Convolutional Neural Networks for Classification of Pulmonary Nodules

Gregory Tomy
University of Colorado Boulder
gregory.tomy@colorado.edu

ABSTRACT

This study explores pulmonary nodule detection by developing a deep learning model using a 3D Convolutional Neural Network (CNN), leveraging the LUNA16 challenge dataset. Our methodology centers on addressing the challenges posed by extreme data imbalance in the LUNA dataset. The model achieved a FROC score of 99% and recall score of 95.51%, indicating a strong capability in detecting actual nodules. However, it also presented a higher false positive rate, highlighting the inherent trade-offs when addressing data imbalance. We discuss the implications of these findings, highlighting the balance between recall and precision in the context of medical imaging and diagnostic accuracy.

1 INTRODUCTION

Lung cancer is the leading cause of cancer-related deaths worldwide. The need for early detection is paramount, especially as countries like the United States and others implement screening programs using low-dose CT scans for individuals at high risk. These initiatives, expected to generate millions of CT scans, present a significant challenge for radiologists due to their sheer volume[11]. This has led to growing interest in developing computer algorithms to improve the efficiency of the screening process.

Advancements in lung cancer detection have been significant with the integration of machine learning and computer vision technologies. Initially, detection relied heavily on manual radiological interpretations, a process often time-consuming and susceptible to human error. Recent shifts toward automated techniques have aimed to enhance both accuracy and efficiency. Notably, segmentation techniques have become crucial in isolating lung nodules for analysis [14]. Machine learning algorithms, including Support Vector Machines and Random Forests, have been employed to categorize these nodules [6]. More recently, the application of deep learning, especially Convolutional Neural Networks (CNNs), has revolutionized automated disease diagnosis [5]. These algorithms are adept at identifying complex patterns in sophisticated data types, such as medical images, offering significant advantages over traditional methods.

The primary focus of this project is to develop a classifier that acts as a diagnostic helper tool for medical professionals analyzing CT scans. This initiative is inspired by and aligned with the challenges presented in the LUNA16 challenge [9]. Our goal is to enable medical personnel to efficiently and accurately sift through numerous CT scans, aiding in the identification and classification of lung nodules. To achieve this, we leverage the LUNA Grand Challenge dataset, an extensive collection of annotated CT scans. This dataset provides a rich foundation for training and testing our models, ensuring that they are capable of performing effectively in real-world scenarios.

Choosing this project provided a unique opportunity to apply data analysis and machine learning techniques to a complex, real world dataset in the field of medical imaging. The primary goal of accurately classifying nodule candidates offered a practical and insightful experience in tackling real-world data challenges. This project exemplifies the application of data science in healthcare, underlining the intricacies and practical aspects of implementing machine learning in real-world scenarios.

2 DATA

The LUNA dataset encompasses 888 CT scans accompanied by detailed annotations and the enhanced candidates file. CT scans are 3D X-rays visualized as three-dimensional arrays of single-channel data, similar to stacked grayscale images. Each unit of these scans is a voxel, the 3D counterpart of a 2D pixel, representing a volumetric space. These voxels create a grid with measurable distances in each dimension, depicting a field of data.The numeric value of each voxel approximates the mass density of its contents, with high-density materials like bones appearing white, and low-density substances like air and lung tissue appearing black. This creates a grayscale image, more detailed than a traditional X-ray due to the retention of the third dimension [2]. The CT data is available in two formats: the .mhd files which contain metadata header information, and the .raw files, which store the raw bytes of the 3D image array. Each set of files is uniquely identified by a series UID, conforming to the Digital Imaging and Communications in Medicine (DICOM) standards.

The annotation file is a critical part of the dataset, listing each nodule's finding in a csv format. Each line details the SeriesInstanceUID of the scan, the x, y, and z coordinates in world coordinates, and the diameter of the nodule in millimeters. This file comprises 1,186 nodules, providing an structurral view of the identified nodules in the scans.

Equally essential to our project is the enhanced candidates file ($candidates\_V2$) file, which contains approximately 700,000 data points of potential nodular formations. This file is fundamental in compiling our list of nodule candidates for training and validation datasets. The enhanced file was developed for the false positive reduction track of the ISBI 2016 challenge, enhancing the original candidates file by integrating additional data from full CAD systems. This enhancement considerably increased the detection sensitivity, capturing 1,166 out of the 1,186 nodules from the annotation file. Each entry in this file includes the scan name, x, y, and z coordinates in world coordinates, and the class of the nodule candidate[9].

2.1 Candidates and Annotations Data

To achieve a balanced representation of nodule sizes in both the training and test sets, we sorted nodules based on their size and selected every N^{th} nodule for the validation set. This process required integrating data from both the candidates and annotations files. We encountered a slight misalignment of location coordinates between these files. Given the limited number of positive cases in our dataset (Table 1) and the minor nature of these mismatches, eliminating the mismatched data was not a feasible option. To address this challenge, we employed a *fuzzy matching* technique. This method involved comparing the center coordinates of each candidate in the candidates file with those of the annotated nodules in the annotations file, with a focus on nodules within the same CT scan series. The matching criterion was based on the proximity of these coordinates. A match was established when the absolute difference between the coordinates of a candidate and a nodule was less than or equal to a quarter of the nodule's diameter in any of the three dimensions (x, y, and z). When a candidate closely matched an annotated nodule based on these criteria, it was marked as a likely match and assigned the nodule's diameter. Thus, we obtained a clean list of candidates, with information integrated from both the candidates and annotations data.

