**SRI SIDDHARTHA INSTITUTE OF TECHNOLOGY**

**MARALURU, TUMAKURU-572103**

**(A Constituent college of Sri Siddhartha Academy of Higher Education Deemed to be University)**

**DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING**



**Mini Project Report**

**On**

**“Tuberculosis Detection using Python”**

Submitted in partial fulfillment of the requirement for the completion of V semester of

**BACHELOR OF ENGINEERING**

**Submitted by:**

KOUSHIK PAI :20IS044

LIKITH GOWDA M J :20IS050

**Under the guidance of:**

Mrs. Champakamala S

Assistant Professor

Dept. of ISE

**DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING**

**2022-23**

**SRI SIDDHARTHA INSTITUTE OF TECHNOLOGY**

**MARALURU, TUMAKURU-572103**

**(A Constituent college of Sri Siddhartha Academy of Higher Education, Deemed to be University)**

**DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING**



**CERTIFICATE**

Certified that mini project work entitled “**Tuberculosis Detection using Python”,** is a bonafide work carried out by **KOUSHIK PAI (20IS044)** and **LIKITH GOWDA M J (20IS050)**, in partial fulfilment of the requirement for the completion of VI semester in **Information Science and Engineering** of Siddhartha Academy of Higher Education- Tumakuru during the year 2022-23. It is certified that all corrections/suggestions indicated for Internal Assessment has been incorporated in the report. The report has been approved as it satisfies the academic requirements with respect to Mini Project work.

Signature of the guide Signature of the HOD

### Mrs. Champakamala S Dr. Karunakar K

Assistant Professor Professor & Head

Dept. of Information Science and Engg. Dept. of Information Science and Engg.

Sri Siddhartha Institute of Technology Sri Siddhartha Institute of Technology

Tumakuru-572105 Tumakuru-572105

**Name of the Examiners: Signature:**

**1.** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**2.** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**DECLARATION**

We, **KOUSHIK PAI (20IS044)** and **LIKITH GOWDA M J (20IS050)** of Sixth semester, Department of Information Science and Engineering of Sri Siddhartha Institute of Technology, Tumakuru, hereby declare that this Mini project titled, **“Tuberculosis Detection using Python”**, has been carried out by us under the supervision of  **Champakamala S**, Assistant Professor, Department of Information Science and Engineering , Sri Siddhartha Institute of Technology , Tumakuru in partial fulfilment of the requirement for the completion of VI semester in Information Science and Engineering.

Date: 26/12/22

Place: Tumakuru

KOUSHIK PAI (20IS044)

LIKITH GOWDA M J (20IS050)

Abstract

This abstract summarizes an automated tuberculosis (TB) detection program that utilizes artificial intelligence (AI) to improve the accuracy and efficiency of TB diagnosis. The program integrates machine learning algorithms with medical imaging analysis to identify TB-related abnormalities in chest X-ray and CT scans. By providing a fast and reliable screening method, especially in regions with limited healthcare resources, the program aims to assist healthcare professionals in detecting TB and making informed treatment decisions. To facilitate ease of use and accessibility, the program is designed to have a user-friendly interface, enabling healthcare providers to easily upload and analyze medical images.

Upon analysis, the program generates a detailed report highlighting the areas of concern and providing a probability score for the presence of TB. Initial evaluations show promising results, but further validation studies are required. Overall, this program has the potential to revolutionize TB screening and contribute to global efforts in controlling and eradicating the disease. Initial evaluations of the program show promising results, demonstrating high sensitivity and specificity in TB detection.

Table of Contents Pg. No.

