

A Comparative Exploration of Neural Networks for Brain Tumor, Melanoma, and X-ray Image Analysis

Abstract

In this research, we investigate the application of sequential neural network architectures – Xception, EfficientNetB7, VGG16, and ResNet – for Brain Tumor, Melanoma, and X-ray Image Analysis through MRI. Leveraging unique strengths of each architecture, our study systematically compares their performance on a benchmark dataset using rigorous evaluation metrics. To address the challenge of AI system explainability in healthcare decision-making, we incorporate XAI techniques. These approaches enhance transparency and interpretability, crucial for gaining trust in machine-driven medical decisions. The outcomes of our study have the potential to guide the selection of optimal neural network architectures, contributing to the enhancement of diagnostic capabilities in computer-aided systems within the healthcare domain.

Introduction

Brain tumors, categorized as malignant or benign, arise from abnormal cell growth within the brain. Primary tumors originate in the brain, while secondary tumors result from metastasis. Symptoms, such as headaches, seizures, vision issues, vomiting, and cognitive changes, vary based on the affected brain region. Our study investigates the effectiveness of three state-of-the-art pre-trained deep learning models for precise and efficient brain tumor detection. The comparative analysis aims to discern the performance nuances and inherent strengths of each model, providing valuable insights into optimizing CNN architectures for accurate diagnosis and prognosis of brain tumors.

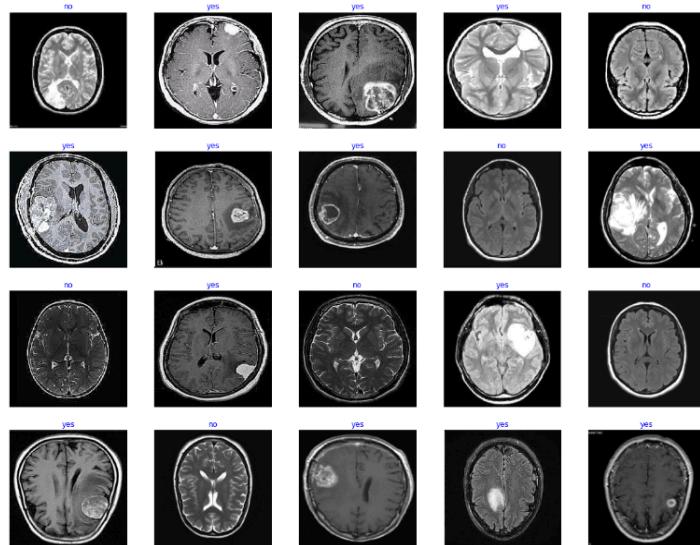
Skin cancer is the most prevalent type of cancer. Melanoma, specifically, is responsible for 75% of skin cancer deaths, despite being the least common skin cancer. When cells responsible for pigment production called melanocytes grow uncontrollably, people are affected by melanoma. According to CNN, there is an estimation of 151,000 cases each year by 2030 which was just above 96,000 in 2019 if it continues with current trend. Melanoma accounts for the highest mortality rate caused due to skin cancer. The chance of survival is very low at advanced stage. Dermoscopy has greatly helped experts in early detection and has increased diagnosis accuracy. So, dermoscopic images have been used. Melanoma is often mistaken as other benign cancer type seborrheic keratosis. So, computer aided methods can greatly assist dermatologists to accurately diagnose melanoma.

The chest X-ray is a widely used diagnostic tool for screening and identifying various lung conditions, such as Infiltration, Effusion, Atelectasis, Nodule, Mass, Pneumothorax, Consolidation, Pleural Thickening, Cardiomegaly, Emphysema, Edema, Fibrosis, Pneumonia, and Hernia . In the United States, annually, more than 35 million chest X-ray images are taken, and radiologists are tasked with interpreting over 100 X-ray studies per day. India, similar to the USA, has a significant volume of chest X-ray data, reflecting the country's population and growing health awareness. Specifically trained tools can automatically categorize chest X-ray images into normal and abnormal, allowing radiologists to focus on the latter. Furthermore, these tools can classify images into various disease categories and assist in disease localization and visualization.

Dataset used:

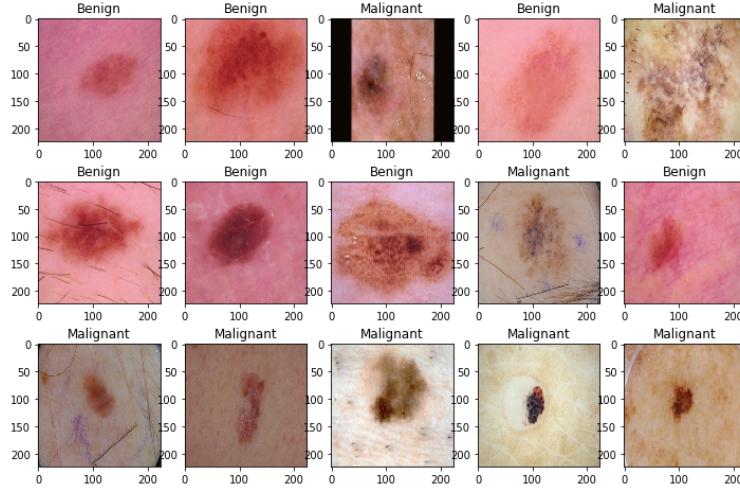
1. Brain Tumor Dataset:

The dataset employed in this research consists of a total of 3060 Brain MRI Images meticulously categorized into distinct folders based on tumor presence. This comprehensive dataset forms the cornerstone for the training and testing phases of our Deep Learning models. With a primary objective of enhancing diagnostic efficiency, especially in regions with restricted access to skilled professionals, the dataset is organized into three folders. The "yes" folder encompasses 1500 images representing tumorous Brain MRI scans, the "no" folder includes an equivalent number of 1500 images depicting non-tumorous Brain MRI scans, and the "pred" folder is designated for predictive analysis. This dataset structure facilitates a robust foundation for our study, allowing for a systematic exploration of Deep Learning techniques in brain tumor detection and classification.



2. Melanoma Dataset:

The dataset comprises medical imaging data in DICOM format, accessible through commonly-used libraries such as pydicom, containing both image and metadata. Additionally, images are available in JPEG and TFRecord formats within the jpeg and tfrecords directories, with TFRecord images uniformly resized to 1024x1024 pixels. Metadata is also provided in CSV files, specifically in train.csv and test.csv. The objective is to predict a binary target for each image, representing the probability (floating point) between 0.0 and 1.0 that the lesion in the image is malignant. In the training data (train.csv), a value of 0 denotes a benign lesion, while a value of 1 indicates malignancy. Relevant columns in the dataset include image_name (unique identifier linked to DICOM filename), patient_id (unique patient identifier), sex (patient's gender, with blanks for unknown), age_approx (approximate patient age at imaging), anatom_site_general_challenge (location of imaged site), diagnosis (detailed diagnosis information, available in the training set only), benign_malignant (indicator of malignancy), and the binarized target variable denoted as 'target'.



3. Chest X-ray Dataset:

The dataset is organized into 3 folders (train, test, val) and contains subfolders for each image category (Pneumonia/Normal). There are 5,863 X-Ray images (JPEG) and 2 categories (Pneumonia/Normal). Chest X-ray images (anterior-posterior) were selected from retrospective cohorts of pediatric patients of one to five years old from Guangzhou Women and Children's Medical Center, Guangzhou. All chest X-ray imaging was performed as part of patients' routine clinical care. For the analysis of chest X-ray images, all chest radiographs were initially screened for quality control by removing all low quality or unreadable scans. The diagnoses for the images were then graded by two expert physicians before being cleared for training the AI system. In order to account for any grading errors, the evaluation set was also checked by a third expert.



Methodology:

Data Preprocessing:

1. Brain Tumor Dataset

Brain tumor data preprocessing is a critical phase encompassing multiple intricate steps. Initially, brain images are retrieved from the dataset, laying the foundation for subsequent analyses. The pre-processing of images involves a meticulous series of operations to enhance their quality and suitability for model training. Following this, feature extraction becomes pivotal, where relevant information is distilled from the images. This involves a two-fold process: feature selection to identify pertinent characteristics and feature reduction to streamline the dataset. Augmenting data is then employed to diversify the dataset and bolster model generalization, leveraging techniques such as rotation, shearing, and zooming. The final step involves further pre-processing of the augmented data, ensuring compatibility and optimizing

the dataset for subsequent model training. This comprehensive approach to brain tumor data preprocessing is a testament to the nuanced consideration of each step, showcasing a deliberate fusion of image enhancement, feature extraction, and data augmentation for optimal model performance. The Kaggle dataset, based on MRI scans, is utilized with different training, testing, and validation set ratios. The pre-processing involves resizing images to 128x128 pixels, shuffling data, and applying image enhancement techniques like salt noise addition or monochrome deformation.

2. Melanoma Dataset

The preprocessing pipeline for the melanoma dataset involves systematic steps to prepare the data for machine learning. Two distinct folders, representing benign and malignant classes, are accessed for both training and testing datasets. Images are read and converted into RGB format, creating arrays for each class. Labels are assigned, with benign labeled as 0 and malignant as 1. The datasets are merged, creating comprehensive training and testing sets. To facilitate model training, one-hot encoding is applied to the categorical labels, transforming them into binary vectors. Additionally, normalization is performed on the testing set. Finally, the data is shuffled to introduce variability, ensuring an unbiased representation in both training and testing datasets. This meticulous preprocessing strategy ensures the data is structured and formatted optimally for subsequent machine learning model training and evaluation on melanoma detection.

3. Chest X-ray Dataset

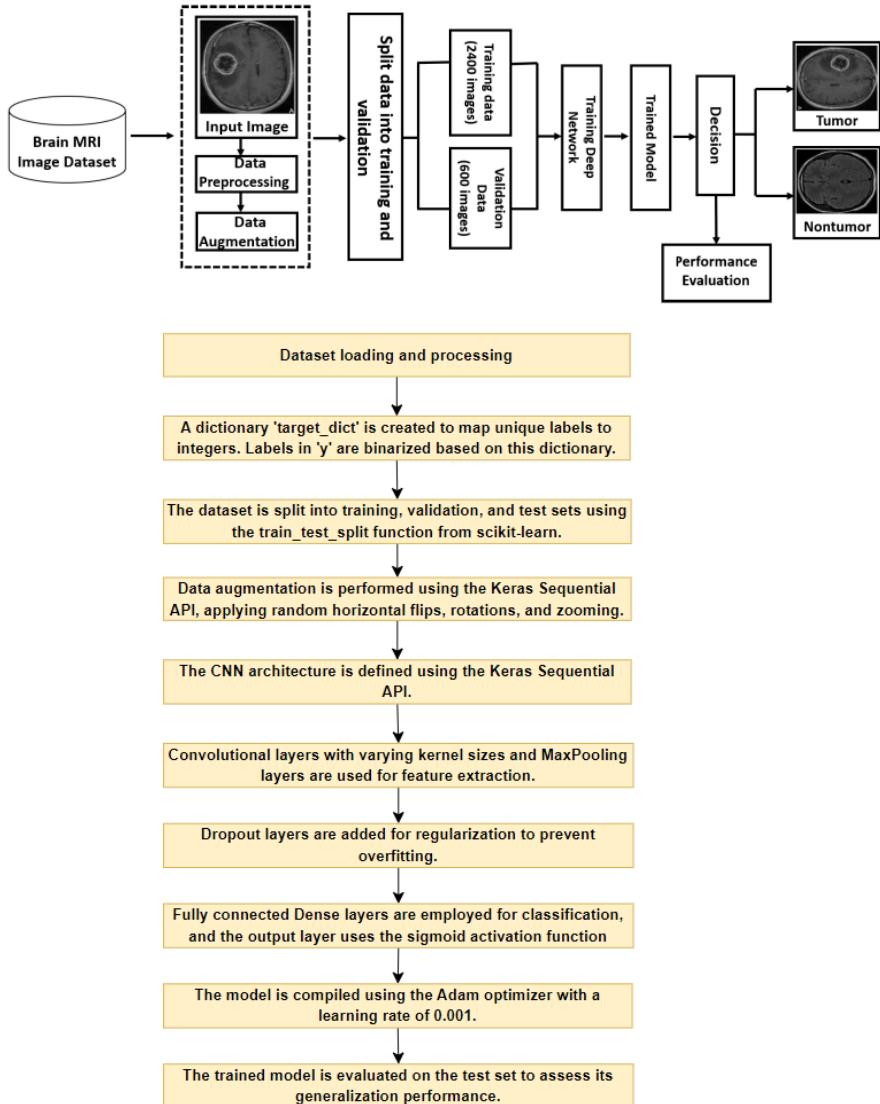
The initial exploration of the chest X-ray dataset revealed an imbalance between pneumonia and normal cases. To address this, data augmentation techniques were employed to increase the number of training examples. The grayscale images were normalized to a range of [0, 1], facilitating faster convergence of the convolutional neural network (CNN). Subsequently, the dataset was resized for deep learning. Augmentation strategies such as random rotation (up to 30 degrees), zooming (up to 20%), horizontal shifting (up to 10% width), and horizontal flipping were applied to diversify the training data and mitigate overfitting. These transformations effectively doubled or tripled the dataset size, enhancing the robustness of the model. Finally, the data augmentation process was completed by fitting the training dataset to the model.

