International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 16, No.1 (Sep-24)

http://dx.doi.org/10.12785/ijcds/1601112

Optimized AlexNet For Enhanced Tuberculosis Classification Using Deep Learning

Shanmuga Sundari M[1], Vidyullatha Sukhavasi[2], Swapna D[3], Durga Kbks[4], Poonam Shaylesh Lunawat[5] and Mayukha Mandya Ammangatambu[6]

[1,2,3,4,5,6]*Computer Science and Engineering, BVRIT HYDERABAD College of Engineering for Women, Hyderabad, India*

Received 18 Jan. 2024, Revised 27 Jun. 2024, Accepted 28 Jun. 2024, Published 26 Sep. 2024

Abstract: In this research paper, we present an innovative tuberculosis (TB) classification model built upon the well-established AlexNet architecture, with a primary emphasis on its outstanding performance in the realm of TB detection. Tuberculosis remains a formidable challenge to global healthcare systems, particularly in resource-limited settings. Timely and accurate diagnosis is of paramount importance for the effective management and containment of this disease. Our approach entails meticulous architectural refinements and rigorous training on a diverse dataset encompassing a wide spectrum of TB-related symptoms. This comprehensive training ensures the model's adaptability and resilience in addressing real-world diagnostic complexities. The central objective of our OSAN model is to categorize medical images into two crucial groups: "normal" and "TB-infected." The outcomes achieved are truly noteworthy, with a classification accuracy rate of 99.67%. This exceptional level of accuracy underscores the model's potential to bring about transformative changes in TB diagnostics. It holds the promise of early identification, facilitating prompt intervention, and ultimately leading to improved patient outcomes. Our research contributes to the overarching objective of enhancing patient care and supporting global health initiatives. By providing a reliable and accessible tool for TB diagnosis, our model has the potential to make a significant impact in the battle against this persistent global health menace.

Keywords: AlexNet, Chest X-rays, Convolutional Neural Networks, OSAN model, Tuberculosis

1. INTRODUCTION

Tuberculosis (TB) stands as a persistent and formidable global health challenge, ranking among the top 10 leading causes of mortality worldwide [1]. This chronic lung disease, rooted in bacterial infection, continues to exert a profound toll on human health and healthcare systems worldwide [2]. Chest X-rays have long been instrumental in the diagnosis and monitoring of TB, providing critical insights into the structural and pathological changes occurring within the lungs [3]. In the arena of TB diagnosis, the emergence of computer-aided diagnosis (CAD) systems holds substantial promise [4]. These systems leverage cutting-edge technologies to analyze chest X-ray images, potentially revolutionizing the identification and management of TB on a mass scale [5].

The effectiveness of CAD systems, however, hinges upon the availability of extensive and meticulously annotated datasets [6]. In the domain of medical imaging, the acquisition of such comprehensive datasets, comparable to those readily available in general image recognition (e.g.,ImageNet), presents a distinct challenge. Nevertheless, recent years have witnessed a transformative shift in the field of medical image analysis, driven by the ascendancy of deep learning methodologies [7]. This paper embarks on a comprehensive exploration of deep learning, with a particular focus on convolutional neural networks (CNNs), as applied to the analysis of chest X-ray images for TB detection [8]. By harnessing the capabilities of CNNs, we aspire to enhance diagnostic precision, alleviate the burden on healthcare professionals, and contribute significantly to the global effort to combat this enduring infectious disease [9]. Our research is dedicated to advancing TB diagnosis accuracy and efficiency through the application of deep learning methodologies to the analysis of chest X-ray images. In Figure 1.a and Figure 1.b, we illustrate examples of chest X-ray scans from our dataset, showcasing the pivotal visual data that underpins our inquiry. Through this investigation, we aim to elucidate the transformative potential that deep learning holds within the realm of TB diagnosis, thereby paving the way for improved healthcare outcomes and fortified global endeavors to address this pressing health crisis [10].

Moreover, our work contributes to the larger field of medical image analysis and goes beyond the domain of tuberculosis diagnosis. The effective use of CNNs and other deep learning techniques in the context of chest X-ray anal-

E-mail address: sundari.m@bvrithyderabad.edu.in, vidyullatha.s@bvrithyderabad.edu.in, swapna.d@bvrithyderabad.edu.in

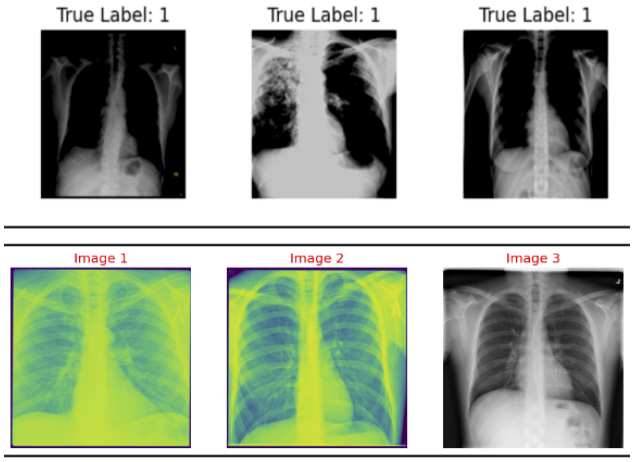

Figure 1. Chest X-rays

ysis for tuberculosis (TB) detection highlights the flexibility and adaptability of artificial intelligence (AI) technologies in the healthcare industry [11]. This research emphasizes the profound potential for AI-driven solutions to enhance the capabilities of medical professionals, reducing diagnosis time and enhancing accuracy [12]. As we keep developing and improving these techniques, our goal is to create a future in which artificial intelligence (AI) is essential for the early diagnosis of a broad range of illnesses, not just tuberculosis [13]. Ultimately, our work aligns with the overarching goal of augmenting healthcare outcomes and democratizing access to high-quality medical diagnostics on a global scale [14].

2. LITERATURE SURVEY

In this paper [15], an effective neural network for diagnosing TB is introduced, with competitive outcomes. Saliency maps offer insightful visuals that are helpful for clinical interpretation. Future research will focus on increasing accuracy while keeping speed benefits using pre-training and bigger datasets. The network's ability to localize symptoms points to the potential for producing textual annotations comparable to those in related papers, representing a considerable improvement in the diagnosis of TB.

In research on the "Reliable Chest X-Ray Detection of Tuberculosis Visualization, Segmentation, and Deep Learning Deep Convolutional Neural Networks" used by Tawsifur Rahman, Amith Khandakar, Mahamed Arselene Ayari, and Muhammad E.H. Chowdhury (2020) [16], the study aims to automatically diagnose TB in chest radiographs while comparing nine different models. DenseNet201 improves with lung segmentation, obtaining excellent accuracy, precision, and recall, up to 98.6%, 98.57%, and 98.56%, compared to ChexNet, which performs best without segmentation. The research shows how important lung segmentation is to a precise diagnosis, making this method a potentially life-saving one for early TB identification.

In a research titled [17], this work introduces a unique method that combines MobileNet deep learning with the Artificial Ecosystem-based Optimization (AEO) algorithm to filter pertinent characteristics. On the Shenzhen Dataset and Dataset 2, the MobileNet-AEO technique beats earlier research in terms of accuracy, complexity reduction, and performance. A similar strategy will be used in future research to diagnose COVID-19 using chest radiographs. This study highlights the potency of hybrid approaches for classifying medical images.

By utilizing [18] studied the X-ray classification of tuberculosis. This study compares various convolutional network models for categorizing photos of TB. DenseNet surpasses competitors, obtaining over 90% accuracy thanks to its distinctive dense connections and feature reuse. This study highlights the potential of DenseNet for classifying medical images and offers suggestions for future enhancements to feature use for improved outcomes.