ABSTRACT iv

LIST OF FIGURES vi

LIST OF TABLES vi

**Chapter 1: Introduction**

* 1. Background 1
  2. Problem Statements 1

1.3 Aim and Objectives 2

**Chapter 2: Survey**

2.1 Literature Review 3

**Chapter 3: System Requirement and Specification**  4

**Chapter 4: Implementations**

4.1 Source code 5

4.2 System Module 11

**Chapter 5: Result and Discussions**

5.1 Sample Pictures 12

5.2 Tables 16

**Chapter 6: Conclusion & Future Scope**

6.1 Conclusion 17

6.2 Future Scope 17

**References** 18

**LIST OF FIGURES**

**Fig.No Description Pg. No.**

Figure 6.1.1 Home Page 12

Figure 6.1.2 About us page 12

Figure 6.1.3 List of features 13

Figure 6.1.4 Locate the hospitals 13

Figure 6.1.5 Experts Doctors Page 14

Figure 6.1.6 Ambulance Service Page 14

Figure 6.1.7 Contact Information 15

Figure 6.1.8 Locate hospital by entering the Pincodeeq 15

**LIST OF TABLES**

**Table. No Description Pg. No.**

Table 6.2.1 List of Doctors available 16

Table 6.2.2 List of ambulance available 16

**Chapter 1**

**Introduction**

The TB detection program utilizes artificial intelligence (AI) to enhance the accuracy and efficiency of tuberculosis (TB) diagnosis. Traditional methods have limitations, and AI offers automated screening in resource-constrained settings. The program utilizes machine learning algorithms to analyze chest X-rays and CT scans for TB-related abnormalities. It aims to improve diagnosis speed and accessibility.

The program's user-friendly interface allows healthcare providers to upload images and receive detailed reports. Further validation studies are needed to assess its real-world performance and compare it with existing methods. Overall, the program has the potential to revolutionize TB diagnosis and contribute to global TB control efforts.

* 1. **Background:**

TB detection involves the use of artificial intelligence (AI) and machine learning algorithms to analyze medical images, such as chest X-rays and CT scans, for TB-related abnormalities. These AI-based programs aim to improve the accuracy, speed, and accessibility of TB diagnosis, especially in resource-constrained settings. By training the AI models on annotated TB image datasets, they can learn to identify TB-related patterns and provide detailed reports to assist healthcare professionals in making informed decisions. Implementing AI-based TB detection programs holds promise for enhancing healthcare workflows, reducing diagnostic delays, and improving patient outcomes.

**1.2 Problem Statement:**

The problem statement for TB detection is the need for an accurate, efficient, and accessible method to detect tuberculosis. Traditional diagnostic methods have limitations, leading to delayed diagnosis, increased transmission, and poorer outcomes. There is a shortage of skilled radiologists and limited access to advanced imaging technologies in many areas. Leveraging artificial intelligence and machine learning, there is a need to develop automated TB detection programs that can analyze medical images, such as chest X-rays and CT scans, to identify TB-related abnormalities with high accuracy and efficiency. These programs should be easily integrated into healthcare systems and enable timely diagnosis and effective management of TB cases.

* 1. **Aim and Objective:**

**Aim:**

The aim of TB detection is to develop an accurate, efficient, and accessible method for the detection and diagnosis of tuberculosis, enabling early intervention, reducing transmission, and improving patient outcomes. earning program

**Objectives:** through the intervention of planned play activities that

* **Develop an AI-based TB detection program**: Develop a program that utilizes artificial intelligence (AI) and machine learning algorithms to analyze medical images, such as chest X-rays and CT scans, for the detection of TB-related abnormalities.
* **Enhance accuracy and efficiency**: Improve the accuracy and efficiency of TB detection by leveraging advanced image analysis techniques and AI algorithms, minimizing false-positive and false-negative results.
* **Improve accessibility**: Develop a user-friendly interface that allows healthcare providers to easily upload medical images for analysis, making TB detection accessible even in resource-constrained settings.
* **Provide timely and reliable results:** Enable the program to generate detailed reports indicating areas of concern and providing probability scores for the presence of TB, assisting healthcare professionals in making informed decisions.
* **Validate and evaluate the performance:** Conduct rigorous evaluations, validation studies, and comparative analyses to assess the performance of the TB detection program, comparing it with existing diagnostic methods and ensuring its reliability and accuracy.
* **Optimize the program for scalability**: Develop the TB detection program to handle large volumes of medical images efficiently, ensuring scalability and adaptability to various healthcare settings.
* **Integrate with existing healthcare systems**: Ensure compatibility and seamless integration of the TB detection program into existing healthcare systems and workflows, enabling easy adoption and utilization by healthcare providers.
* **Ensure data security and privacy**: Implement appropriate security measures to protect patient data and comply with privacy regulations during the operation and utilization of the TB detection program.

**Chapter 2**

**Survey**

**2.1 Literature survey:**

“Tuberculosis Detection from Chest X-Ray Images using Deep Learning Techniques” (2020): Deep learning-based approach using Python and convolutional neural networks (CNNs) for tuberculosis detection from chest X-ray images.

“Automatic Tuberculosis Screening using Deep Learning and Chest Radiographs” (2021): Python-based deep learning model for automatic tuberculosis screening from chest radiographs.

“Tuberculosis Detection using Ensemble Learning and Chest X-ray Images” (2021): Ensemble learning approach in Python for tuberculosis detection from chest X-ray images.