2.2 Preparing CT Data

For our CT scan data preparation, we implemented a clipping technique to isolate relevant data. CT scans are quantified in *Hounsfield units (HU)*, a scale used to describe radiodensity [2]. We restricted our analysis to the -1000 to 1000 HU range[1], thereby trimming any extraneous data beyond this spectrum. We utilized the cleaned candidates list to locate and further process the relevant areas in the CT scans.

Our data preparation required transforming the candidate center data from millimeters to voxel-based coordinates (Index, Row, Column). This conversion was necessary because CT scans are structured around voxel coordinates. Unlike the voxel system, the patient coordinate system, which is commonly used in medical imaging, measures in millimeters and has an arbitrary origin that does not align with the origin of the CT scan's voxel array. This system is typically used for marking important anatomical features across various scans. The reconciliation of the two coordinate systems was achieved using the metadata in the CT scan header files. This metadata contained the necessary transformation matrices for converting coordinates from the patient coordinate system (X, Y, Z) to the voxel coordinate system (I, R, C)[1].

Given our emphasis on nodule classification, processing the entire CT scan was impractical and unnecessary, particularly considering hardware limitations[2]. To efficiently train our model, we adopted a targeted approach by cropping a specific 3D region around the center of each candidate nodule. The dimensions for these cropped regions were set to 36x48x48[3]. This focused method of extracting nodules ensured that our model received the most relevant data, enhancing both the efficiency and accuracy of our classification process [12].

2.2.1 Subsampling.

In our approach to streamline the training process and enable faster experimental iterations, we employed a subsampling strategy. This involved creating a smaller dataset from our original data, henceforth called the *small dataset*, consisting of approximately 50,000 randomly selected data points. A key consideration in this subsampling process was to maintain the original distribution of positive to negative classes, ensuring that the reduced dataset accurately reflected the characteristics of the full dataset.

Using this *small dataset*, we were able to conduct model testing and experimentation more efficiently. This method significantly reduced the computational load and time required for each training iteration, allowing us to explore various model configurations, parameters, and techniques with greater agility. Once we identified promising models and techniques with the *small dataset*, we applied these insights to the full dataset for comprehensive training and validation. This approach balanced the need for rapid experimentation with the requirement to maintain data integrity and representativeness.

2.3 Handling Imbalanced Data

In the development of our machine learning model for nodule classification, we faced a significant challenge due to the highly imbalanced nature of our dataset (Table 1). The dataset was skewed with a ratio of approximately 480:1 in favor of negative samples over positive ones. This stark imbalance posed a critical challenge for accurate classification, as it predisposed the model to bias, often erroneously classifying nodules as negative.

Our initial model iterations, without special consideration for this imbalance, produced deceptively high accuracy scores. Despite an accuracy rate near 99% , the model failed to correctly identify any positive samples, resulting in a precision and recall of 0%. This was a clear indication that the model was defaulting to the prediction of the negative class, reinforced by the observation in a smaller subset of our data where, out of 552 samples, 550 were negative and only 2 were positive.

To address this imbalance, our first strategy involved the implementation of Kaiming's focal loss [12]. This approach, a variation of the standard cross-entropy loss, focuses the model on difficult-to-classify samples by lessening the impact of those that are easier to classify [8]. While this method did yield some improvement, our results still fell short of the top performance levels in the LUNA challenge, indicating a possible shortfall in our application of this technique.

2.3.1 Oversampling The Positive Class.

Subsequently, we adopted a simpler, yet effective approach: oversampling the positive class [10]. We adjusted our data processing to present equal proportions of positive and negative samples in each batch, effectively achieving a 1:1 ratio through the repetition of positive samples. This method effectively countered the model's tendency to predict every sample as negative. By intermixing positive and negative classes within the training batches, the model's weight updates were forced to distinguish between the two classes.

2.3.2 Shorter Training Epochs.

This approach necessitated a rethinking of our training methodology. We adopted a subset-based approach, focusing on balanced batches containing an equal number of positive and negative samples, with each class contributing 10,000 samples per epoch. This method not only helped mitigate the

[1] -1000 HU is the density of air and 1000+ the density of bone.
[2] See A.2
[3] In voxel coordinates (IRC).

Table 1: Candidates nodules.

Not Nodules	Actual Nodules
753,418	1,557

model's bias towards the majority class but also made the training process more manageable and efficient, considering our hardware constraints. By varying the subsets in successive epochs, we ensured diverse exposure and cumulative learning over time, crucial for the model's ability to generalize effectively.

