Pneumonia Detection Using ResNet50 and VGG16

Shikha Tiwari
Khoury College of Computer Sciences
Northeastern University
Boston, USA
tiwari.shi@northeastern.edu

***Abstract*—Medical imaging, particularly in pneumonia detection from X-ray images, demands accurate interpretation and diagnosis. Deep learning, notably convolutional neural networks (CNNs), presents a promising avenue for automating this process. However, training CNNs from scratch necessitates substantial computational resources and labeled data, posing challenges in practical implementation. In this study, we investigate the efficacy of pre-trained CNN models, specifically VGG16 and ResNet-50, for pneumonia detection tasks. Notably, these models demonstrate promising accuracy rates in pneumonia detection, suggesting their potential for clinical deployment. This study underscores the practicality of leveraging pre-trained CNNs for medical image analysis, offering an efficient approach to enhance diagnostic capabilities in healthcare practice.**

Keywords—Deep Learning, Medical Imaging, Pneumonia Detection, ResNet-50, VGG-16

I. Introduction

Accurate medical image segmentation holds paramount importance in various clinical scenarios, facilitating subsequent research and diagnosis. Thus, the development of automated segmentation algorithms for precise annotation of medical images is imperative, given the expertise and time required for manual segmentation. Convolutional neural networks (CNNs), widely embraced in computer vision, have emerged as a cornerstone in various domains, including medical image analysis, owing to their adeptness at discerning features across diverse abstraction levels.

However, training deep neural networks from scratch is contingent upon data quality and can be hindered by imbalanced datasets, where normal samples outnumber malignant ones, as well as the scarcity of large, expert-annotated datasets. Fine-tuning pre-trained CNNs using extensive, labeled datasets from unrelated applications provides a pragmatic alternative to starting from scratch. These pre-trained models have showcased success as transfer learning baselines across many computer vision tasks.

This study aims to compare the performance of pre-trained models ResNet-50 and VGG-16 for medical image segmentation, with a specific focus on chest X-ray images of pneumonia sourced from the Kaggle repository. Through experimental evaluation and comparing accuracy for number of epochs, the efficacy of VGG-16 and ResNet-50 in accurately segmenting chest X-ray images for pneumonia diagnosis is assessed, offering insights into their suitability for clinical applications.

II. Related work

A. Deep Residual Learning for Image Recognition[1]

The paper "Deep Residual Learning for Image Recognition" introduces a novel approach to deep learning called residual learning, which addresses the challenge of training very deep neural networks by introducing shortcut connections that skip one or more layers. This allows the network to learn residual mappings, making it easier to optimize and train extremely deep architectures. The proposed ResNet architecture achieves state-of-the-art performance on various image recognition tasks, including ImageNet, CIFAR-10, and COCO, while enabling the training of networks with over a hundred layers. Moreover, they compared ResNet with plain architectures and other established models like VGG, demonstrating superior performance in terms of accuracy and training efficiency.

Drawing from the insights gained from this research, I made an informed decision to leverage the ResNet-50 architecture and compare it with VGG-16 for my medical image segmentation project focused on pneumonia detection.

B. A Comprehensive Survey on Transfer Learning[2]

The paper "A Comprehensive Survey on Transfer Learning" authored by Fuzhen Zhuang et al. provides a thorough investigation into transfer learning, a vital technique in machine learning where knowledge gained from one domain or task is utilized to enhance learning and performance in a different but related domain or task. The authors conduct an extensive survey covering various aspects of transfer learning, including its fundamental concepts, methodologies, applications, and recent advancements. It also explores the diverse applications of transfer learning across domains like natural language processing, computer vision, healthcare, and social media analysis.

In the project, transfer learning has been effectively employed to train the model for medical segmentation tasks. Both the ResNet-50 and VGG16 architectures possess a remarkable ability to extract intricate hierarchical features from images, honed through their training on extensive image datasets. By leveraging transfer learning, these pre-trained models serve as powerful feature extractors, adept at capturing relevant patterns indicative of pneumonia from chest X-ray images.

C. Brain Tumour Analysis Using Deep Learning and VGG-16 Ensebmling learning approaches [3]

The paper "Brain Tumor Analysis Using Deep Learning and VGG-16 Ensembling Learning Approaches" explores the application of deep learning techniques, specifically using the VGG-16 architecture, for the analysis of brain tumor images. The study investigates ensembling learning approaches to enhance the performance of brain tumor detection and classification tasks.