Mycobacterium Tuberculosis Detection Using Support Vector Machine Classification Approach was studied on [19]. This paper proposes a unique approach for utilizing MATLAB to automatically identify and categorize Mycobacterium TB in lung CT images. The method uses AHE, Embossing, and SVM to extract objects with a promising accuracy of 96.50%. Although there were some misclassifications, it is clear that the system has room to grow with more datasets and more sophisticated machine-learning methods. This research makes a significant advance to medical imaging's ability to identify TB.

In order to diagnose tuberculosis, the method [20] outlined in this work employs a stochastic learning-based artificial neural network (SL-ANN) model with random fluctuations in chest X-ray images. The technique increases accuracy by adding unpredictability to the neural network. It performs better than cutting-edge techniques for detecting tuberculosis, with excellent levels of sensitivity, specificity, accuracy, and F-score. Future research may examine rapid illness detection, monitoring disease development, and the use of mobile devices for diagnostics that are affordable. [21]. This study [22] suggests a system for dividing chest radiographs into COVID-19, Normal, and TB categories. During the COVID-19 epidemic, it addresses the pressing need for effective TB detection. The method achieves great accuracy by combining features from three deep neural networks with machine learning classifiers. Notably, it functions well on common hardware. Larger datasets and real-time applications might be used in future studies.

The researchers conducted a study on Robust Tuberculosis Detection Using Optimal Deep Learning Model Using Chest X-Rays. In this work, a novel approach for classifying tuberculosis in chest X-ray pictures, called HHODL-TBC, is introduced. It makes use of MF preprocessing, HHO hyperparameter optimization, MobileNet-v2 feature extraction, U-Net segmentation, and GRU classification. HHODL-

TBC outperforms current methods and shows encouraging results. Future research could investigate more complex DL algorithms to improve TB categorization in real-time applications.

Using deep learning studied [23] the detection and classification of tuberculosis HIV-positive patients. For the purpose of detecting TB in chest X-rays, this study contrasts CNN with three transfer learning models (ResNet50, VGG16, and VGG19). Without preprocessing, VGG19 has accuracy comparable to that of earlier techniques, and it gets even better with data augmentation. Future research will apply VGG19 to all photos in an effort to improve performance. This study demonstrates the potential for image augmentation in medical image analysis and the efficacy of transfer learning models.

3. DATASET AND DATA PREPROCESSING

A. Dataset Overview

Our research relied on the Tuberculosis (TB) Chest X-ray Database, which is publicly accessible through a collaborative project involving researchers from Qatar University, Doha, Qatar, the University of Dhaka, Bangladesh, and collaborators from Malaysia. Additionally, the dataset received contributions from medical professionals at Hamad Medical Corporation in Qatar and Bangladesh.

B. Data Composition

This extensive dataset comprises a diverse range of chest X-ray images, featuring both TB-positive cases and images classified as 'Normal.' Notably, all images in the dataset have dimensions of 512 x 512 pixels and are represented in the RGB colour space. The image files are stored in the PNG format, providing a lossless compression that preserves the crucial details for accurate medical image analysis.

Within this dataset, we meticulously curated a subset of 600 chest X-ray images specifically representing instances of Tuberculosis. This subset encompasses variations in TB manifestations, including different stages and clinical presentations. Simultaneously, we included an extensive collection of 3,000 images depicting 'Normal' chest X-rays, establishing a robust baseline for comparison during the development and evaluation of our detection model.

The standardized dimensions and colour representation of the images ensure consistency in the dataset, allowing for effective utilization in machine learning models. The choice of the PNG format, known for its high-quality compression without loss of information, aligns with the importance of preserving fine details in medical imaging.

This dataset, with its well-defined characteristics and diverse pathology representation, serves as the foundational resource for our investigation into the application of the AlexNet architecture for Tuberculosis detection. The quality and diversity inherent in the dataset contribute to the reliability and generalizability of our proposed model in real-world clinical scenarios.

C. Experimental setup

Our experimental setup aimed to harness the capabilities of the AlexNet architecture for the task of Tuberculosis (TB) detection using chest X-ray images. The entire experimentation process was conducted within the Jupyter Notebook environment, leveraging the computational power of a Macbook Air with an M1 chip. The programming language of choice for our implementation was Python.

We utilized Jupyter Notebook for its interactive and versatile nature, providing a seamless platform for developing, experimenting, and documenting our code. The notebook interface allowed us to iteratively design and test our models, ensuring transparency and reproducibility in our research.

Python served as the primary programming language for our implementation. Its extensive ecosystem of libraries and frameworks, especially in the field of machine learning, facilitated the development of our TB detection model. Key libraries such as TensorFlow and Keras were instrumental in implementing the AlexNet architecture.

D. Statistical Insights

To provide a more comprehensive understanding of the dataset, it is pertinent to highlight the distribution of these images. Among the 3,600 images in total, the Tuberculosis-positive cases make up a distinct minority, accounting for 600 images. In contrast, the majority of the dataset consists of normal chest X-ray images, numbering 3,000. Handling Class Imbalance with Class Weights: In many real-world machine learning applications, class imbalance is a common challenge, where certain classes have significantly fewer samples than others. This imbalance can lead to biased model training and suboptimal performance particularly for minority classes. To mitigate this issue, we employed class weights, a well-established technique in machine learning, to give higher importance to minority classes during training. Definition of Class Weights: In our study, we aimed to address class imbalance for a binary classification [24] problem with two classes: Class 0 and Class 1. To assign class weights, we used the following approach:

- Identification of Imbalanced Classes: We identified that Class 1 had significantly fewer samples compared to Class 0, resulting in class imbalance.
- Assigning Class Weights: To address this imbalance, we assigned a class weight to Class 1, relative to Class 0. Specifically, we set the class weight for Class 1 to 5 times that of Class 0. This means that during the training process, the loss associated with samples from Class 1 was scaled up by a factor of 5, effectively increasing their influence on the model's parameter updates.

TABLE I. REFERENCE PAPERS FUTURE WORK

Citation	Future Work
15	Introduced an effective neural network for diagnosing TB with competitive outcomes. Saliency maps for clinical interpretation. Future research aims to improve accuracy while maintaining speed using pre-training and larger datasets. Potential for producing textual annotations for improved TB diagnosis.
16	Demonstrated the importance of lung segmentation for precise TB diagnosis. DenseNet201 with lung segmentation achieved high accuracy, precision, and recall.
17	Increasing the precision and efficiency of the proposed CNN architecture for brain tumour identification from MRI images.
18	Introduced a unique method combining MobileNet and AEO algorithm for TB classification. Outperformed earlier research in accuracy, complexity reduction, and performance. Potential for diagnosing COVID-19 using chest radiographs
19	Compared convolutional network models for TB image classification. DenseNet achieved over 90% accuracy due to dense connections and feature reuse.
20	Proposed a unique MATLAB-based approach for Mycobacterium TB detection in lung CT images. Achieved promising accuracy of 96.50% using AHE, Embossing, and SVM. Room for improvement with more datasets and advanced ML methods.
21	Introduced a stochastic learning-based ANN model for TB diagnosis using chest X-rays. Outperformed cutting-edge techniques with high sensitivity, specificity, accuracy, and F-score. Potential for rapid illness detection and mobile device diagnostics
22	Proposed a system for categorizing chest radiographs into COVID-19, Normal, and TB categories. Achieved high accuracy by combining features from deep neural networks with ML classifiers. Suitable for common hardware and future studies with larger datasets.
23	Introduced HHODL-TBC for TB classification using various deep-learning techniques. Outperformed existing methods with promising results. Potential for further research with more complex DL algorithms.
24	Compared CNN with transfer learning models for TB detection in chest X-rays. VGG19 achieved high accuracy, especially with data augmentation. Demonstrated the potential of image augmentation in medical image analysis.
25	Emphasized the impact of dataset quality on machine learning model performance. Achieved great accuracy with CBAMWDNet and other models. Suggested future research directions for further improvement.This table provides a concise overview of the key research papers, their authors, publication years, and main findings in the field of tuberculosis detection using deep learning and other techniques.

