Improvised VGG16 CNN Architecture for Predicting Tuberculosis Using the Frontal Chest X-Ray Images

Student Name: Rushi Parmar
Student ID: 225817210

Student Name: Ayush Patel
Student ID: 225817200

Abstract

Although Lungs are mainly impacted by the infectious bacterial disease named Tuberculosis, other parts of the body like brain, spine, kidney could also be attacked by Tuberculosis bacterias. TB could damage any of the above-listed body parts and it could also lead to death if it is not detected early. In contrast to other infectious diseases, the diagnosis of tuberculosis is highly doughty as a number of checks are usually required while chest X-rays are mostly used to detect TB. This project implements the research paper[16] in which a deep learning-based method is proposed to detect TB from a chest X-ray image. For the project, an improvised VGG16 machine-learning algorithm is used to detect Tuberculosis. The model is trained and tested on a publicly available dataset of Chest X-ray images from Shenzhen Hospital [7] in which 87% Accuracy was obtained which is close to the accuracy proposed in the research paper.

1. Introduction

Malignant and benign lung nodules are tiny lumps of tissue that may or may not be cancerous. For medical imaging, feature extraction, and object categorization, deep learning has proven to be a reliable method. Various research has proposed and examined a wide range of specialized architectures, mainly the Convolutional Neural Network (CNN) and its variations. South Africa, which has the third-highest rate of new TB cases worldwide, is currently experiencing an explosive TB epidemic. This high rate of tuberculosis infection is caused by a number of factors, including exposure to silica dirt, the origin of the miners from rural areas with a high TB prevalence, crowding, and HIV. [11]. As per WHO, TB is number two after HIV/AIDS as the pinnacle murdering disease. Many South African mining corporations generally screen their miners for TB by the use of chest x-rays as they are fast to perform, non-invasive, and surprisingly inexpensive. However, this leads to a situation where a large extent of x-rays continuously needs to be analyzed. With the help of numerous researchers, a wide variety of deep learning architectures are introduced to categorize TB. The VGG16 network proposes a categorization scheme for benign and malignant nodules in this project.

2. Related Work

Due to the presence of many chest x-ray manifestations, including opacity, cavities, nodules, localized lesions, etc., detecting tuberculosis is a very difficult task [13]. Recent research has retrieved hand-crafted features and continued the classification process to categorize chest x-rays. The most crucial aspect of feature extraction is identifying patterns of a particular kind. Deep learning-based techniques have gained popularity recently, as some of them are illustrated below. First, Hwang et al. [4] suggested a deep learning method for tuberculosis detection that makes use of the Alexnet network and transfer learning. They evaluated the model using the Montgomery and Shenzhen datasets and trained the model using a new dataset in this, achieving ACU scores of 0.88 and 0.93.

Using the graph cut method, Jaeger et al. [6] segmented chest x-rays to separate the lung module from the x-ray pictures. They have classified the chest x-rays using a variety of categorization approaches, and the results are encouraging. The deep learning model for the categorization of tuberculosis diagnosis from a chest x-ray was developed by Cao et al. [1]. They suggested a mobile solution for the detection of tuberculosis from a chest x-ray in which the chest x-ray will upload and the output generated, using a bespoke dataset that comprised several sorts of the manifest. They suggested GoogleNet in this for the classification of chest x-rays. They achieved 62.07% accuracy for multi-class classification and 89.6% accuracy for binary classification.

A comparison of the benefits and drawbacks of various Deep Learning approaches:

1. CNN (Convolutional neural networks) [1]

CNN extracts spatial information from an image. Pixel placement and their connections within an image are referred to as spatial points. They help us recognize objects accurately, as well as their location and relationships to other objects in an image.

However, The location and orientation of the object are not encoded by CNN. It indicates that in order to fully comprehend and categorize an image, it requires different rotations and diverse angles.

2. DNN(Deep neural networks) [7]

Due to the network's input layer's versatility, DNNs may readily add question characteristics and item facets, which can help identify a user's individual interests and increase the relevancy of suggestions. But DNN requires larger data set to be train-tested more extensively.