Models Used:

1. SCNN

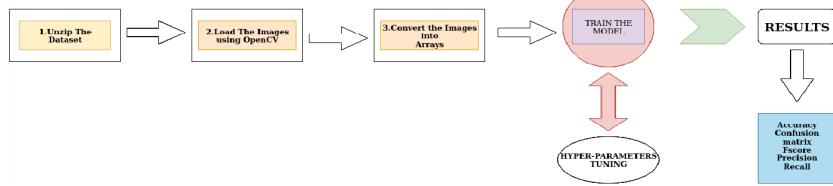
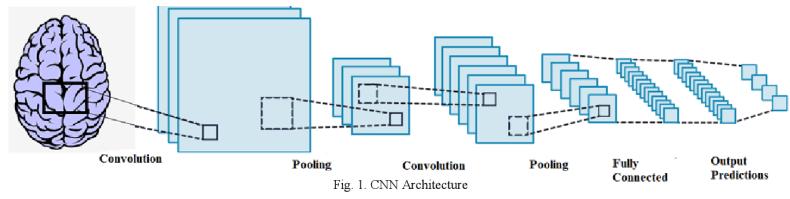
A Simple Convolutional Neural Network (Simple CNN) is a streamlined architecture tailored for image classification tasks, characterized by its shallow structure and computational efficiency. In the convolutional layer of a CNN, key operations include the application of filters, creation of feature maps, and padding. Filters analyze nearby pixel influences by moving across the image, calculating values through convolution operations. Filters, representing specific features like noses in medical images, aid in reducing neural network weights and accommodating feature location changes. Feature maps are generated post-filter application, undergo activation functions, and can be further abstracted through additional layers, including pooling layers. Padding prevents downsampling effects at image edges by maintaining feature map sizes. Pooling layers, such as max pooling and average pooling, downscale feature maps to reduce pixel density. Global pooling layers, which consider entire inputs, help in dimensionality reduction and replace fully connected layers. The convolutional layer involves filters, feature maps, and padding, contributing to the spatial hierarchy. Pooling and convolution operations, despite differing in cost, collectively build effective spatial hierarchies in CNNs. The network's

architecture also includes fully connected layers, akin to MLP output layers, for flattening results before classification.



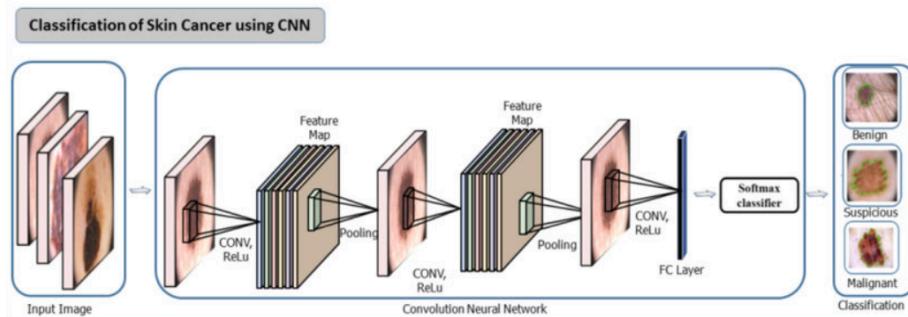
1. Brain Tumor Detection

The sequential CNN employed for brain tumor detection relies solely on MRI scans, excluding accompanying masks. The initial layer comprises a Convolutional layer with filters for feature abstraction, producing feature maps representing filter responses to input images. A subsequent pooling layer reduces feature map size, retaining essential features and mitigating parameter count to prevent overfitting. The compressed outputs from both layers feed into fully connected layers for feature classification. The CNN outputs a probability distribution for possible class labels, with a threshold determining tumor presence. Training optimizes Convolutional and fully connected layer weights via backpropagation and stochastic gradient descent to minimize differences between predicted and ground truth classes. The convolutional layer crucially extracts features, while the pooling layer, with max pooling predominantly employed, prevents overfitting. Incorporating three fully connected layers, fine-tuning involves layer-wise adjustments, though marginal accuracy improvement is noted compared to the pre-trained CNN network.



2. Melanoma Detection

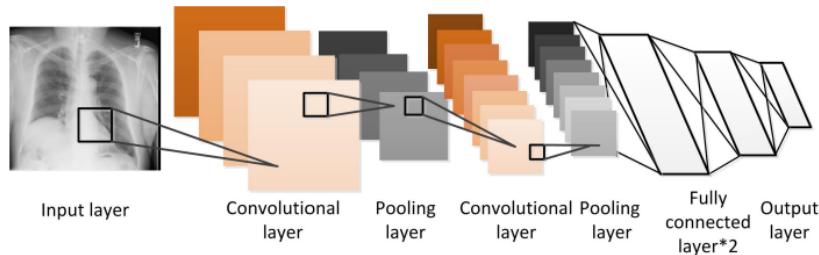
The fundamental element constituting this architecture is the Conv2D layer, characterized by a plethora of parameters, including filters, kernel_size, strides, and padding. The 'filters' parameter governs the quantity of trainable convolutional filters, with deeper layers mastering intricate features crucial for accurate detection. 'Kernel_size' dictates the dimensions of the convolution window, often set to (3, 3) or (5, 5). The 'strides' parameter defines the step of convolution along the x and y axes, profoundly influencing downsampling, while 'padding' options of 'valid' or 'same' dynamically control spatial dimensions. The CNN blueprint intricately weaves convolutional layers in tandem with max-pooling layers, strategically placed to downsample feature maps. The pivotal Flatten layer orchestrates the transformation of feature maps into a singular 1D vector, facilitating seamless integration with fully connected layers. Incorporation of dropout layers acts as a regularization mechanism, adeptly thwarting overfitting concerns. The activation functions, particularly Rectified Linear Unit (ReLU), inject indispensable non-linearity into the architecture. The conclusive layer, employing the 'sigmoid' activation function, culminates in a binary classification model meticulously fine-tuned for melanoma detection. The model's training regimen, governed by binary crossentropy loss and optimized through the RMSprop algorithm, incorporates dynamic adjustments to the learning rate via the sophisticated ReduceLROnPlateau technique. Implementation intricacies involve the systematic stacking of Conv2D, MaxPooling2D, and Dense layers, culminating in a comprehensive CNN architecture meticulously tailored for melanoma detection.



3. Chest X-ray Dataset:

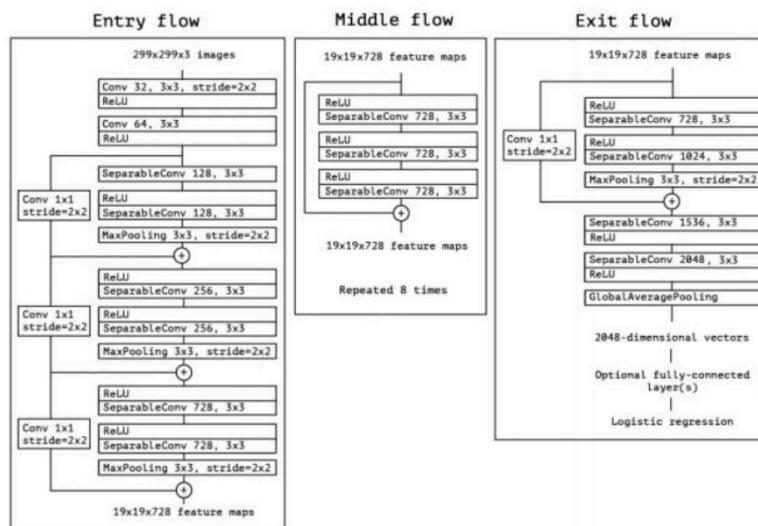
The model's architecture starts with an input layer, followed by a groundbreaking convolutional layer with 96 filters of size 11x11 and a Rectified Linear Unit (ReLU) activation function, complemented by max-pooling for spatial downsampling. Subsequent layers introduce batch normalization for convergence and regularization, and further convolutional layers with varying filter sizes systematically increase the depth of learned features. Global average pooling reduces spatial dimensions before feeding data into densely connected layers. The fully connected layers employ rectified linear units, dropout for regularization, and batch normalization for stability. The output layer, with a sigmoid activation

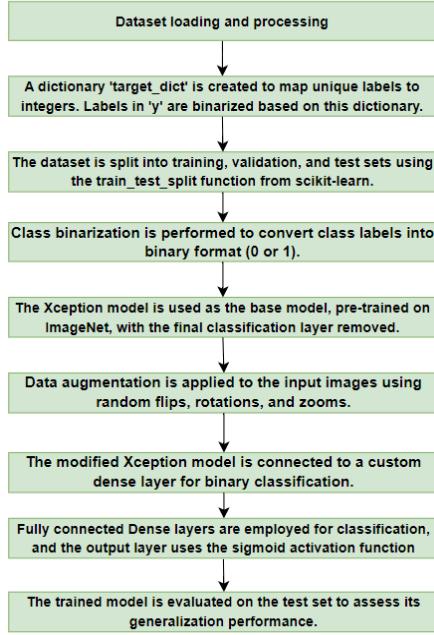
function, facilitates multi-class classification across 14 disease categories. The model, compiled with the Adam optimizer and categorical cross-entropy loss, stands poised to unravel intricate patterns in chest X-ray images, contributing decisively to the precise identification of diseases in a new era of medical diagnostics. <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8575127>



2. Xception Model:

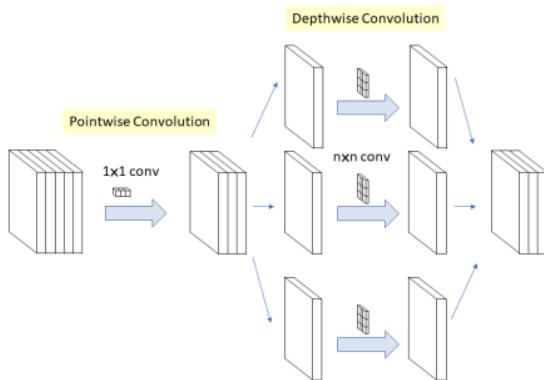
Xception is an extension of the Inception model, wherein it substitutes the conventional inception modules with depth-wise separable convolutions. Inception-v3, a widely employed image recognition model, enhances the precision of ImageNet datasets and allows for fine-tuning using low-level functions. Xception, having undergone training on the ImageNet database encompassing over a million images, offers the advantage of capturing comprehensive features from diverse images. The Xception model, a pioneering architecture in deep learning, employs a unique and efficient approach to image classification, including brain tumor detection. The depthwise convolution independently applies a single filter to each input channel, while the pointwise convolution combines results across channels, substantially reducing computational complexity and parameters compared to conventional convolutions. Xception's architecture is structured into three main stages: Entry Flow, Middle Flow, and Exit Flow. The Entry Flow initializes the process with convolutional and residual blocks, where the latter facilitates information flow through skip connections, addressing vanishing gradient issues. The Middle Flow intensifies feature extraction through repeated depthwise separable convolutions, and the Exit Flow encapsulates high-level features and spatial reduction through residual blocks and average pooling, ultimately converging to a 1x1 feature map. Efficiency is further enhanced by techniques like truncated normal weight initialization and batch normalization after each convolutional layer, ensuring stable and effective training.





