**Mini Project Report on**



**MACHINE LEARNING MODEL FOR DISEASE PREDICTION**  


**Submitted in partial fulfilment of the requirement for the award of the degree of**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE & ENGINEERING**

**Submitted by:**

**Student Name: Shubhangi Gupta**  **University Roll No. 2017055**

***Under the Mentorship of***

**Dr. Sharon Christa**

**Associate Professor, Dept of Computer Science & Engineering**

Logo

Description automatically generated

**Department of Computer Science and Engineering**

**Graphic Era (Deemed to be University)**

**Dehradun, Uttarakhand**

**January 2023**

A picture containing text

Description automatically generated

**DECLARATION**

I hereby certify that the work which is being presented in the project report entitled **“Machine Learning Model For Disease Prediction”** in partial fulfilment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science and Engineeringof the Graphic Era (Deemed to be University), Dehradun shall be carried out by the under the mentorship of **Dr. Sharon Christa, Associate professor**, Department of Computer Science and Engineering, Graphic Era (Deemed to be University), Dehradun.

**Name: Shubhangi Gupta University roll no. 2017055**

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Description** | **Page No.** |
| Chapter 1 | Introduction | 4 |
| Chapter 2 | Literature Survey | 5 |
| Chapter 3 | Methodology | 6 |
| Chapter 4 | Result and Discussion | 12 |
| Chapter 5 | Conclusion and Future Work | 14 |
|  | References | 15 |

**Chapter 1**

**Introduction**

One of the major factors associated with pneumonia in children is indoor air pollution. Apart from this, under-nutrition, lack of safe water, sanitation and basic health facilities are also major factors.

Pneumonia is an interstitial lung disease caused by bacteria, fungi, or viruses. It accounted for approximately 16% of the 5.6 million under-five deaths, killing around 880,000 children in 2016 [1]. Affected victims were

mostly less than two years old. Timely detection of pneumonia can help to prevent the deaths of children.

This project presents convolutional neural network models to accurately detect pneumonic lungs from chest X-rays, which can be utilized in the

real world by medical practitioners to treat pneumonia [2].

Classification model were built using CNN to detect pneumonia from chest X-ray images to help control this deadly infection in children and other age groups. Accuracy of the model is directly correlated with the size of the dataset, that is, the use of large datasets helps improve the accuracy of the model, but there is no direct correlation between the number of convolutional layers and the accuracy of the model.

To obtain the best results, a certain number of combinations of convolution layers, dense layers, dropouts and learning rates must be trained by evaluating the models after each execution. The objective of the project is to develop CNN model from scratch which can classify and thus detect pneumonic patients from their chest Xray with high validation accuracy.

**Chapter 2**

**Literature Survey**

Many researchers have tackled the problem of classifying images with high accuracy.

Here are some citations related to our paper:

Rubin et al. [3] developed a CNN model to detect common thorax disease from frontal and lateral chest X-ray images. MIMIC-CXR dataset was used to perform large-scale automated recognition of these images. The dataset was split into training, testing and validation sets as 70%, 20% and 10%, respectively. Data augmentation and pixel normalization were used to improve overall performance. Their DualNet CNN model achieved an average AUC of 0.72 and 0.688 for PA and AP, respectively.

A deep convolutional neural network to classify pulmonary tuberculosis was developed by Lakhani et al. [4]. Transfer learning models such as AlexNet and GoogleNet were also used to classify chest X-ray images. The dataset was split into training, testing and validation sets as 68%, 14.9% and 17.1%, respectively. Data augmentation and pre-processing techniques were employed to get the best performing model achieving an AUC of 0.99. Precision and recall of the model were 100 and 97.3%.

An AG-CNN model was developed by Guan et al. [5] to recognize thorax disease. ChestX-ray14 dataset was used to detect thorax disease from chest X-ray images. Global and local branch attention-guided CNN was used for classification purposes. Their model was better than other models, achieving an AUC of 0.868.