3. FFNN (Feed Forward Neural Network) or ANN (Artificial Neural Network) [10]

Any nonlinear function can be successfully learned by an artificial neural network. As a result, these networks are known as Universal Function Approximators. ANNs are capable of researching weights that map any input to any output. The activation function is one of the primary drivers behind extensive approximation.

An image's spatial details are lost when using ANN. The arrangement of the pixels in an image is referred to as spatial elements. ANN is unable to capture the consecutive records needed to handle sequence data in the entered records.

4. DBN(Deep belief networks) [10] [12]

Just a small labeled dataset is required. On machines with GPU power, training times are comparatively short. Very precise as opposed to a shallow net. It is a remedy for the vanishing gradient difficulty. The weakness of DBN is that errors in the initial layers can render them ineffective.

3. Methodology

Using X-ray pictures of the lungs that can be used by laboratories for faster detection of TB with the use of machine learning and deep learning, we are attempting to predict whether the patient has TB using transfer learning. The use of computer vision for TB diagnosis can minimize the need for human involvement and improves accuracy. For our model, we are employing the VGG-16 technique, which is also known as "Very Deep Convolutional Networks for Large-Scale Image Recognition" [15]. First, we have the VGG-16 base model with "imagenet" weights and a 299x299x3 input shape for the images. The fundamental VGG-16 model is composed of a number of layers. Following the 299x299x3 picture input, the first and second convolutional layers with 64 feature maps of size 33 are applied. The new image size is 299x299x64 [8].

In the maximum pooling layer/sub-sampling layer, VGG16 uses a filter of size 33 and a stride value of two, as illustrated in figure 1. The produced image is transformed to 149x149x64 size. Then, two convolutional layers with 128 feature mappings in stride 1 and size 33 are applied to the output image. Then it moves on to a layer with a maximum pooling capacity of size 33 and stride 2. Similar to the initial pooling layer, this pooling layer has 128 feature maps, therefore the output is reduced to 74x74x128. This vector is then sent through three convolutional layers, each of which uses 256 feature maps and has a stride of 1 and a size of 33. Three convolutional layers are applied to this output, and then a maximum pooling layer is applied after that. The convolutional layers that are mentioned here are stride 1, size 33, and have 512 filters. The final output vector is thus 9x9x512 in size [8]