1. Brain Tumor Detection

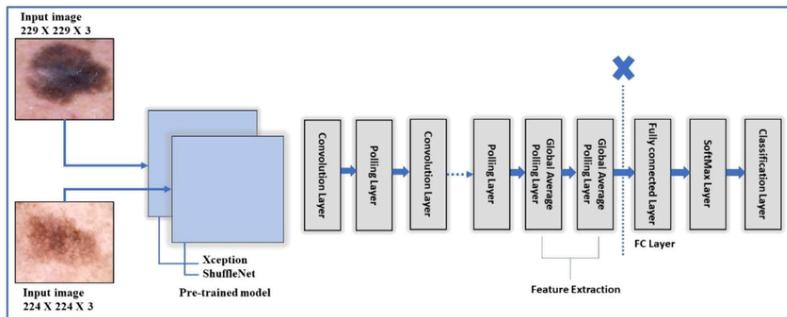
To enhance efficiency in brain tumor detection, the model incorporates techniques such as truncated normal weight initialization and batch normalization after each convolutional layer, fostering stability and effectiveness during training. The EfficientNetB model is employed with pre-trained weights from the ImageNet database, enhancing its ability to recognize complex patterns and features. The model is configured to exclude the top layer, allowing for further customization. The output from the EfficientNetB base model is passed through a batch normalization layer to standardize and normalize the activations. Subsequently, a dense layer with 256 units is added, incorporating regularization techniques such as L2 kernel regularization, L1 activity regularization, and L1 bias regularization to prevent overfitting. To introduce non-linearity, the ReLU activation function is applied. A dropout layer with a rate of 0.45 is included to selectively deactivate neurons during training, promoting model robustness and preventing overfitting. The final layer consists of a dense layer with a softmax activation function, which is suitable for multi-class classification tasks like identifying different types of brain tumors. The overall model is defined using the Keras Model API, specifying the input as the EfficientNetB7 base model input and the output as the constructed neural network's final layer. The model is compiled using the Adamax optimizer with a learning rate of 0.001, employing categorical cross-entropy as the loss function and accuracy as the evaluation metric. This comprehensive architecture, leveraging EfficientNetB and incorporating regularization techniques, demonstrates a sophisticated approach to brain tumor detection, capitalizing on the strengths of pre-trained models for efficient and effective feature extraction.



2. Melanoma Detection

The Xception is configured with specific parameters, including the exclusion of the top layer (fully connected layer), pre-trained weights from 'imagenet,' and an input shape of (224, 224, 3) representing a standard RGB image size. Subsequently, the model's output is processed through a Global Average Pooling layer, which spatially aggregates the extracted features. A final Dense layer with a softmax activation function is added to facilitate three-class skin cancer classification. The comprehensive model is defined using the Keras functional API, combining the base Xception model and the newly added layers, forming an integrated architecture for robust skin cancer detection.

The ensuing methodological architectural setup demonstrates the model's readiness for training through the specified compilation parameters. The categorical cross-entropy loss function is chosen to optimize the model's classification capabilities with respect to multiple classes. The Adam optimizer, a popular choice for deep learning tasks, is configured with a learning rate of 0.0001 to regulate the update steps during training. Additionally, model evaluation during training and testing is facilitated by the accuracy metric, providing insights into the classification performance. This highly technical formulation encapsulates a state-of-the-art approach to skin cancer detection, combining the strengths of the Xception architecture, meticulous layer design, and effective training configuration to achieve accurate and efficient results in skin cancer classification tasks.



3. Chest X-ray Dataset:

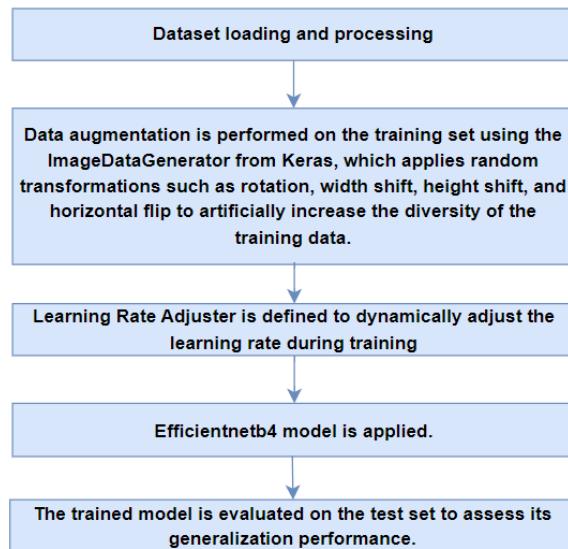
Employing the Xception architecture for analyzing the NIH Chest X-ray dataset involves a sophisticated deep learning model. Xception is a multi-layered convolutional neural network (CNN) with distinctive depth-wise and pointwise convolutional layers. The architecture comprises three main stages: entry flow, middle flow (repeated eight times), and exit flow. The data progresses through these stages, starting with the entry flow (depicted in blue), moving through the middle flow (depicted in grey), and concluding with the exit flow. Following the fully connected layer, all vectors undergo a SoftMax function, and the subsequent application of binary cross-entropy loss facilitates the derivation of binary classification results. This amalgamation of the Xception architecture with the NIH Chest X-ray dataset enhances the model's ability to recognize intricate patterns and achieve accurate disease classification in medical diagnostics.

3. EfficientNetB Models:

EfficientNet architecture utilizes compound coefficients for effective scaling, expanding the width, resolution, and depth of resources in a constant ratio without compromising efficiency. Leveraging the AutoML MNAS framework for neural architecture search, a new baseline network was developed, dependent on the compound scaling method, enhancing both accuracy and efficiency (FLOPS). This architecture employs the mobile inverted bottleneck convolution (MBConv), and by consistently

applying the compound scaling technique, a series of models ranging from EfficientNet-B1 to EfficientNet-B7 were obtained.

Model	Width Multiplier	Depth Multiplier	Resolution	Parameters (Millions)	FLOPs (Billions)
EfficientNet-B0	1.0	1.0	224×224	5.3	0.77
EfficientNet-B1	1.0	1.1	240×240	7.8	1.15
EfficientNet-B2	1.1	1.2	260×260	9.2	1.37
EfficientNet-B3	1.2	1.4	300×300	12.0	2.24
EfficientNet-B4	1.4	1.8	380×380	19.0	4.20
EfficientNet-B5	1.6	2.2	456×456	30.0	10.78
EfficientNet-B6	1.8	2.6	528×528	43.0	19.96
EfficientNet-B7	2.0	3.1	600×600	66.0	37.66



1. Brain Tumor Detection

The Generalized Methodology for EfficientNet Series in Brain Tumor Detection involves the use of three different pre-trained models from the EfficientNet series, namely EfficientNetB5, EfficientNetB6, and EfficientNetB7. The methodology employs transfer learning, leveraging the pre-trained weights on ImageNet for feature extraction and utilizing a custom neural network for the final classification task. For each model, the process begins by initializing the base model with EfficientNet architecture, excluding the top classification layer. The inclusion of top layers is set to False, and the weights are initialized with those pre-trained on the ImageNet dataset. The input shape is defined according to the desired image dimensions for brain tumor detection, and max pooling is applied to downsample the spatial dimensions of the output. After obtaining the base model's output, batch normalization is applied to enhance training stability. Subsequently, a densely connected layer with 256 units is added, incorporating kernel regularization, activity regularization, and bias regularization to mitigate overfitting. The activation function used is rectified linear unit (ReLU).

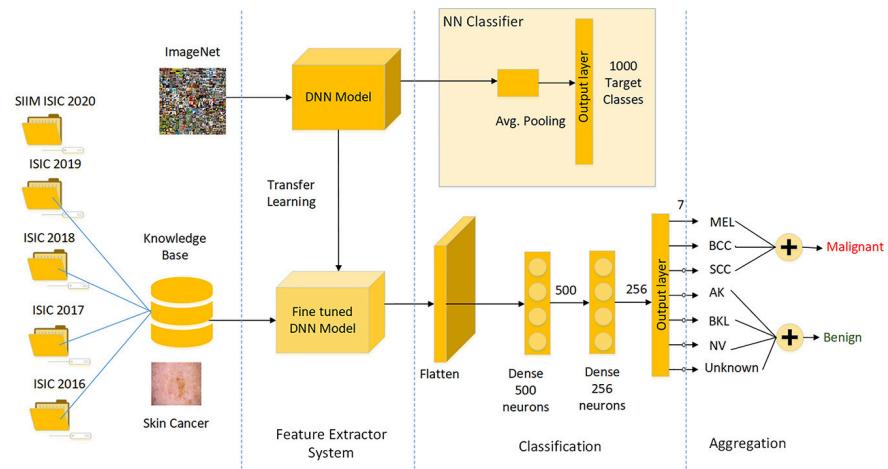
A dropout layer is introduced with a dropout rate of 0.45 to further prevent overfitting. The final classification layer consists of a dense layer with the number of units equal to the target class count (presumably the number of brain tumor classes). The activation function for this layer is softmax,

facilitating multi-class classification. The entire model is constructed using the Keras Functional API, with the input as the base model's input and the output as the final classification layer. Finally, the model is compiled using the Adamax optimizer with a learning rate of 0.001. The loss function chosen is categorical crossentropy, suitable for multi-class classification tasks. The evaluation metric for model performance is accuracy. This methodology provides a systematic approach to leveraging EfficientNet models for brain tumor detection, allowing for scalability and adaptability across different model sizes within the EfficientNet series.

2. Melanoma Detection

The methodology for skin cancer detection adopts the EfficientNet-B6 model, a sophisticated convolutional neural network design that deviates from traditional CNN architectures. Unlike typical CNNs that primarily focus on optimizing layer architecture, EfficientNet seeks to strike a balance between accuracy and computational efficiency by expanding network depth, width, and input resolution. The baseline network, EfficientNet-B0, is derived through a multi-objective neural architecture search inspired by MnasNet. The subsequent generation of EfficientNet-B6 utilizes the compound scaling method, incorporating user-specified coefficients for network depth (d), width (w), and resolution (r). The superior performance of EfficientNet-B6 is attributed to its deeper, wider, and higher-resolution architecture, capturing richer features and patterns.