By leveraging the insights and methodologies presented in the research paper, I sought to adapt the VGG-16 architecture for analyzing medical images in my own project. The successful application of VGG-16 in brain tumor analysis highlighted its potential for accurately detecting and classifying abnormalities in medical image.

III. METHOD

A. Dataset Description

The dataset is organized into 2 folders (train, test) 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.

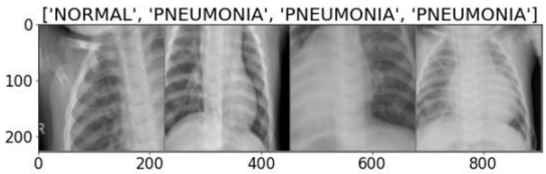


Fig. 1. Sample Chest X-ray images from the dataset.

The dataset is highly imbalance with more of pneumonia cases versus normal cases, hence data augmentation was used to balance the dataset, thereby eliminated the possibility of overfitting the model. The idea is to alter the training data with small transformations to reproduce the variations. Approaches that alter the training data in ways that change the array representation while keeping the label the same are known as data augmentation techniques. Some popular augmentations people use are grayscales, horizontal flips, vertical flips, random crops, color jitters, translations, rotations. I have utilized random horizontal flip and random crop in the project to introduce data augmentation.

B. ResNet-50

ResNet50 is a deep convolutional neural network architecture introduced by Microsoft Research in 2015. It consists of 50 layers and features an innovative use of residual blocks, which allows the network to skip connections and mitigate the vanishing gradient problem during training. This design facilitates the training of very deep networks, enabling better performance in various computer vision tasks. It achieves 92.2% top-5 classification accuracy on the ImageNet benchmark dataset.

For ResNet-50, I began by loading the model using the torchvision library. Following this, I proceeded to freeze the layers of the ResNet model to prevent them from being trained during subsequent training phases. Next, I appended a fully connected layer to the model with an output of 2, corresponding to the number of classes in our dataset (normal and pneumonia).Subsequently, I conducted data transformations, resizing the images to 224x224 pixels and incorporating data augmentation techniques such as random horizontal flipping and random cropping. Additionally, I normalized the images to ensure consistency in the input data. After loading the dataset, I set the hyperparameters, including a learning rate of 0.0001 and momentum of 0.9. For the loss criterion, I opted for the cross-entropy loss function. I specified the number of epochs as 30 to iterate over the dataset multiple times during training.

During each epoch, I executed both forward pass and backward propagation through the network using the optimizer and loss criterion defined earlier. Throughout this process, I tracked and stored the accuracy and loss values for further analysis.

Finally, saved the model dictionary after training, which can subsequently be utilized for testing purposes or further fine-tuning. This comprehensive approach ensures robust training of the ResNet-50 model for pneumonia detection, leveraging its deep learning capabilities to achieve accurate results.

C. VGG-16

VGG16, developed by the Visual Geometry Group at the University of Oxford, is a deep convolutional neural network known for its simplicity and uniformity. Consisting of 16 layers, 13 convolutional layers with 3x3 filters, and 3 fully connected layers, it demonstrated remarkable performance in the 2014 ImageNet Large-Scale Visual Recognition Challenge (ILSVRC), achieving 92.7% classification accuracy. It is able to capture fine-grained details due to small filter sizes. Although computationally intensive, VGG16 set the stage for deeper networks and influenced subsequent models.

The code workflow was very similar to ResNet-50,I began by loading the model using the torchvision library. Following this, I proceeded to freeze the layers of the model to prevent them from being trained during subsequent training phases. The fully connected layers of the VGG-16 model are replaced with a custom classifier to adapt the model for pneumonia detection. Subsequently, I conducted data transformations, resizing the images to 224x224 pixels and incorporating data augmentation techniques such as random horizontal flipping and random cropping. Additionally, I normalized the images to ensure consistency in the input data. After loading the dataset, I set the hyperparameters, including a learning rate of 0.001 and momentum of 0.9. For the loss criterion, I opted for the cross-entropy loss function. I specified the number of epochs as 5 to iterate over the dataset multiple times during training.

During each epoch, I executed both forward pass and backward propagation through the network using the optimizer and loss criterion defined earlier. Throughout this process, I tracked and stored the accuracy and loss values for further analysis.