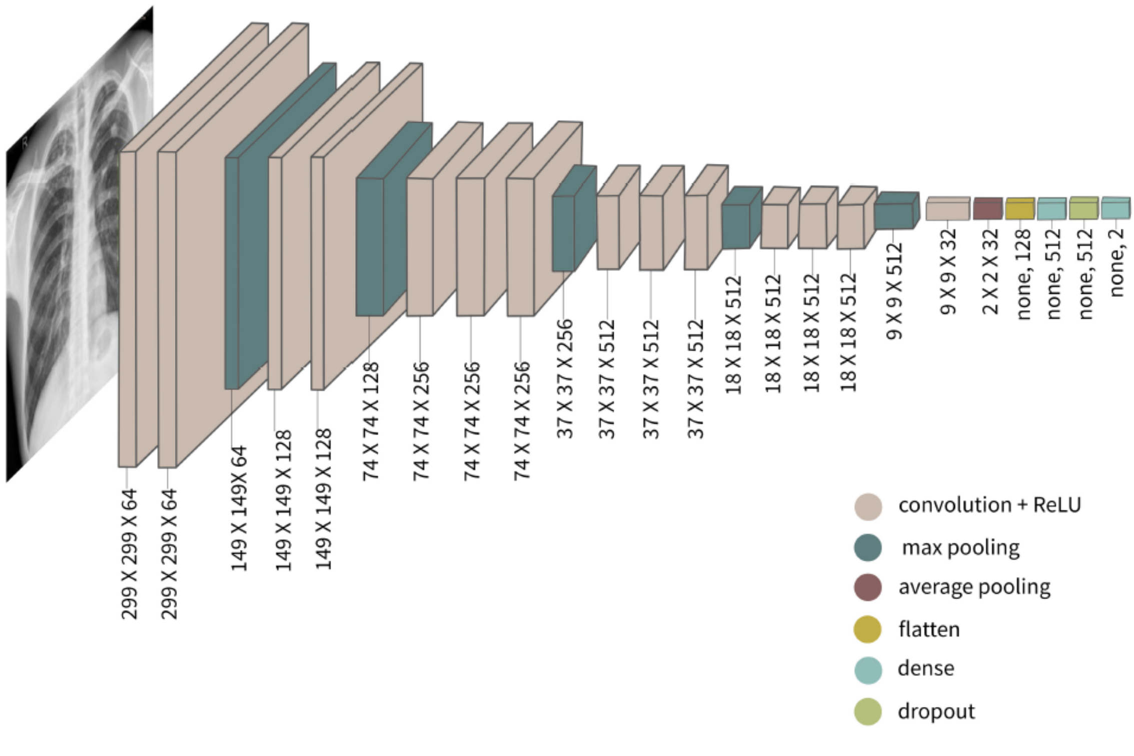

Fig. 1: Improvised VGG16 System Architecture [16]

The output size of the following layer, which is a convolutional layer with the same padding, activation ReLU, kernel size 3, and 32 filters, is 9x9x32. After this layer, the size is reduced to 2x2x32 by an average pooling layer with a pool size of 4x4. A fatten layer with a dimension of 128 and a dense layer with a dimension of 512 follow this. Then, to reduce overfitting, a dropout layer with a dropout value of 0.5 is added. The dense layer with dimension 2 and the activation function softmax makes up the final layer.

For the pre-processing of the chosen dataset, we make use of the Keras package. It makes it reasonable to augment them to numerous random modifications in order to maximize the use of the dataset and ensure that no two input images are identical. It lessens overfitting and aids in increasing accuracy. The Keras ImageDataGenerator class can be used to carry it out. Entropic capacity is one parameter that can be used to gauge how well the model fits the data. The quantity of information that may be stored in the model is determined by its entropic capacity. As seen in figure 2, we sent a vector with a dimension of 299x299x3 into the VGG model, which then applied its output to the convolutional layer. To input the vector into the following layer, we flatten it to make its dimension 1D. The vector is input to all of the neurons in the dense feature. Because it can employ more features, a model (like the one in figure 2) that can retain a lot of data can be more accurate, but it can also lose accuracy by storing unimportant features. Entropic capacity can also be modified via weight regularisation like L1 or L2. Dropout is used to lessen overfitting since it keeps a layer from seeing the same pattern repeatedly. We also employ the flatten feature, which passes a single column of the pool-featured map onto a fully connected layer. The fully connected layer is added to the neural network via dense.

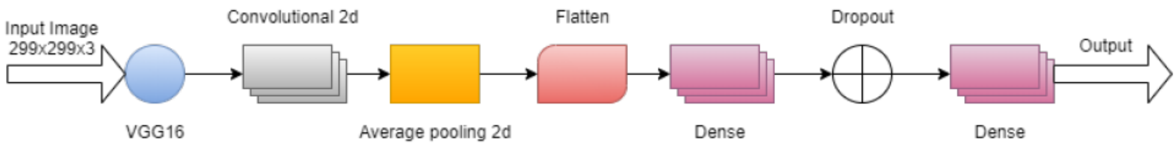

Fig. 2: Flowchart of System Architecture [16]

4. Experiments and Results

4.1 Dataset Selection

In the related research work [16], there are the following 2 publicly available datasets mentioned:

1. Shenzhen Dataset [7]: Chest X-ray images have been collected from Shenzhen No.3 Hospital, Shenzhen, China. The dataset contains 662 chest X-rays images in PNG format and the size of all images is around 3K X 3K pixels. The images are evenly distributed with 326 Normal and 326 Abnormal x-rays.

2. Montgomery Dataset [7]: Montgomery dataset contains the chest x-ray images collected from the Department of Health

and Human Services, Montgomery County, Maryland, USA. The dataset contains 138 chest X-rays images in PNG format and the size of all images is 4892 X 4020 pixels.