In the transfer learning phase, the last layer of EfficientNet-B6 is employed to extract high-level feature representations from input skin lesion images. The classifier built atop these features consists of four fully connected layers. Transfer learning proves beneficial in this context, accelerating training and aiding the model in finding a better convergence state for inference. Leveraging pre-trained EfficientNet weights facilitates the extraction of fine-grained features from dermoscopic skin images, a task demanding substantial computational resources. The four less complex fully connected layers are initialized using the Xavier method and trained from scratch, contributing to the model's overall efficacy in skin cancer detection. This highly technical approach underscores the sophistication of the EfficientNet-B6 model and its application in extracting and classifying intricate features crucial for accurate diagnosis.



3. Chest X-ray Dataset:

EfficientNetB4, a variant of the EfficientNet architecture, is adeptly employed for chest X-ray detection, showcasing a harmonious blend of depth, width, and resolution scaling. The model parameters are fine-tuned to strike an optimal balance, achieving impressive computational efficiency with superior

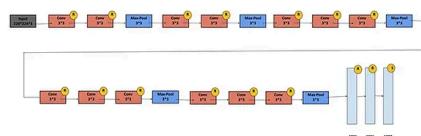
accuracy. Leveraging transfer learning, the initial layers of EfficientNetB4, responsible for general feature extraction, are retained from pre-trained models on ImageNet, while the classifier is tailored for the specific task of disease classification. The classifier is redefined with densely connected layers, including ReLU activations and dropout for regularization, ultimately culminating in an output layer employing sigmoid activation for multi-label classification. EfficientNetB4's methodology for chest X-ray detection relies on a strategic combination of convolutional layers and global average pooling to extract intricate features. Its distinctive architecture minimizes the risk of overfitting while maximizing the model's capacity to discern nuanced patterns within chest X-ray images. The incorporation of EfficientNetB4, coupled with meticulous fine-tuning and transfer learning, manifests as a potent framework for medical diagnostics, demonstrating the fusion of efficiency, adaptability, and precision in the pursuit of robust chest X-ray disease detection.

4. VGG Model:

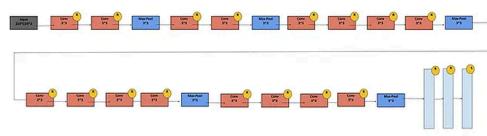
VGG16 and VGG19 are actually an improvement over AlexNet. They replaced the large sized filters of AlexNet with multiple 3x3 filters. They are convolutional neural network architectures. Introduced as part of the ImageNet Large Scale Visual Recognition Challenge, these models are renowned for their simplicity and effectiveness in image classification tasks. VGG16 consists of 16 layers, including 13 convolutional and 3 fully connected layers, while VGG19 extends the depth to 19 layers. The networks are characterized by the use of small 3x3 convolutional kernels, ReLU activation functions, and max pooling for spatial downsampling. VGG models have served as influential benchmarks in the field of computer vision, laying the groundwork for subsequent architectures.

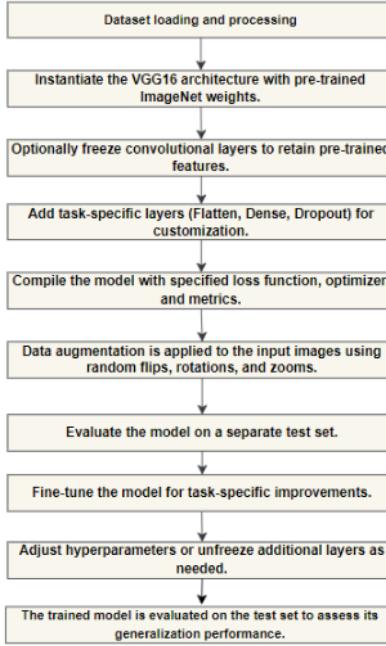
Feature	VGG16	VGG19
Number of Layers	16 layers	19 layers
Convolutional Layers	13 layers	16 layers
Fully Connected Layers	3 layers	3 layers
Convolutional Kernels	3x3	3x3
Pooling	Max pooling with 2x2 windows, stride of 2	Max pooling with 2x2 windows, stride of 2
Activation Function	ReLU	ReLU
Fully Connected Units	4096 units	4096 units
Dropout	Used in the fully connected layers	Used in the fully connected layers
Batch Normalization	Applied to convolutional layers	Applied to convolutional layers

VGG-16



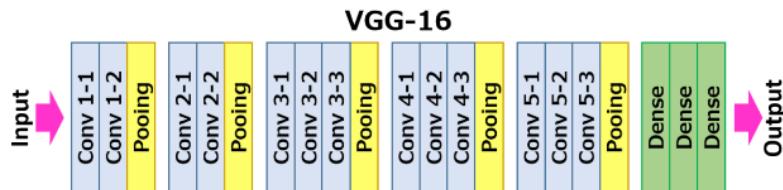
VGG-19





1. Brain Tumor Detection

Brain tumor detection using the VGG-16 architecture involves leveraging its capabilities in image classification for medical imaging tasks. The process begins with the acquisition of magnetic resonance imaging (MRI) scans of the brain, where the VGG-16 model is applied to analyze these images. The model, pre-trained on large datasets like ImageNet, has learned to recognize a wide range of features and patterns in images. In the context of brain tumor detection, the VGG-16 model is fine-tuned on a dataset specifically curated for this task. The dataset includes MRI scans with labeled images indicating the presence or absence of brain tumors. During the fine-tuning process, the pre-trained weights of VGG-16 are adjusted to adapt to the distinctive features and characteristics of brain tumor images. The trained VGG-16 model is then employed to predict whether a given MRI scan contains indications of a brain tumor. The model outputs class probabilities for each category (presence or absence of a tumor), and a threshold is set to classify the scan based on these probabilities. If the probability surpasses the threshold, the model detects the presence of a brain tumor; otherwise, it predicts the absence. The detection process involves analyzing the learned features in the MRI scans, enabling the model to recognize patterns indicative of brain tumors. The use of deep learning models like VGG-16 in brain tumor detection allows for more accurate and efficient analysis of medical imaging data, assisting healthcare professionals in early diagnosis and treatment planning. The trained model can be integrated into clinical workflows, providing valuable support to radiologists and enhancing the overall diagnostic process for brain tumors.

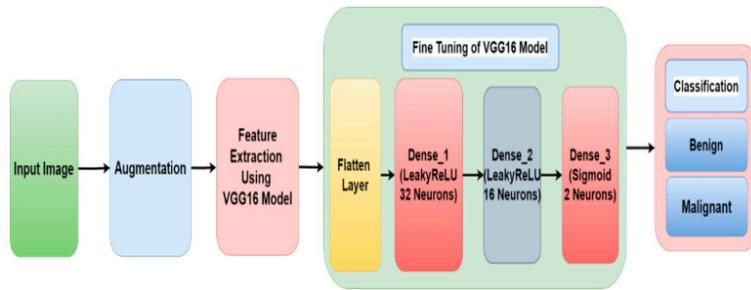


The approach for brain tumor detection utilizing the VGG19 model involves a systematic methodology focused on model architecture, customization, and training. To implement this, the VGG19 model is selected and instantiated using TensorFlow's Keras API. The pre-trained VGG19 base model is loaded with weights from ImageNet and configured to exclude the top (fully-connected) layers, with the input shape specified for subsequent customization. The architecture is then tailored by adding dense layers, batch normalization for stabilization during training, and dropout layers to mitigate overfitting. The final

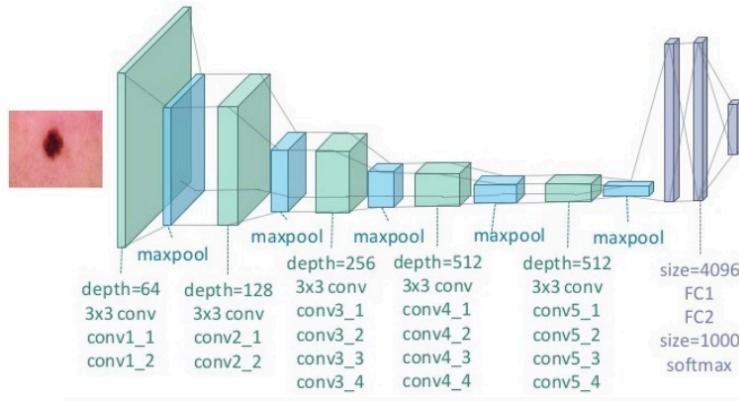
dense layer employs the softmax activation function to output predictions for the specified number of classes. The model is compiled using the Adam optimizer with a learning rate suitable for the task, and categorical cross-entropy is chosen as the loss function, given the multi-class classification nature of the brain tumor detection task. The chosen evaluation metric is accuracy. This methodology harnesses the transfer learning capabilities of the VGG19 model, fine-tuning it for brain tumor detection by adapting the architecture and training on relevant datasets. The regularization techniques, optimizer choice, and metric selection contribute to the creation of a robust and effective model for medical image analysis.

2. Melanoma Detection

The modified VGG-16 architecture comprises convolutional layers with trainable parameters reduced for memory efficiency, specifically using FC-512 instead of FC-4096 in the fully connected layers. The total number of parameters is 27,560,769. To leverage transfer learning, pre-trained weights from the ImageNet dataset are used. The fine-tuning process involves training the fully connected layers initially with the new dataset to avoid randomly initialized weights that could disrupt learned convolutional layer weights. After this phase, the trained FC weights and pre-trained VGG-16 weights are combined for further fine-tuning. In the fine-tuning process, only the top layers of the network are trained, as they contain more dataset-specific features. The first four convolutional layers remain frozen, utilizing pre-trained VGG-16 weights from the ImageNet dataset. The last convolutional and fully connected layers are trained using the pre-trained weights from the fully connected layers. The number of trainable weights is reduced to 19,925,505. Adam optimization is employed for training the network, with an initial learning rate set to 4e-5 and a learning rate decay of 8e-6 at each epoch. This meticulous methodology ensures efficient training, leveraging the strengths of pre-trained weights for feature extraction while adapting the model for skin cancer detection.

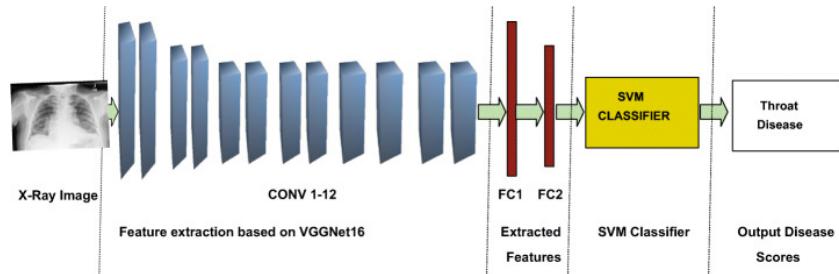


The implemented VGG19-based skin cancer detection model begins by loading the VGG19 architecture with pre-trained weights from ImageNet. To focus on fine-tuning the model for skin cancer detection, the last four layers of VGG19 are set as trainable, while the remaining layers are frozen. This modification allows the model to adapt its learned features to the specific characteristics of skin cancer images. The architecture of the model is extended by adding a flattening layer followed by two dense layers with 4096 neurons and ReLU activation functions. The final layer consists of a single neuron with a sigmoid activation function, producing binary classification probabilities. Data augmentation is employed during training using the ImageDataGenerator, enhancing the model's ability to generalize from limited training data. The model is compiled using binary cross-entropy loss and an RMSprop optimizer with a learning rate of 1e-4. The training process involves fitting the model to the training data using a generator, with a specified number of steps per epoch and epochs. Additionally, validation is performed on a separate test dataset, allowing for performance evaluation. The training history, including accuracy and loss metrics, is stored for subsequent analysis and model assessment.