2.4 Augmenting the Data

To fortify our 3D Convolutional Neural Network against overfitting and enhance its generalization capabilities, we implemented a data augmentation strategy during the preprocessing phase. Data augmentation artificially generates new training samples from existing data by applying random, yet systematic, transformations. This method enriches the training dataset, enabling the model to learn from a broader array of examples and enhancing its generalization to unseen data[12][7]. Four data augmentations were used as follows:

(1) **Rotation**: Rotations are applied in the X-Y plane. Due to the uniform spacing in the X and Y axes and non-cubic nature of the Z-axis in CT slices, we limit rotations to the X-Y plane to maintain anatomical accuracy.
(2) **Scale**: Scaling the images, either enlarging or reducing them.
(3) **Rotation**: Mirroring of images across various axes—left to right, front to back, and up and down.
(4) **Shift**: Shifting the images by a few voxels, introducing subtle spatial translations.

3 MODEL ARCHITECTURE

We employ a 3D Convolutional Neural Network (CNN) utilizing PyTorch for the classification of nodule candidates in CT scans (Figure 1). This model was informed by the top two submissions to the LUNA16 challenge, incorporating key insights and strategies that proved effective in those high-performing models [12][3]. Our CNN architecture is conventionally structured with three main components: a tail (input layer), a backbone (core layers), and a head (output layer).

3.1 Tail

(1) **Augmentation layer**: The entry point of our model is the augmentation layer. Although strictly speaking augmentation is a data preprocessing step, due to the large size of our dataset and persistent hardware limitations, particularly concerning CPU memory capacity, this approach proved to be impractical. To optimize our workflow, we decided to leverage the GPU's memory, which is generally more efficient for handling large-scale data transformations compared to the CPU. This led us to integrate an augmentation layer directly into our model architecture. The augmentation layer in our model is distinct in that it carries no model states and is dedicated solely to data transformations. This layer dynamically applies selected augmentations to the data as it passes through the network, offloading this computational task to the GPU. Depending on the augmentation flags activated, the layer performs transformations, enhancing the diversity and robustness of the training process.

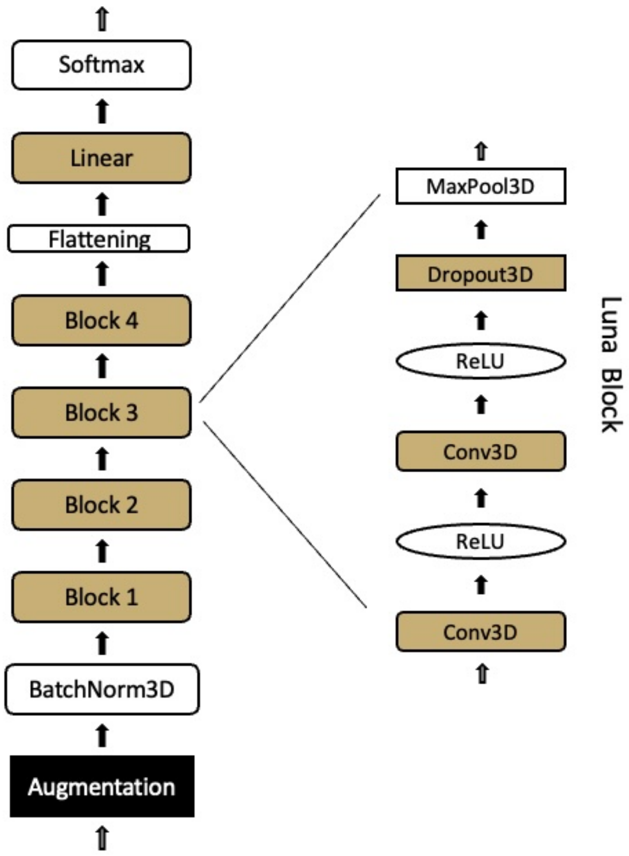


Figure 1: LunaModel architecture

(2) **Normalization layer**: The augmentation layer is followed by a batch normalization layer. This layer plays a crucial role in stabilizing the learning process by normalizing the input batch. Given the manageable size of our images, we intentionally omitted any downsampling layers at this stage to preserve the integrity of the image data.

3.2 Backbone

The backbone of the model is constructed from four repeating convolutional blocks. Each block is structured as follows:

(1) **Convolutional Layers**: A pair of 3x3 convolutions form the primary components of each block. These layers are responsible for feature extraction from the input data.
(2) **ReLU Activation**: Post each convolutional layer, a ReLU activation function is applied.

(3) **Dropout layer**: A dropout layer is introduced after the second ReLU activation within each block. This layer randomly deactivates a fraction of neurons during training, preventing overfitting and encouraging a more robust feature learning.

(4) **Max pooling**: Each block concludes with a max-pooling layer, which reduces the spatial dimensions of the output.

3.3 Head

The final stage of our model, the head, is responsible for producing the classification output. The process involves the following steps.

(1) **Flattening layer**: The output from the backbone is first flattened to transform the multidimensional feature map into a one-dimensional array, ready to be fed into the fully connected layer.

(2) **Linear layer**: The flattened output is then passed through a fully connected linear layer to map the extracted features to the binary classification output. In contrast to some of the top submissions in the LUNA16 challenge, which employed additional fully connected layers, our model utilizes a single linear layer. This design decision is based on the relative simplicity of the nodule structures in the images, where additional complexity did not yield significant benefits.

(3) **Softmax layer**: Finally, a softmax layer is appended. This layer converts the linear outputs into probabilities, facilitating the classification between nodule and non-nodule classes.