In this project, the Shenzhen Dataset is used for the training and testing of the Machine Learning model.

4.2 Data Preprocessing

Data preprocessing [2] is the cleaning up and preparing of the data before building the learning model. We have applied the following data preprocessing techniques

1. Resizing all the images to 299 x 299 pixels.

2. Extract the class from the filename, this helps us create labels for classification.

3. Perform one hot encoding of all the labels to represent categorical variables as binary vectors.

4. We have also done a little bit of data augmentation to generate more data from what we already have.

5. Split the data into 80:20 splits. 80% for training and 20% for testing.

4.3 Evaluation Measures

Our model can be evaluated on the following parameters: accuracy, sensitivity, and specificity. Sensitivity is the proportion of true positives that are correctly predicted by the model, while specificity is the proportion of true negatives that are correctly predicted by the model.

Sensitivity = (True Positive) / (True Positive + False Negative)

Specificity = (True Negative) / (True Negative + False Positive)

4.4 Configuration Setup and Execution

Python programming has been used to carry out the project work. Additionally, the same has been done in the Kaggle notebooks, which offer a system combination with a 2.30GHz Intel Xeon processor and 16GB of NVIDIA Tesla P100 GPU. We kept a learning rate of 1e-4 (0.0001) for the Shenzhen dataset while we trained our model across 100 iterations.

4.5 Result

Models	Project Results	Research Paper Proposed Model	Research Proposed model without dropout	Researched Proposed model without convolutional 2d, dense(2nd last layer) & dropout
Accuracy	0.87	0.91	0.86	0.84
F1 Score (Normal)	0.854	0.90	0.85	0.83
F1 Score (TB)	0.885	0.92	0.88	0.85
Precision (Normal)	0.847	0.86	0.83	0.77
Precision (TB)	0.891	0.96	0.89	0.92
Recall (Normal)	0.862	0.95	0.86	0.91
Recall (TB)	0.88	0.88	0.87	0.79

Table 1: Results

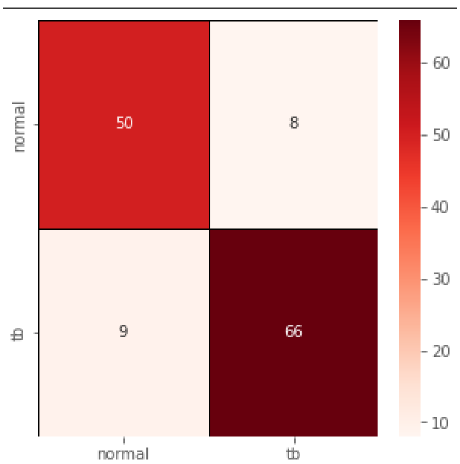

Fig. 3: Confusion Matrix for Shenzhen Dataset

Model	Accuracy	Sensitivity	Specificity
Project Results	0.87	0.88	0.86
Research Paper Model (VGG16+) [16]	0.91	0.95	0.88
VGG16 [5]	0.84	0.96	0.72
VGG19 [5]	0.80	0.76	0.84
Res-Net-50 [5]	0.86	0.84	0.88
GoogleNet [9]	0.82	-	-
Res-Net [9]	0.81	-	-
VggNet [9]	0.82	-	-

Table 2: Comparison of different deep learning based solutions trained on the Shenzhen dataset [7]. A horizontal line means that corresponding results were not provided in the literature.