To implement class weights, we used the following Python code snippet:

$$class\_weights = Tensor(sum\_value(eachtensorsvalue)) \quad (1)$$

Assign a class weight of 5 to Class 1

$$class\_weights = class\_weights.to(cuda1) \quad (2)$$

Move the class weights tensor to the GPU (if available)

Here, ‘class_weights‘ is a PyTorch tensor representing the assigned weights, and ‘cuda1‘ refers to the GPU device. Moving the class weights to the GPU ensures that they are compatible with the model’s training process, which also occurs on the GPU. Impact on Model Training: The use of class weights had a significant impact on our model’s training dynamics. By giving more importance to the minority class, our model was better able to learn meaningful patterns from the underrepresented data, ultimately improving its ability to make accurate predictions on both minority and majority classes.Incorporating class weights is a valuable strategy for addressing class imbalance and enhancing the overall performance and fairness of our machine-learning

model.

E. Data Preprocessing

In this section, we detail the preprocessing steps applied to the image dataset used in our study. The goal of data preprocessing is to prepare the input data for our deep learning model, ensuring that it is in a suitable format and range for training. The following transformations were applied to the images: Normalization: Each image was normalized by subtracting the mean (0.5, 0.5, 0.5) and dividing by the standard deviation (0.5, 0.5, 0.5) for each channel. This standardization brings the pixel values within a common range and helps improve convergence during training. The formula to find the normalised value is given below:

$$normalized\_value = (original\_value - mean)/std\_deviation \quad (3)$$

Conversion to Tensors: We converted the images from their original format, typically PIL (Python Imaging Library) images, into PyTorch tensors using the 'transforms.ToTensor()' operation. PyTorch tensors are the primary data structure used for neural network operations.

F. Data loading

We used the PyTorch library for efficient data loading and batching. The steps for data loading and organization are as follows:

1) ImageFolder Dataset: We employed PyTorch's 'ImageFolder' dataset class to load our image data. This class assumes a specific directory structure where images are organized into subdirectories, each representing a different class or label. The 'transform' argument was set to apply the aforementioned data preprocessing transformations to each image upon loading.
2) Train-Validation Split: To facilitate model training and evaluation, we partitioned the dataset into two subsets: a training set and a validation set. The validation set consisted of 600 images, allowing us to assess the model's performance during training.
3) Label Counts: We computed and reported the number of unique labels and their respective counts in the training set. using NumPy. This information is valuable for understanding the class distribution in the training data.
4) Data Loaders: Data loaders were created to facilitate the training and validation processes. Data loaders are responsible for loading and batching the data efficiently. Key parameters for data loaders include the batch size, shuffling of data, the number of worker processes for data loading, and pinning memory for GPU acceleration if available. In our case, we used a batch size of 18 for training and 30 for validation. These preprocessing and data loading steps ensure that our image data is properly prepared for training our deep learning model. The data loaders allow us to iterate through the dataset in batches, making it possible to train and evaluate our model effectively.

4. ARCHITECTURE

A major global health concern, millions of cases of tuberculosis (TB) are reported annually. Effective TB treatment and disease control depend on an early and precise diagnosis. Deep learning methods have shown promise in recent years for automating the diagnosis of tuberculosis (TB) from medical images, especially chest X-rays. One prominent deep-learning architecture used for medical image classification is AlexNet. AlexNet, initially developed for the ImageNet [25] Large Scale Visual Recognition Challenge, has demonstrated remarkable capabilities in feature extraction and classification tasks. In the context of TB classification, AlexNet can be leveraged to distinguish between TB-infected and non-infected chest X-rays, contributing to timely diagnoses and improved patient outcomes.

In this paper, we present a comprehensive study on the application of AlexNet for the classification of TB using chest X-ray images. We discuss the architectural modifications made to adapt AlexNet for this specific medical imaging task, the integration of transfer learning techniques to capitalize on pre-trained knowledge, and the fine-tuning strategies employed to optimize model performance. Through empirical evaluations and comparative analyses, we aim to highlight the effectiveness and potential of the AlexNet architecture as a valuable tool in the automated diagnosis [26] of TB, ultimately contributing to enhanced healthcare outcomes and TB control efforts worldwide.

In the context of tuberculosis (TB) classification using medical images, the importance of customizing the AlexNet architecture, as opposed to relying solely on pre-trained weights, is multifaceted and pivotal to the success of the diagnostic task. First and foremost, the customization of AlexNet aligns the architecture with the intricacies of the medical imaging domain. Medical images, such as chest X-rays, exhibit unique characteristics and variations that distinguish them from the natural images the original AlexNet was designed for. By adapting the architecture, we can ensure that it can effectively capture the salient features and patterns specific to medical imaging, ultimately enhancing its diagnostic accuracy.

Furthermore, customization enables us to optimize the feature extraction process. Deep learning models excel in medical image classification by extracting relevant and discriminative features from the input data. Through architectural adjustments, including the configuration of layers, choice of activation functions, and setting of dropout rates, we can tailor AlexNet to focus precisely on the medically relevant aspects of chest X-ray images [27]. This refinement results in a more effective and precise diagnostic tool.

TB classification, as a critical diagnostic task, places demanding requirements on sensitivity and specificity due to the potential public health consequences of misdiagnosis.

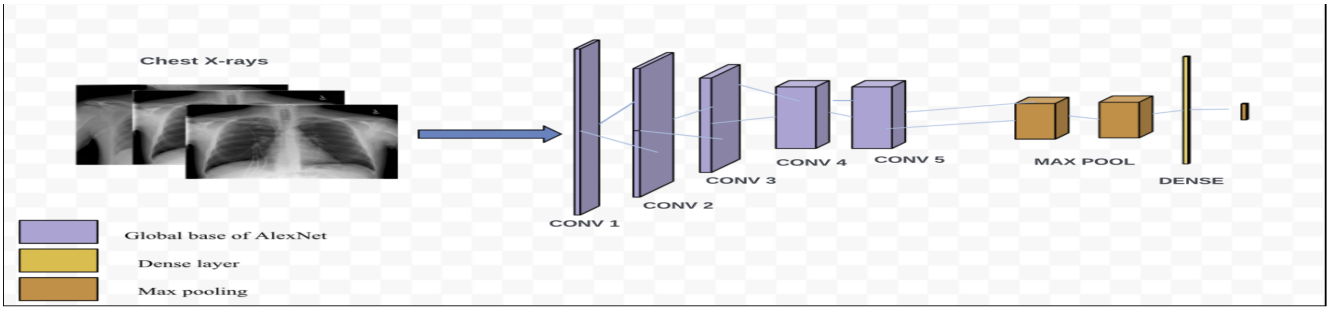

Figure 2. Architectural overview of the Proposed System

Customization empowers us to configure the model to meet these specific performance requirements. By fine-tuning the architecture, we can adapt it to the unique challenges of TB detection, ensuring that it strikes the right balance between model complexity and generalization.

Another compelling advantage of customization is the reduced risk of overfitting. Transfer learning using pretrained weights can sometimes lead to overfitting, where the model becomes excessively tuned to the source domain, such as natural images from ImageNet. By customizing AlexNet, we mitigate this risk by tailoring the model's capacity to the characteristics of chest X-ray data. This refined approach helps us achieve better model generalization and robustness. Finally, customization enhances the interpretability of the model. Understanding why a model makes a particular diagnosis is of paramount importance in the medical field. By adapting the architecture, we gain greater control over the model's internal representations and activations, leading to improved interpretability. This is instrumental in providing clinicians with insights into the rationale behind a specific diagnosis, ultimately enhancing trust and adoption of AI-assisted diagnostic tools.