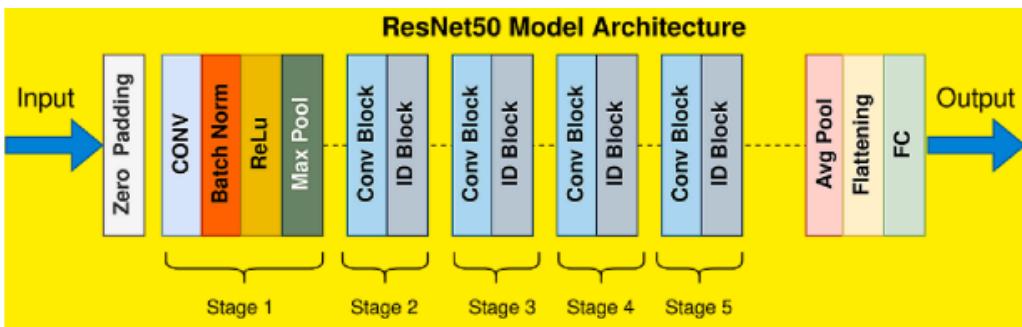
3. Chest X-ray Dataset:

Utilizing the VGG architecture for chest X-ray analysis is exemplified through a meticulously constructed image data generator in the provided code snippet. Employing the Keras framework, the VGG16 pre-trained model, denoted as PTModel, is harnessed for feature extraction. The image data generator, with tailored configurations, encompasses critical transformations such as horizontal and vertical flips, height and width shifts, brightness adjustments, rotations, shearing, and zooming. These augmentations contribute to a more robust and diverse dataset, crucial for training a highly accurate model. The preprocessing function preprocess_input aligns the image data with VGG16 expectations, ensuring seamless integration. With an image size slightly smaller than the typical VGG16 input, set at (512, 512), this approach not only adheres to the model's specifications but also enhances its adaptability to chest X-ray images. In essence, the incorporation of the VGG architecture, coupled with meticulous data augmentation, establishes a powerful foundation for precise feature extraction and subsequent disease classification in the realm of medical diagnostics.



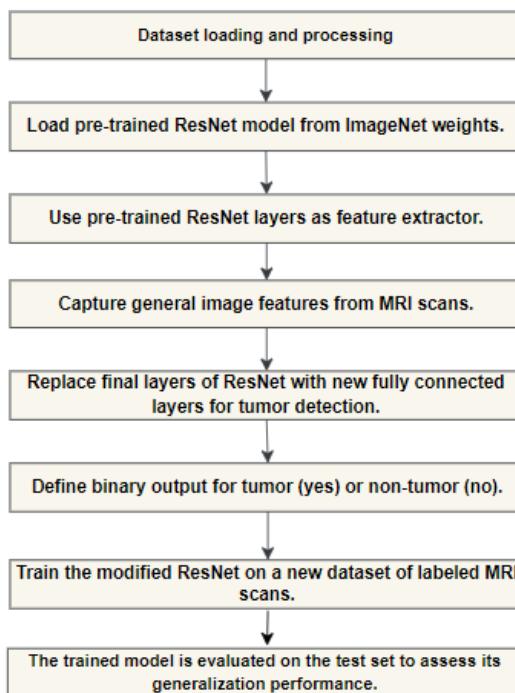
5. ResNet:

ResNet models are a deep learning algorithmic approach that can be used to detect brain tumors . The ResNet (Residual Network) architecture was introduced to address the vanishing gradient problem in deep neural networks. ResNet uses skip connections, also known as residual connections, to enable the flow of information across layers more efficiently. The model can be fine-tuned on a new dataset of MRI scans to adapt it to the specific task. ResNet50 is a CNN model trained on the large-scale ImageNet dataset for object recognition tasks. It contains different layers, including convolutional, pooling, and fully connected layers. This model can be used as a feature extractor for the brain tumor detection task. The fundamental building block of ResNet is the residual block. Each block consists of two main paths: a shortcut connection (skip connection) and a main convolutional path. The shortcut connection bypasses one or more layers and directly connects the input to the output of the residual block. The main convolutional path typically includes two 3x3 convolutional layers (with batch normalization and ReLU activation functions) and sometimes a 1x1 convolutional layer for adjusting dimensions.



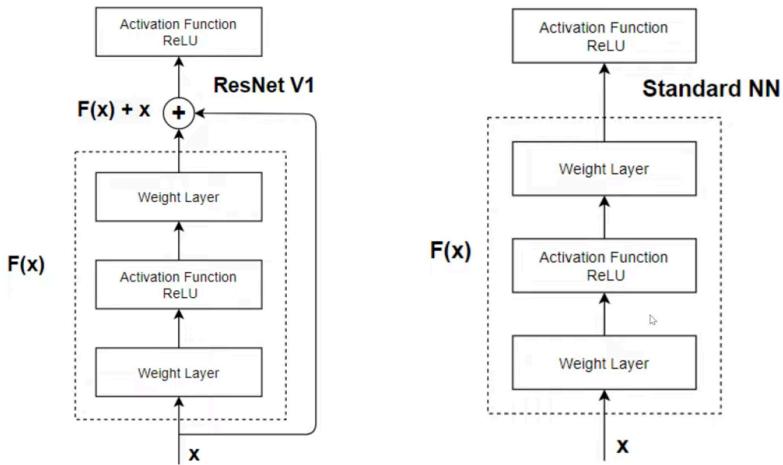
The shortcut connection can either be an identity mapping (when the input and output dimensions are the same) or a projection (when the dimensions change). If the dimensions change, a 1x1 convolutional layer is used in the shortcut to match the dimensions of the main path. Traditional networks learn functions $H(x)$, where H is the mapping to be learned. ResNet learns residual functions $F(x)=H(x)-x$. The output of the residual block is then $F(x)+x$, allowing for the easy flow of gradients during backpropagation. ResNet comes in various depths, such as ResNet-18, ResNet-34, ResNet-50, ResNet-101, and ResNet-152. ResNet typically employs global average pooling. GAP reduces the spatial dimensions to 1x1 and computes the average value for each feature map, resulting in a compact representation of the entire input.

ResNet models are often pre-trained on large datasets like ImageNet for generic feature extraction. The pre-trained model's weights can then be fine-tuned for specific tasks, such as brain tumor detection in your case. Fine-tuning involves replacing the final layers of the pre-trained ResNet model with task-specific layers. During fine-tuning, the entire model is trained on the new dataset, adapting the learned features to the specific characteristics of brain tumor detection. ResNet50 is used as a feature extractor for MRI scans. The lower layers capture general image features, while the final layers are fine-tuned for brain tumor detection. The output is a probability distribution, and a threshold is applied to make the final decision on tumor presence or absence. ResNet's architecture with residual blocks and skip connections allows for the training of very deep neural networks.



1. Brain Tumor Detection

The ResNet50 model, known for handling deep network training challenges, is utilized for its ability to address deterioration issues and vanishing boundaries. The feature extraction phase employs ResNet50 for accurate brain cancer identification, considering various ML parameters and dataset characteristics. Augmentation is introduced to diversify and enhance the training data, involving geometric alterations and changes in tone space. Feature extraction is crucial for CNN feature vector extraction, and a leave-one-out cross-validation technique is employed. The model training phase utilizes the augmented data, and the CNN model is trained with leave-one-out cross-validation to predict tumor areas accurately. The methodology is evaluated using numerical results, including accuracy, precision, sensitivity, and specificity. The proposed model demonstrates significant improvements after augmentation, as reflected in the numerical results. The process is illustrated using bar charts for image ratios, confusion matrices, and graphs depicting model accuracy and loss. Overall, the proposed methodology combines data pre-processing, transfer learning, feature extraction, and augmentation to enhance brain tumor detection accuracy.

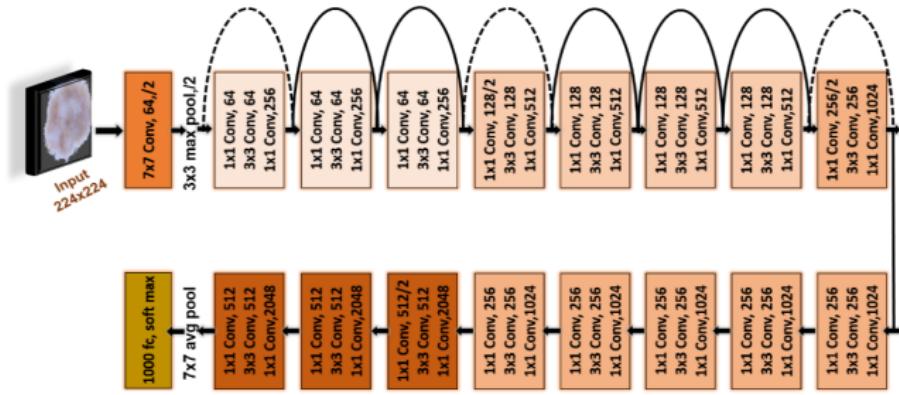


2. Melanoma Detection:

The skin cancer detection model employs a Convolutional Neural Network (CNN) based on the ResNet50 architecture. The convolutional base, initialized with pre-trained weights from ImageNet, is used as a feature extractor. This base is then extended with custom dense layers to adapt it for skin cancer classification. The model includes dropout layers to prevent overfitting, and the final output layer uses the sigmoid activation function for binary classification. The Convolutional Base (conv_base) is constructed using the ResNet50 architecture, pre-trained on ImageNet. This base serves as a feature extractor due to its ability to capture intricate patterns in images. The model summary is printed to provide an overview of the architecture. A Sequential model is then created, incorporating the conv_base as the first layer. The model is extended with additional layers: a Flatten layer, two Dropout layers with a dropout rate of 0.5, a Dense layer with ReLU activation and L2 regularization, another Dropout layer, and the final Dense layer with a sigmoid activation function for binary classification.

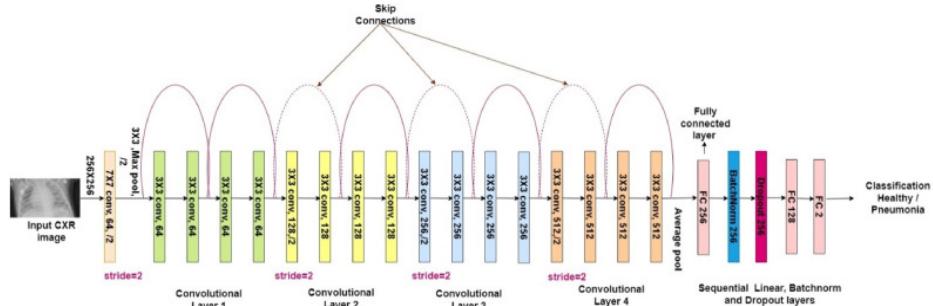
The model is compiled using the Adam optimizer, binary cross-entropy loss function, and accuracy as the evaluation metric. The training process involves using data generators to augment the training dataset, enhancing the model's ability to generalize. The model is trained for a specified number of epochs, with the training and validation datasets prepared using the ImageDataGenerator. During training, only the top layers of the model are updated, preserving the pre-trained weights of the convolutional base. This approach ensures that the learned features from the ImageNet dataset are not

discarded during the random initialization of new weights. The training history, including accuracy and loss metrics, is recorded for subsequent analysis.