A deep convolutional neural network model was developed by Rajpurkar et al. [6] to classify chest X-ray images into pneumonia and other 14 diseases. ChestX-ray14 dataset was used for training the model. They compared their ChXNet model (121 layered model) with practicing

academic radiologists. Their ChXNet model achieved an F1 score (95% CI) of 0.435

outperforming radiologists which achieved an F1 score (95% CI) of 0.387.

A deep convolutional neural network model having five convolutional layers some followed by max-pooling layers, having three fully connected layers was trained by Krizhevsky et al. [7]. This network contained 60 million different parameters. By employing dropout, this model achieved a top-five error percent of 17%.

**Chapter 3**

**Methodology**

CNN model has been created from scratch and trained on Chest X-Ray Images

(Pneumonia) dataset on Kaggle.

Using some libraries like Matplotlib and Seaborn for data visualization, Keras neural network library with TensorFlow backend has been used to implement the models, Sklearn for calculation (evaluation metrics), CV2 for data processing, OS for accessing the data as data directories, NumPy and Pandas for processing and Gradio for user interface.

Data pre-processing has been applied for getting training, validation and testing dataset and performing simple grayscale normalization to reduce the effect of illumination differences and then resizing the data for CNN training and testing.

Data augmentation has been applied to achieve better results from the dataset. The models have been trained on the training dataset, each with different number of convolutional layers ( Normalization layer, Pooling Layer, Dense Layer).

Performing evaluation metrices using the confusion matrix and checking the accuracy and precision.

The following sub-headings further explain the above stages in depth.

**3.1 CNN Architecture**

CNN models are feed-forward networks with convolutional layers, pooling layers,

flattening layers and fully connected layers employing suitable activation functions.

**Convolutional layer**. It is the building block of the CNNs. Convolution operation is done in mathematics to merge two functions. In the CNN models, the input image is first converted into matrix form. Convolution filter is applied to the input matrix which slides over it, performing element-wise multiplication and storing the sum. This creates a feature map. 3 × 3 filter is generally employed to create 2D feature maps when images are black and white. Convolutions are performed in 3D when the input image is represented as a 3D matrix where the RGB colour represents the third dimension. Several feature detectors are operated with the input matrix to generate a layer of feature maps which thus forms the convolutional layer.

**Activation functions**. Model presented in this project use two different activation functions, namely ReLU activation function and SoftMax activation function. The ReLU activation function stands for rectified linear function. It is a nonlinear function that outputs zero when the input is negative and outputs one when the input is positive. The ReLU function is given by the following formula:

F(x) = max (0, x) ………………………. (3.1)

This type of activation function is broadly used in CNNs as it deals with the

problem of vanishing gradients and is useful for increasing the nonlinearity of layers.

ReLU activation function has many variants such as Noisy ReLUs, Leaky ReLUs

and Parametric ReLUs. Advantages of ReLU over other activation functions are

computational simplicity and representational sparsity. SoftMax activation function

is used in the model presented in this project. This broadly used activation

function is employed in the last dense layer of the model. This activation

function normalizes inputs into a probability distribution. Categorical cross-entropy

cost function is mostly used with this type of activation function.

**Pooling layer**. Convolutional layers are followed by pooling layers. The type of

pooling layer used in all models is max-pooling layers. The max-pooling layer

having a dimension of 2 × 2 selects the maximum pixel intensity values from the

window of the image currently covered by the kernel. Max-pooling is used to down

sample images, hence reducing the dimensionality and complexity of the image.

Two other types of pooling layers can also be used which are general pooling and

overlapping pooling. The model presented in this project use max-pooling technique

as it helps recognize salient features in the image.

**Flattening layer and fully connected layers**. After the input image passes

through the convolutional layer and the pooling layer, it is fed into the flattening

layer. This layer flattens out the input image into a column, further reducing its

computational complexity. This is then fed into the fully connected layer/dense layer.

The fully connected layer has multiple layers, and every node in the first layer

is connected to every node in the second layer. Each layer in the fully connected

layer extracts features, and on this basis, the network makes a prediction.

This process is known as forward propagation. After forward propagation, a cost

function is calculated. It is a measure of performance of a neural network model.