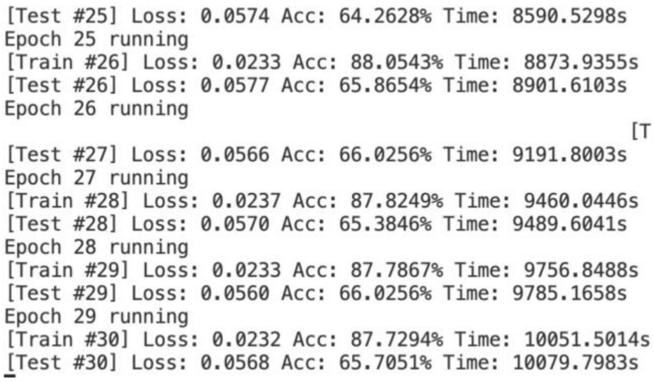
Finally, saved the model dictionary after training, which can subsequently be utilized for testing purposes or further fine-tuning. This comprehensive approach ensures robust training of the VGG-16 model for pneumonia detection, leveraging its deep learning capabilities to achieve accurate results.

IV. EXPERIMENT AND RESULTS

The results were based on performing transfer learning usingVGG-19 and ResNet-50, pretrained models to discriminate pneumonia images from normal chest X-ray images taken from the anterior to posterior (AP/PA) at high resolution. The choice of ResNet-50 over ResNet-101 was to compensate for limited resources at our disposal. As a standard technique, the image was reduced into smaller sizes which is then passed into the convolutional neural network for classification.

A. *ResNet-50*

In terms of training the model, due to limitation in compute power, I ran the model on my local system only and to run 30 epoch it took 10080 sec which is huge to train a pretrained model and only learn the last fully connected layer.



```
[Test #25] Loss: 0.0574 Acc: 64.2628% Time: 8590.5298s
Epoch 25 running
[Train #26] Loss: 0.0233 Acc: 88.0543% Time: 8873.9355s
[Test #26] Loss: 0.0577 Acc: 65.8654% Time: 8901.6103s
Epoch 26 running
                                                       [T
[Test #27] Loss: 0.0566 Acc: 66.0256% Time: 9191.8003s
Epoch 27 running
[Train #28] Loss: 0.0237 Acc: 87.8249% Time: 9460.0446s
[Test #28] Loss: 0.0570 Acc: 65.3846% Time: 9489.6041s
Epoch 28 running
[Train #29] Loss: 0.0233 Acc: 87.7867% Time: 9756.8488s
[Test #29] Loss: 0.0560 Acc: 66.0256% Time: 9785.1658s
Epoch 29 running
[Train #30] Loss: 0.0232 Acc: 87.7294% Time: 10051.5014s
[Test #30] Loss: 0.0568 Acc: 65.7051% Time: 10079.7983s
```

Fig. 2. RunTime and Accuracy for last few Epochs ResNet-50 model

In terms of accuracy, the test accuracy of our model was slightly lower when compared to train accuracy which can be because of the complexity of model. The dataset was comparably smaller and model was complex which led to overfitting. So in future we can try by reducing the complexity of the model or try to train the model on more images, images can be created using data augmentation.

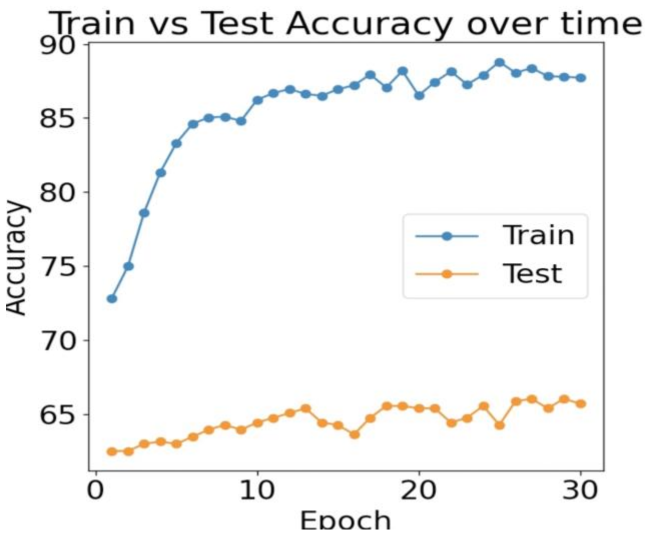
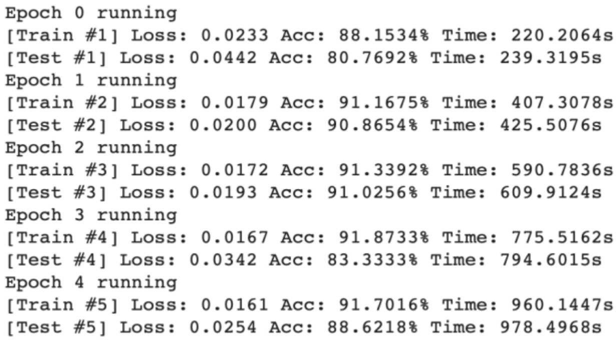