4 TRAINING METHODOLOGY

In the preliminary phase of our training methodology, we utilized the *small dataset* to conduct crucial trial runs. These trials aimed to refine our approach through assessing the impact of different model parameters and data augmentation strategies. Our trial runs encompassed the following scenarios:

(1) **No Augmentation Trial (*Base*):** A baseline trial without any augmentation was conducted to establish a performance benchmark for comparison.

(2) **Individual Augmentation Trials**: We tested each augmentation technique independently to pinpoint its unique contribution to model performance.

(3) **Combined Augmentation Trial (*All*)**: All augmentation techniques were activated simultaneously, providing insights into their collective impact on the model's generalization and adaptability.

Results from these initial trials on the *small dataset* were useful in determining effective parameter and augmentation combinations. Insights gained guided the strategy for training on the *full dataset*, allowing us to progress with configurations that promised better performance. The final model was trained on the *full dataset* with all augmentations for 40 epochs.

4.1 Loss Function

For training our model, we utilize the Cross Entropy Loss function, a standard choice for classification tasks. This function is effective for our nodule classification task due to its ability to handle class imbalances. It operates by first converting model outputs into probabilities (using softmax activation) and then calculating the negative log likelihood of the true class. This approach not only provides a probabilistic view of the model's predictions but also directly optimizes classification accuracy by penalizing confident incorrect predictions [13].

4.2 Evaluation Metric

The Free-Response Receiver Operating Characteristic (FROC) score is the primary metric for evaluating the performance of lung nodule detection models in the LUNA challenge. It assesses the model's sensitivity to detect true positives over a range of decision thresholds. FROC analysis involves calculating sensitivity and the average number of false positives per scan at multiple threshold levels. The model's predictions above each threshold are considered, and the corresponding sensitivity is plotted against the average false positives, beginning at (0,0)—indicating no false positives—and spanning through all distinctive points determined by the thresholds. To synthesize the model's performance, the average sensitivity at predefined false positive rates of 0.125, 0.25, 0.5, 1, 2, 4, and 8 per scan is computed. This average FROC score serves as a consolidated measure of the model's capacity to maintain high sensitivity while controlling for false positives across these specified rates [9].

In addition to the FROC score, this study expands the evaluation framework by incorporating a range of metrics: Recall, F2 score, Precision, and False Positive Rate (FPR). Recall, or sensitivity, measures the model's effectiveness in correctly identifying true lung nodules. The F2 score is a variation of the F-score with a stronger emphasis on recall over precision. In medical imaging, this is vital as the cost of missing a true positive generally outweighs the impact of a false positive detection.

$$F2 = (1 + 2^2) \cdot \frac{precision \cdot recall}{2^2 \cdot precision + recall} \tag{1}$$

Precision assesses the proportion of correctly identified nodules out of all predictions classified as nodules. FPR gauges the frequency of false positives relative to the total number of actual negatives. Combining these metrics offered a comprehensive evaluation. FROC focuses on the model's accuracy in nodule detection by examining sensitivity and false positive management. F2 score underscores the importance of detecting true positives to minimize missed diagnoses. Precision and FPR further elucidate the model's diagnostic capabilities, allowing for a nuanced understanding of its results and the trade-offs involved.

4.3 Early Stopping

Early stopping was integrated into our training regimen to optimize computational resources. We continuously monitored the FROC metric during the validation phase. If, over a sequence of five consecutive validation cycles, no discernible improvement in the FROC score was observed, the training process was automatically halted. In such instances, the most recent model with the highest attained FROC score was retained. This strategy not only conserved valuable computing time but also mitigated the risk of overfitting and model divergence. By preserving the last best model, we ensured that the culmination of our training efforts was encapsulated, striking a balance between computational efficiency and optimal model performance.

4.4 Weight Initialization

In our model we implemented Kaiming initialization, an approach particularly suited for networks using ReLU activation functions. Developed by Kaiming He et al., this method sets initial weights to maintain controlled variance across layers, addressing the common issues of vanishing and exploding gradients in deep neural networks. Kaiming initialization, optimized for ReLU, ensures efficient gradient propagation and stabilizes the learning process, especially crucial during early training epochs. This approach not only facilitates faster convergence, saving time and computational resources, but also enhances overall model performance by enabling more effective learning from the start[4].

4.4.1 Initialization Strategy. In developing our deep learning model for detecting pulmonary nodules, we faced a challenge with the model's initial weight setup. Neural networks typically begin with randomly initialized weights, which can cause variability and bias in the early training stages, especially in complex or imbalanced datasets. Our model initially showed a tendency to converge to extreme recall values (0 or 1), reflecting a strong bias towards one class. To address this, we employed a strategic approach to weight initialization. We aimed for a starting point where the model showed balanced sensitivity between classes. Through preliminary training runs of one epoch each, under different initializations, we evaluated the model's recall on a validation set. We selected the weights from an initialization that yielded a recall neither 0 nor 1, indicating a balanced start. This method of selective weight initialization helped the model begin training from a more neutral position, reducing the risk of early bias and promoting balanced learning. The chosen initial weights were then used for the full training, leading to more stable early training dynamics and improved overall predictive performance, as shown by consistent FROC scores.