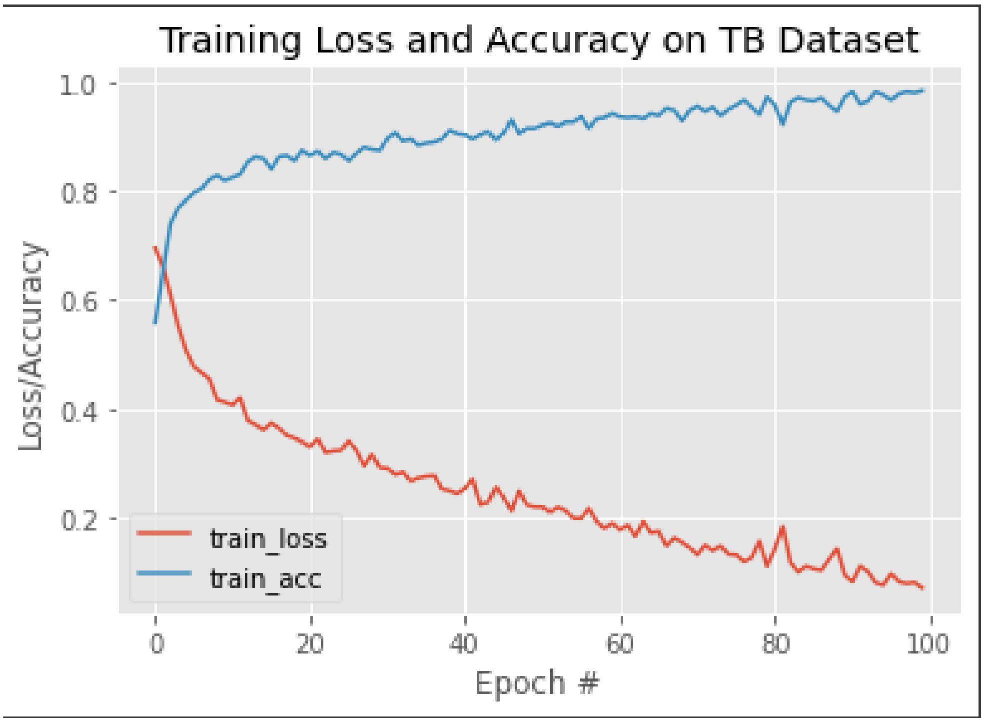

Fig. 4: Loss graph for training on Shenzhen dataset

5. Discussion on Results

Table 1 displays the outcomes we were able to attain using the Shenzhen dataset. We have tested the research proposed model in this project. We experimented with several layer combinations to see how they affected the outcomes that the model produced. While the relevant research paper claims 91% accuracy, our project's findings were 87% accurate, which was close to the stated values in the original research paper. According to the observation, our model performed a better evaluation of these datasets than regular VGG16 and the rest of the models.

Table 2 compares our research on the Shenzhen dataset with that of the other researcher. Among all of the deep learning models suggested by the researcher listed in table 2, our model has produced the best results.

In contrast to the standard VGG16 model, our model employs additional dropout and a dense mechanism to decrease over-fitting and hence increase accuracy. The confusion matrix from the Shenzhen dataset's training data is shown in Figure 3. Figure 4 displays the accuracy and loss for training and validation on the Shenzhen dataset.

Epoch denotes how frequently the training vectors are used to update the neural network's weights. The accuracy vs. epoch graph in figure 4 shows that for the given dataset, our accuracy increases with increasing epoch.

6. Conclusion

This deep learning model's goal is to more accurately detect tuberculosis using medical photos more than other models currently on the market. This model would decrease human involvement and help decrease human error in the identification of TB.

The foundation model for the project is VGG16. The Shenzhen dataset [7] was used in which the model's accuracy is 0.91. Because dropout eliminates irrelevant information, the suggested model performs more accurately than other models. In order to further increase the model's accuracy, previous medical records of the patients can also be used as input. IoT devices can track the patient's breathing patterns over time and add that data into the model, which could furthermore increase accuracy. Our model might aid different healthcare facilities in enhancing the accuracy of their TB detection, which might lead to fewer fatalities and more people receiving treatment.

References

1. Y. Cao, C. Liu, B. Liu, M. J. Brunette, N. Zhang, T. Sun, P. Zhang, J. Peinado, E. S. Garavito, L. L. Garcia, and W. H. Curioso. Improving tuberculosis diagnostics using deep learning and mobile health technologies among resource-poor and marginalized communities. In 2016 IEEE First International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE), pages 274–281, 2016.

