Low Level Design (LLD)

Disease Recognition Using X-ray Plates

Revision Number - 1.2
Last Date of Revision - 24/05/2024

FARHAN AHMAD

Document Control

Date	Version	Description	Author
14/04/2024	1.0	Introduction, Problem Statement	FARHAN AHMAD
18/05/2024	1.1	Dataset Information, Architecture Description	FARHAN AHMAD
24/05/2024	1.2	Final Revision	FARHAN AHMAD

Contents

1. Introduction

1.1 What is Low Level Design Document?

The goal of the Low-level design document (LLDD) is to give the internal logic design of the actual program code for the Disease Recognition Using X-ray plates . LLDD describes the class diagrams with the methods and relations between classes and programs specs. It describes the modules so that the programmer can directly code the program from the document.

1.2 What is Scope?

Low-level design (LLD) is a component-level design process that follows a step-by-step refinement process. The process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work.

1.3 Project Introduction

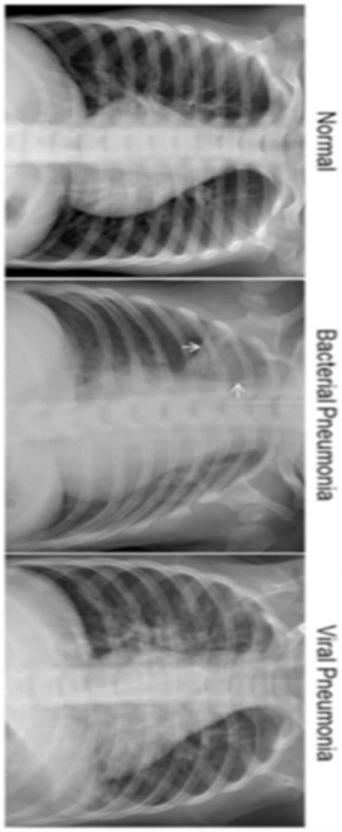
An X-ray is a common imaging test that's been used for decades. It can help your doctor view the inside of your body without having to make an incision. This can help them diagnose, monitor, and treat many medical conditions. Different types of X-rays are used for different purposes. For example, your doctor may order a mammogram to examine your breasts. Or they may order an X-ray with a barium enema to get a closer look at your gastrointestinal tract.

There are some risks involved in getting an X-ray. X-rays are standard procedures. An X-ray technologist or radiologist can perform an X-ray in a hospital's radiology department, a dentist's office, or a clinic that specializes in diagnostic procedures.

2. Problem Statement

X-rays are used to examine many parts of our body. Sometimes a human doctor might miss slightest part of disfunctionality. An automated vision based system which is trained with huge amount of data which can recognize the slightest amount of disfunctionality in any part of the body which can be visible through X-ray Plates.

3. Dataset Information



Normal: Nomral X-ray of human body, depicts clear lungs without any areas of abnormal opacification in the image

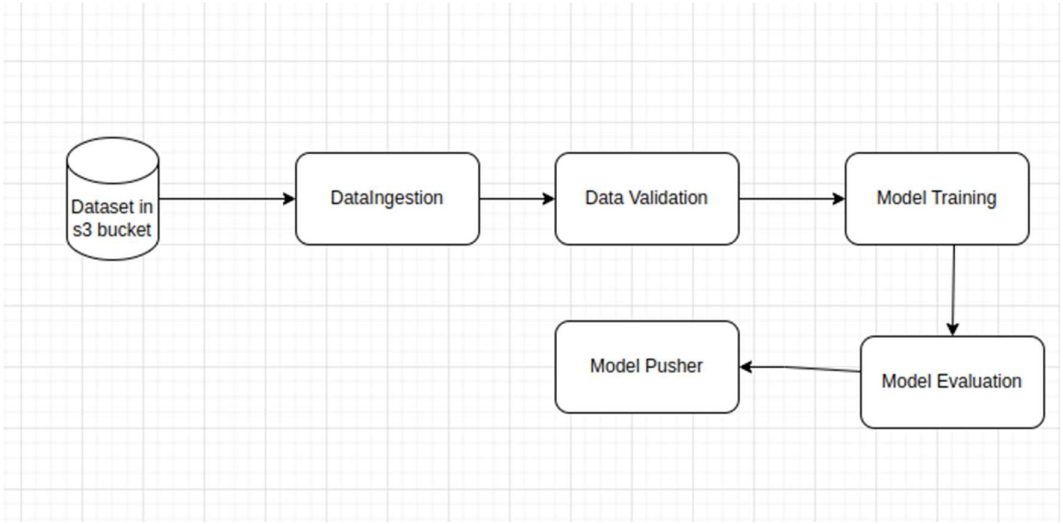
Bacterial pneumonia : typically exhibits a focal lobar consolidation, in this case in the right upper lobe (white arrows), whereas viral pneumonia (right) manifests with a more diffuse ‘‘interstitial’’ pattern in both lungs.