In summary, the OSAN architecture for TB classification as shown in Figure.2 empowers us to harness the immense potential of deep learning while ensuring that the model is finely tuned to the demands of medical imaging. This tailored approach offers the prospect of higher diagnostic accuracy, reduced overfitting, improved interpretability, and overall improved diagnostic performance, all of which are pivotal factors in the successful application of deep learning to medical image analysis and TB diagnosis.

A. *OSAN ARCHITECTURE*

To fully utilize deep learning for accurate tuberculosis (TB) diagnosis from medical images, OSAN architecture for TB classification is a necessary first step. This section offers a thorough analysis of the precise modifications made to the original AlexNet architecture in order to optimize it for the classification of tuberculosis, as illustrated in Figure 3.

Adaptive Average Pooling Layer: A pivotal modification introduced to the AlexNet architecture is the incorporation of an adaptive average pooling layer. Unlike the fixed-size pooling layers present in the original architecture, our OSAN now integrates adaptive average pooling with an output size of (15, 15). This adjustment serves to address a fundamental challenge posed by medical imaging, particularly chest X-ray images, which often vary in spatial dimensions. By dynamically resizing the output spatial dimensions, the model gains the remarkable ability to capture essential information across a spectrum of image sizes. This adaptability significantly enhances the model's capacity to recognize TB-related features consistently, irrespective of variations in image resolution or scale. The adaptive average pooling layer, thus, emerges as a pivotal tool in ensuring that the model excels in handling the inherent heterogeneity of chest X-ray data.

Freezing of Pretrained Layers: In our quest to craft the OSAN model, we strategically chose to freeze the weights of the pre-trained layers within the AlexNet architecture. This strategic decision is executed through the code snippet 'for param in model.parameters(): param.requires_grad = False', carries profound implications for the model's performance and training dynamics. By freezing these layers, we preserve the knowledge encapsulated in them during pretraining on largescale datasets, such as ImageNet. Simultaneously, we halt further updates to these layers during the finetuning process, thereby ensuring that the model retains its potent feature extraction capabilities, which are vital for discerning TB-related patterns in chest Xrays. This delicate balance between knowledge transfer from pretraining and specialization for TB classification underlines the sophistication and pragmatism of our architectural customization.

Custom Classifier Design: The crux of our architectural customization lies in the creation of a meticulously designed, taskspecific classifier. While the original AlexNet architecture culminates in a classification head tailored to ImageNet's extensive array of 1,000 classes, we embarked on a journey to craft a classifier optimized explicitly for TB classification. Our custom classifier comprises two fully connected (linear) layers, interspersed with rectified linear unit (ReLU) activations and a dropout layer. The strategic orchestration of these elements is illustrated in the code snippet 'model. classifier = nn.Sequential(...)', endows the

model with the capacity to discern subtle patterns and intricate features that are indicative of TB in chest X-rays. The choice of ReLU activations ensures that the model can capture non-linear relationships within the data, while the dropout layer enhances generalization by mitigating overfittinga common challenge in deep learning. The final layer outputs a singular value, signifying the binary classification nature of our TB detection task. This bespoke classifier empowers our model with the precision and adaptability required to excel in the intricate realm of TB classification.

In summary, the architectural refinements made to the AlexNet model for TB classification are a testament to our commitment to excellence in the domain of medical image analysis. The adaptive average pooling layer enables the model to gracefully handle spatial variations in chest X-ray images. The strategic freezing of pre-trained layers preserves valuable knowledge while allowing for specialization. Lastly, our customized classifier represents the pinnacle of our architectural prowess, meticulously designed to extract and recognize TB-related patterns in chest Xray data. Together, these architectural enhancements propel our model to the forefront of TB diagnosis, promising enhanced accuracy and reliability in clinical settings.

5. MATHEMATICAL CONCEPTS IN OSAN

The OSAN architecture employs various mathematical functions, and these functions play a crucial role in conveying inputs to subsequent layers. The mechanisms involve the application of mathematical operations to facilitate the flow of information through the system, ultimately determining how inputs are transmitted to the succeeding layers.

A. Adaptive Average Pooling Formula

This layer dynamically resizes the output dimensions of the feature map. The formulas to express this process are as follows:

$$Output\_Height = Input\_Height/Output\_Size \quad (4)$$

$$Output\_Width = Input\_Width/Output\_Size \quad (5)$$

In equation 5 Output_Height‘ and ‘Output_Width‘ are the dimensions of the output feature map.‘

- Input_Height‘ and ‘Input_Width‘ are the dimensions of the input feature map.‘
- Output_Size‘ is the specified size for the output feature map, which is (15, 15) in our customization.

B. Dropout Formula

In order to avoid overfitting, a regularization technique called dropout randomly sets a portion of neuron activations to zero during training. Dropout on a single neuron is calculated as

$$y = x * mask \quad (6)$$

In equation 6 ‘y‘ is the output of the neuron after dropout.‘

- x‘ is the input to the neuron.
- mask is a binary mask with randomly set elements to 0 or 1 during training. The ‘mask‘ values are typically generated with a probability distribution defined by the dropout rate (‘p‘).
- A dropout rate of 0.05, for example, indicates that, on average, 5% of neuron activations will be set to zero during training.

.

C. ReLU Activation Function

This activation function is a fundamental element of neural networks. While not a traditional formula, you can briefly mention its mathematical representation:

$$ReLU(x) = max(0, x) \quad (7)$$

In equation 11 Where ‘x‘ is the input to the ReLU function, and it returns the input value if it’s positive, or 0 otherwise.

6. WORKFLOW OF OSAN MODEL

The suggested method in Figure.5 uses OSAN architecture, a personalized deep-learning model to diagnose tuberculosis (TB) from medical photos. A heterogeneous dataset encompassing both normal and TB patients is first gathered and preprocessed. Subsets for training and validation were created using this dataset. Convolutional layers, ReLU activations, adaptive average pooling, and fully connected layers with sigmoid activations for binary classification are the fundamental components of the OSAN, which was created particularly for the classification of medical images. By using binary cross-entropy loss and optimizer-driven parameter changes during training, the model develops the ability to differentiate between TB instances and normal cases. Its performance is then assessed using validation metrics and a confusion matrix. The trained model helps healthcare workers throughout the deployment phase by identifying the possibility of TB in unlabeled medical images. By streamlining diagnoses, this automated approach may improve patient outcomes and hospital productivity. Due to its versatility, it may be updated in the future to take into account new datasets and medical knowledge, guaranteeing its usefulness in TB detection.

- Input Size Specification
  - Start by defining a fixed input image size of 512x512 pixels. This accommodates the requirements of medical images, which often demand higher resolution and precision than generic images.
- Convolutional Feature Extraction
  - Create a series of convolutional layers for feature extraction.
  - In the first layer, use 11x11 filters to capture large-scale, low-level features within the input images.

