Computer Methods and Programs in Biomedicine Update 6 (2024) 100169

Contents lists available at ScienceDirect

Computer Methods and Programs in Biomedicine Update

journal homepage: www.sciencedirect.com/journal/computer-methods-and-programs-in-biomedicine-update

An X-ray image-based pruned dense convolution neural network for tuberculosis detection

Edna Chebet Too [a,*], David Gitonga Mwathi [a], Lucy Kawira Gitonga [b], Pauline Mwaka [a], Saif Kinyori [a,c]

[a] Computer Science, Faculty of Science and Technology, Chuka University, Chuka, Kenya
[b] Nursing, School of Nursing and Public Health, Chuka, Kenya
[c] Information Communication and Technology, Faculty of Science and Technology, Chuka University, Chuka, Kenya

Emails: echebet@chuka.ac.ke; dgmwathi@chuka.ac.ke; lgitonga@chuka.ac.ke; skinyori@chuka.ac.ke

ARTICLE INFO

Keywords:
Pruned CNN
Tuberculosis detection
Densely convolution neural network
Deep learning
Image processing
Pruning

ABSTRACT

According to the Ministry of Health in Kenya, tuberculosis (TB) is the fifth greatest cause of death and the main infectious disease killer in Kenya and across the world. In Kenya and throughout Africa, TB continues to wreak havoc on many vulnerable populations, homes, and communities despite being preventable and treatable. Common TB diagnostics, like blood and skin tests, frequently fail to identify the precise kind of TB. As a result, the World Health Organization (WHO) advises expanding the use of X-rays, for screening. In TB-prevalent regions of Kenya, a shortage of radiologists hampers effective screening and diagnosis, highlighting the need for scalable solutions for accurate X-ray analysis.

Recent advancements in deep learning techniques have shown promise in the healthcare sector, particularly in radiology. However, many deep convolutional neural network (CNN) architectures are computationally intensive due to their size and resource requirements. This study designed and developed a Pruned CNN to address this issue by applying pruning techniques to baseline architectures. This approach significantly reduced model sizes while maintaining accuracy levels. Specifically, the pruned version of the DenseNet model achieved an impressive 99 % accuracy with a reduction rate of 65.8 %. These results highlight the potential of this pruned CNN as an effective and efficient tool for TB detection, particularly in resource-constrained environments. This study addresses the shortage of radiological expertise in many regions by providing a tool that can assist in the interpretation of X-ray images. This capability can help healthcare providers deliver timely and accurate diagnoses, thereby improving patient care.

1. Introduction

According to the World Health Organization, tuberculosis (TB) is one of the leading infectious disease killers worldwide and remains a significant cause of death in many countries [1]. It disproportionately impacts vulnerable populations, households, and communities, both in Africa and globally, despite being preventable and treatable. In Kenya, the Ministry of Health reports that TB is the largest infectious disease killer in the country and ranks as the fifth leading cause of death overall [2]. Although TB is preventable and treatable, it continues to have a devastating impact on vulnerable communities.

Conventional TB tests, such as the skin and blood tests, are often inadequate for determining the specific type of TB. The WHO recommends using chest X-rays for enhanced screening [1]. However, many TB-affected areas in Kenya face a shortage of radiological interpretation skills, which can hinder effective screening and follow-up efforts [3]. This shortage underscores the necessity for automated solutions that can assist healthcare providers in accurately diagnosing TB from X-ray images, ultimately improving patient outcomes in regions with limited access to expert radiologists. Implementing an automated and cost-effective technology in Kenya could significantly improve screening capabilities and enable earlier disease detection. Consequently, there has been growing interest in employing computer-aided diagnosis to analyze chest X-rays for TB diagnosis, leading to various innovative approaches.

Computer-assisted diagnostics solutions have mostly relied on traditional machine-learning approaches. Deep learning has grown rapidly over the previous decade, attracting a lot of attention in the field

* Corresponding author.
E-mail address: echebet@chuka.ac.ke (E.C. Too).

https://doi.org/10.1016/j.cmpbup.2024.100169

Available online 2 December 2024

of image processing and classification. Deep learning approaches can extract characteristics automatically without the need for human interaction, and they have shown excellent outcomes for real-world issues. Deep learning has recently gotten a lot of attention as a way to create a system that is autonomous, fast, and accurate. Deep Convolution Neural Network (CNN) is a deep learning system that is accurate for image categorization and computer vision in general. CNNs have repeatedly outperformed other traditional recognition algorithms for image-based classification and recognition issues, thanks to breakthroughs in deep learning. GoogLeNet [4], VGGNet [5], Deep Residual Network (ResNet) [6], Densely Connected Convolutional Network (DenseNet) [7], MobileNet [8], Xception [9], and EfficientNet [10] are examples of advanced CNN architectures that have significantly advanced the field of computer vision. Several of these architectures have won prestigious competitions, including the ImageNet Large Scale Visual Recognition Challenge, showcasing their effectiveness in image classification tasks.

Deep learning techniques have only recently been introduced to the healthcare sector, and their adoption has been limited. Deep learning has emerged as a powerful tool in the medical field, particularly in the diagnosis of complex diseases such as schizophrenia, Attention Deficit Hyperactivity Disorder (ADHD) [11], epilepsy [12], and COVID-19 [13, 14]. In schizophrenia, deep learning techniques are applied to neuroimaging data such as MRI and fMRI scans to identify subtle structural and functional brain abnormalities associated with the disorder. These models have shown promise in improving early diagnosis by automating the detection of patterns that are difficult to discern through traditional analysis. However, due to recent successes and promising outcomes, deep learning is gaining popularity in the field of radiology. Notable applications include the diagnosis of pleural effusion through chest radiography [15], detection of breast and cervical cancers [16] lung cancer detection [17], and brain tumor segmentation and detection [18, 19].

Despite recent advancements, training CNNs remains challenging due to high computational costs, making it an expensive endeavor. Even with relatively small image datasets, the training process can still be resource-intensive and time-consuming, often requiring substantial training time that can extend from days to weeks. This substantial demand for computational resources can limit the broader adoption and realization of deep learning techniques. Further, the demand to utilize lots of data slows down the advancement and the viability of exploiting deep learning. Additionally, the high inference time of the trained model limits its realization [7]. To reduce the time that it takes to develop deep learning models with eminent precision there is a bid to lessen the time linked to the training of deep learning networks. Furthermore, the problem of overfitting due to the depth of the network is still challenging the implementation of deep learning in real-world problems. Moreover, practical applications in real-world problems are still lacking since more work is focused on improving the networks using some benchmarking datasets [20].

Attempts to address the challenges associated with training deep learning models have led to various techniques, one of which is pruning. Pruning is a method used to reduce the size and complexity of neural networks by systematically removing weights, neurons, or entire layers that contribute minimally to the model's performance. This process aims to enhance computational efficiency, reduce memory usage, and improve inference speed, particularly in environments with limited resources, such as mobile and edge devices. By identifying and eliminating redundant parameters, pruning offers several advantages. First, it leads to a reduced model size, making it easier to store and deploy. Second, with fewer parameters to process, the model can deliver predictions more quickly, which is crucial for real-time applications. Additionally, smaller models consume less power, making pruning especially important for battery-operated devices. Furthermore, pruning can improve generalization by mitigating overfitting, as it removes noisy or irrelevant parameters and allows the model to focus on the most impactful features [21].

In this study, we develop a pruned dense convolutional neural network (PDenseNet) specifically for detecting Tuberculosis (TB). This method reduces the trained model size, making it more suitable for deployment. Additionally, it enhances inference speed and conserves energy without significantly compromising accuracy. The pruning technique is also applied to other state-of-the-art pre-trained CNN architectures, including Xception, EfficientNet, MobileNet, ResNet50, and DenseNet with 201 layers.

This study makes several key contributions to the field of Tuberculosis (TB) detection using deep learning:

- The development of a convolutional neural network specifically designed for TB detection, which enhances model efficiency.
- The introduction of a pruning method that effectively reduces model size while maintaining high accuracy, addressing a critical need for deployment in resource-limited settings.
- A systematic application of the pruning technique to various state-of-the-art pre-trained CNN architectures, including Xception, MobileNet, ResNet50, and DenseNet providing valuable insights into the effectiveness of pruning across different models.
- A prototype for detecting normal and TB cases has been created using real-world image data from hospitals in Kenya, showcasing the model's effectiveness in facilitating timely and accurate diagnoses in a clinical environment.

2. Literature review

Convolutional Neural Networks (CNNs) are the backbone of most deep learning approaches to image classification. The seminal work by Krizhevsky et al. [22]. introduced AlexNet, which achieved unprecedented accuracy on the ImageNet dataset, demonstrating the effectiveness of deep architectures and GPU acceleration. This was followed by significant innovations, such as VGGNet [5] and GoogLeNet [4], which introduced deeper architectures and more efficient designs, respectively. Subsequent advancements in CNN architectures have further enhanced their applicability in various domains. ResNet [6] further advanced the field by introducing residual connections, allowing for the training of very deep networks without the vanishing gradient problem. DenseNet [7] built upon this concept by connecting each layer to every other layer, promoting feature reuse and improving efficiency. MobileNet [8] introduced depthwise separable convolutions, allowing for lightweight models that are particularly suited for mobile and embedded applications. Xception [9] extended this concept with extreme inception modules, achieving state-of-the-art performance while maintaining efficiency. EfficientNet [10] introduced a compound scaling method that optimally balances depth, width, and resolution, setting new benchmarks in both accuracy and efficiency. These architectures have become standard benchmarks in image classification tasks. Deep learning has significantly advanced various applications, showcasing its versatility and effectiveness. In face recognition, architectures like DeepFace and FaceNet have improved accuracy through techniques such as facial embedding and triplet loss [23]. Object identification has benefited from models like YOLO and Faster R-CNN, enabling real-time detection with enhanced speed and accuracy [24]. In agriculture, deep learning has transformed plant disease classification, allowing for precise detection through specialized datasets and transfer learning [25,26]. Additionally, autonomous vehicles utilize deep learning for perception tasks, processing data from sensors to ensure safe navigation and decision-making [21]. Together, these advancements demonstrate how deep learning continues to drive innovation and improve efficiency across diverse fields.

The success of deep learning techniques has led to significant applications in the medical field, transforming various areas, including medical imaging, diagnosis, and personalized treatment. For instance, deep learning models are being used to analyze medical images such as

X-rays [17], MRIs [27], and CT scans [13], enabling more accurate detection of conditions like tumors and fractures as shown in table 1. These advancements have facilitated the application of CNNs in diagnosing a range of conditions, including schizophrenia [11], Attention Deficit Hyperactivity Disorder (ADHD) [11], epilepsy [12], and COVID-19 [13,14]. By leveraging deep learning, healthcare providers can identify these conditions with greater precision, reducing misdiagnoses and ensuring that patients receive the appropriate care more swiftly. The integration of these technologies not only enhances diagnostic capabilities but also streamlines workflows in medical settings, ultimately leading to improved patient management and a higher quality of care.

In recent years, deep learning has emerged as a promising approach for the automated detection of tuberculosis (TB) from chest X-ray images, addressing the limitations of traditional diagnostic methods that rely heavily on expert interpretation. Various CNN architectures have been employed to enhance the accuracy and efficiency of TB diagnosis [28–30]. Notable models, such as ResNet, DenseNet, and pre-trained networks like VGG16 and MobileNet, have been utilized, with transfer learning proving particularly effective in leveraging knowledge from larger datasets to improve performance on smaller, specific TB datasets. Several studies have demonstrated the efficacy of these approaches. For example, Hansun et al. developed a CNN-based architecture that achieved over 98 % accuracy, underscoring the capability of CNNs to learn critical features indicative of TB [31]. Sufian et al. focused on transfer learning with DenseNet121, achieving a accuracy of 98 % highlighting the method's effectiveness in addressing data limitations [32]. Additionally, Rajpurkar et al. reported that their deep learning model, ChexNet performed comparably to expert radiologists, showcasing its potential for generalization across diverse populations [33].