“Tuberculosis Detection using Convolutional Neural Networks and Mobile Applications” (2020): Mobile application developed in Python using convolutional neural networks for tuberculosis detection.

“Tuberculosis Detection using Machine Learning and Audio Cough Analysis” (2021): Python-based machine learning techniques applied to audio cough analysis for tuberculosis detection.

**Chapter 3**

**System Requirement and Specification**

The system requirements for tuberculosis (TB) detection can vary depending on the specific approach and technology used.

* **Storage:** Adequate storage capacity to store the dataset, models, and any intermediate or output files generated during the TB detection process.
* **Computer or server:** A capable computing device with sufficient processing power and memory to handle the computational demands of the TB detection algorithm.
* **Network Connectivity:** Stable internet connection for uploading and downloading medical images, as well as accessing updates or additional resources.
* **Image Processing Tools**: Software or libraries for image preprocessing and analysis, including handling medical images such as chest X-rays or CT scans.
* **Programming Language:** Depending on the chosen approach, familiarity with programming languages such as Python, R, or MATLAB may be required.
* **Datasets:** An appropriately labeled dataset of TB samples, including both positive (TB-infected) and negative (non-TB) samples, for training and testing the TB detection model.

These are general system requirements, and the specific requirements may vary depending on the chosen approach, dataset, and technology used for TB detection. It is essential to consider the specific requirements mentioned in the literature or documentation of the chosen TB detection method or algorithm.

**Chapter 4**

**System Design**

**4.1 System Architecture**

Conventional Image Processing Pipeline: Preprocess images, extract relevant features, and classify using algorithms like SVM or decision trees.

Deep Learning Convolutional Neural Network (CNN) Architecture: Preprocess images, pass through CNN layers for feature extraction, use fully connected layers for classification.

Hybrid Approaches: Combine multiple imaging modalities or integrate clinical data with imaging data for improved TB detection.

Mobile and Point-of-Care Architectures: Develop lightweight models or mobile applications for on-device TB detection with limited computational resources.

**4.2 Data flow diagrams**

* **Context-Level DFD**: Provides an overview of the system, showing external entities and data flows.
* **Level 0 DFD**: Illustrates major processes and high-level data flows in the system.
* **Level 1 DFD:** Decomposes processes into sub-processes and shows detailed data flows.
* **Detailed DFDs**: Provide a granular representation with lower-level processes, data stores, and data flow annotations.
* **Symbols**: Processes represent functions, data flows show data movement, external entities are entities outside the system, and data stores represent data repositories.
* DFDs help visualize data flow, understand system interactions, and identify potential issues. They aid in communication and system analysis.

**Chapter 5**

**Implementation**

**5.1 System Modules**

The modules are designed to perform specific tasks or provide specific functionalities within the system. The specific modules required in a system can vary depending on its nature, purpose, and requirements.

* **User Interface Module:** Handles interaction between the system and users, providing a graphical or command-line interface for user input and output.
* **Data Management Module:** Handles the storage, retrieval, and manipulation of data within the system, including database management, data validation, and data integrity.
* **Processing Module:** Executes core processing tasks or algorithms specific to the system's purpose. This module performs calculations, transformations, or operations on input data to generate desired outputs.
* **Reporting and Analytics Module:** Generates reports, visualizations, or analytics based on system data, providing insights or summaries to users or administrators.
* **Communication Module:** Facilitates communication and data exchange between different components or external systems, such as APIs, network protocols, or messaging services.
* **Configuration and Administration Module:** Manages system configuration settings, system parameters, and administration tasks such as user management, system backups, and updates.

The specific modules required in a system will depend on its functionality, complexity, and specific requirements. System analysis, design, and domain expertise are essential for identifying and defining the appropriate modules for a particular system.