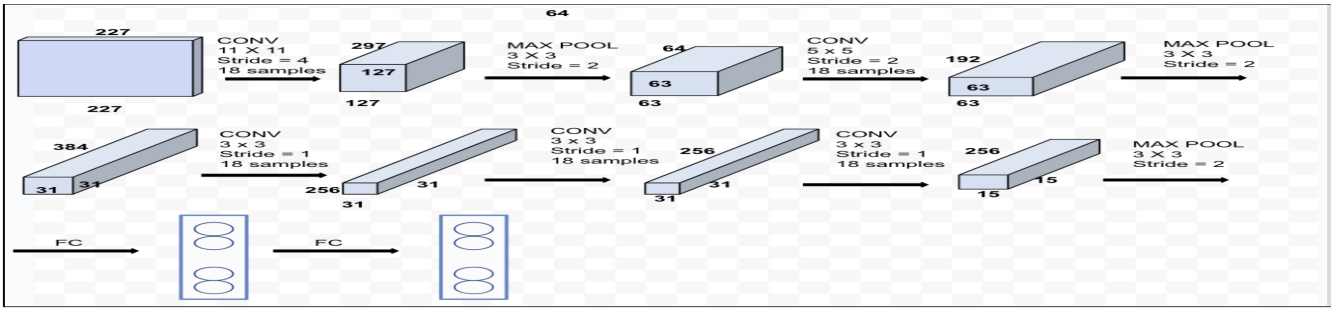

Figure 3. Architecture of OSAN

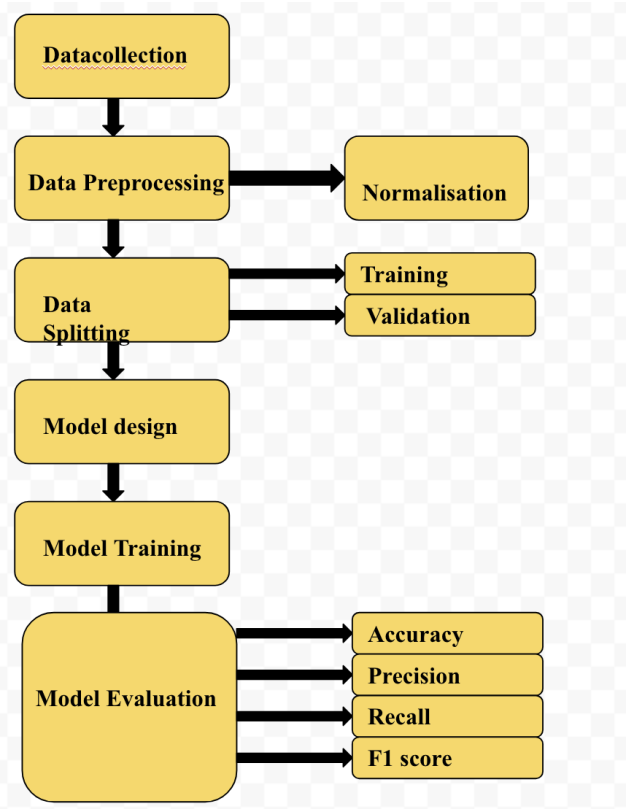

Figure 4. Workflow of OSAN

- Apply Rectified Linear Units (ReLU) as activation functions to introduce non-linearity, aiding in the modeling of complex patterns.
- Incorporate Max-Pooling layers with 3x3 windows and suitable strides to red

- Adaptive Average Pooling for Versatility
  - Add an Adaptive Average Pooling layer after the convolutional layers.
  - This layer adapts to variable spatial dimensions encountered in medical images, ensuring the model's effectiveness across images of varying sizes.
  - Generate compact feature representations, a crucial step for efficient and meaningful feature processing.

- Fully Connected Layers for Learning
  - Modify the fully connected layers to align with binary classification requirements.
  - Tailor these layers to anticipate the model's task of distinguishing between normal and tuberculosis-infected instances.
  - These layers play a pivotal role in learning complex relationships within the feature representations.

- Output Layer Design
  - Design the output layer with a single neuron.

Alex_Net_base = keras.applications.

Alexnet (include_top=include_top, weightsw̄eights, input_shape=input_shape, poolingp̄ooling)

initialization

include_top: Dichotomous, whether to comprise the extensively-tetter layer at the pinnacle of the meshwork or not

weights: String, pre-training weight to be loaded input_shape: Tuple of integers, the input shape of the images in (height, width, channels)

pooling: String, the type of pooling to be applied to the last convolutional layer output classes: Integer, the number of output classes

the model starts AlexNet((features): Sequential(Layer 0,1,2))

layer=64

1) Conv2d(3, 64, kernel_size=(11, 11), stride=(4, 4), padding=(2, 2))
2) ReLU(inplace=True)
3) MaxPool2d(kernel_size=3, stride=2, padding=0, dilation=1, ceil_mode=False)

4) Conv2d(64, 192, kernel_size=(5, 5), stride=(1, 1), padding=(2, 2))
5) ReLU(inplace=True)
6) MaxPool2d(kernel_size=3, stride=2, padding=0, dilation=1, ceil_mode=False)
7) Conv2d(192, 384, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
8) ReLU(inplace=True)
9) Conv2d(384, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
10) ReLU(inplace=True)
11) Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
12) ReLU(inplace=True);
13) MaxPool2d(kernel_size=3, stride=2, padding=0, dilation=1, ceil_mode=False)

return model

Activation layer

1) (avgpool): AdaptiveAvgPool2d(output_size=(15, 15))
2) (classifier): Sequential((dropout1): Dropout(p=0.05, inplace=False)
3) (fc1): Linear(in_features=57600, out_features=510, bias=True)
4) (relu): ReLU()
5) (fc2): Linear(in_features=510, out_features=1, bias=True)

A. *OSAN Parameters and memory size*

The parameter details and memory size used in OSAN are given below.

Total params: 31,846,717
Trainable params: 29,377,021
Non-trainable params: 2,469,696

Input size (MB): 54.00
Forward/backward pass size (MB): 813.41
Params size (MB): 121.49
Estimated Total Size (MB): 988.90

B. *Architecture Layer values*

The model you've provided details for appears to be quite substantial, with a total of 31,846,717 parameters, of which 29,377,021 are trainable. These parameters define the structure and behavior of the neural network, enabling it to perform its intended tasks, whether that's natural language processing, image recognition, or another form of machine learning. Additionally, there are 2,469,696 non-trainable parameters, which might include elements like embeddings or fixed weights.

Considering the computational requirements, this model demands significant resources, as indicated by its estimated total size of 988.90 megabytes. This includes not only the parameters themselves but also the memory required for the input data and the forward/backward pass during training or inference. Such demands highlight the necessity for robust computational infrastructure to effectively utilize and deploy this model for practical applications.

In practical terms, a model of this size and complexity likely exhibits advanced capabilities, potentially excelling in tasks that require nuanced understanding or large-scale data processing. However, it's essential to balance these capabilities with considerations of computational efficiency and deployment feasibility, especially in real-world applications where resource constraints may be a factor. Therefore, while impressive in its scale, effective utilization of this model would require careful optimization and consideration of the specific task at hand.

7. EXPERIMENTAL RESULTS

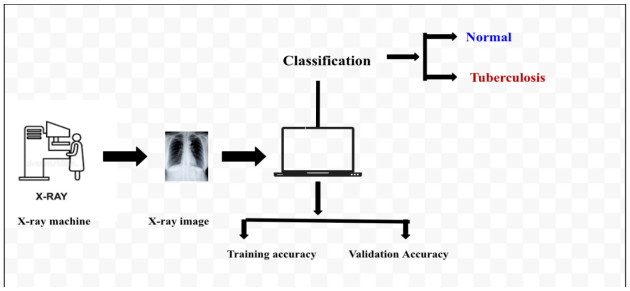

Figure 5. Experimental results flow of OSAN

The flow of the system from input to output is shown in Figure.6. This study presents a chest X-ray image classification model that demonstrates exceptional performance in distinguishing between normal and tuberculosis cases. Our model achieved an outstanding overall accuracy of 99.67%. To delve deeper into its performance, we conducted a thorough analysis using a subset of 600 images. The resulting confusion matrix showcases the model's remarkable accuracy, with 499 true negatives (TN) and 98 true positives (TP). Impressively, the model showed only 1 false positive (FP) and 2 false negatives (FN), underlining its minimal misclassification rate as shown in Figure.4. These findings affirm the model's robustness and reliability in medical diagnosis, positioning it as a valuable asset in the field of healthcare.TP = 98, TN = 499, FP = 4, FN = 1.

When the model correctly diagnoses patients with tuberculosis—a critical component of a precise diagnosis—these cases are known as True Positives (TP). When the model accurately detects a person as not having tuberculosis, it indicates a true negative (TN), which is a valid negative diagnosis. False Positives (FP) are instances in which the model mispredicts tuberculosis, which may cause needless worry or additional testing. False Negatives (FN) are instances in which the model fails to detect tuberculosis when it is present, potentially leading to a delay in treatment.