The success of CNNs in various applications is accompanied by a significant increase in the computation and parameter storage costs. Recent efforts toward reducing these overheads involve pruning and compressing the weights of various layers without hurting original accuracy. However, magnitude-based pruning of weights reduces a significant number of parameters from the fully connected layers and may not adequately reduce the computation costs in the convolutional layers due to irregular sparsity in the pruned networks. We present an acceleration method for CNNs, where we prune filters from CNNs that are identified as having a small effect on the output accuracy. By removing whole filters in the network together with their connecting feature maps, the computation costs are reduced significantly. In contrast to pruning weights, this approach does not result in sparse connectivity patterns. Hence, it does not need the support of sparse convolution libraries and can work with existing efficient BLAS libraries for dense matrix multiplications. We show that even simple filter pruning techniques can reduce inference costs for VGG-16 by up to 34 % and ResNet-110 by up to 38 % on CIFAR10 while regaining close to the original accuracy by retraining the networks [27]. Deep learning has made significant strides across various fields, but it also faces several challenges that can hinder its effectiveness and application. One major issue is the requirement for large amounts of labeled data for training. Collecting, annotating, and maintaining high-quality datasets can be time-consuming and costly, particularly in specialized domains like healthcare [33]. Additionally, deep networks are prone to overfitting, especially when trained on limited datasets, resulting in models that perform well on training data but poorly on unseen data, thereby reducing their generalizability [34]. The computational resources required for training deep learning models present another significant hurdle. Substantial power and memory are needed, leading to high costs and accessibility issues, particularly for smaller organizations or researchers with limited resources [35]. Furthermore, the integration and deployment of deep learning models pose complexities, as transitioning from research to practical application can involve challenges related to scalability, compatibility, and maintenance [36]. Deployment in resource-limited settings is also a critical challenge. Many deep learning applications require substantial infrastructure, which may not be available in regions with limited technological resources or unstable internet connectivity [37]. This can hinder the implementation of models that could otherwise improve healthcare and other services in underserved areas. Addressing these challenges is essential for maximizing the potential of deep learning and ensuring its responsible and effective application across various domains.

To address these issues, model pruning techniques have emerged as a promising solution. Pruning aims to reduce the size and computational requirements of CNNs while preserving their performance [25]. By systematically removing less important weights or neurons from the model, pruning can make deep learning models more efficient and accessible, particularly in settings where resources are constrained. Pruning techniques aim to reduce the size and complexity of CNNs while maintaining their performance. Methods such as magnitude-based pruning, which removes less important weights, and structured pruning, which eliminates entire channels or layers, have gained traction [36–40]. These approaches help in reducing the computational load, making CNNs more feasible for real-time applications.

Pruning has been effectively employed in various medical applications. For instance, Kaur and Mittal [41] demonstrated that pruning could enhance the efficiency of CNNs for chest X-ray classification, achieving comparable accuracy while significantly reducing inference time. Additionally, in segmentation tasks, UrRheiman et al. [42]) utilized structured pruning to streamline a CNN for lung nodule detection, illustrating the potential of pruning to maintain high performance while decreasing resource requirements.

Table 1
Deep learning architectures performance.

Pruning Technique	Architecture	Dataset	Accuracy (Top-1)	Accuracy (Top-5)	Pruning %
Filter Pruning [43]	ResNet	ImageNet	72 %	–	38
Channel Pruning [44]	VGGNet	ImageNet	–	~89 %	80
Dynamic Pruning [45]	ResNet	ImageNet	76 %	–	71.85
Dynamic Pruning [45]	MobileNet v2	ImageNet	71 %	–	37.24
Dynamic Pruning [45]	VGG	Cifar 10	94.93 %	–	80.51
Layer-wise Pruning [10]	EfficientNet	ImageNet	84.3 %	–	88
Network Pruning [46]	SqueezeNet	ImageNet	60.4 %	–	98
Channel Pruning [47]	Xception	Cifar-10	91.8	–	80
layer-wise pruning +Skip Connections [25]	LightNet	PlantVillage Dataset	98.7	–	50
Structural Pruning [48]	ResNet,MobileNet	Cifar-10	93	–	50
Class wise Pruning [49]	ALexNet,VGG,ResNet	Cifar-10,Cifar-100	92	–	87
Filter-Level Pruning [50]	VGG16,ResNet	SVHN,Cifar-10,Cifar-100	–	–	60
Cluster-Based pruning [51]	VGG16	Cifar-10	92.68	–	86.27
Multi-stage pruning [52]	CNN	ECG Data	97.7	–	60
Iterative pruning [53]	VGG	Pediatric, RSNA, Twitter, Montreal Datasets	99.01	–	–
Efficient Channel Attention [30]	L-TBNET	Montgomery and Shenzhen datasets	96	–	96
Efficient Channel Attention [29]	E-TBNet	Montgomery and Shenzhen datasets	85	–	58.1
stacking model [54]	TB-CXRNet	QU-MLG-TB	93.32	–	–

The deployment of deep learning models for real-time image classification has become increasingly feasible due to advancements in model optimization techniques, such as pruning [36]. These methods reduce model size and inference time while maintaining accuracy, making it possible to run complex models on edge devices. Research by [36] on model pruning highlights the potential for optimizing deep learning models for mobile and embedded systems, expanding their applicability in various settings.

Several lightweight models for the detection of tuberculosis have been developed to enhance diagnostic efficiency and accessibility. Notably, the works of An et al. [29] and Razid et al. [30] illustrate innovative approaches that prioritize computational efficiency while maintaining accuracy in identifying TB cases. Table 1 provides an overview of the pruning techniques utilized in image classification for a range of classification tasks. It presents the reported accuracies achieved with these techniques, alongside the percentage of pruning applied, highlighting the trade-offs between model size reduction and classification performance.

CNNs combined with pruning techniques can significantly enhance tuberculosis (TB) detection in real-world settings. CNNs improve diagnostic accuracy by extracting relevant features from medical images, while pruning reduces model size and computational requirements, enabling faster inference and making the technology accessible on lower-end devices. Using real-world datasets for training ensures that models can generalize across diverse patient demographics, improving robustness. These efficient models can be deployed in resource-limited environments, facilitating on-site screening and timely diagnoses. Integrating CNNs into healthcare workflows streamlines the diagnostic process, allowing for quicker treatment decisions and supporting healthcare professionals with valuable second opinions. Ultimately, these advancements lead to timely interventions, better patient outcomes, and improved public health initiatives for controlling tuberculosis.

3. Methodology

3.1. Proposed architecture

The study aims to enhance the efficiency of Convolutional Neural Networks (CNNs) for detecting tuberculosis (TB) using X-rays, specifically focusing on optimizing the computational costs associated with deep learning methods. To achieve this, the study introduces a novel Pruned Convolutional Neural Network (PDenseNet) architecture, as illustrated in Fig. 1. The proposed methodology encompasses several key steps: X-ray image acquisition, preprocessing, and classification using CNN architectures. This structured approach is designed to improve both the efficiency and accuracy of TB detection in medical imaging.

The proposed PDenseNet architecture is based on the Densely connected architecture (DenseNet) and its transformation involves the application of the pruning technique as illustrated in Fig. 2. The pruning process selectively removes neurons or connections that have little or no significance to the network by ranking them based on the absolute value of weights or filters. Neurons with lowest ranking are then eliminated from the network, resulting in a more compact network with fewer neurons hence parameter efficient and less complex.

To introduce sparsity in the original DenseNet architecture, the concept of skip connections was employed. This process involves skipping or dropping some connections in the dense layer as illustrated in Fig. 3 and Fig. 4. The basic idea behind the proposed PDenseNet architecture was to preserve connections from the beginning, middle, and the deepest connection in each sparse block (originally dense block). This was achieved by concatenating or adding up only those previous layers with modulus equal to zero denoted as (i modulus 2==0) where i represents the sparse layers to be concatenated, for instance $i = 2, i = 4, i = 6, i = 8, i = 10 \ldots\ldots i = n$ as shown in Fig. 4. Layers where i modulus 2 was not equal to zero were dropped or skipped.

3.2. Dataset

A publicly available chest X-ray dataset was used to train and validate the proposed model (Awsifur, R. et al., 2020). The dataset comprised 4200 X-ray images categorized into two classes: tuberculosis and non-tuberculosis (normal). The normal class included 3500 images, while the tuberculosis class contained 700 images. Subsequently, they images were resized to dimensions of 150 × 150 pixels before being employed in the training process. For model training, 80 % of the data was allocated for training, with the remaining 20 % set aside for testing as suggested. Typically, a division such as 80 % for training and 20 % for validation is widely used to ensure that the model can generalize well to unseen data [9,34]. Additionally, the training set was further divided, with 10 % reserved for validation during training. Additionally, the training set was further divided, with 10 % reserved for validation during training to monitor for overfitting. This approach helps ensure that the model maintains its ability to generalize to unseen data, thereby enhancing overall performance [55]. Furthermore, the developed model was tested using anonymized chest X-ray images collected from level 4 and level 5 hospitals in Embu, Tharaka-Nithi, and Meru counties. These X-ray images underwent initial annotation, analysis, and digitization with the assistance of a radiologist. Following this, the images were resized to dimensions of 150 × 150 pixels before being used in the testing process. Using data from hospitals in Embu, Tharaka-Nithi, and Meru counties provides several key advantages for the study. It offers a diverse range of patient demographics, enhancing the model's ability to generalize across different populations. The data reflects real-world conditions, making the findings more relevant for similar healthcare

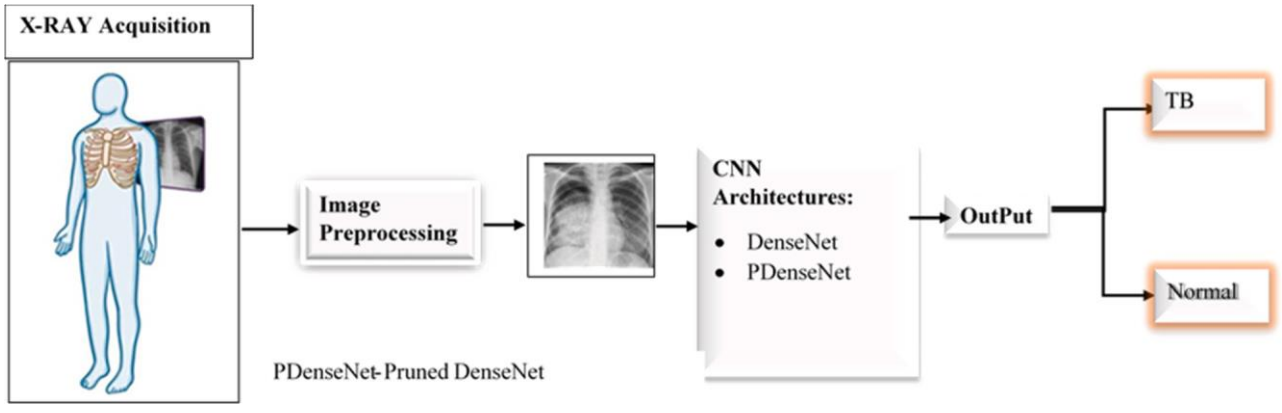

Fig. 1. Proposed PDenseNet architecture for detecting TB.

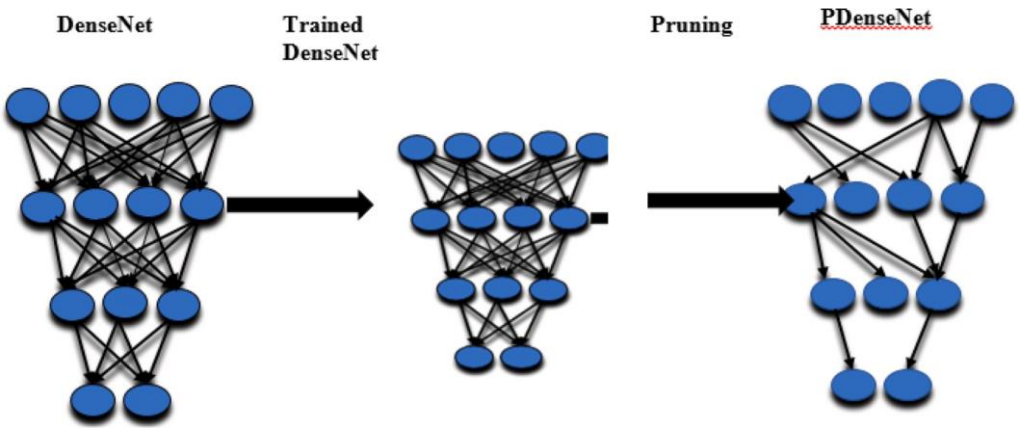

Fig. 2. Process of transformation from baseline CNN (DenseNet) to Pruned CNN (PDenseNet).

