuberculosis in chest X-ray images using hybrid feature descriptors. Phys Eng Sci Med. 2021 Mar;44(1):183-194. doi: 10.1007/s13246-020-00966-0. Epub 2021 Jan 18. PMID: 33459996; PMCID: PMC7812355.*

[6] Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE international conference on computer vision (pp. 618-626).

[7] Ribeiro, M. T., Singh, S., & Guestrin, C. (2016, August). " Why should i trust you?" Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1135-1144).

[8] Nori, H., Jenkins, S., Koch, P., & Caruana, R. (2019). Interpretml: A unified framework for machine learning interpretability. arXiv preprint arXiv:1909.09223.