**5.2 Sample Code**

**import warnings**

**warnings.filterwarnings('ignore')**

**from tensorflow import keras**

**from keras.layers import Input, Lambda, Dense, Flatten**

**from keras.models import Model**

**from keras.applications.vgg16 import VGG16**

**from keras.applications.vgg16 import preprocess\_input**

**from keras.preprocessing import image**

**from keras.preprocessing.image import ImageDataGenerator**

**from keras.models import Sequential**

**import numpy as np**

**from glob import glob**

**import matplotlib.pyplot as plt**

**IMAGE\_SIZE = [224, 224]**

**train\_path = 'Datasets/train'**

**valid\_path = 'Datasets/test'**

**vgg = VGG16(input\_shape=IMAGE\_SIZE + [3], weights='imagenet', include\_top=False)**

**for layer in vgg.layers:**

**layer.trainable = False**

**folders = glob('Datasets/train/\*')**

**x = Flatten()(vgg.output)**

**prediction = Dense(len(folders), activation='softmax')(x)**

**# create a model object**

**model = Model(inputs=vgg.input, outputs=prediction)**

**# view the structure of the model**

**model.summary()**

**model.compile(**

**loss='categorical\_crossentropy',**

**optimizer='adam',**

**metrics=['accuracy']**

**)**

**from keras.preprocessing.image import ImageDataGenerator**

**train\_datagen = ImageDataGenerator(rescale = 1./255,**

**shear\_range = 0.2,**

**zoom\_range = 0.2,**

**horizontal\_flip = True)**

**test\_datagen = ImageDataGenerator(rescale = 1./255)**

**# Make sure you provide the same target size as initialied for the image size**

**training\_set = train\_datagen.flow\_from\_directory('Datasets/train',**

**target\_size = (224, 224),**

**batch\_size = 10,**

**class\_mode = 'categorical')**

**test\_set = test\_datagen.flow\_from\_directory('Datasets/test',**

**target\_size = (224, 224),**

**batch\_size = 10,**

**class\_mode = 'categorical')**

**r = model.fit\_generator(**

**training\_set,**

**validation\_data=test\_set,**

**epochs=1,**

**steps\_per\_epoch=len(training\_set),**

**validation\_steps=len(test\_set)**

**)**

**import tensorflow as tf**

**from keras.models import load\_model**

**model.save('chest\_xray.h5')**

**from keras.models import load\_model**

**from keras.preprocessing import image**

**from keras.applications.vgg16 import preprocess\_input**

**import numpy as np**

**model=load\_model('chest\_xray.h5')**

**img=image.load\_img('D:\\Machine Learning DataSet\\Cheast\_xray\\Datasets\\val\\NORMAL\\NORMAL2-IM-1431-0001.jpeg',target\_size=(224,224))**

**x=image.img\_to\_array(img)**

**x=np.expand\_dims(x, axis=0)**

**img\_data=preprocess\_input(x)**

**classes=model.predict(img\_data)**

**result=int(classes[0][0])**

**if result==0:**

**print("Person is Affected By PNEUMONIA")**

**else:**

**print("Result is Normal")**

**Chapter 6**

**Results and Discussion**

**6.1 Result**

**6.2 Conclusion**

TB detection has benefited from advancements in deep learning, machine learning, and integration of clinical data. Methods like CNNs have shown promise in accurately detecting TB from medical images. Mobile applications and point-of-care solutions have improved accessibility and rapid screening. Continued research is needed to enhance sensitivity and specificity in TB detection methods using artificial intelligence and data analytics.

**6.2 Future Scope**

The proposed application aims to create a friendly working environment for any child health care centers and to overcome the drawbacks in existing system of child health care management. This system is very reliable and flexible from all aspects, so new features and modules can be easily integrated into the system in future.

**References**

1. Cassedy A, Fairbrother G, Newacheck P. The impact of insurance instability on children's access, utilization, and satisfaction with health care. Ambul Pediatr 2008 Sep/Oct;8(5):3218.PMID:18922506. <http://www.sciencedirect.com/science/article/pii/S1530156708001172>. Accessed March 22, 2016.
2. Guevara J, Moon J, Hines E, et al. Continuity of public insurance coverage: a systematic review of the literature. Med Care Res Rev 2014;71(2):115-37. PMID: 24227811. http://mcr.sagepub.com/content/71/2/115.long. Accessed March 22, 2016.
3. Kenney G, Pelletier J. Monitoring duration of coverage in Medicaid and CHIP to assess program performance and quality. Acad Pediatr 2011;May/Jun;11(3 Suppl):S3441. <http://www.sciencedirect.com/science/article/pii/S1876285910001257> . Accessed March 22, 2016.
4. U.S. Census Bureau. Community facts: 2010 profile of general population and housing characteristics: 2010 demographic profile data. <http://factfinder2.census.gov/faces/tableservices/jsf/pages/productview.xhtml?pid=DEC_10_DP_DPDP1>. Accessed March 22, 2016.
5. Centers for Disease Control and Prevention. Meningitis. <http://www.cdc.gov/meningitis/index.html>. Accessed March 22, 2016.
6. Centers for Disease Control and Prevention. Meningococcal Disease. <http://www.cdc.gov/meningococcal/index.html>. Accessed March 22, 2016.