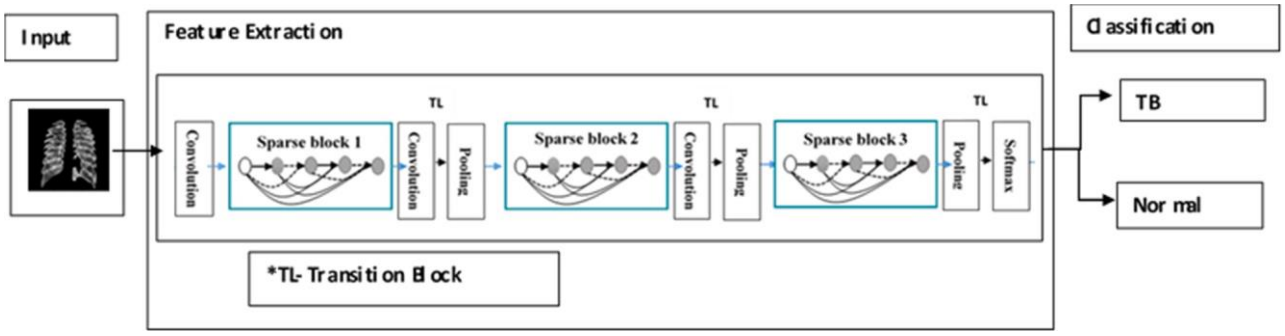

Fig. 3. PDenseNet Architecture.

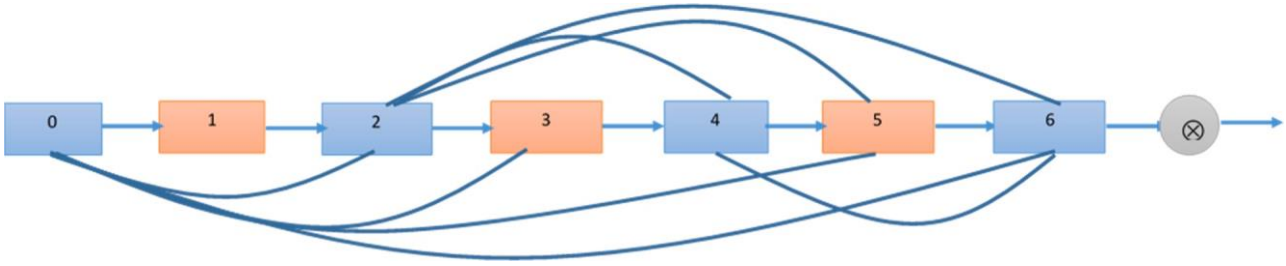

Fig. 4. PDenseNet Structure that concatenates evenly distributed connections from previous layers.

environments. Additionally, these regions may have higher tuberculosis prevalence, helping to fill gaps in larger datasets. Overall, leveraging this data enriches the model's training and aligns it with local needs, ultimately leading to more effective tuberculosis detection solutions.

3.3. Training and evaluation

The proposed PDenseNet was developed using a DenseNet architecture with 121 layers, pre-trained on the ImageNet dataset as the base model. To enhance efficiency and speed, the TensorFlow Model Optimization Toolkit (TF MOT) was employed for pruning the DenseNet network. This toolkit provides a set of techniques and tools designed to optimize model performance while maintaining accuracy.

For the experimental evaluation, the proposed pruning technique was applied to several state-of-the-art neural network architectures in image classification, including Xception, EfficientNet, MobileNet, ResNet50, and DenseNet with 201 layers. This selection of diverse architectures allows for a comprehensive assessment of the pruning technique's effectiveness across different model designs and complexities. All networks were trained using the Stochastic Gradient Descent (SGD) optimizer for 50 epochs, with an initial learning rate of 1E–2. The SGD optimizer is well-regarded for its robustness and effectiveness in training deep-learning models, making it a suitable choice for this evaluation. The decision to train for 50 epochs strikes a balance between allowing sufficient time for convergence while preventing overfitting. Due to memory constraints, a batch size of 32 was utilized. This batch size is commonly used in practice as it offers a good trade-off between computational efficiency and memory usage. It allows for manageable updates to the model weights while ensuring that the training process remains stable and efficient.

During image preprocessing, data augmentation was applied to artificially increase the diversity of the training set, helping the model to generalize better. The augmentation parameters were adjusted based on the characteristics of the dataset and the nature of the X-ray images. Basic augmentation like zooming, shearing, brightness, and vertical and horizontal shifting were done on the images to strike a balance between augmentation and maintaining the clinical relevance of the data. All images were resized to 150×150 pixels in our experiments to maintain

consistency in data preprocessing.

To assess the performance of the developed model, five performance evaluation metrics namely accuracy, precision, recall, F1-score, and confusion matrix evaluation metrics were employed.

4. Results and discussions

The six CNN architectures namely DenseNet with layers 121,169 and 201, Xception, MobileNet, and ResNet50 were initially trained and then subsequently pruned to reduce their size by removing unimportant weights.

4.1. Training and validation loss and accuracy learning curves

During training, the training and validation loss and accuracy learning curves were monitored and plotted. The architectures were able to learn the mapping between the X-ray images and their respective class with little overfitting as shown by the learning curves depicted in Figs. 5–10. The DenseNet models—DenseNet121, DenseNet169, and DenseNet201—demonstrated consistent decreases in both training and validation loss, indicating effective learning. The accuracy curves for these models were closely aligned, suggesting minimal overfitting and a strong ability to generalize to unseen data. However, despite their strong performance, the larger model sizes, particularly DenseNet201, could pose challenges in resource-constrained environments, limiting their practical deployment in areas with limited computational power. Xception showed similar learning trends, with steady decreases in training and validation losses and improvements in accuracy. The proximity of the training and validation accuracy curves indicated effective generalization. Nevertheless, Xception's complexity may lead to longer training times and increased computational resource requirements, which could hinder its use in real-time applications or settings with limited infrastructure. MobileNet exhibited a clear learning pattern, with training loss decreasing steadily and validation loss remaining stable. While the model effectively classified X-ray images without significant overfitting, its performance may be slightly lower in terms of accuracy compared to larger models. This could limit its effectiveness in high-stakes situations where precision is paramount. ResNet50 showcased comparable performance, with both training and validation losses decreasing and accuracy increasing steadily. However, like Xception, the ResNet architecture may be more complex and resource-intensive, which could restrict its applicability in settings with constrained computational resources.

4.2. A comparative analysis based on the performance metrics and model size

A comparative analysis based on the four performance metrics was conducted between the six CNN models and their pruned variants as shown in Table 2. Additionally, the size of each set of architectures was compared. The pruning process successfully reduced the size of the DenseNet121 model from 84 MB to 29 MB while maintaining a comparable performance in classifying the X-ray images. The same was evident in the other selected architectures

The DenseNet family of models consistently outperforms other architectures, with DenseNet201 standing out due to its perfect accuracy score of 100 %. This indicates that it correctly classifies every instance, showcasing its reliability in identifying true positives while minimizing false positives, with precision and recall values both at 99 %. Similarly, our pruned DenseNet201 (PDenseNet201), although slightly lower in accuracy at 99.76 %, maintains excellent precision (100 %) and solid recall (98 %). This combination makes it an effective and compact choice for applications where performance and efficiency are crucial.

PDenseNet121 offers comparable accuracy to baseline DenseNet121 but at a significantly reduced size of just 29 MB. It maintains high precision and recall, making it particularly well-suited for environments where both performance and resource constraints are critical. This model strikes an excellent balance, especially for mobile devices or real-time systems.

PDenseNet169 shares the same accuracy as DenseNet169 but is smaller at 52.6 MB. However, it shows a decrease in recall compared to

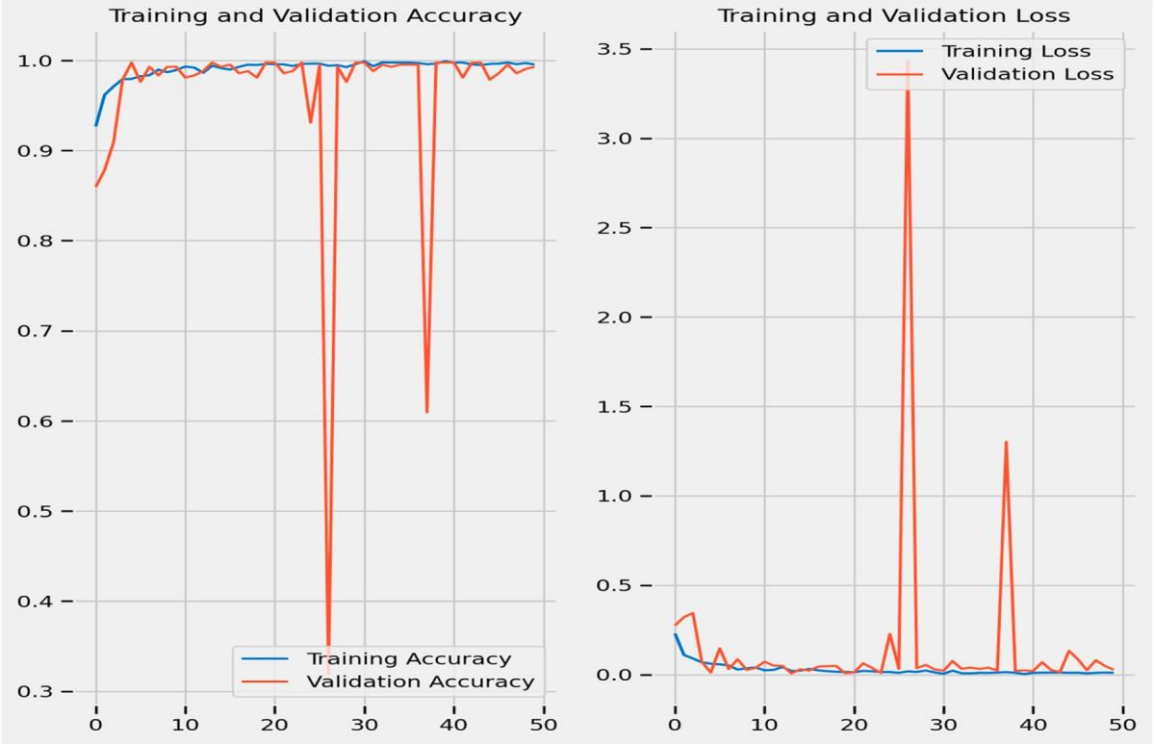

Fig. 5. DenseNet121 Training and Validation Loss and Accuracy Curves.

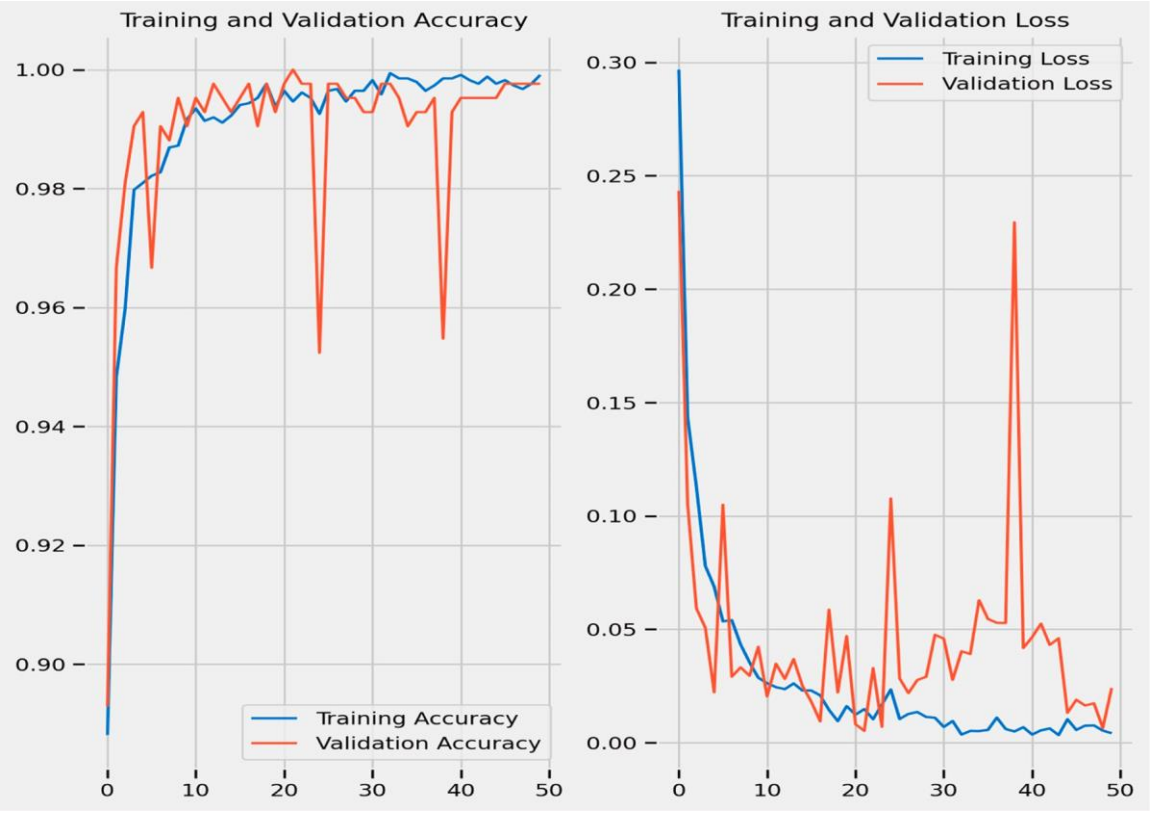

Fig. 6. DenseNet169 Training and Validation Loss and Accuracy Curves.