5 RESULTS

Trial runs with the *small dataset* pinpointed augmentations that enhanced the model performance. Considering the constraints that allowed for only one *full dataset* run, we chose to implement *All* augmentations, targeting the highest recall score AppendixA.

The results from the *full dataset* run were promising. As demonstrated in, the model achieved an almost perfect FROC score of 99.27%, indicating exceptional sensitivity across various false positive rates. The recall score was high at 95.51% , underscoring the model's ability to correctly identify the majority of true nodules. This high recall is further validated by the confusion matrix in , with 149/156 true positives identified.

The precision of the model stood at 2.78%. The false positive rate (FPR) of 6.91% indicates that the mode, while robust in identifying nodules, does so with a tendency to over-classify.

6 ANALYSIS

Our model's performance, in the context of a heavily imbalanced dataset, highlights several key insights. The high FROC and recall scores are significant achievements, indicating the model's capability to detect the majority of true positives amid a vast number of negatives[4]. This focus on recall is a response to the dataset's imbalance, emphasizing the importance of not missing true positives in lung cancer detection. However, the model's lower precision and higher false positive rate (FPR) reveal the complexities and potential drawbacks of our approach. These metrics suggest a model inclined to over-classify nodules, a possible consequence of the strategies employed to address data imbalance.

Table 2: Confusion Matrix

		Predicted	
		Positive	Negative
Actual	Positive	149	7
	Negative	5207	70135

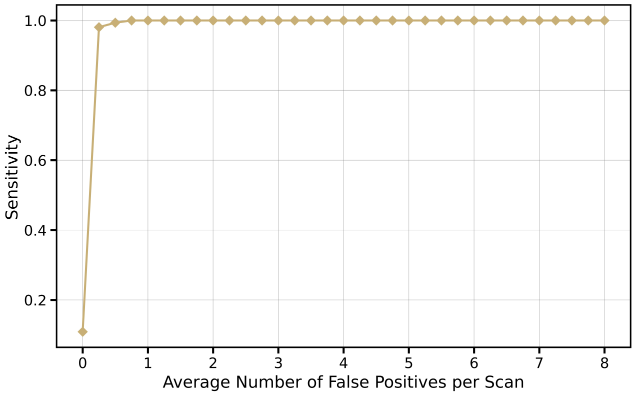


Figure 2: FROC Curve

Table 3: Model Performance Metrics

Metric	Value
FROC	99.27%
$F2$	12.46%
Recall	95.51%
FPR	6.91%
Precision	2.78%

The decision to oversample the positive class had a marked effect on the model's performance metrics.The impact of this approach is evidenced by the model's high recall score. However, while this approach led to improved recall, it is possible that it also contributed to an increase in the model's false positive rate. Presenting the model with an equal representation of positive and negative cases may have inadvertently caused it to overgeneralize nodule characteristics. This would lead to a higher incidence of false positives. Such an outcome is a recognized drawback of oversampling: the repeated exposure to positive samples can sometimes reduce the model's ability to discern the specific features of nodules accurately.

[4] See A.1 for comments on FROC metric.

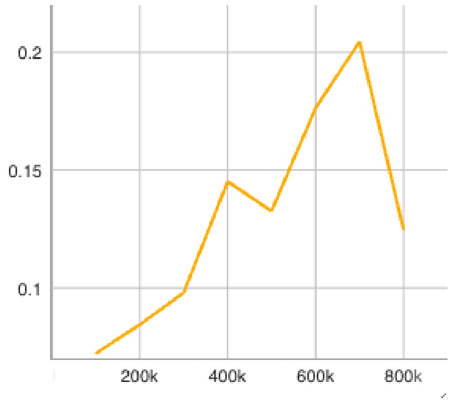


Figure 3: F2 scores for *full dataset.*

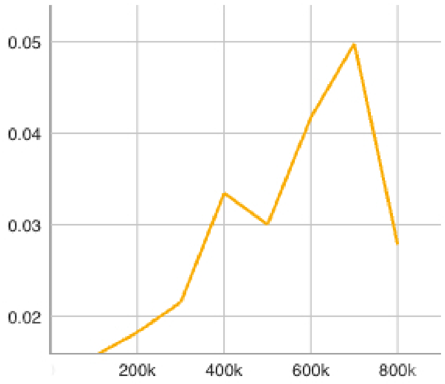


Figure 4: Precision scores for *full dataset.*

The adoption of shorter training epochs was a response to both the imbalanced nature of the dataset and the limitations imposed by our computational resources. The subset-based approach allowed for a more focused and efficient training process. However, the use of shorter epochs also meant that the model was exposed to a smaller portion of the dataset in each training cycle. While we attempted to mitigate this by varying the subsets in successive epochs to provide diverse exposure, there is a potential risk that the model may not have fully learned the complex patterns inherent in the broader dataset. Additionally, the shorter epochs, while efficient, may have contributed to the higher false positive rate observed in our results. The model's exposure to repeated examples of nodules — due to the oversampling of the positive class — within these shortened training cycles may have led to an overfitting to certain nodule characteristics, thereby increasing the likelihood of false positives.