Number of X-ray reports: 5,863 X-Ray images incorporated

Train Data :

Validate Data :

4. Architecture



4.1 Architecture Description

1. **Raw Data Collection** The Dataset has been taken from Kaggle database

2. Data Ingestion

Data ingestion is the process of importing large, assorted data files from multiple sources into a single, cloud-based storage medium—a data warehouse, data mart or database—where it can be accessed and analyzed. As data may be in multiple different forms and come from hundreds of sources, it is sanitized and transformed into a uniform format using an extract/transform/load (ETL) process.

3. Data Validation

Data validation is the process of verifying and validating data that is collected before it is used. Any type of data handling task, whether it is gathering data, analyzing it, or structuring it for presentation, must include data validation to ensure accurate results. Sometimes it can be tempting to skip validation since it takes time. However, it is an essential step toward garnering the best results possible.

4. Model Training

Exploratory A training model is a dataset that is used to train an ML algorithm. It consists of the sample output data and the corresponding sets of input data that have an influence on the output. The training model is used to run the input data through the algorithm to correlate the processed output against the sample output. The result from this correlation is used to modify the model.

This iterative process is called “model fitting”. The accuracy of the training dataset or the validation dataset is critical for the precision of the model.

5. Model Evaluation

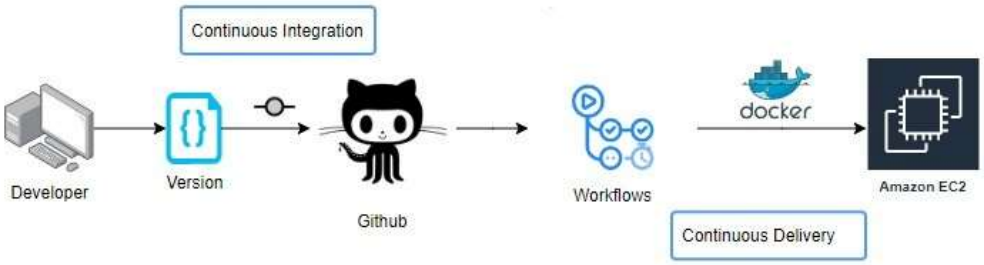
Model evaluation is the process that uses some metrics which help us to analyze the performance of the model. As we all know that model development is a multi-step process and a check should be kept on how well the model generalizes future predictions. Therefore evaluating a model plays a vital role so that we can judge the performance of our model. The evaluation also helps to analyze a model’s key weaknesses.

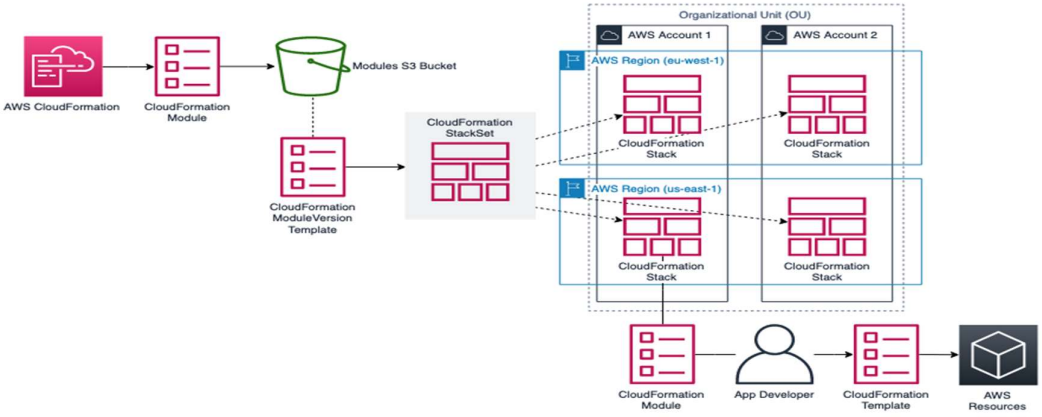
6. Model Pusher

Data Modelling is the process of analysing the data objects and their relationship to the other objects. It is used to analyse the data requirements that are required for the business processes. The data models are created for the data to be stored in a database. The Data Model's main focus is on what data is needed and how we have to organize data rather than what operations we have to perform.

7. Deployment

We have used and created an Amazon EC2


Continuous Integration
Developer
Version
Github
Workflows
docker
Amazon EC2
Continuous Delivery


AWS CloudFormation
CloudFormation Module
Modules S3 Bucket
CloudFormation ModuleVersion Template
CloudFormation StackSet
Organizational Unit (OU)
AWS Account 1
AWS Account 2
AWS Region (eu-west-1)
AWS Region (us-east-1)
CloudFormation Stack
CloudFormation Module
App Developer
CloudFormation Template
AWS Resources

***