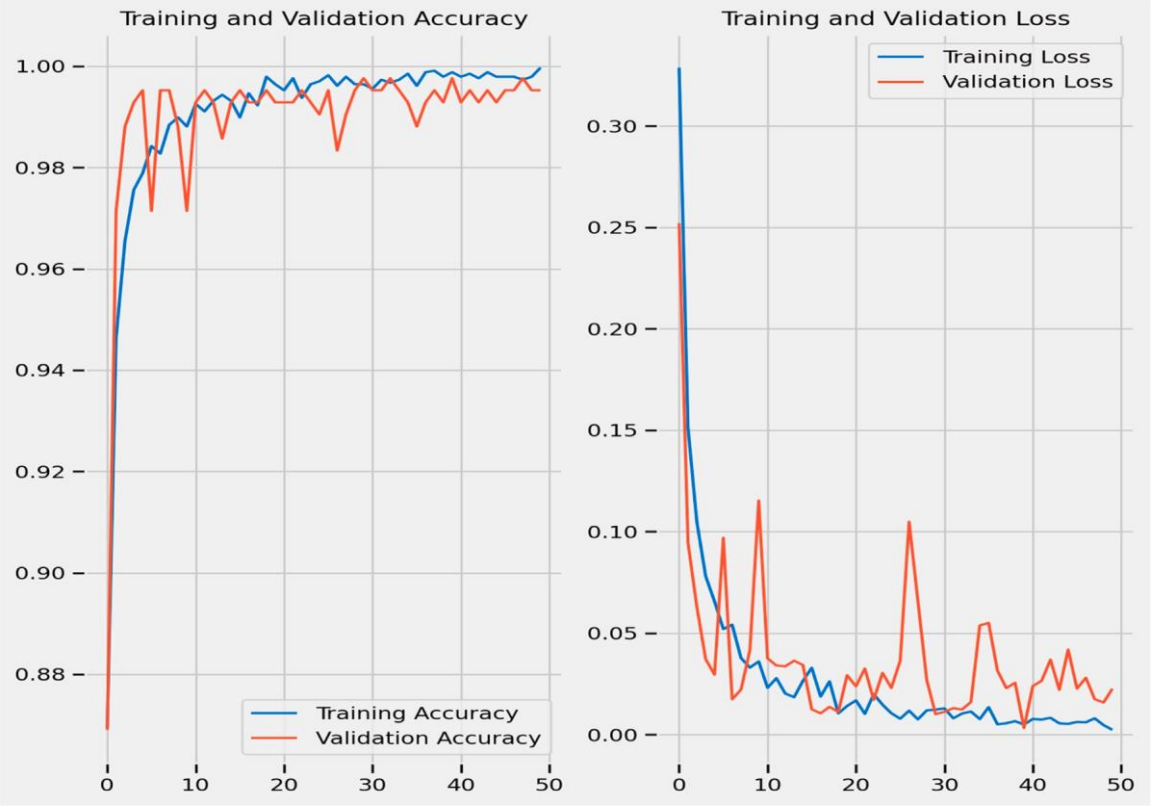

Fig. 7. DenseNet201 Training and Validation Loss and Accuracy Curves.

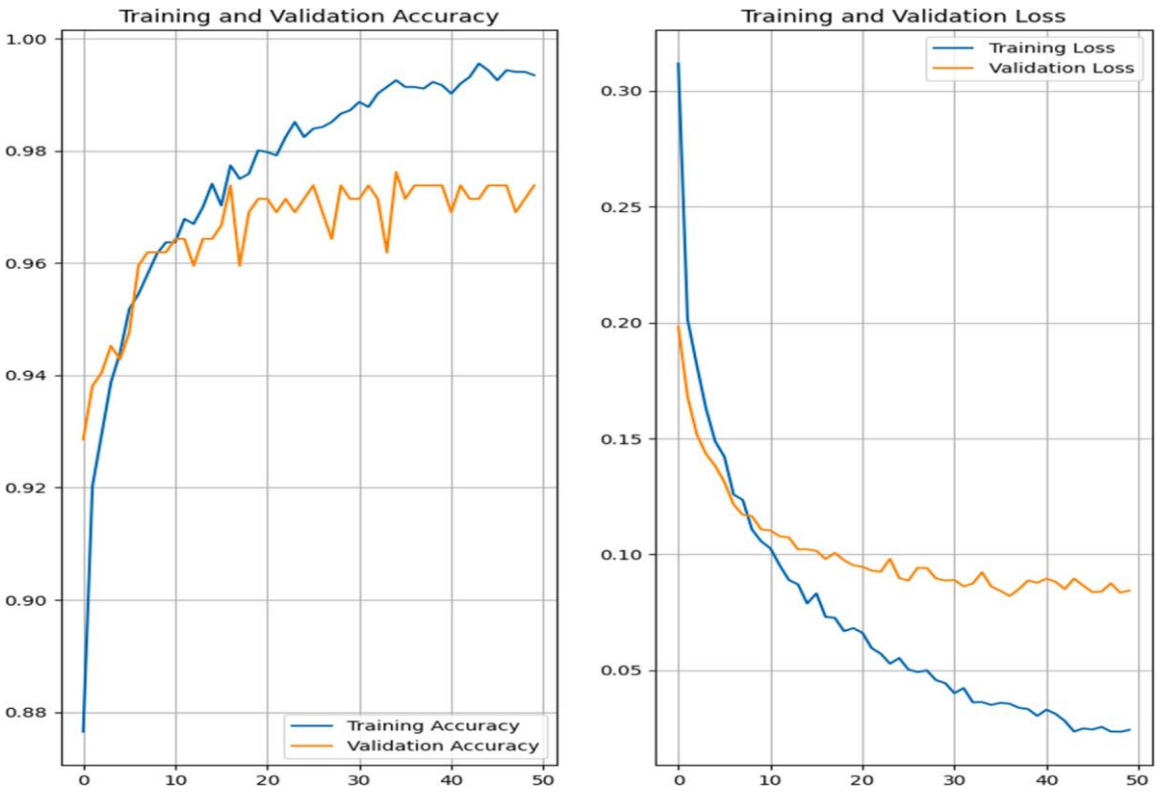

Fig. 8. Xception Training and Validation Loss and Accuracy Learning Curves.

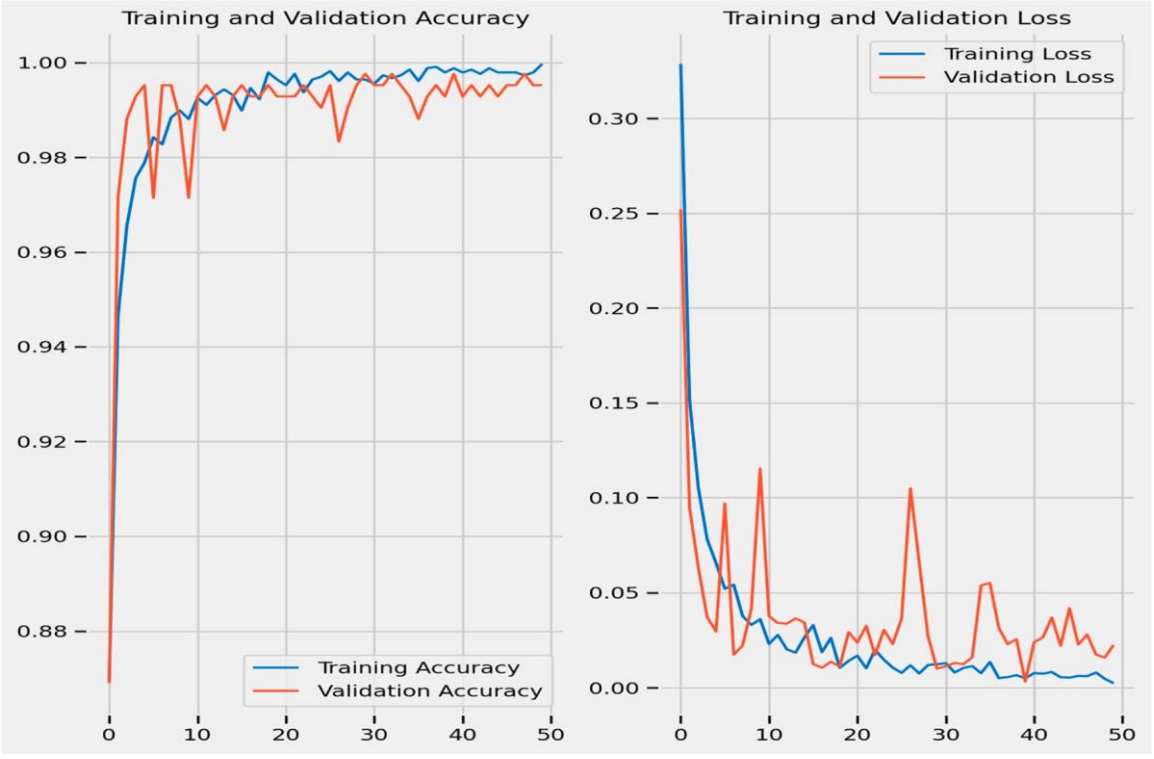

Fig. 9. MobileNet Training and Validation Loss and Accuracy Learning Curves.

both DenseNet169 and PDenseNet121, suggesting it may miss some actual positive cases. Despite this, its precision remains strong, making it a good option for scenarios that require a balance between size and performance.

In contrast, the MobileNet series exhibits a different performance profile. MobileNet has the lowest accuracy at 93.33 %, but it is light-weight at 30.1 MB. Despite its lower metrics, its precision and recall are respectable, indicating it can perform adequately in less demanding

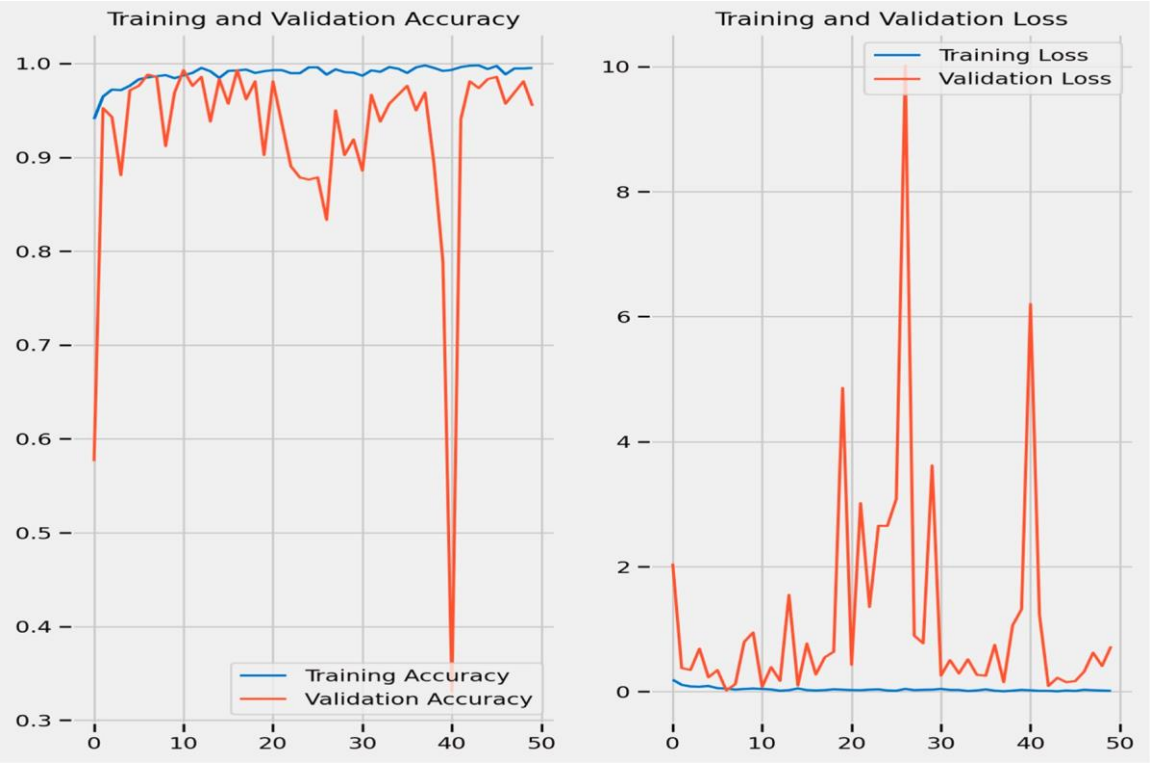

Fig. 10. ResNet50 Training and Validation Loss and Accuracy Learning Curves.