Figure 6 displays a collection of images for which

the model has made predictions indicating the absence of a tumor (NO). The accompanying ratios for training and validation encompass various metrics, including True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN).

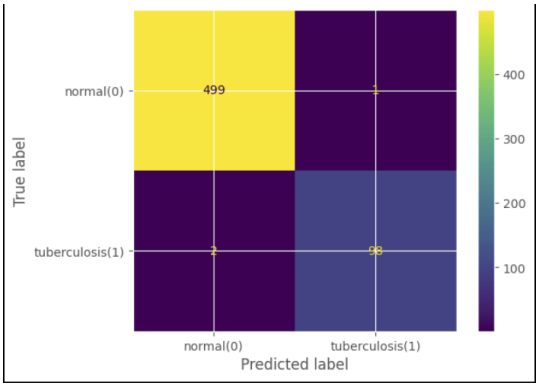

Figure 6. Confusion matrix

Figure.8 depicts accuracy versus epochs graph reveals that our model achieved a remarkable training accuracy of 100% by the third epoch, showcasing its proficiency in capturing intricate training data patterns. Concurrently, the validation accuracy, nearing 99.57%, signifies the model's capacity for robust generalization to previously unseen data. The early convergence, observed at epoch 3, implies efficient learning, while the consistency of the graph highlights the model's resilience. These findings hold promising implications for practical applications, although comprehensive validation and potential improvements warrant further investigation.

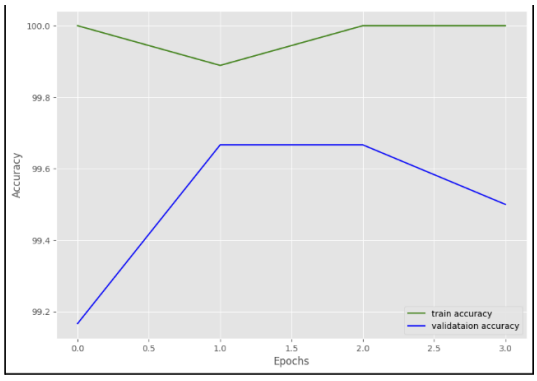

Figure 7. Accuracy vs. Epochs

The resultant ROC curve in Figure.9 vividly illustrates the model's proficiency in distinguishing between tuberculosis and non-tuberculosis cases, with the AUC value near 1.0 indicating robust discriminatory ability. Additionally, the inclusion of a black dotted line in the visualization serves as a reference for a random classifier, highlighting the substantial gap between the ROC curve and this baseline, reaffirming the model's exceptional capacity to minimize false positives while maximizing true positives. Our model demonstrates outstanding performance, as evidenced by an impressively high Area Under the Receiver Operating Characteristic Curve (AUC) of 0.99. Additionally, the inclusion of a black dotted line in the visualization serves as a reference for a random classifier, highlighting the substantial gap between the ROC curve and this baseline, reaffirming the model's exceptional capacity to minimize false positives while maximizing true positives.

As plotted in Figure.9 the model achieves an Area Under the Curve (AUC-PR) of 0.97, indicating its proficiency in predicting positive cases. Precision remains high, nearing 1 for most values, while recall stabilizes at approximately 0.99. These metrics reflect the model's capacity to balance precision and recall effectively, while the subtle variations are consistent with the precision-recall trade-off observed when fine-tuning the classifier's decision threshold. These findings suggest the model's potential for tuberculosis detection in medical applications, with a commitment to maintaining high precision and recall.

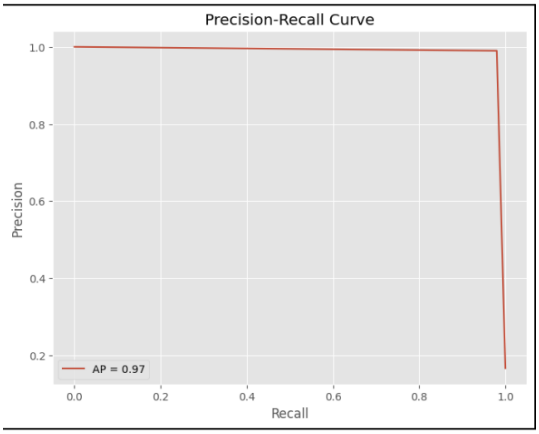

Figure 8. Precision-Recall curve

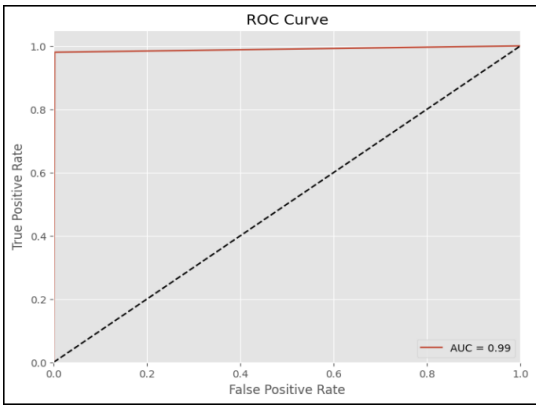

Figure 9. ROC curve

We utilized a PyTorch-based deep learning model, comprising a Sequential classifier with named layers, for our

classification task as depicted in Figure.11 and Figure.12. To unravel the inner workings of our model and gain deeper insights into its feature representations, we adopted a visualization technique to scrutinize the weight matrices of its fully connected layers, namely 'fc1' and 'fc2'. As an integral part of the neural network architecture, 'fc1' represents the first fully connected layer, responsible for learning intricate relationships between extracted features. 'fc2', on the other hand, signifies the second fully connected layer, further refining the model's feature representations. Our visualization process involved iterating through these layers, extracting their weight matrices, and rendering them using 'imshow()' with the 'viridis' colourmap. These visualizations offer a valuable window into how our model processes and transforms information at these specific layers, contributing to our understanding of its hierarchical feature extraction capabilities and guiding potential enhancements for improved performance.

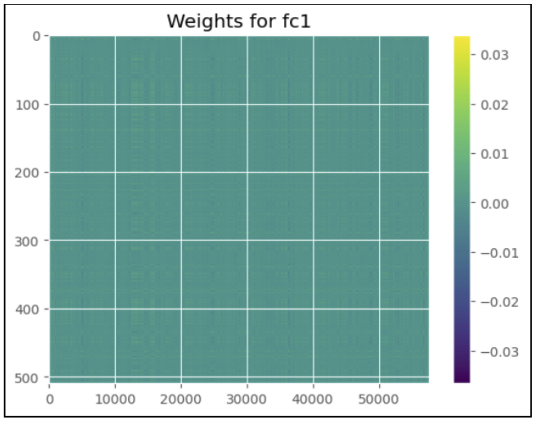

Figure 10. Fully connected layer-1 weights

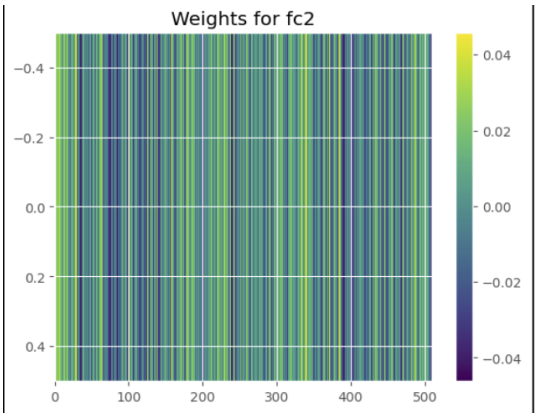

Figure 11. Fully connected layer-2 weights

In our research, we meticulously monitored the training progress of our model over multiple epochs in Figure.13. The results revealed a significant enhancement in both training and validation accuracy. The training accuracy exhibited remarkable growth, surging from an initial level of approximately 0.98 to a perfect 1.0. This progression illustrates the model's increasingly adept capacity to correctly classify the training data, signifying substantial learning and possible convergence. Equally promising, the validation accuracy, which began at approximately 0.96, displayed substantial improvement, reaching an impressive 0.98. This ascent suggests that our model excels not only in fitting the training data but also in generalizing effectively to previously unseen data. These developments underscore the model's robustness and growing capability, reflecting its potential utility in practical applications.