Fig. 3. Train vs Test Accuracy over number of Epoch for ResNet-50 model

B. *VGG-16*

In terms of training the model, I used Google Collab GPU as for ResNet-50 it took lot of time to run. While running on GPU it took 978 seconds to run 5 epoch.



```
Epoch 0 running
[Train #1] Loss: 0.0233 Acc: 88.1534% Time: 220.2064s
[Test #1] Loss: 0.0442 Acc: 80.7692% Time: 239.3195s
Epoch 1 running
[Train #2] Loss: 0.0179 Acc: 91.1675% Time: 407.3078s
[Test #2] Loss: 0.0200 Acc: 90.8654% Time: 425.5076s
Epoch 2 running
[Train #3] Loss: 0.0172 Acc: 91.3392% Time: 590.7836s
[Test #3] Loss: 0.0193 Acc: 91.0256% Time: 609.9124s
Epoch 3 running
[Train #4] Loss: 0.0167 Acc: 91.8733% Time: 775.5162s
[Test #4] Loss: 0.0342 Acc: 83.3333% Time: 794.6015s
Epoch 4 running
[Train #5] Loss: 0.0161 Acc: 91.7016% Time: 960.1447s
[Test #5] Loss: 0.0254 Acc: 88.6218% Time: 978.4968s
```

Fig. 4. Runtime and Accuracy for 5 Epochs for VGG-16 model

In terms of accuracy, for both train and test accuracy were much better if we compare it with ResNet-50 model. This can occur because of complexity of ResNet-50 is more as compared to VGG-16 and with limited and imbalanced dataset VGG-16 performed better.

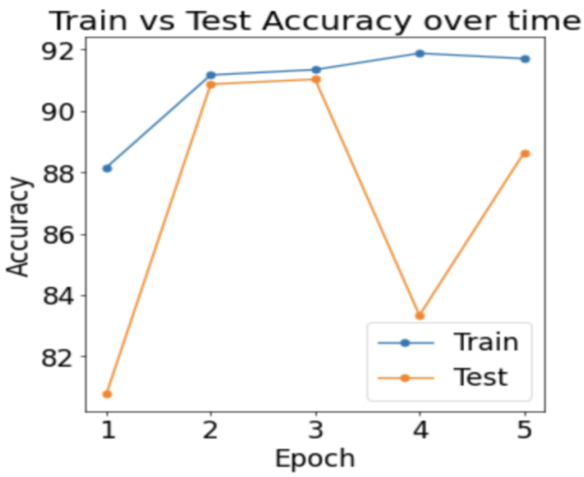


Fig. 5. Train vs Test Accuracy over number of Epochs for VGG-16 model

V. DISCUSSION AND SUMMARY

It is evident from the results obtained with both ResNet-50 and VGG-16 that both models achieved satisfactory accuracy during training. However, VGG-16 exhibited superior performance on the test dataset compared to ResNet-50. This discrepancy could be attributed to ResNet-50 higher model complexity, which may lead to overfitting by memorizing the training data rather than truly learning from it. Additionally, the longer training time required for both models may be attributed to computational constraints, indicating a need for more efficient processing resource.

For future endeavors, expanding the dataset size could be beneficial, as it may enhance the models' generalization capabilities and reduce overfitting. Furthermore, exploring the development of custom convolutional neural network (CNN) architectures rather than relying solely on pre-trained models could offer valuable insights and potentially improve performance. By designing CNN architectures tailored specifically to the task of pneumonia detection, we can potentially achieve better results and gain a deeper understanding of the underlying data characteristics.

REFERENCES

[1] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770-778

[2] F. Zhuang et al., "A Comprehensive Survey on Transfer Learning," in Proceedings of the IEEE, vol. 109, no. 1, pp. 43-76, Jan. 2021, doi: 10.1109/JPROC.2020.3004555.

[3] Younis A, Qiang L, Nyatega CO, Adamu MJ, Kawuwa HB. Brain Tumor Analysis Using Deep Learning and VGG-16 Ensembling Learning Approaches. Applied Sciences. 2022; 12(14):7282. https://doi.org/10.3390/app1214728