3. Chest X-ray Dataset:

The implementation of ResNet for chest X-ray analysis is exemplified through a highly modularized and versatile ResNet layer, encapsulated within the `resnet_layer` function. This function systematically constructs a 2D Convolution-Batch Normalization-Activation stack, allowing for seamless customization of key parameters. The user-defined inputs, encompassing the input tensor, number of filters, kernel size, strides, activation function, and the inclusion of batch normalization, empower adaptability for varied experimental configurations. The `conv_first` parameter dictates the layer order, facilitating conv-bn-activation when set to True, or bn-activation-conv when set to False. The `Conv2D` layer within employs parameters conducive to robust feature learning, including a square kernel, consistent padding, He-normal initialization, and L2 regularization. This modularized ResNet layer serves as a foundational building block, affording flexibility and precision in crafting intricate neural network architectures for chest X-ray analysis.



<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8913041/>

Evaluation Metrics:

- Accuracy:** Accuracy is a metric that measures how often a machine learning model correctly predicts the outcome. It is calculated by dividing the number of correct predictions by the total number of predictions.
- Precision:** The ratio of true positive predictions to the total predicted positives, measuring the accuracy of positive predictions and indicating how often the model is correct when it predicts the positive class.
- Recall (Sensitivity):** The ratio of true positive predictions to the total actual positives, assessing the ability of the model to capture all relevant instances of the positive class.

4. **Specificity**: The ratio of true negative predictions to the total actual negatives, indicating the model's ability to correctly identify instances of the negative class.
5. **F-score (F1 Score)**: The harmonic mean of precision and recall, providing a single metric that balances the trade-off between precision and recall. It is particularly useful when there is an uneven class distribution.
6. **Confusion Matrix**: The confusion matrix is a tabular representation of these metrics, breaking down the performance of a classification model into counts of true positives, true negatives, false positives, and false negatives.

1. BRAIN TUMOR DATASET

MODEL NAME	ACCURACY	PRECISION	RECALL	F1 SCORE	SPECIFICITY	AUC
Simple	0.9633	0.96	0.9673	0.9636	0.9615	0.9644
Xception	0.9733	0.9675	0.9667	0.9671	0.9674	0.967
Efficientnet	0.9967	0.9933	1	0.9967	0.9934	0.9967
VGG16	0.93	0.9451	0.9043	0.9242	0.9424	0.9234
ResNet	0.98	0.9679	0.98	0.9739	0.9675	0.9738

2. MELANOMA DATASET

MODEL NAME	ACCURACY	PRECISION	RECALL	F1 SCORE	SPECIFICITY	AUC
Simple	0.9366	0.9375	0.9736	0.9205	0.9026	0.9321
Xception	0.9421	0.9399	0.9491	0.9437	0.9431	0.9385
Efficientnet	0.9852	0.9834	0.9782	0.9867	0.9634	0.9842
VGG16	0.9810	0.9418	0.9512	0.9546	0.9348	0.9635
ResNet	0.9588	0.9615	0.9437	0.9425	0.9315	0.9417

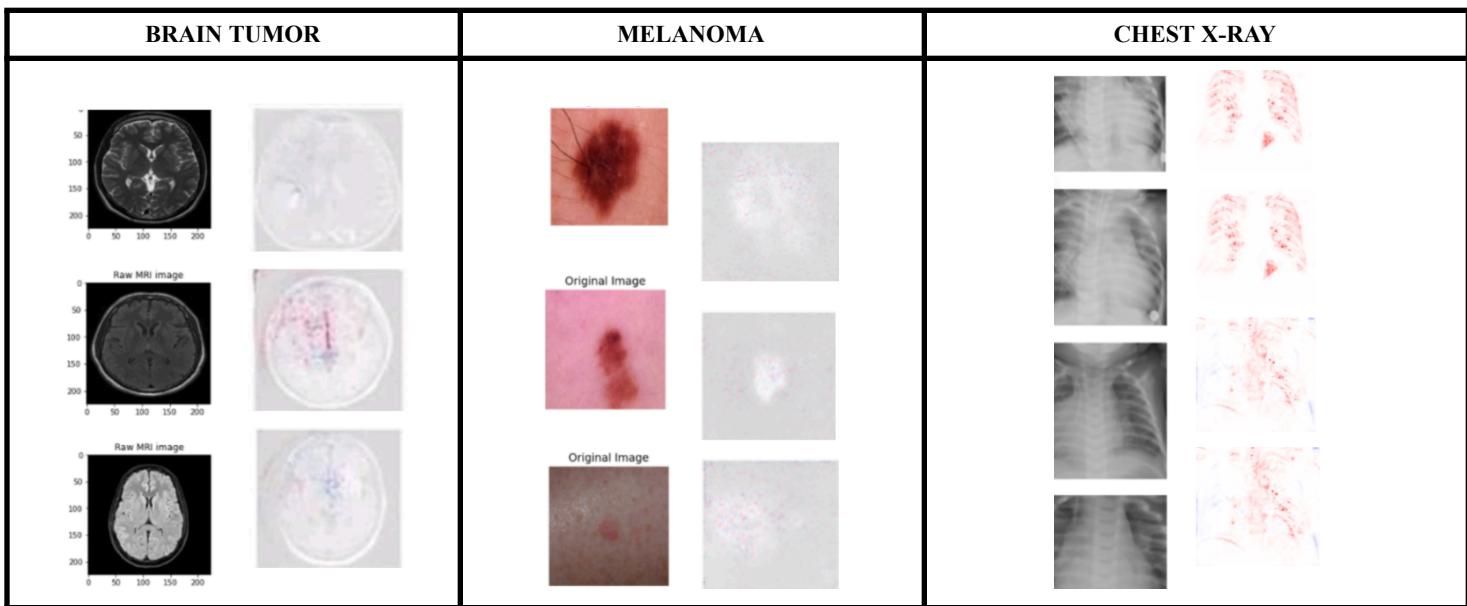
3. CHEST X-RAY DATASET

MODEL NAME	ACCURACY	PRECISION	RECALL	F1 SCORE	SPECIFICITY	AUC
Simple	0.9012	0.8914	0.9019	0.9001	0.8896	0.9502
Xception	0.9692	0.9633	0.9600	0.9445	0.9611	0.9682
Efficientnet	0.9751	0.9774	0.9770	0.9771	0.9679	0.9692
VGG16	0.9397	0.9412	0.9381	0.9302	0.9269	0.9221
ResNet	0.9166	0.9002	0.9051	0.9105	0.9016	0.9078

Explainable AI Models for interpretation:-

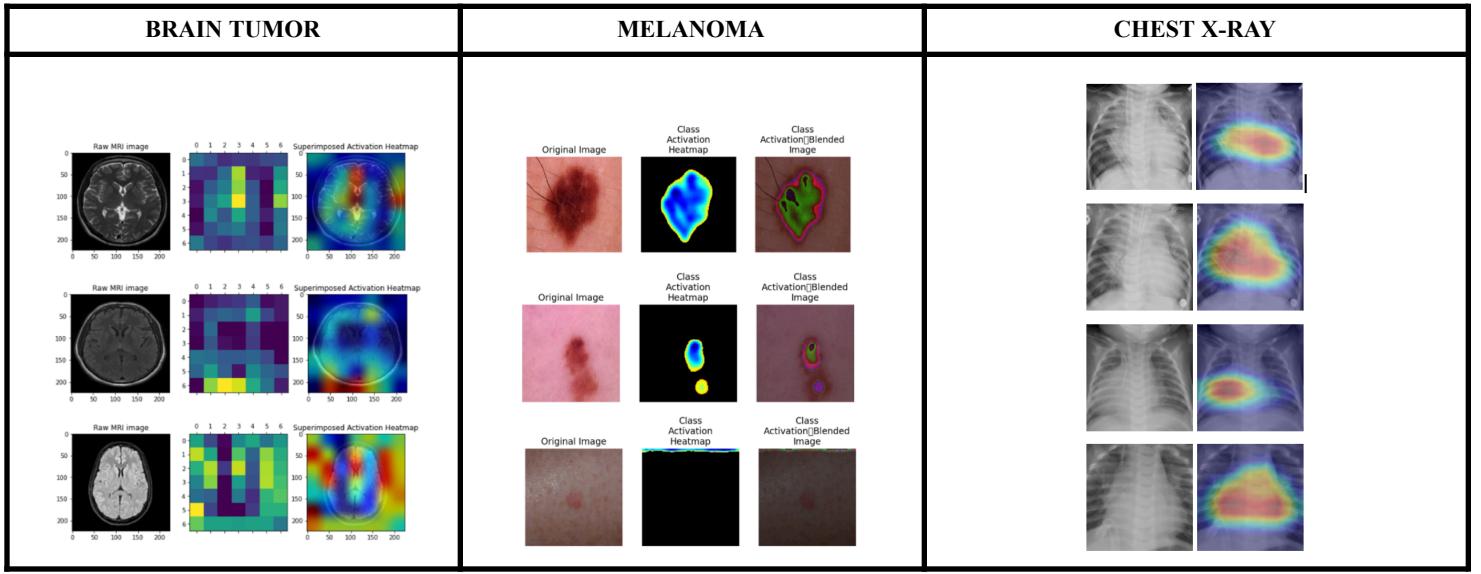
1. LRP

Layer-wise Relevance Propagation (LRP) is a technique in deep learning designed for interpretability by assigning relevance scores to individual neurons in a neural network, aiding in understanding model decisions. The process involves initializing relevance at the output layer, propagating it backward through the network based on neuron contributions, and ensuring conservation of relevance. Various redistribution rules prevent vanishing or exploding relevance, providing insights into the significance of input features in the model's output. LRP has been applied to diverse neural network architectures, including CNNs and RNNs, making it valuable in applications where transparency and interpretability are crucial for building user trust and understanding model decisions.