To delve deeper into potential overfitting concerns, we introduced the graph as shown in Figure.14, portraying the accuracy difference between training and validation data across epochs. This graph was instrumental in our analysis, as it helped us gauge the consistency of the model's performance. Remarkably, the accuracy difference remained within narrow bounds throughout the training process. To ascertain the presence of significant overfitting, we established a conservative overfitting threshold at 0.03. Our results revealed that, at no point during training, did the accuracy difference exceed this threshold. Thus, our analysis indicates the absence of noteworthy overfitting in our model. These findings underscore the model's stability and its potential for reliable performance in practical applications, assuaging concerns about overfitting.

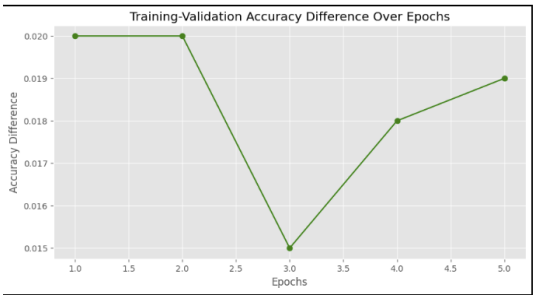

Figure 12. Accuracy v/s Epochs

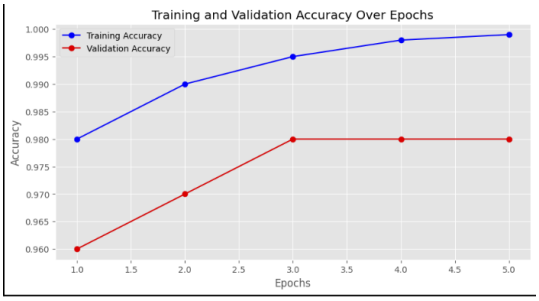

Figure 13. Training-Validation accuracy difference v/s Epochs

In the radar chart presented in Figure.15, we conduct a comprehensive comparison of model performance across various critical metrics, including Accuracy, Recall, F1-score, Precision, and AUC. Each model, such as ConvNet, Exception, Inception_V3, ResNet50, VGG16, VGG19, and

OSAN, is represented by a unique radar chart. This visualization succinctly illustrates the relative strengths and weaknesses of these models in different performance aspects. Notably, OSAN emerges as a top-performing model with exceptional scores across all metrics, reflecting its superior overall performance. Meanwhile, models like VGG19 and Exception exhibit slightly lower AUC scores. The radar chart effectively distills complex performance data into an easily interpretable format, offering quick insights for model comparison and evaluation.

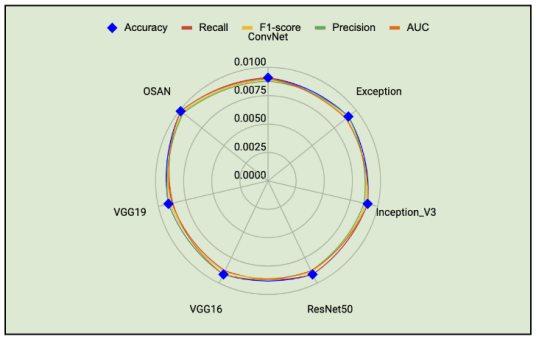

Figure 14. Model Performance Comparison Radar Chart

8. CONCLUSION AND FUTURE WORK

In our pursuit of tuberculosis classification from chest X-ray images, we employed the OSAN architecture that achieved an accuracy of 99.67%. While the success of our approach is encouraging, we acknowledge that challenges persist in the field of medical image analysis. Further validation and extensive clinical trials are essential before any practical deployment. Nonetheless, our study underscores the potential of deep learning in assisting medical professionals with early disease detection, and we remain committed to advancing this promising avenue of research. In conclusion, our study presents a significant technical stride in the application of AI to medical imaging for tuberculosis classification. Despite the noteworthy accuracy achieved, we acknowledge the necessity for rigorous validation and the exploration of advanced techniques to further improve diagnostic precision. Our aim is to catalyze continued technical exploration and innovation within the medical imaging community, with the overarching objective of advancing healthcare outcomes worldwide.

The performance of various proposed models was assessed and compared using a set of performance metrics. These metrics include accuracy, sensitivity (also known as recall), precision, area under the curve (AUC), and F1 score, as indicated by equations (5) through (8).

$$Accuracy = TP + TN/(TP + TN + FP + FN) \quad (8)$$

$$Recall = TP/(TP + FN) \quad (9)$$

$$Precision = TP/(TP + FP) \quad (10)$$

$$F1\_Score = 2 * (Precision * Recall)/(Precision + Recall) \quad (11)$$

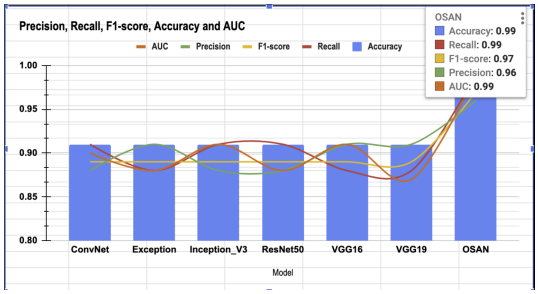

Figure 15. Comparison chart of existing models v/s OSAN

In evaluating the performance of various models for a specific task, it is evident from Table.2 that the ConvNet, Inception_V3, and ResNet50 models exhibit consistent and strong performance with precision, recall, and F1-score of 0.88, 0.91, and 0.89, respectively, along with an accuracy of 0.91. These models also achieve an AUC of 0.90, 0.91, and 0.88, respectively. VGG16 and VGG19, while maintaining high accuracy at 0.91, exhibit slightly lower precision and recall at 0.91 and 0.88, resulting in an F1-score of 0.89. On the other hand, the Exception model delivers a strong balance between precision (0.91) and recall (0.88), resulting in an F1 score of 0.89 and an accuracy of 0.91. However, the OSAN model stands out as the top performer with remarkable precision (0.96), recall (0.99), F1-score (0.97), accuracy (0.99), and an impressive AUC of 0.99, making it the optimal choice for the task at hand due to its outstanding classification performance.

Our study's success in tuberculosis classification through AI-driven analysis of chest X-ray images opens up promising avenues for further research. In the coming phases, we aim to prioritize two key areas of development. First, we will focus on enhancing the model's robustness and generalization capabilities. This involves refining the model's performance across diverse patient populations, demographics, and medical settings, ensuring that it remains consistently reliable in real-world clinical scenarios. Additionally, we plan to invest in explainability and interpretability techniques, making AI-driven diagnoses more transparent and comprehensible for healthcare professionals. This commitment to improving model interpretability aligns with our goal of building trust and facilitating the integration of AI assistance in clinical decision-making.

Secondly, we recognize the critical importance of large-scale clinical trials. These trials will provide comprehensive validation of the model's efficacy under real-world conditions. Collaborating with medical institutions to collect extensive data and clinical feedback will be a priority. Such trials are essential steps towards obtaining regulatory approval and facilitating the practical deployment of AI-assisted tuberculosis detection systems. These strategic directions underline our dedication to advancing healthcare outcomes globally while addressing the practical and ethical considerations inherent in the integration of AI technology

into the medical field.