2. Hisham El-Amir and Mahmoud Hamdy. Data Wrangling and Preprocessing, pages 147–206. Apress, Berkeley, CA, 2020.

3. Thi Kieu Khanh Ho, Jeonghwan Gwak, Om Prakash, Jong-In Song, and Chang Min Park. Utilizing pretrained deep learning models for automated pulmonary tuberculosis detection using chest radiography. In Ngoc Thanh Nguyen, Ford Lumban Gaol, Tzung-Pei Hong, and Bogdan Trawi´nski, editors, Intelligent Information and Database Systems, pages 395–403, Cham, 2019. Springer International Publishing.

4. Sangheum Hwang, Hyo-Eun Kim, Jihoon Jeong M.D., and Hee-Jin Kim. A novel approach for tuberculosis screening based on deep convolutional neural networks. In Georgia D. Tourassi and Samuel G. Armato III, editors, Medical Imaging 2016: Computer-Aided Diagnosis, volume 9785, pages 750 – 757. International Society for Optics and Photonics, SPIE, 2016.

5. Mohammad Tariqul Islam, Md Abdul Aowal, Ahmed Tahseen Minhaz, and Khalid Ashraf. Abnormality detection and localization in chest x-rays using deep convolutional neural networks, 2017.

6. S. Jaeger, A. Karargyris, S. Antani, and G. Thoma. Detecting tuberculosis in radiographs using combined lung masks. In 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pages 4978–4981, 2012.

7. Stefan Jaeger, Sema Candemir, Sameer Antani, Y`ı-Xi´ang J. W´ang, Pu-Xuan Lu, and George Thoma. Two public chest x-ray datasets for computer-aided screening of pulmonary diseases. Quantitative imaging in medicine and surgery, 4(6):475–477, Dec 2014.

8. G. Labhane, R. Pansare, S. Maheshwari, R. Tiwari, and A. Shukla. Detection of pediatric pneumonia from

chest x-ray images using cnn and transfer learning. In 2020 3rd International Conference on Emerging Technologies in Computer Engineering: Machine Learning and Internet of Things (ICETCE), pages 85–92, 2020.

9. U.K. Lopes and J.F. Valiati. Pre-trained convolutional neural networks as feature extractors for tuberculosis detection. Computers in Biology and Medicine, 89:135 – 143, 2017.

10. Zhihan Lv and Liang Qiao. Deep belief network and linear perceptron based cognitive computing for collaborative robots. Applied Soft Computing, 92:106300, 2020.

11. Juergen Noeske and Petrus Nkamsse Nguenko. Impact of resistance to anti-tuberculosis drugs

on treatment outcome using world health organization standard regimens. Transactions of the

Royal Society of Tropical Medicine and Hygiene, 96(4):429 – 433, 2002.

12. Yara Rizk, Nadine Hajj, Nicholas Mitri, and Mariette Awad. Deep belief networks and cortical

algorithms: A comparative study for supervised classification. Applied Computing and

Informatics, 15(2):81 – 93, 2019.

13. Megan Jill Russell, Andre Nel, and Tshilidzi Marwala. Arma analysis of chest x-rays for

computer assisted detection of tuberculosis. In Mian Long, editor, World Congress on Medical

Physics and Biomedical Engineering May 26-31, 2012, Beijing, China, pages 896–899,

Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

14. Seelwan Sathitratanacheewin, Panasun Sunanta, and Krit Pongpirul. Deep learning for automated

classification of tuberculosis-related chest x-ray: dataset distribution shift limits diagnostic

performance generalizability. Heliyon, 6(8):e04614, 2020.

15. Srikanth Tammina. Transfer learning using vgg-16 with deep convolutional neural network for classifying images. International Journal of Scientific and Research Publications (IJSRP), 9:p9420, 10 2019.

16. Patel, S.B., Patel, P.H., Jain, V.D., Verma, J.P. (2022). Improvised VGG16 CNN Architecture for Predicting Tuberculosis Using the Frontal Chest X-Ray Images. In: Somani, A.K., Mundra, A., Doss, R., Bhattacharya, S. (eds) Smart Systems: Innovations in Computing. Smart Innovation, Systems and Technologies, vol 235. Springer, Singapore. https://doi.org/10.1007/978-981-16-2877-1_7