Table 2
Perfomance comparative table for the Original pretrained CNN models and their pruned variants.

Model	Accuracy %	Precision %	Recall %	F1-score %	Model Size (MB)
DenseNet121	99.52	99	99	100	84.9
PDenseNet121	99	99	99	99	29
DenseNet169	99.52	98	100	99	102.4
PDenseNet169	99.52	99	96	97	52.6
DenseNet201	100	99	99	99	146
PDenseNet201	99.76	100	98	99	75
MobileNet	93.33	95	74	80	30.1
PMobileNet	94.9	97	86	90	20
ResNet50	97.38	99	93	95	276.4
PResNet50	97.62	98	96	97	92.5
Xception	97.14	98.34	91.43	91	79.87
PXception	96.19	97.81	88.57	87.1	40

contexts. PMobileNet improves slightly on accuracy at 94.9 % while emphasizing resource efficiency, making it well-suited for mobile and embedded applications.

The ResNet models present a mixed-performance landscape. ResNet50 boasts strong precision (99 %) and reasonable recall (93 %), but it is significantly larger at 276.4 MB. In contrast, PResNet50 achieves better recall (96 %) while being more compact at 92.5 MB, effectively balancing performance and size.

Examining Xception and its variant PXception reveals moderate performance. Xception delivers decent results but falls short compared to the DenseNet family, particularly in the recall. PXception offers slightly lower accuracy at 96.19 % but maintains good precision and recall, making it practical for scenarios where model size is a consideration.

Overall, a clear trade-off exists between model size and performance. Smaller models like PMobileNet may not match the high accuracy and F1 scores of larger models like DenseNets, but they are better suited for applications with limited computational resources. The choice of model should ultimately depend on the specific context of deployment. For high-stakes applications, such as medical imaging, the superior performance of DenseNet variants may be essential. Conversely, for real-time applications on mobile devices, PMobileNet may be the preferable option, despite its lower metrics.

4.3. Reduction rates

The reduction rates of the pruned models indicate significant efficiency gains compared to their baseline architectures as illustrated in Table 3. These reductions are particularly important for deploying models in resource-constrained environments such as mobile devices or embedded systems. PDenseNet121 exhibits a remarkable reduction rate of approximately 65.8 % compared to DenseNet121. This substantial decrease in size allows for easier deployment without compromising much on performance. Given that DenseNet121 already has high accuracy and F1-score, the pruned version remains a strong candidate for applications needing both efficiency and effectiveness. PDenseNet169 achieves a reduction rate of about 48.7 % compared to DenseNet169. While this reduction is slightly lower than that of PDenseNet121, it still indicates a meaningful improvement in efficiency. The pruned model retains competitive accuracy, making it suitable for various classification tasks. The balance between model size and performance makes PDenseNet169 an attractive option for scenarios where both factors are critical. PDenseNet201 shows a reduction rate of approximately 48.6 %

Table 3
reduction rates.

CNN Architecture	Reduction Rates
PDenseNet121	~65.8 %
PDenseNet169	~48.7 %
PDenseNet201	~48.6 %
PResNet50	~66.6 %
PMobileNet	~33.5 %
PXception	~49.9 %

compared to DenseNet201. This efficient pruning allows the model to remain highly effective while significantly reducing its size. Given that DenseNet201 already achieves perfect accuracy, the pruned version's ability to retain strong performance metrics while being much smaller makes it ideal for applications requiring high reliability without excessive resource demands. PResNet50 stands out with an impressive reduction rate of about 66.6 % compared to ResNet50. This reduction is noteworthy, as it makes the model highly suitable for deployment in resource-constrained environments. The pruned model retains strong precision and recall, indicating that it can perform effectively in real-world applications while benefiting from a smaller footprint. PMobileNet shows a reduction rate of approximately 33.5 % compared to the original MobileNet. While this reduction is less pronounced than those seen in the DenseNet and ResNet variants, it still reflects meaningful optimization. Given that MobileNet was originally designed for efficiency, PMobileNet enhances this aspect further, making it well-suited for real-time applications on mobile and embedded devices. PXception achieves a reduction rate of about 49.9 % compared to the original Xception model. This reduction strikes a balance between maintaining performance and achieving a smaller model size. Although PXception shows slightly lower accuracy and recall than Xception, it still provides good precision, making it a viable choice for applications where model size is a priority.

The reduction rates across these pruned models highlight the effectiveness of pruning techniques in achieving compact architectures while retaining competitive performance. Each model demonstrates varying degrees of size optimization, with PDenseNet121 and PResNet50 leading the way in reduction rates.

The choice of model should consider the specific requirements of the application, including the importance of accuracy, precision, recall, and size. While DenseNet and ResNet variants exhibit significant reductions without compromising performance, MobileNet and its pruned variant excel in efficiency, making them ideal for real-time scenarios.

In conclusion, these insights underscore the trade-offs and benefits of pruning within CNN architectures, providing valuable options for developers and researchers in diverse fields, from mobile applications to high-stakes environments requiring reliable performance.

4.4. Comparison of confusion matrices

4.4.1. DenseNet121 vs. PDenseNet121

The original DenseNet121 demonstrates high accuracy with very few misclassifications, particularly among true positives as depicted in Fig. 11. The pruned variant, PDenseNet121, closely mirrors this performance, indicating that pruning has not significantly affected its ability to correctly classify instances. The minimal performance gap suggests that substantial reductions in model size can be achieved without compromising classification accuracy. Additionally, PDenseNet121 demonstrated superior performance due to its densely connected layers, which promote feature reuse and enable more efficient learning of the X-ray images. The model's ability to capture both low- and high-level features likely contributed to its enhanced accuracy compared to the other networks.

4.4.2. DenseNet169 vs. PDenseNet169

Similar to the DenseNet121 comparison, the DenseNet169 model showcases strong performance with accurate classification of most true positives as illustrated in Fig. 12. The pruned PDenseNet169 maintains this high level of accuracy, again showing a small performance gap relative to the original model. This is particularly impressive given the reduction in model size, which enhances its usability in resource-constrained environments.

4.4.3. DenseNet201 vs. PDenseNet201

In the case of DenseNet201, the original model achieves perfect accuracy of 100 %, effectively classifying all instances without errors as indicated in Fig. 13. The pruned PDenseNet201, while slightly lower in accuracy at 99.76 %, still demonstrates outstanding performance. The confusion matrix indicates that PDenseNet201 maintains high precision (100 %) and recall (98 %), suggesting that it is nearly as effective as its baseline counterpart. This close alignment in performance, despite the significant reduction in model size (approximately 48.6 %), further highlights the effectiveness of pruning techniques.

The results from the confusion matrices indicate that all pruned DenseNet models—PDenseNet121, PDenseNet169, and PDenseNet201—retain the essential characteristics of their baseline counterparts. This makes them suitable for applications requiring both accuracy and efficiency. The close alignment in performance suggests that pruning techniques can effectively reduce model size while preserving the ability to classify accurately.

Generally, the analysis of the confusion matrices reinforces the conclusion that pruning can be a valuable strategy in model optimization. By achieving significant size reductions with only minimal performance trade-offs, the pruned DenseNet variants present compelling

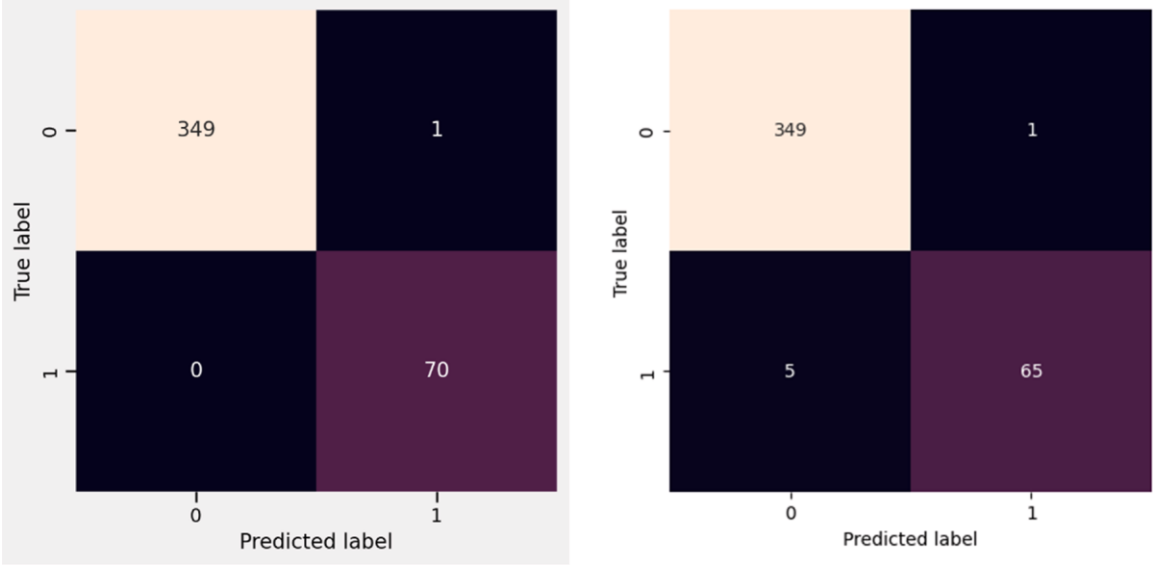

Fig. 11. Confusion Matrices for both the original DenseNet121 model and its pruned variant respectively.

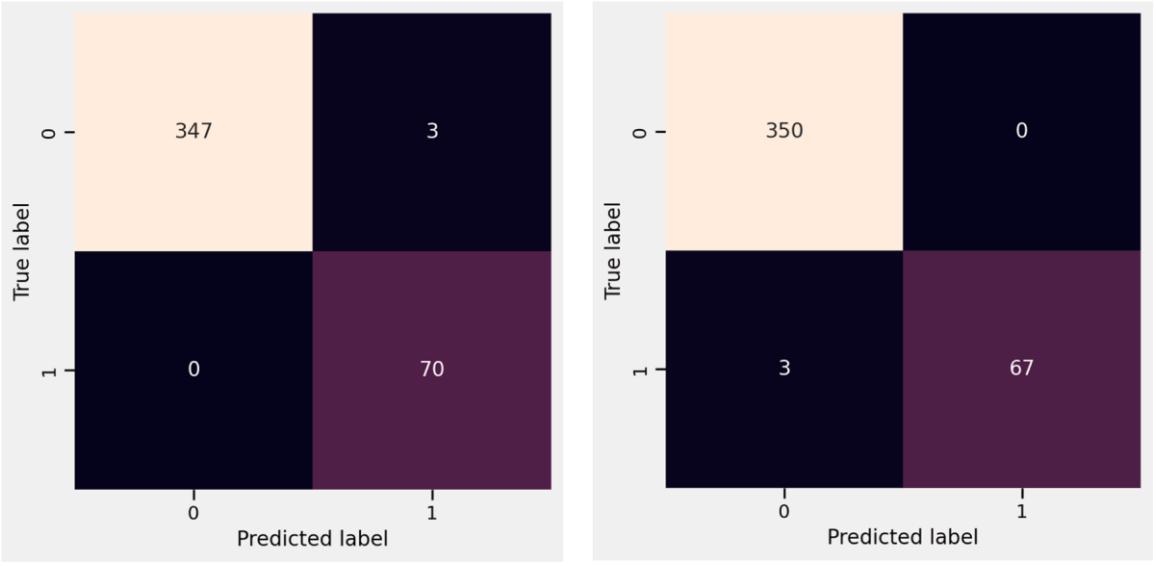

Fig. 12. Confusion Matrices for both the original DenseNet169 model and its pruned variant respectively.