The cost function used in model is categorical cross-entropy. After the cost function is calculated, back propagation takes place. This process is repeated until the network achieves optimum performance.

**Reducing overfitting**. The first model exhibits substantial overfitting; hence,

dropout technique was employed in the later model. Dropout technique helps to

reduce overfitting and tackles the problem of vanishing gradients. Dropout technique

encourages each neuron to form its own individual representation of the input data.

This technique on a random basis cuts connections between neurons in successive

layers during the training process. Learning rate of model was also modified,

to reduce overfitting. Data augmentation technique can also be employed to reduce

overfitting.

**Algorithm of CNN classifiers**. The algorithms used in the convolutional neural

network classifiers have been explained in Figs. 1 and 2. Figure 3 shows the flowchart

of the overall schema of project. The number of epochs for all the classifier models

presented in this project was fixed at 10 after training and testing several CNN models

over the course of project. Classifier models are trained for a greater number of epochs

have showed overfitting. Several optimizer functions were also trained and studied.

Adam optimizer function was finalized to be used for all classifiers after it gave the

best results. Initially, a simple classifier model with convolutional layer of image

size set to 64 \* 64, 32 feature maps and employing ReLU activation function was

trained. Fully connected dense layer with 128 perceptron’s was utilized. To improve

the result, the second classifier model was trained with one more convolutional

layer of 64 feature maps for better feature extraction. The number of perceptron’s

in dense layer was also doubled to 256, so that better learning could be achieved.

The third model was trained for three convolutional layers with 128 feature maps in

third convolutional layer for more detailed feature extraction. Dense layer was kept

unchanged. Dropout layer was introduced at 0.3 and learning rate of optimizer was lowered to 0.0001 to reduce the overfitting. The final fourth classifier model was

trained for four convolutional layers with 256 feature maps in fourth convolutional

layer. Dense layer, dropout layer and learning rate were kept same as third classifier

model. The results have been summarized in the subsequent section of this paper.



**Fig. 1** Algorithms of CNN classifier model 1 (left) and model 2 (right)



**Fig. 2** Algorithms of CNN classifier model 3 (left) and model 4 (right)



**Fig. 3** Detailed schema of the experiment conducted

**Dataset**. Chest X-Ray Images (Pneumonia) dataset of 1.24 GB size has been

imported from Kaggle, with total of 5856 jpeg images split into Train, Test and

Val folders each divided into category Pneumonia and Normal. Chest X-ray images

(Front and back) were selected from paediatric patients of one- to five-year olds from

Guangzhou Women and Children’s Medical Centre, Guangzhou. Figure 4 provides the sample images from the dataset used during the research.



**Fig. 4** Left image depicts normal lungs and right image depicts pneumonic lungs

**Chapter 4**

**Result and Discussion**

To study the performance of each CNN classifier model, validation accuracy and loss, recall, F1-score, and Support were evaluated as the performance measures. Accuracy and loss graphs were also studied. The confusion matrix was also computed for the

model.

**4.1 Comparison of Performance of Model**

Below Figures show the confusion matrices, accuracy, and loss CNN classifier model. The Accuracy, recall, F1-scores, and Support are also showed. It achieved the least overfitting along with highest accuracy and recall. Several attempts were made to better the performance by adding more convolutional layers and changing the parameters.

In the following equations, tp = true positive, tn = true negative, fp = false positive and

fn = false negative.

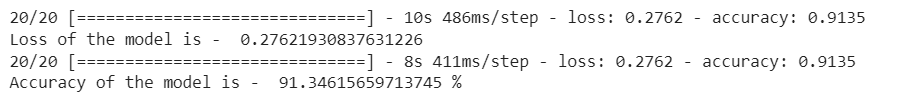
Accuracy = (tp + tn)/ (tp + tn + fp + fn) …………………………………. (4.1)

Precision = tp/ (tp + fp) …………………………………………………... (4.2)

Recall = tp/ (tp + fn) ……………………………………………………… (4.3)

F1 Score = 2(Precision ∗ Recall)/ (Precision + Recall) …………………... (4.5)