The weight initialization strategy, chosen to avoid early biases, also influenced the model's results. By starting from a balanced sensitivity to both classes, we aimed to avoid the model favoring either class excessively. This approach likely aided in achieving the high recall but may have also led to a propensity for over-classification, as indicated by the precision and FPR.

It is interesting to note that the model achieved higher F2 and Precision scores in the penultimate validation epoch, as indicated in Figure 3 and Figure 4, before experiencing a significant drop in the final validation epoch. On the other hand, there was only a small difference in the FROC score (Figure 6), but the Recall increased to 0.95 in the final epoch, compared to 0.91 in the penultimate epoch (Figure 7). This indicates a more balanced configuration of the model in the penultimate validation epoch. In clinical settings, where the minimization of false positives is as crucial as the detection of true positives, such a balanced approach could be preferable. The trade-off between recall and precision is a critical consideration in medical diagnostics, especially in scenarios like pulmonary nodule classification where missing a true positive (a potential malignancy) can have serious consequences. In this context, our model's inclination towards high recall appears to be a justifiable compromise. Nonetheless, the potential for a more balanced model configuration is a valuable insight. It opens up possibilities for tailoring the model to different clinical scenarios, where the emphasis might shift depending on the specific requirements of the task at hand. For instance, in situations where reducing false positives is a priority, a configuration with a higher emphasis on precision and F2 score could be more suitable.

We have also initiated a *full dataset* run with a model variant focusing on maximizing the F2 score instead of prioritizing FROC and Recall. The results of this are still pending and will be provided in a future update to the github repository..

7 CONCLUSION

Our model's performance, characterized by high recall but accompanied by a high false positive rate, underscores the challenge of balancing sensitivity and specificity in machine learning for medical imaging. In clinical settings, a high FPR could lead to an increase in unnecessary follow-up procedures, potentially affecting healthcare efficiency and patient experience. Nonetheless, considering the overwhelming task of manually reviewing 700,000 potential nodules, our model's ability to narrow this down to approximately 5,000, while capturing most true positives, marks a significant enhancement in screening efficiency. This reduction significantly eases radiologists' workload, enabling them to focus on a more manageable subset of flagged nodules.

In working with the LUNA dataset, we encountered the multifaceted challenges of preparing real-world data for machine learning applications. Our exploration into the data creation process was particularly enlightening, offering valuable lessons in the complexities of data cleaning and wrangling. This deeper dive enabled us to effectively tackle the nuances of medical imaging data, aligning different data formats and addressing inconsistencies, which was crucial for the thorough preparation of our dataset. Additionally, overcoming hardware limitations and creatively addressing big data challenges were key components of our work. We implemented strategies such as shorter training epochs and strategic weight initialization to efficiently handle computational demands, demonstrating the need for varied approaches in processing large-scale data. Furthermore, the project's interdisciplinary nature, blending machine learning and medical imaging, underscored the value and skill involved in combining machine learning and data mining knowledge with domain-specific expertise.

7.1 Future Work

As we look towards potential future enhancements of our lung nodule detection model, our primary objective will be to improve precision while maintaining the high recall we have achieved. To this end, several key areas of development may be pursued. Investigating different model architectures could provide new avenues for performance enhancement. In particular, the use of ensemble models, which combine the strengths of multiple learning algorithms, could offer a way to balance precision and recall more effectively. Our project faced certain limitations due to hardware constraints, leading to workarounds that might have introduced biases. Utilizing more powerful hardware will enable us to avoid these compromises in the future. While we have already utilized data augmentation techniques, there is scope to explore more complex and varied augmentations. Experimenting with additional combinations and variations of augmentations could further improve the model's ability to generalize and perform accurately on unseen data. Our current approach involves cropping to extract identified candidate nodules. Future work will focus on automating this process, potentially through the development of an end-to-end model that can automatically identify and segment nodule candidates.

REFERENCES

[1] Nikolas Adaloglou. 2020. Understanding coordinate systems and DICOM for deep learning medical image analysis. https://theaisummer.com/ (2020). https://theaisummer.com/medical-image-coordinates/

[2] WE Brant. 2018. Brant and Helms' Fundamentals of Diagnostic Radiology. Lippincott Williams & Wilkins, Philadelphia, USA.

[3] Fonova. 2017. 3D Deep Convolution Neural Network Application in Lung Nodule Detection on CT Images. (2017). https://rumc-gcorg-p-public.s3.amazonaws.com/f/challenge/71/fda79ef2-f2a9-4a04-b90e-efebf928edb6/20170915_095225_LUNA16FONOVACAD_FPRED.pdf

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In Proceedings of the IEEE international conference on computer vision. 1026–1034.

[5] Seyed Hesamoddin Hosseini, Reza Monsefi, and Shabnam Shadroo. 2023. Deep learning applications for lung cancer diagnosis: a systematic review. Multimedia Tools and Applications (2023), 1–31.

[6] Timor Kadir and Fergus Gleeson. 2018. Lung cancer prediction using machine learning and advanced imaging techniques. Translational lung cancer research 7, 3 (2018), 304.

[7] Shoji Kido, Yasusi Hirano, and Noriaki Hashimoto. 2018. Detection and classification of lung abnormalities by use of convolutional neural network (CNN) and regions with CNN features (R-CNN). In 2018 International workshop on advanced image technology (IWAIT). IEEE, 1–4.