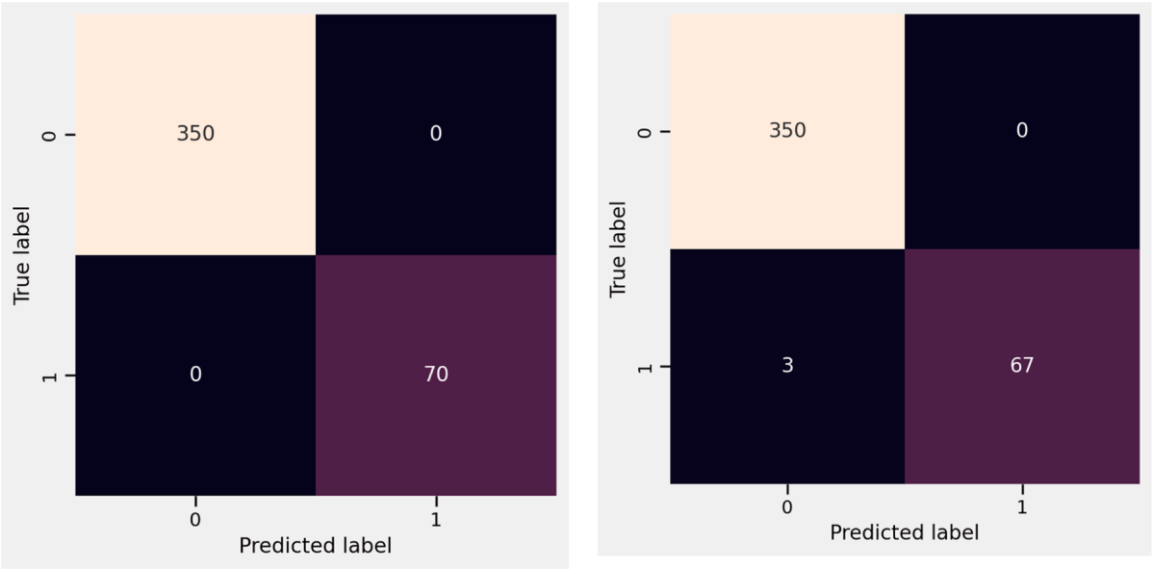

Fig. 13. Confusion Matrices for both the original DenseNet169 model and its pruned variant respectively.

options for real-world applications, especially in scenarios where computational resources are limited. The findings emphasize the potential for leveraging pruned models in various fields, from mobile computing to large-scale data analysis. The successful performance of PDenseNet201 alongside its baseline model illustrates that high accuracy and efficiency can coexist, making these pruned models a strong choice for modern applications.

4.5. Prediction

The pruned DenseNet121 model successfully predicted X-ray images from our collected dataset, which included both normal and TB-infected images. The model demonstrated impressive performance in accurately classifying both classes, as illustrated in Figs. 14 and 15.

In Fig. 14, we see the correctly predicted images for the normal class (0), showcasing the model's ability to identify healthy X-ray images with a high degree of accuracy. This reflects its effectiveness in distinguishing normal cases from those affected by tuberculosis. Similarly, Fig. 15 presents the correctly predicted images for the tuberculosis class (1), further emphasizing the model's competence in identifying TB-infected images.

The successful predictions made by the pruned DenseNet121 model highlight its potential for practical applications in medical imaging. Its ability to accurately classify both normal and TB-infected images indicates that it can be a valuable tool for healthcare professionals, assisting in the diagnosis of tuberculosis through X-ray analysis.

Largely, these results underscore the effectiveness of pruning techniques in maintaining high classification performance while significantly reducing model size. This optimization makes the pruned DenseNet121 model particularly suitable for deployment in resource-constrained settings, where efficient and accurate diagnostics are essential. The visual confirmations of accurate predictions reinforce the model's reliability and its promising application in clinical environments.

Some images in the collected dataset had poor quality and low resolution, presenting a challenge for the model in achieving accurate

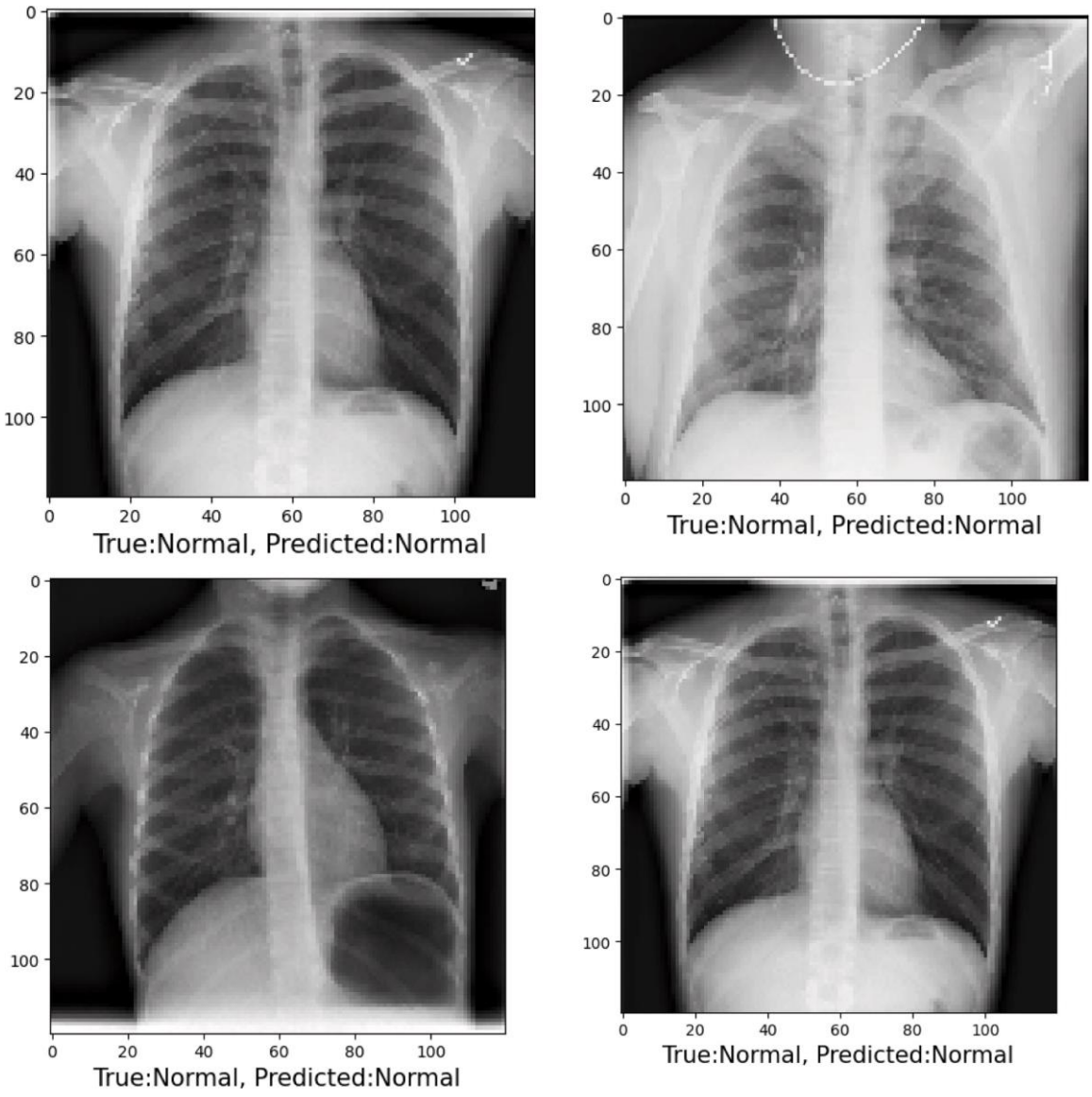

Fig. 14. Correctly predicted images for Normal Class (0).

classification. Despite these challenges, the model demonstrated robust performance, successfully classifying the majority of images. However, a few misclassifications occurred, as depicted in Fig. 16. This only occurred in those images with very poor quality.

4.6. Prototype

The model prototype was developed to demonstrate the practical application of Pruned DenseNet (PDenseNet) in a real-world healthcare platform. As shown in Figs. 17 to 19, the prototype is designed to efficiently classify chest X-ray images as either indicative of tuberculosis (TB) infection or as normal (healthy). These images were collected from Kenyan hospitals, where they were carefully annotated by medical experts, ensuring the highest quality of training data for the model. The X-ray images used for this prototype were sourced from healthcare facilities in Kenya, providing a diverse set of real-world data that reflects the conditions encountered in clinical settings. The images were thoroughly annotated by trained radiologists and medical experts to label the presence or absence of TB. This high-quality, expert-curated dataset forms the foundation for training and evaluating the PDenseNet model, ensuring its applicability to the local healthcare context.

In Fig. 18, the prototype demonstrates the model's ability to accurately classify chest X-rays showing TB cases. The model effectively identifies and differentiates X-ray images with signs of TB infection, underscoring its potential to assist healthcare professionals in diagnosing this critical condition. The correct classification of these images is a testament to the model's robustness in recognizing subtle patterns indicative of TB in chest radiographs. In Fig. 19, the model also correctly classifies normal X-ray images (healthy cases), highlighting its ability to distinguish between infected and non-infected images. This capability is vital in a clinical setting, as it enables healthcare professionals to quickly identify patients without TB, thus streamlining the diagnostic process and reducing the risk of false positives.

The integration of PDenseNet into this prototype highlights its potential for real-world medical applications. By utilizing annotated X-ray images from a region where access to timely diagnostic resources can be limited, the model offers a practical solution to the challenges faced by healthcare professionals. Specifically, it can aid in the rapid detection of TB, enabling earlier intervention and better patient outcomes, especially in resource-constrained environments. The model's ability to accurately classify both TB-positive and normal X-rays demonstrates its utility as a diagnostic tool that can be deployed in clinical settings, particularly in

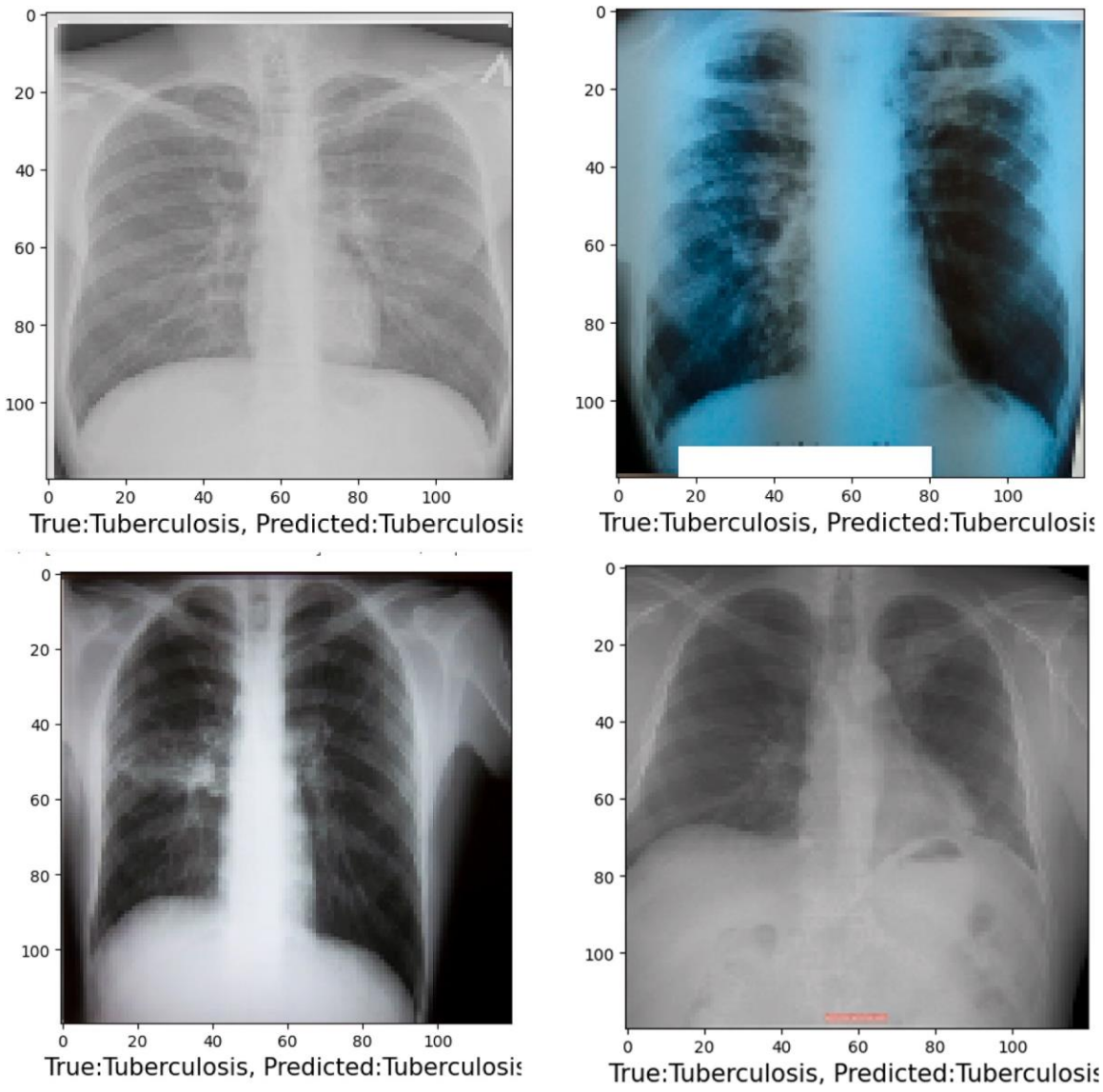

Fig. 15. Correctly predicted images for Tuberculosis Class (1).