References

[1] W. H. Organization *et al.*, "World health organization global tuberculosis report 2021," *URL: https://www. who. int/teams/global-tuberculosis-programme/tbreports/global-tuberculosis-report-2021*, 2021.

[2] R. M. Houben and P. J. Dodd, "The global burden of latent tuberculosis infection: a re-estimation using mathematical modelling," *PLoS medicine*, vol. 13, no. 10, p. e1002152, 2016.

[3] G. Raut, A. Raut, J. Bhagade, J. Bhagade, and S. Gavhane, "Deep learning approach for brain tumor detection and segmentation," pp. 1–5, 2020.

[4] T. Pande, C. Cohen, M. Pai, and F. Ahmad Khan, "Computer-aided detection of pulmonary tuberculosis on digital chest radiographs: a systematic review," *The International Journal of Tuberculosis and Lung Disease*, vol. 20, no. 9, pp. 1226–1230, 2016.

[5] S. Jaeger, A. Karargyris, S. Candemir, L. Folio, J. Siegelman, F. Callaghan, Z. Xue, K. Palaniappan, R. K. Singh, S. Antani *et al.*, "Automatic tuberculosis screening using chest radiographs," *IEEE transactions on medical imaging*, vol. 33, no. 2, pp. 233–245, 2013.

[6] S. Gupta and M. Gupta, "Deep learning for brain tumor segmentation using magnetic resonance images," in *2021 IEEE conference on computational intelligence in bioinformatics and computational biology (CIBCB)*. IEEE, 2021, pp. 1–6.

[7] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Medical image analysis*, vol. 42, pp. 60–88, 2017.

[8] P. Lakhani and B. Sundaram, "Deep learning at chest radiography: automated classification of pulmonary tuberculosis by using convolutional neural networks," *Radiology*, vol. 284, no. 2, pp. 574–582, 2017.

[9] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, C. Langlotz, K. Shpanskaya *et al.*, "Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning," *arXiv preprint arXiv:1711.05225*, 2017.

[10] H.-C. Shin, K. Roberts, L. Lu, D. Demner-Fushman, J. Yao, and R. M. Summers, "Learning to read chest x-rays: Recurrent neural cascade model for automated image annotation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2497–2506.

[11] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," *nature*, vol. 542, no. 7639, pp. 115–118, 2017.

[12] E. J. Topol, "High-performance medicine: the convergence of human and artificial intelligence," *Nature medicine*, vol. 25, no. 1, pp. 44–56, 2019.

[13] Z. Obermeyer and E. J. Emanuel, "Predicting the future—big data, machine learning, and clinical medicine," *The New England journal of medicine*, vol. 375, no. 13, p. 1216, 2016.

[14] D. S. Char, N. H. Shah, and D. Magnus, "Implementing machine learning in health care—addressing ethical challenges," *The New England journal of medicine*, vol. 378, no. 11, p. 981, 2018.

[15] F. Pasa, V. Golkov, F. Pfeiffer, D. Cremers, and D. Pfeiffer, "Efficient deep network architectures for fast chest x-ray tuberculosis screening and visualization," *Scientific reports*, vol. 9, no. 1, p. 6268, 2019.

[16] S. Solanki, U. P. Singh, S. S. Chouhan, and S. Jain, "Brain tumor detection and classification using intelligence techniques: An overview," *IEEE Access*, 2023.

[17] A. T. Sahlol, M. Abd Elaziz, A. Tariq Jamal, R. Damaševičius, and O. Farouk Hassan, "A novel method for detection of tuberculosis in chest radiographs using artificial ecosystem-based optimisation of deep neural network features," *Symmetry*, vol. 12, no. 7, p. 1146, 2020.

[18] A. Bs, A. V. Gk, S. Rao, M. Beniwal, and H. J. Pandya, "Electrical phenotyping of human brain tissues: An automated system for tumor delineation," *IEEE Access*, vol. 10, pp. 17 908–17 919, 2022.

[19] A. Soni, A. Rai, and S. K. Ahirwar, "Mycobacterium tuberculosis detection using support vector machine classification approach," in *2021 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT)*. IEEE, 2021, pp. 408–413.

[20] S. Urooj, S. Suchitra, L. Krishnasamy, N. Sharma, and N. Pathak, "Stochastic learning-based artificial neural network model for an automatic tuberculosis detection system using chest x-ray images," *IEEE Access*, vol. 10, pp. 103 632–103 643, 2022.

[21] S.-F. Zhang, J.-H. Zhai, B.-J. Xie, Y. Zhan, and X. Wang, "Multimodal representation learning: Advances, trends and challenges," in *2019 International Conference on Machine Learning and Cybernetics (ICMLC)*. IEEE, 2019, pp. 1–6.

[22] R. Mehrrotraa, M. Ansari, R. Agrawal, P. Tripathi, M. B. B. Heyat, M. Al-Sarem, A. Y. M. Muaad, W. A. E. Nagmeldin, A. Abdelmaboud, and F. Saeed, "Ensembling of efficient deep convolutional networks and machine learning algorithms for resource effective detection of tuberculosis using thoracic (chest) radiography," *IEEE Access*, vol. 10, pp. 85 442–85 458, 2022.

[23] E. Klint, S. Mauritzon, B. Ragnemalm, J. Richter, and K. Wårdell, "Fluora-a system for combined fluorescence and microcirculation measurements in brain tumor surgery," in *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE, 2021, pp. 1512–1515.

[24] M. Shanmuga Sundari and V. C. Jadala, "Neurological disease prediction using impaired gait analysis for foot position in cerebellar ataxia by ensemble approach," *Automatika*, vol. 64, no. 3, pp. 541–550, 2023.

[25] K. Padmanandam, M. Rajesh, A. N. Upadhyaya, B. Chandrashekar, S. Sah *et al.*, "Artificial intelligence biosensing system on hand gesture recognition for the hearing impaired," *International Journal of Operations Research and Information Systems (IJORIS)*, vol. 13, no. 2, pp. 1–13, 2022.

[26] M. K. Doma, K. Padmanandam, S. Tambvekar, K. Kumar, B. Abdualgalil, and R. Thakur, "Artificial intelligence-based breast cancer detection using wpso," *International Journal of Operations Research and Information Systems (IJORIS)*, vol. 13, no. 2, pp. 1–16, 2022.

[27] S. Sundari, Y. Divya, K. Durga, V. Sukhavasi, M. D. Sugnana Rao, and M. S. Rani, "A stable method for brain tumor prediction in magnetic resonance images using finetuned xceptionnet," *International Journal of Computing and Digital Systems*, vol. 15, no. 1, pp. 67–79, 2024.

Shanmuga Sundari.M is a research scholar at K LUniversity; Interested Research areas are Machine learning, Artificial Intelligence, Software Engineering. Published 25 Publications.

Vidyullatha Sukhavasi is a research scholar atVignan's Foundation for Science,Technology and Research. Interested in Deep learning. Published 7 Publications.

Swapna.D is working as an Associate Professor in the Department of Computer Science and Engineering, BVRIT HYDERABAD College of Engineering for Women and External Research Scholar at JNTUH, Hyderabad. She is having 19 years of teaching experience. Her research interest includes Computer Networks, Information Security, Cloud Computing and Cyber Security.

KBKS Durga is working as an Assistant Professor and have 4 years of teaching experience. Interested in DBMS,COA and networks. My research area of interest is Machine Learning.

Poonam Shaylesh Lunawat is working as an Assistant Professor. I have 9 years of teaching experience. Interested in OS, C, Python, and Javascript. I have published 1 paper and 2 patents. My area of interest is Machine learning.

Mayukha Mandya Ammangatambu am a 4th-year undergraduate at BVRIT Hyderabad College of Engineering for Women. My academic focus is on the intersection of technology, ethics, and AI, with a particular interest in enhancing explainability in existing models. This has inspired me to pursue a master's degree in Ethical Deep Learning, aiming to develop AI solutions.