[8] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In Proceedings of the IEEE international conference on computer vision. 2980–2988.

[9] LUNA16. 2023. LUNA16: LUng Nodule Analysis 2016. https://luna16.grand-challenge.org

[10] Roweida Mohammed, Jumanah Rawashdeh, and Malak Abdullah. 2020. Machine learning with oversampling and undersampling techniques: overview study and experimental results. In 2020 11th international conference on information and communication systems (ICICS). IEEE, 243–248.

[11] Virginia A Moyer and US Preventive Services Task Force*. 2014. Screening for lung cancer: US Preventive Services Task Force recommendation statement. Annals of internal medicine 160, 5 (2014), 330–338.

[12] China Ping An Technology (Shenzhen) Co., Ltd. 2017. 3DCNN for False Positive Reduction in Lung Nodule Detection. (2017). https://rumc-gcorg-p-public.s3.amazonaws.com/f/challenge/71/f1dddec3-4421-4154-a203-23eb3e894a1c/20171220_083208_PAtech_FPRED.pdf

[13] Pang-Ning Tan Michael Steinbach Vipin et al. 2006. Introduction to data mining.

[14] Lulu Wang. 2022. Deep learning techniques to diagnose lung cancer. Cancers 14, 22 (2022), 5569.

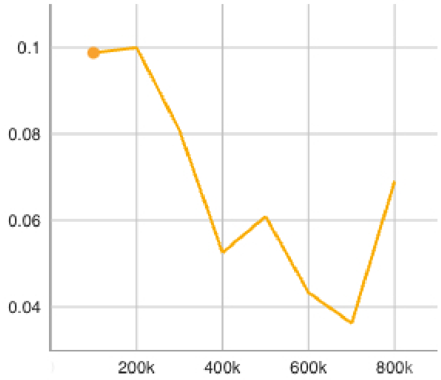


Figure 5: FPR scores for *full dataset*.

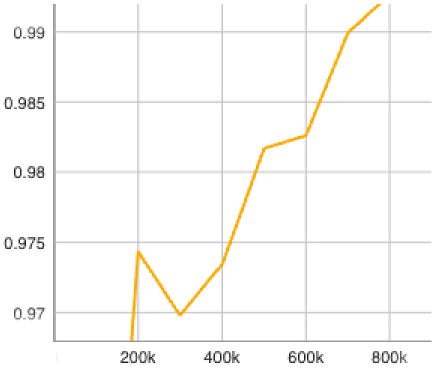


Figure 6: FROC scores for *full dataset*.

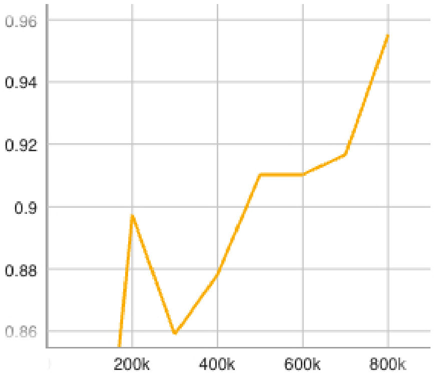
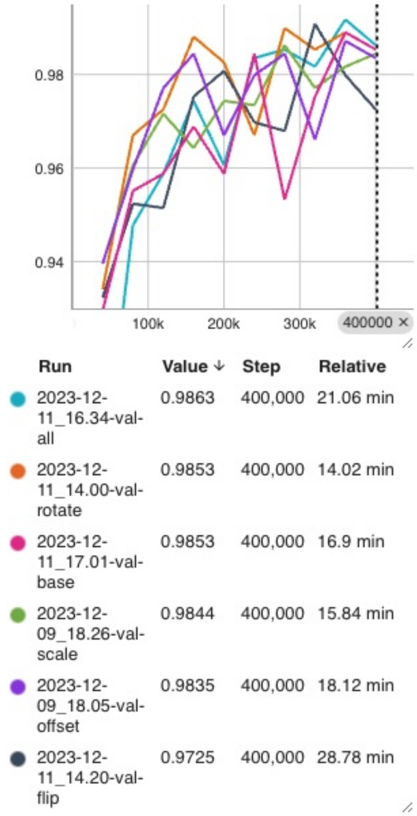


Figure 7: Recall scores for *full dataset*.



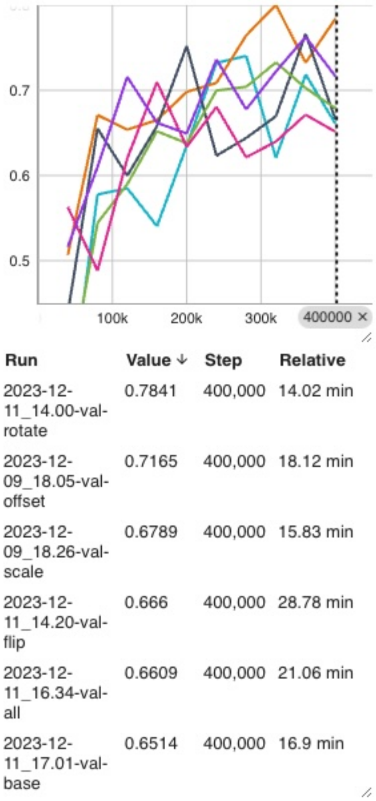
Run	Value ↓	Step	Relative
● 2023-12-11_16.34-val-all	0.9863	400,000	21.06 min
● 2023-12-11_14.00-val-rotate	0.9853	400,000	14.02 min
● 2023-12-11_17.01-val-base	0.9853	400,000	16.9 min
● 2023-12-09_18.26-val-scale	0.9844	400,000	15.84 min
● 2023-12-09_18.05-val-offset	0.9835	400,000	18.12 min
● 2023-12-11_14.20-val-flip	0.9725	400,000	28.78 min