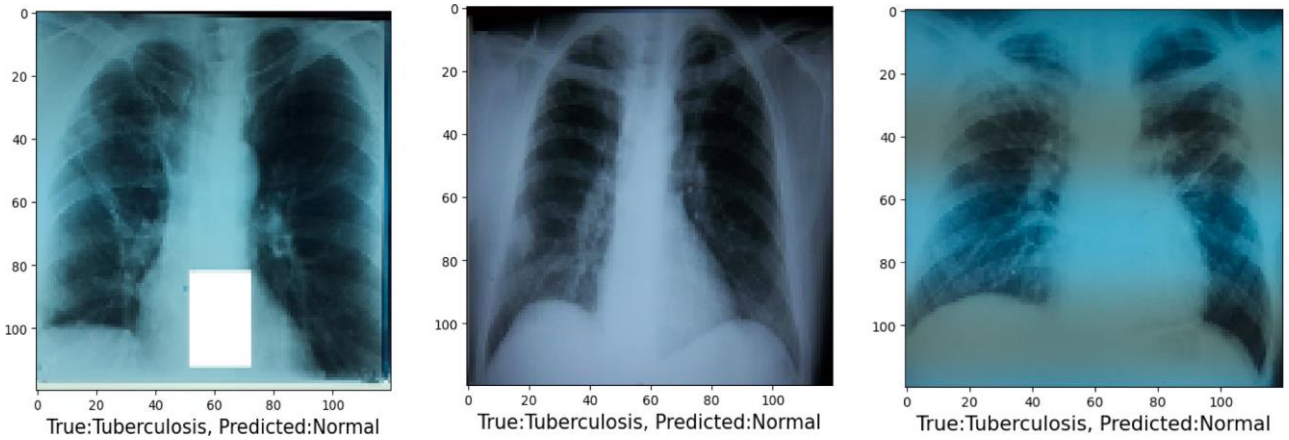

Fig. 16. Pruned DenseNet121 Misclassification.

areas with limited access to expert radiologists. The speed and accuracy of PDenseNet in identifying TB cases have the potential to accelerate diagnostic workflows, reducing waiting times for results and improving patient care.

This prototype exemplifies how techniques, like PDenseNet, can enhance the diagnostic process by automating the analysis of medical

Fig. 17. Loading an Image into Our Prototype.

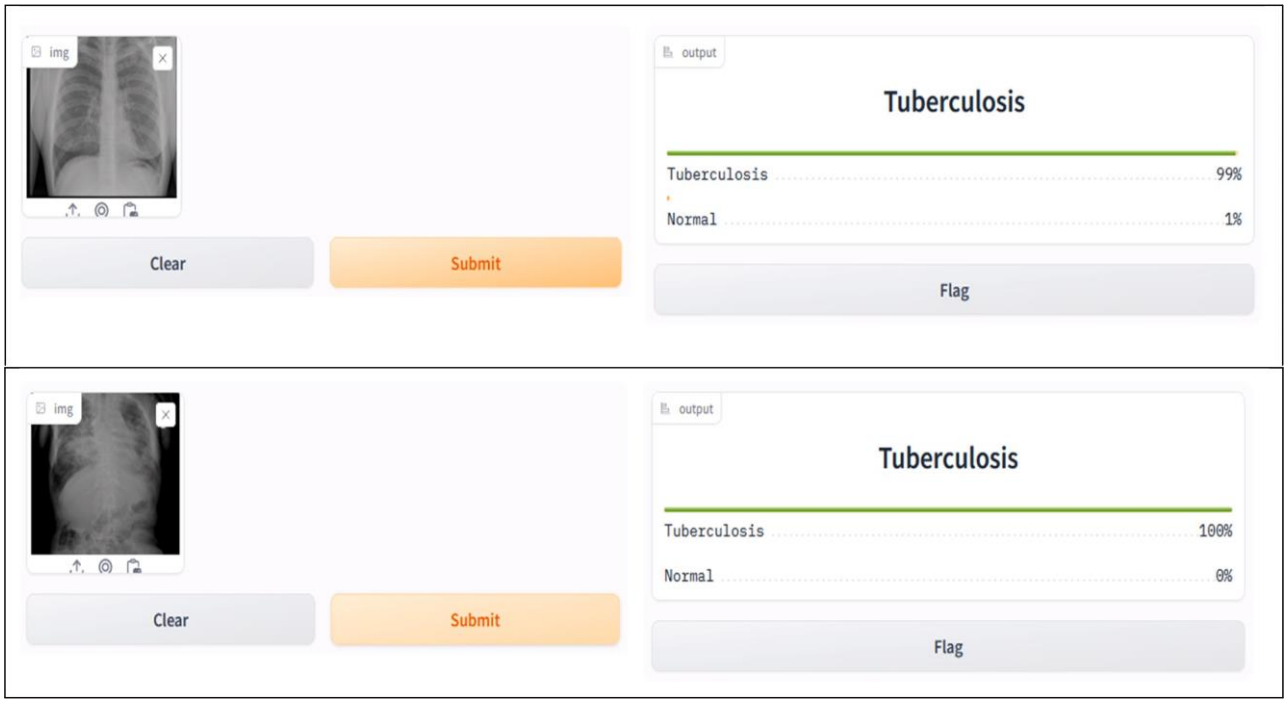

Fig. 18. Correctly classified as Tuberculosis using our porotype.

imaging. In clinical environments, where timely diagnoses are crucial, such tools can significantly improve decision-making efficiency. By providing objective, data-driven insights, the model supports healthcare workers in making informed decisions, particularly in regions where radiology expertise may be scarce. Furthermore, the use of PDenseNet can reduce the workload on healthcare professionals, allowing them to focus on more complex cases or direct patient care. The model not only demonstrates its ability to classify X-ray images accurately but also serves as a powerful tool for augmenting human decision-making in the detection of infectious diseases such as tuberculosis. Overall, the successful integration of the PDenseNet model into this prototype is a significant step forward in applying artificial intelligence (AI) to medical diagnostics. The system demonstrates high accuracy in distinguishing between tuberculosis-infected and normal chest X-rays, showcasing its potential for real-world deployment in healthcare settings, particularly in regions with limited diagnostic resources. By leveraging annotated X-ray images, the model not only proves its diagnostic accuracy but also highlights the role of advanced deep learning algorithms in improving patient outcomes and supporting healthcare professionals. The development of this prototype paves the way for further innovations in medical imaging and disease detection, with the potential to make a profound impact on global health.

5. Conclusion and future research

Tuberculosis (TB) is the leading infectious disease killer in Kenya, and while conventional tests are common, the World Health Organization (WHO) recommends incorporating X-ray screening for improved detection. However, a shortage of radiologists presents challenges, necessitating effective alternatives for screening X-ray images.

In response, we developed and tested several deep learning architectures for TB detection using chest X-ray images. The pruned DenseNet121 model proved particularly effective, reducing model size by approximately 65.8 % while maintaining strong performance metrics, including accuracy, precision, recall, and F1 score. Other models also demonstrated significant size reductions, with PDenseNet169 (~48.7 %), PDenseNet201 (~48.6 %), PResNet50 (~66.6 %), PMobileNet (~33.5 %), and PXception (~49.9 %).

The results highlight the pruned DenseNet (PDenseNet) model's effectiveness in classifying normal and TB-infected images, supported by confusion matrices and visual evidence. This model's reduced size and minimal performance trade-offs make it a viable solution for resource-

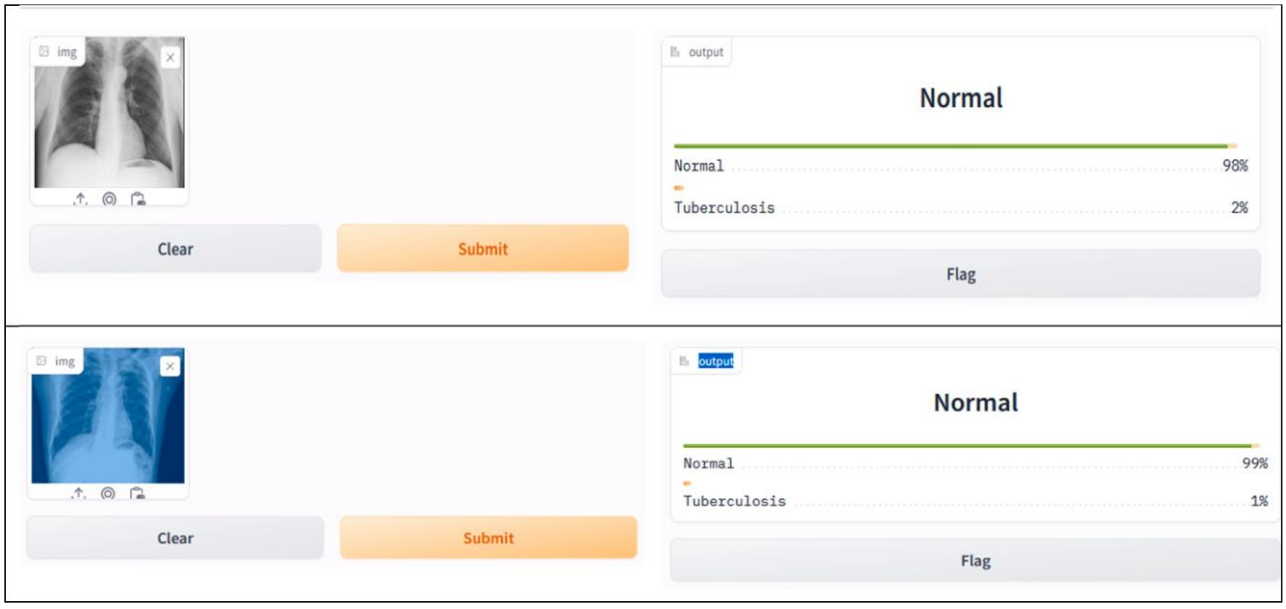

Fig. 19. Correctly classified as Normal using our porotype.

constrained environments, such as rural hospitals in Kenya, enhancing diagnostic processes.

Additionally, the successful integration of PDenseNet into a working prototype underscores its practical applicability for healthcare professionals in diagnosing TB. While this study focused on TB in the chest region, further research is needed to explore detection in other parts of the body, which could improve overall TB screening effectiveness, especially in areas with limited access to radiologists.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work the author(s) used ChatGPT in order for grammatical and sentence restructuring. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

CRediT authorship contribution statement

Edna Chebet Too: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **David Gitonga Mwathi:** Writing – review & editing, Validation, Supervision, Project administration, Funding acquisition, Conceptualization. **Lucy Kawira Gitonga:** Writing – review & editing, Investigation, Conceptualization. **Pauline Mwaka:** Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Saif Kinyori:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Methodology, Investigation, Formal analysis, Data curation, Conceptualization.

Declaration of competing interest

The authors, Edna Too and team, declare that they have received minigrant funding from the Research and Innovation Support for Africa (RISA) in collaboration with the University of Nairobi for this research. Other than this, the authors declare no other conflicts of interest related to this manuscript.

Acknowledgment

We extend our gratitude to Research and Innovation Systems for Africa (RISA) for funding the TBNet Project, which made this research possible. We also wish to express our special appreciation to the management of P.C.E.A Chogoria Hospital, Chuka Level 5 Hospital and Marimanti Level 4 Hospital for generously providing access to the necessary data for this project. Special appreciation goes to Julius Muchoki Clinician and Radiologist from the medical department of Chuka University. Their collaboration and support were invaluable in enabling us to conduct this study and achieve our objectives in advancing tuberculosis detection and diagnosis through innovative technologies.