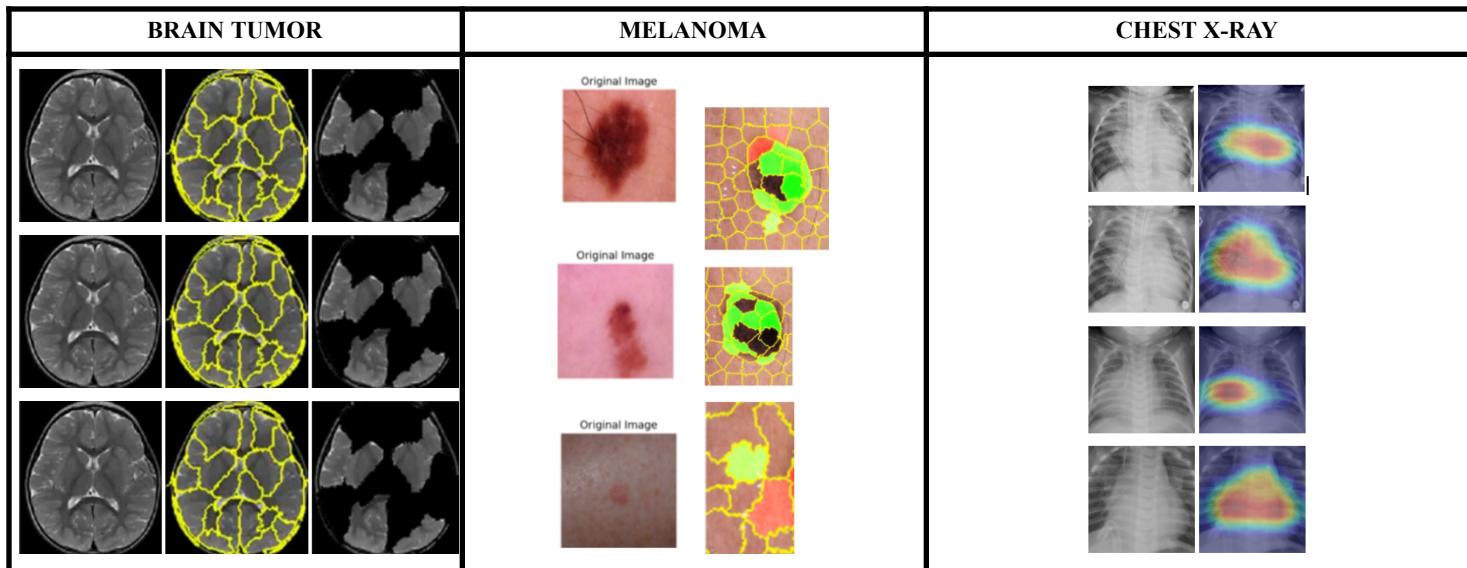
2. Grad CAM

Gradient-weighted Class Activation Mapping is a technique used in deep learning for visualizing and understanding the decision-making process of a neural network, particularly in the context of convolutional neural networks (CNNs). Grad-CAM helps identify the regions in an input image that contribute most to a specific class prediction. The method leverages the gradients of the predicted class score with respect to the feature maps of the last convolutional layer in the network. By computing these gradients, Grad-CAM highlights the importance of different spatial locations in the feature maps, producing a heatmap that visually represents the regions that strongly influence the final prediction.



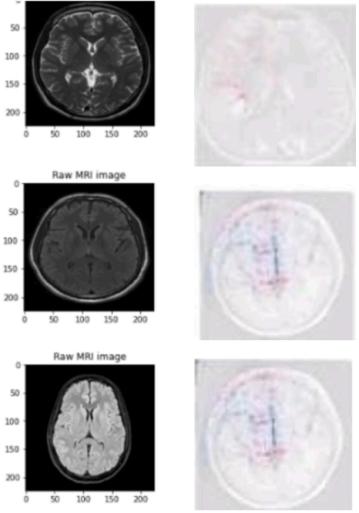
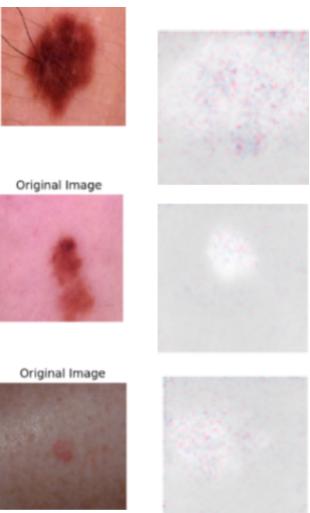
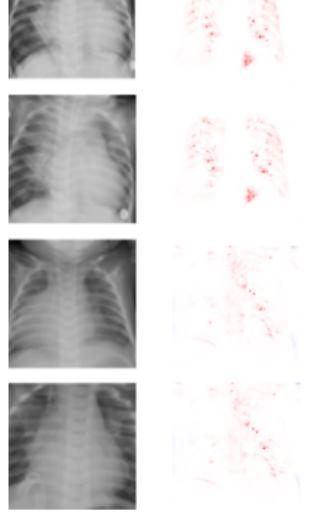
3. Local interpretable Model-agnostic Explanations

LIME, which stands for Local Interpretable Model-agnostic Explanations, is a crucial technique in the field of Explainable Artificial Intelligence (XAI). It provides detailed insights into the predictions made by machine learning models. The main advantage of this approach is its capacity to offer detailed explanations at a local level, focusing on specific occurrences rather than the overall model. LIME is a model-agnostic solution that may be used with different machine learning methods due to its versatility. This is accomplished by simplifying the behavior of the black-box model through training interpretable models, usually linear ones, using modified samples of the original data. These disturbances allow for the analysis of the significance of features, revealing the fundamental reasoning behind particular forecasts. By utilizing visual representations like heatmaps or bar charts, LIME enhances comprehension of the contributions made by different features, hence promoting transparency and confidence in AI systems.



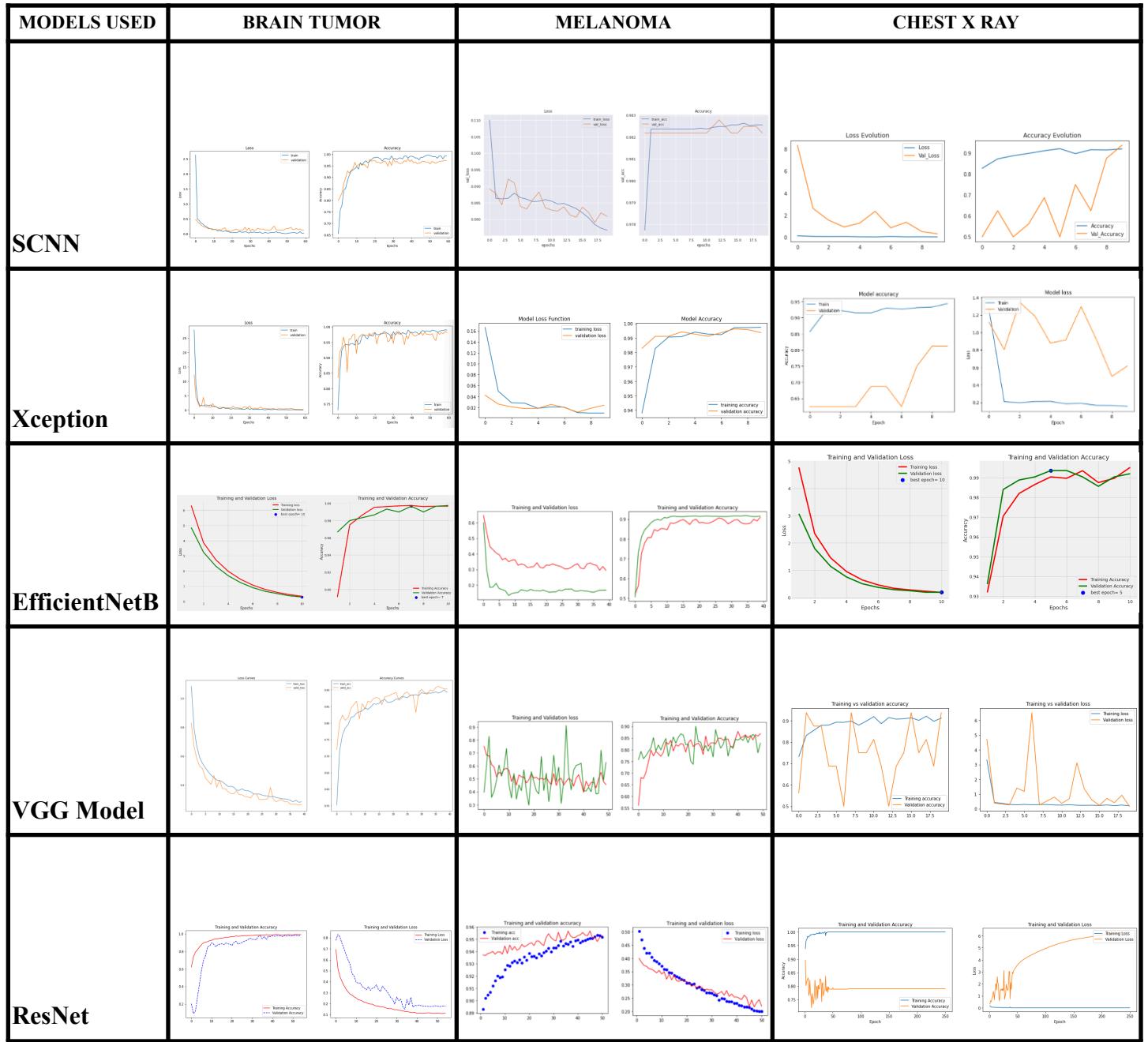
4. SHapley Additive exPlanations

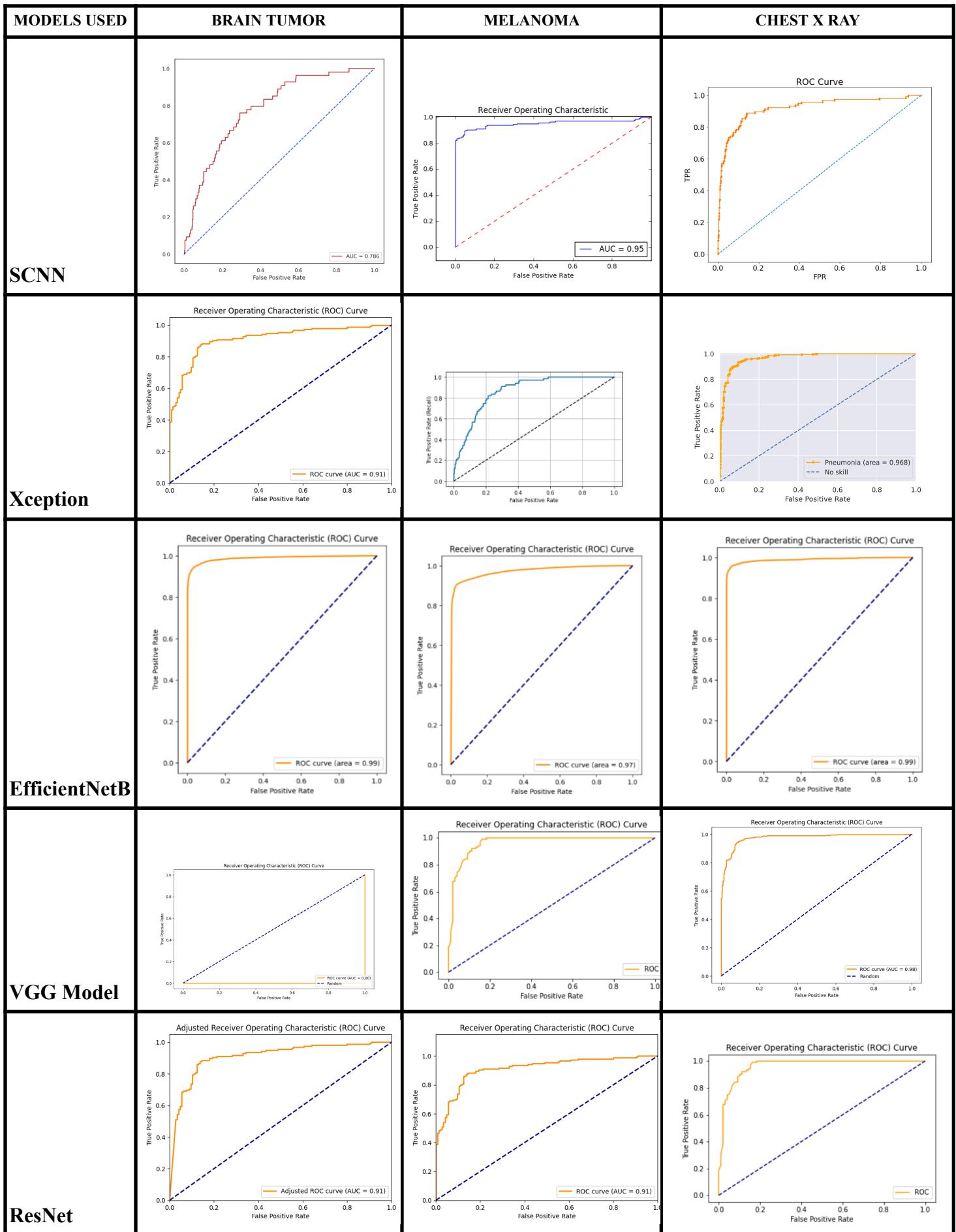
SHAP (SHapley Additive exPlanations) is a very adaptable technique used in the field of machine learning. It is well-known for its ability to offer detailed explanations for model predictions, both on a local and global scale. The main advantage of this approach is its reliance on cooperative game theory, particularly the concept of Shapley values. This allows for the accurate assessment of the contributions made by each feature to the model's output. SHAP calculates the Shapley values for features, which helps to clarify the influence of individual variables on predictions. This process makes it easier to analyze and build confidence in complex models. This methodology provides both quantitative explanations and visual representations, including summary plots and force charts, which improve understanding in various applications and models. As a result, it plays a crucial role in making AI more explainable.