Support = freq (0,1)/ n ……………………………………………………. (4.6)



**Fig. 5** Loss and Accuracy of the model

Table

Description automatically generated

**Fig. 6** Precision, Recall, F1-Scrore and Support of the model

Table

Description automatically generated Graphical user interface

Description automatically generated with medium confidence

**Fig. 7** Result in the form of Confusion Matrix

Chart, treemap chart

Description automatically generated

**Fig. 8** Graph of the model

**Chapter 5**

**Conclusion and Future Work**

The validation accuracy, recall and F1 score of CNN classifier model with three

convolutional layers are 91%, 91% and 91%, respectively, which are quite high

compared to other models that were trained.

The paper by Chakraborty [8] achieved the overall accuracy of 95.62%

and recall of 95% trained on the same dataset. The paper by Liang [9] achieved

recall of 96.7% on the same dataset. The models presented by us at best could

achieve 92.31% accuracy which is lower, but 98% recall has been achieved.

High recall values will ensure that the number of false-negative instances is lower, hence

lowers the risk to the patient’s life.

Thus, it is concluded that CNN classifier model can, therefore, be effectively used by medical officers for diagnostic purposes for early detection of pneumonia in children as well as adults. Many X-ray images can be processed very quickly to provide highly precise diagnostic results, thus helping healthcare systems provide efficient patient care services and reduce mortality rates. This convolutional neural networks’ model was successfully achieved by employing various methods of parameter tuning like adding dropout, changing learning rates, changing the batch size, number of epochs, adding more complex fully connected layers and changing various stochastic gradient optimizers.

In the future, it is hoped that transfer learning models would be trained on this dataset that would outperform this CNN model. It is intended that larger datasets will also be trained using the model presented in the project. It is also expected that neural network models based on GAN [10], generative adversarial networks, would also be trained and compared with the existing models.

**References**

[1] https://data.unicef.org/topic/child-health/pneumonia/. Accessed on 15 July 2019

[2] Jaiswal, A.K., Tiwari, P., Kumar, S., Gupta, D., Khanna, A., Rodrigues, J.J.: Identifying

pneumonia in chest x-rays: a deep learning approach. Measurement **145**, 511–518 (2019)

[3] Rubin, J., Sanghavi, D., Zhao, C., Lee, K., Qadir, A., Xu-Wilson, M.: Large Scale Automated Reading of Frontal and Lateral Chest X-Rays Using Dual Convolutional Neural Networks

(2018). arXiv preprint arXiv:1804.07839

[4]Lakhani, P., Sundaram, B.: Deep learning at chest radiography: automated classification of

pulmonary tuberculosis by using convolutional neural networks. Radiology **284**(2), 574–582

(2017)

[5] Guan, Q., Huang, Y., Zhong, Z., Zheng, Z., Zheng, L., Yang, Y.: Diagnose Like a Radiologist: Attention Guided Convolutional Neural Network for Thorax Disease Classification (2018).*arXiv preprint* arXiv:1801.09927

[6]Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Langlotz,C., Shpanskaya,K., Lungren,M.P.: Chexnet: Radiologist-Level PneumoniaDetection on ChestX-rays with Deep Learning (2017). arXiv preprint arXiv:1711.05225

[7] Krizhevsky, A., Sutskever, I., Hinton, G.E: Imagenet classification with deep convolutional

neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105(2012)

[8] Chakraborty, S., Aich, S., Sim, J.S., Kim, H.C.:Detection of pneumonia from chest x-rays using a convolutional neural network architecture. In: International Conference on Future Information & Communication Engineering, vol. 11, no. 1, pp. 98–102 (2019)

[9] Liang, G., Zheng, L.: A transfer learning method with deep residual network for paediatric

pneumonia diagnosis. In: Computer Methods and Programs in Biomedicine (2019)

[10] Du, S. S., Zhai, X., Poczos, B., Singh, A.: Gradient Descent Provably Optimizes Over-

Parameterized Neural Networks (2018). arXiv preprint arXiv:1810.02054