References

[1] World Health Organization, "Global tuberculosis report 2022," 2022.
[2] Ministry of Health, Kenya, "Kenya latent tuberculosis infection policy 2020," Ministry of Health, Kenya, 1, Mar. 2020. [Online]. Available: http://www.health.go.ke.
[3] N.P. Mnyambwa, et al., Gaps related to screening and diagnosis of tuberculosis in care cascade in selected health facilities in East Africa countries: a retrospective study, J. Clin. Tuberc. Mycobact. Dis. 25 (2021) 100278, https://doi.org/10.1016/j.jctube.2021.100278. Dec.
[4] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," 2015, *arXiv*. doi: 10.48550/ARXIV.1512.00567.
[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv*. doi: 10.48550/ARXIV.1409.1556.
[6] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Las Vegas, NV, USA, 2016, pp. 770–778, https://doi.org/10.1109/CVPR.2016.90. Jun.
[7] G. Huang, Z. Liu, L. van der Maaten, and K.Q. Weinberger, "Densely connected convolutional networks," 2016, *arXiv*. doi: 10.48550/ARXIV.1608.06993.
[8] A.G. Howard et al., "MobileNets: efficient convolutional neural networks for mobile vision applications," 2017, *arXiv*. doi: 10.48550/ARXIV.1704.04861.
[9] F. Chollet, "Xception: deep learning with depthwise separable convolutions," 2016, *arXiv*. doi: 10.48550/ARXIV.1610.02357.
[10] M. Tan and Q.V. Le, "EfficientNet: rethinking Model scaling for convolutional neural networks," 2019, doi: 10.48550/ARXIV.1905.11946.
[11] A. Shoeibi, et al., Automatic diagnosis of schizophrenia and attention deficit hyperactivity disorder in rs-fMRI modality using convolutional autoencoder model and interval type-2 fuzzy regression, Cogn. Neurodyn. 17 (6) (2023) 1501–1523, https://doi.org/10.1007/s11571-022-09897-w. Dec.
[12] A. Shoeibi, et al., Epileptic seizures detection using deep learning techniques: a review, Int. J. Environ. Res. Public. Health 18 (11) (2021) 5780, https://doi.org/10.3390/ijerph18115780. May.

[13] A. Shoeibi, et al., Automated detection and forecasting of COVID-19 using deep learning techniques: a review, Neurocomputing 577 (2024) 127317, https://doi.org/10.1016/j.neucom.2024.127317. Apr.
[14] N. Ghassemi, et al., Automatic diagnosis of COVID-19 from CT images using CycleGAN and transfer learning, Appl. Soft Comput. 144 (2023) 110511, https://doi.org/10.1016/j.asoc.2023.110511. Sep.
[15] C.-H. Tsai, et al., Automatic deep learning-based pleural effusion classification in lung ultrasound images for respiratory pathology diagnosis, Phys. Med. 83 (2021) 38–45, https://doi.org/10.1016/j.ejmp.2021.02.023. Mar.
[16] P. Xue, et al., Deep learning in image-based breast and cervical cancer detection: a systematic review and meta-analysis, Npj Digit. Med. 5 (1) (2022), https://doi.org/10.1038/s41746-022-00559-z. Art. no. 1Feb.
[17] A. Shimazaki, et al., Deep learning-based algorithm for lung cancer detection on chest radiographs using the segmentation method, Sci. Rep. 12 (1) (2022), https://doi.org/10.1038/s41598-021-04667-w. Art. no. 1Jan.
[18] A. Gurunathan, B. Krishnan, A Hybrid CNN-GLCM classifier for detection and grade classification of brain tumor, Brain Imaging Behav. 16 (3) (2022), https://doi.org/10.1007/s11682-021-00598-2. Art. no. 3Jun.
[19] J. Ramasamy, R. Doshi, K.K. Hiran, Segmentation of brain tumor using deep learning methods: a review, in: Proceedings of the International Conference on Data Science, Machine Learning and Artificial Intelligence, ACM, Windhoek Namibia, 2021, pp. 209–215, https://doi.org/10.1145/3484824.3484876. Aug.
[20] S. Kornblith, J. Shlens, and Q.V. Le, "Do better imagenet models transfer better?," 2018, *arXiv*. doi: 10.48550/ARXIV.1805.08974.
[21] M. Gupta, S. Aravindan, A. Kalisz, V. Chandrasekhar, and L. Jie, "Learning to prune deep neural networks via reinforcement learning," 2020, *arXiv*. doi: 10.48550/ARXIV.2007.04756.
[22] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, Commun. ACM 60 (6) (2017) 84–90, https://doi.org/10.1145/3065386. May.
[23] Z. Yu, Y. Dong, J. Cheng, M. Sun, F. Su, Research on face recognition classification based on improved googleNet, Secur. Commun. Netw. 2022 (2022) 1–6, https://doi.org/10.1155/2022/7192306. Jan.
[24] R. Yang, Y. Yu, Artificial convolutional neural network in object detection and semantic segmentation for medical imaging analysis, Front. Oncol. 11 (2021) 638182, https://doi.org/10.3389/fonc.2021.638182. Mar.
[25] E.C. Too, LightNet: pruned sparsed convolution neural network for image classification, Int. J. Comput. Sci. Eng. 26 (3) (2023), https://doi.org/10.1504/IJCSE.2023.131508. Art. no. 3.
[26] E.C. Too, Y. Li, P. Kwao, S. Njuki, M.E. Mosomi, J. Kibet, Deep pruned nets for efficient image-based plants disease classification, J. Intell. Fuzzy Syst. 37 (3) (2019), https://doi.org/10.3233/JIFS-190184. Art. no. 3Oct.
[27] S.E. Sorour, A.A.A. El-Mageed, K.M. Albarrak, A.K. Alnaim, A.A. Wafa, E. El-Shafeiy, Classification of Alzheimer's disease using MRI data based on deep learning techniques, J. King Saud Univ. - Comput. Inf. Sci. 36 (2) (2024), https://doi.org/10.1016/j.jksuci.2024.101940. Art. no. 2Feb.
[28] E. Kotei, R. Thirunavukarasu, Computational techniques for the automated detection of mycobacterium tuberculosis from digitized sputum smear microscopic images: a systematic review, Prog. Biophys. Mol. Biol. 171 (2022) 4–16, https://doi.org/10.1016/j.pbiomolbio.2022.03.004. Jul.
[29] L. An, et al., E-TBNet: light deep neural network for automatic detection of tuberculosis with X-ray DR imaging, Sensors 22 (3) (2022) 821, https://doi.org/10.3390/s22030821. Jan.
[30] A. Raziq, N. Ahmed, S. Khan, M. Bizanjo, N. Uddin, R. Baloch, Development of light-weight convolutional neural network model to diagnose tuberculosis, VFAST Trans. Softw. Eng. 10 (3) (2022) 43–50, https://doi.org/10.21015/vtse.v10i3.1148. Sep.
[31] S. Hansun, A. Argha, S.-T. Liaw, B.G. Celler, G.B. Marks, Machine and deep learning for tuberculosis detection on chest X-Rays: systematic literature review, J. Med. Internet Res. 25 (2023) e43154, https://doi.org/10.2196/43154. Jul.
[32] M.A. Sufian, et al., AI-driven thoracic X-ray diagnostics: transformative transfer learning for clinical validation in pulmonary radiography, J. Pers. Med. 14 (8) (2024) 856, https://doi.org/10.3390/jpm14080856. Aug.
[33] G. Litjens, et al., A survey on deep learning in medical image analysis, Med. Image Anal. 42 (2017) 60–88, https://doi.org/10.1016/j.media.2017.07.005. Dec.
[34] Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 2016 [Online]. Available, http://www.deeplearningbook.org.
[35] T. Hoefler, D. Alistarh, T. Ben-Nun, and N. Dryden, "Sparsity in deep learning: pruning and growth for efficient inference and training in neural networks",2021, *arXiv*. 10.48550/ARXIV.2102.00554.
[36] T. Liang, J. Glossner, L. Wang, S. Shi, and X. Zhang, "Pruning and quantization for deep neural network acceleration: a survey," Jun. 15, 2021, *arXiv*: arXiv:2101.09671. Accessed: Oct. 08, 2024. [Online]. Available: http://arxiv.org/abs/2101.09671.
[37] S.M. Adil, et al., Deep learning to predict traumatic brain injury outcomes in the low-resource setting, World Neurosurg 164 (2022) e8–e16, https://doi.org/10.1016/j.wneu.2022.02.097. Aug.
[38] S. Han, H. Mao, and W.J. Dally, "Deep compression: compressing deep neural networks with pruning, trained quantization and huffman coding," 2015, *arXiv*. doi: 10.48550/ARXIV.1510.00149.
[39] Z. Li, H. Li, L. Meng, Model compression for deep neural networks: a survey, Computers 12 (3) (2023), https://doi.org/10.3390/computers12030060. Art. no. 3Mar.
[40] M. Zhu and S. Gupta, "To prune, or not to prune: exploring the efficacy of pruning for model compression," 2017, *arXiv*. doi: 10.48550/ARXIV.1710.01878.
[41] N. Kaur, A. Mittal, CheXPrune: sparse chest X-ray report generation model using multi-attention and one-shot global pruning, J. Ambient Intell. Humaniz. Comput. 14 (6) (2023) 7485–7497, https://doi.org/10.1007/s12652-022-04454-z. Jun.
[42] Z. UrRehman, et al., Effective lung nodule detection using deep CNN with dual attention mechanisms, Sci. Rep. 14 (1) (2024) 3934, https://doi.org/10.1038/s41598-024-51833-x. Feb.
[43] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H.P. Graf, "Pruning filters for efficient ConvNets," 2016, *arXiv*. doi: 10.48550/ARXIV.1608.08710.
[44] Y. He, X. Zhang, and J. Sun, "Channel pruning for accelerating very deep neural networks," 2017, *arXiv*. doi: 10.48550/ARXIV.1707.06168.
[45] J.-H. Park, Y. Kim, J. Kim, J.-Y. Choi, and S. Lee, "Dynamic structure pruning for compressing CNNs," 2023, *arXiv*. doi: 10.48550/ARXIV.2303.09736.
[46] F.N. Iandola, S. Han, M.W. Moskewicz, K. Ashraf, W.J. Dally, and K. Keutzer, "SqueezeNet: alexNet-level accuracy with 50x fewer parameters and <0.5MB model size," Nov. 04, 2016, *arXiv*: arXiv:1602.07360. Accessed: Oct. 11, 2024. [Online]. Available: http://arxiv.org/abs/1602.07360.
[47] Y. He, "Pruning very deep neural network channels for efficient inference," 2022, *arXiv*. doi: 10.48550/ARXIV.2211.08339.
[48] A. Ganjdanesh, S. Gao, and H. Huang, "Jointly training and pruning CNNs via learnable agent guidance and alignment," 2024, *arXiv*. doi: 10.48550/ARXIV.2403.19490.
[49] M. Narkhede, S. Mahajan, P. Bartakke, M. Sutaone, Towards compressed and efficient CNN architectures via pruning, Discov. Comput. 27 (1) (2024) 29, https://doi.org/10.1007/s10791-024-09463-4. Sep.
[50] X. Zhao, A. Farjudian, A. Bellotti, Pruning convolutional neural networks for inductive conformal prediction, Neurocomputing 611 (2025) 128704, https://doi.org/10.1016/j.neucom.2024.128704. Jan.
[51] N.A. Khan, A.M.S. Rafat, Pruning convolution neural networks using filter clustering based on normalized cross-correlation similarity, J. Inf. Telecommun. (2024) 1–19, https://doi.org/10.1080/24751839.2024.2415008. Oct.
[52] L. Xiaolin, R.C. Panicker, B. Cardiff, D. John, Multistage pruning of CNN based ECG classifiers for edge devices, in: 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), IEEE, Mexico, 2021, pp. 1965–1968, https://doi.org/10.1109/EMBC46164.2021.9630588. Nov.
[53] S. Rajaraman, J. Siegelman, P.O. Alderson, L.S. Folio, L.R. Folio, S.K. Antani, Iteratively pruned deep learning ensembles for COVID-19 detection in chest X-Rays, IEEE Access 8 (2020) 115041–115050, https://doi.org/10.1109/ACCESS.2020.3003810.
[54] T. Rahman, et al., TB-CXRNet: tuberculosis and drug-resistant tuberculosis detection technique using chest X-ray images, Cogn. Comput. 16 (3) (2024) 1393–1412, https://doi.org/10.1007/s12559-024-10259-3. May.
[55] Y. Tian, Y. Zhang, A comprehensive survey on regularization strategies in machine learning, Inf. Fusion 80 (2022) 146–166, https://doi.org/10.1016/j.inffus.2021.11.005. Apr.