Figure 8: FROC scores for the *small dataset* trials.

A APPENDIX

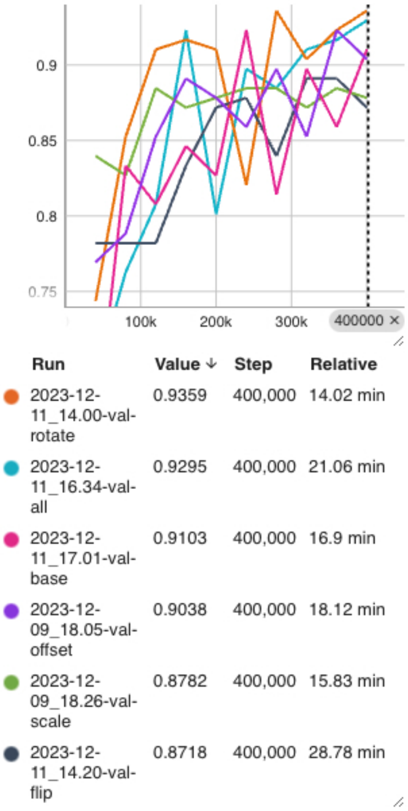
A.1 FROC Score

The near-perfect FROC score attained by our model warrants careful scrutiny. Our model exhibits a high recall, which directly contributes to the FROC score. However, this high recall does not inherently imply high precision. In our case, the model's tendency to identify many false positives (low precision) alongside true positives still allows for a high FROC score, as FROC primarily focuses on the detection of true positives at varying false positive levels. Our understanding is that it is possible for a model to achieve a high FROC score even with a relatively high false positive rate. This situation arises because the FROC curve considers sensitivity across different false positive rates. As long as the model maintains high sensitivity (high recall), it can still achieve a high FROC score despite generating a significant number of false positives. This aspect is particularly relevant when considering the limitations in comparing our results with the LUNA16 challenge. The LUNA16 submissions primarily report FROC scores, which focus on sensitivity without detailed insights into false positive rates. As a result, our model's performance, characterized by its high recall and FROC score, might not be directly comparable to other models in terms of precision and overall false positive rate. Additionally, our approach to calculating the FROC score was manually implemented based on our interpretation of the LUNA16 guidelines. While this method was carefully executed, the manual nature of the calculation introduces a possibility of inaccuracies, underscoring the need for meticulous verification to ensure the reliability and accuracy of our findings.



Run	Value ↓	Step	Relative
● 2023-12-11_14.00-val-rotate	0.7841	400,000	14.02 min
● 2023-12-09_18.05-val-offset	0.7165	400,000	18.12 min
● 2023-12-09_18.26-val-scale	0.6789	400,000	15.83 min
● 2023-12-11_14.20-val-flip	0.666	400,000	28.78 min
● 2023-12-11_16.34-val-all	0.6609	400,000	21.06 min
● 2023-12-11_17.01-val-base	0.6514	400,000	16.9 min

Figure 10: F2 scores for the *small dataset* trials.



Run	Value ↓	Step	Relative
● 2023-12-11_14.00-val-rotate	0.9359	400,000	14.02 min
● 2023-12-11_16.34-val-all	0.9295	400,000	21.06 min
● 2023-12-11_17.01-val-base	0.9103	400,000	16.9 min
● 2023-12-09_18.05-val-offset	0.9038	400,000	18.12 min
● 2023-12-09_18.26-val-scale	0.8782	400,000	15.83 min
● 2023-12-11_14.20-val-flip	0.8718	400,000	28.78 min

Figure 9: Recall scores for the *small dataset* trials.

A.2 Hardware Performance and Limitations

A significant challenge encountered in this project was sourcing hardware capable of running our complex model within a reasonable timeframe. Due to the intensive computational requirements, standard consumer-grade hardware was insufficient for our needs. The initial stages of our work utilized the resources provided by the University of Colorado Research and Computing (CURC). Despite the capabilities of CURC, we faced persistent obstacles related to the specific requirements and limitations of CURC's system architecture. We encountered extended wait times due to queueing and resource allocation protocols. Additionally, data loading errors frequently disrupted our processing, leading to delays and incomplete runs. To overcome these limitations, we shifted our computations to a dedicated high-performance setup. This hardware included an Intel Core i9-13900K processor, 64GB of DDR5 RAM at 5600 MHz, an NVIDIA GeForce RTX 4090 graphics card, and a 2TB SSD for rapid data access and storage.