BRAIN TUMOR	MELANOMA	CHEST X-RAY
		

Results:

MODELS USED	BRAIN TUMOR	MELANOMA	CHEST X RAY																																				
SCNN	<p>Confusion Matrix</p> <table border="1"> <tr><th colspan="2">Predicted Labels</th><th>No Tumor</th><th>Tumor</th></tr> <tr><th>True Labels</th><th>No Tumor</th><td>144</td><td>6</td></tr> <tr><th>Tumor</th><td>5</td><td>145</td><td></td></tr> </table>	Predicted Labels		No Tumor	Tumor	True Labels	No Tumor	144	6	Tumor	5	145		<p>Confusion Matrix Accuracy: 96.60%</p> <table border="1"> <tr><th colspan="2">Predicted Label</th><th>Actual Negative</th><th>Actual Positive</th></tr> <tr><th>Actual Label</th><th>Actual Negative</th><td>7.88e+03</td><td>1.79e+02</td></tr> <tr><th>Actual Positive</th><td>1.05e+02</td><td>1.88e+02</td><td></td></tr> </table>	Predicted Label		Actual Negative	Actual Positive	Actual Label	Actual Negative	7.88e+03	1.79e+02	Actual Positive	1.05e+02	1.88e+02		<p>Confusion Matrix</p> <table border="1"> <tr><th colspan="2">Predicted Label</th><th>PNEUMONIA</th><th>NORMAL</th></tr> <tr><th>Actual Label</th><th>PNEUMONIA</th><td>373</td><td>17</td></tr> <tr><th>NORMAL</th><td>29</td><td>205</td><td></td></tr> </table>	Predicted Label		PNEUMONIA	NORMAL	Actual Label	PNEUMONIA	373	17	NORMAL	29	205	
Predicted Labels		No Tumor	Tumor																																				
True Labels	No Tumor	144	6																																				
Tumor	5	145																																					
Predicted Label		Actual Negative	Actual Positive																																				
Actual Label	Actual Negative	7.88e+03	1.79e+02																																				
Actual Positive	1.05e+02	1.88e+02																																					
Predicted Label		PNEUMONIA	NORMAL																																				
Actual Label	PNEUMONIA	373	17																																				
NORMAL	29	205																																					
Xception	<p>Confusion Matrix</p> <table border="1"> <tr><th colspan="2">Predicted Labels</th><th>No Tumor</th><th>Tumor</th></tr> <tr><th>True Labels</th><th>No Tumor</th><td>345</td><td>5</td></tr> <tr><th>Tumor</th><td>3</td><td>147</td><td></td></tr> </table>	Predicted Labels		No Tumor	Tumor	True Labels	No Tumor	345	5	Tumor	3	147		<p>Confusion Matrix Accuracy: 97.05%</p> <table border="1"> <tr><th colspan="2">Predicted Label</th><th>Actual Negative</th><th>Actual Positive</th></tr> <tr><th>Actual Label</th><th>Actual Negative</th><td>7.59e+03</td><td>2.31e+02</td></tr> <tr><th>Actual Positive</th><td>0.00e+00</td><td>5.00e+00</td><td></td></tr> </table>	Predicted Label		Actual Negative	Actual Positive	Actual Label	Actual Negative	7.59e+03	2.31e+02	Actual Positive	0.00e+00	5.00e+00		<p>Confusion Matrix</p> <table border="1"> <tr><th colspan="2">Predicted Label</th><th>PNEUMONIA</th><th>NORMAL</th></tr> <tr><th>Actual Label</th><th>PNEUMONIA</th><td>162</td><td>72</td></tr> <tr><th>NORMAL</th><td>3</td><td>387</td><td></td></tr> </table>	Predicted Label		PNEUMONIA	NORMAL	Actual Label	PNEUMONIA	162	72	NORMAL	3	387	
Predicted Labels		No Tumor	Tumor																																				
True Labels	No Tumor	345	5																																				
Tumor	3	147																																					
Predicted Label		Actual Negative	Actual Positive																																				
Actual Label	Actual Negative	7.59e+03	2.31e+02																																				
Actual Positive	0.00e+00	5.00e+00																																					
Predicted Label		PNEUMONIA	NORMAL																																				
Actual Label	PNEUMONIA	162	72																																				
NORMAL	3	387																																					
EfficientNetB	<p>Confusion Matrix</p> <table border="1"> <tr><th colspan="2">Predicted</th><th>no</th><th>yes</th></tr> <tr><th>Actual</th><th>no</th><td>149</td><td>1</td></tr> <tr><th>yes</th><td>0</td><td>150</td><td></td></tr> </table>	Predicted		no	yes	Actual	no	149	1	yes	0	150		<p>Confusion Matrix Accuracy: 0.995</p> <table border="1"> <tr><th colspan="2">Predicted label</th><th>Benign</th><th>Malignant</th></tr> <tr><th>True Label</th><th>Malignant</th><td>1.1e+04</td><td>59</td></tr> <tr><th>Benign</th><td>0</td><td>0</td><td></td></tr> </table>	Predicted label		Benign	Malignant	True Label	Malignant	1.1e+04	59	Benign	0	0		<p>Confusion Matrix</p> <table border="1"> <tr><th colspan="2">Predicted Label</th><th>PNEUMONIA</th><th>NORMAL</th></tr> <tr><th>True Label</th><th>PNEUMONIA</th><td>96</td><td>3</td></tr> <tr><th>NORMAL</th><td>0</td><td>310</td><td></td></tr> </table>	Predicted Label		PNEUMONIA	NORMAL	True Label	PNEUMONIA	96	3	NORMAL	0	310	
Predicted		no	yes																																				
Actual	no	149	1																																				
yes	0	150																																					
Predicted label		Benign	Malignant																																				
True Label	Malignant	1.1e+04	59																																				
Benign	0	0																																					
Predicted Label		PNEUMONIA	NORMAL																																				
True Label	PNEUMONIA	96	3																																				
NORMAL	0	310																																					
VGG Model	<p>Confusion matrix</p> <table border="1"> <tr><th colspan="2">Predicted label</th><th>(0, NO)</th><th>(1, YES)</th></tr> <tr><th>True label</th><th>(0, NO)</th><td>189</td><td>11</td></tr> <tr><th>(1, YES)</th><td>20</td><td>180</td><td></td></tr> </table>	Predicted label		(0, NO)	(1, YES)	True label	(0, NO)	189	11	(1, YES)	20	180		<p>Confusion Matrix Accuracy: 98.48%</p> <table border="1"> <tr><th colspan="2">Predicted Label</th><th>Actual Negative</th><th>Actual Positive</th></tr> <tr><th>Actual Label</th><th>Actual Negative</th><td>1e+04</td><td>1.3e+02</td></tr> <tr><th>Actual Positive</th><td>25</td><td>39</td><td></td></tr> </table>	Predicted Label		Actual Negative	Actual Positive	Actual Label	Actual Negative	1e+04	1.3e+02	Actual Positive	25	39		<p>Confusion Matrix</p> <table border="1"> <tr><th colspan="2">predicted label</th><th>0</th><th>1</th></tr> <tr><th>true label</th><th>0</th><td>178</td><td>56</td></tr> <tr><th>1</th><td>12</td><td>378</td><td></td></tr> </table>	predicted label		0	1	true label	0	178	56	1	12	378	
Predicted label		(0, NO)	(1, YES)																																				
True label	(0, NO)	189	11																																				
(1, YES)	20	180																																					
Predicted Label		Actual Negative	Actual Positive																																				
Actual Label	Actual Negative	1e+04	1.3e+02																																				
Actual Positive	25	39																																					
predicted label		0	1																																				
true label	0	178	56																																				
1	12	378																																					
ResNet	<p>Confusion Matrix</p> <table border="1"> <tr><th colspan="2">Predicted Negative</th><th>Actual Positive</th><th>Actual Negative</th></tr> <tr><th>Actual Positive</th><th>Actual Negative</th><td>146</td><td>5</td></tr> <tr><th>Actual Negative</th><td>3</td><td>148</td><td></td></tr> </table>	Predicted Negative		Actual Positive	Actual Negative	Actual Positive	Actual Negative	146	5	Actual Negative	3	148		<p>Confusion Matrix Accuracy: 98.88%</p> <table border="1"> <tr><th colspan="2">Predicted Label</th><th>Actual Negative</th><th>Actual Positive</th></tr> <tr><th>Actual Label</th><th>Actual Negative</th><td>1e+04</td><td>89</td></tr> <tr><th>Actual Positive</th><td>25</td><td>97</td><td></td></tr> </table>	Predicted Label		Actual Negative	Actual Positive	Actual Label	Actual Negative	1e+04	89	Actual Positive	25	97		<p>Confusion Matrix</p> <table border="1"> <tr><th colspan="2">Predicted Label</th><th>Normal</th><th>Pneumonia</th></tr> <tr><th>True Label</th><th>Normal</th><td>165</td><td>69</td></tr> <tr><th>Pneumonia</th><td>2</td><td>388</td><td></td></tr> </table>	Predicted Label		Normal	Pneumonia	True Label	Normal	165	69	Pneumonia	2	388	
Predicted Negative		Actual Positive	Actual Negative																																				
Actual Positive	Actual Negative	146	5																																				
Actual Negative	3	148																																					
Predicted Label		Actual Negative	Actual Positive																																				
Actual Label	Actual Negative	1e+04	89																																				
Actual Positive	25	97																																					
Predicted Label		Normal	Pneumonia																																				
True Label	Normal	165	69																																				
Pneumonia	2	388																																					





Conclusion:

In conclusion, our comprehensive investigation into the application of sequential neural network architectures – Xception, EfficientNetB7 VGG16, and ResNet – for Brain Tumor, Melanoma, and X-ray Image Analysis has provided valuable insights into their unique capabilities. Through rigorous evaluation metrics, including accuracy, sensitivity, specificity, and AUC-ROC, we have systematically compared their performance on a benchmark dataset, paving the way for informed decisions in selecting optimal neural network architectures. Furthermore, our commitment to addressing the crucial challenge of explainability in AI systems led us to employ XAI techniques. This dual approach enhances the transparency and interpretability of machine-driven medical decisions, contributing to the establishment of trust in intelligent systems within healthcare. The outcomes of our study hold significant implications for advancing the diagnostic capabilities of computer-aided systems, providing a foundation for further integration of AI technologies into the intricate realm of medical diagnostics and treatment decision-making.