SRS

1. INTRODUCTION

1.1 Purpose

The purpose of SRS is to provide a detailed overview of our product. This document would specify the functional and non-functional requirements of the software. It would also provide a clear and concise description of the system, including how it interfaces with healthcare providers and medical equipment, the expected accuracy and sensitivity of the system, and the regulatory requirements that must be met.

The SRS would establish a common understanding of the project among stakeholders, including developers, healthcare professionals, and regulatory bodies. It would also serve as a basis for testing and quality assurance to ensure that the system meets the expected standards for accuracy and performance.

Overall, the SRS would facilitate project management and development by providing a clear roadmap for the development team to follow, allowing them to build a system that can help healthcare professionals diagnose and treat patients with COVID-19.

1.2 Scope

The scope of the SRS is to describe all the requirements of the software system required for detection of COVID-19 using X-Ray Scans. The SRS document for a project focused on COVID-19 detection from chest X-ray imaging using the ResNet-18 model will provide a detailed roadmap for the development of the system, as well as clear guidance for its use and maintenance.

1.3 References

1. Rahimzadeh M, Attar A. Convolutional neural network for COVID-19 detection from chest X-ray images using transfer learning. Preprints. 2020:2020030452.
2. Apostolopoulos ID, Mpesiana TA. Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks. Phys Eng Sci Med. 2020;43(2):635-40.
3. Khan AI, Shah JL, Bhat MM. CoroNet: A deep neural network for detection and diagnosis of COVID-19 from chest x-ray images. Comput Methods Programs Biomed. 2020;196:105581.

1.4 Overview

The other section in this SRS document describes the other aspects of project. Section 2 will give the general description of the project.

2. The Overall Description

2.1 Product Perspective

From a product perspective, this project could be useful in multiple ways. The system could be deployed as a web application or a mobile application, allowing healthcare professionals to access it easily. The project has the potential to provide an efficient and accurate solution to the diagnostic challenges of COVID-19, and could potentially improve patient outcomes.

2.1.1 System Interfaces

Our project will have the following system interfaces:

1. Data collection interface: This interface is responsible for collecting and storing the Chest X-Ray scans and their corresponding labels.
2. Preprocessing interface: This interface is responsible for preprocessing the collected data, such as resizing, normalization, and data augmentation.
3. Model training interface: This interface is responsible for training the ResNet-18 model on the preprocessed data.
4. Model evaluation interface: This interface is responsible for evaluating the trained model's performance on a separate test set using appropriate metrics.
5. Prediction interface: This interface is responsible for making predictions on new Chest X-Ray scans using the trained model.
6. User interface: This interface is responsible for providing an interactive interface for the end-users to input and view the results of the prediction.
7. Deployment interface: This interface is responsible for deploying the trained model and its associated components to the production environment.

2.1.2 Interfaces

We will use a Graphical User Interface (GUI) for this project. The GUI will have two main functionalities: uploading Chest X-ray scans and viewing prediction results. The GUI should provide clear instructions on the format and resolution requirements for the uploaded image and will also display the prediction in a clear and easy to understand way.

2.1.3 Hardware Interfaces

Our project requires a hardware interface to allow users to upload Chest X-ray scans to the system. The hardware interface will be compatible with the devices used by the end-users, such as laptops, desktops, tablets, or smartphones. The interface will support various image formats, including JPEG, PNG, and TIFF, and have a minimum resolution requirement of 1024 x 1024 pixels.

2.1.4 Software Interfaces

The project uses Python, PyTorch, a popular deep learning framework that's used for developing and training the ResNet-18 model and python libraries such as OpenCV or PIL to handle image processing tasks.

2.2 Product Functions

- Upload Chest X-ray scans: The system should allow users to upload Chest X-ray scans in various formats, such as JPEG, PNG, and TIFF.
- Preprocess images: The system should preprocess the uploaded images by resizing them to the required resolution of 1024 x 1024 pixels and converting them to grayscale.
- Classify images: The system should classify the preprocessed images into one of the three classes - Normal, Viral Pneumonia, and COVID-19 - using the trained ResNet-18 model.
- Display prediction results: The system should display the prediction results to the user in a user-friendly and informative way.

2.3 User Characteristics

The covid-19 detection system using X-ray scans will be used by healthcare professionals such as doctors, radiologists, pneumologists and other medical professionals. It can also be used by general public, students, researchers to learn more about Chest X-ray scans and the classification of pneumonia and COVID-19. They may not have any technical or medical background.

2.4 Constraints

The model will require a internet connection to run and will be functional on the devices which meet the specified hardware requirements.

2.5 Assumptions and Dependencies

Assumptions:

1. The Chest X-ray scans in the dataset are correctly labeled as Normal, Viral Pneumonia, or COVID-19.
2. The dataset is representative of a diverse range of Chest X-ray scans and does not suffer from any significant biases.
3. The ResNet-18 model architecture is appropriate for the classification task and has been adequately trained on the dataset.

Dependencies:

1. The project requires access to a dataset of Chest X-ray scans that have been labeled as Normal, Viral Pneumonia, or COVID-19.
2. The project relies on the PyTorch deep learning framework to train the ResNet-18 model and perform image classification.

3. The project requires sufficient computational resources, including a CPU or GPU capable of handling the deep learning computations and adequate storage for the dataset and model files.

3. Specific Requirements

3.1 External Interface

1. Dataset source: The project may depend on external sources of Chest X-ray scans labeled as Normal, Viral Pneumonia, or COVID-19. You may need to download or access this data from a website or a database.
2. PyTorch: The project depends on the PyTorch deep learning framework, which is an external interface. We may need to install and configure PyTorch on your system to use it for training and testing the ResNet-18 model.
3. File storage: Your project may require external interfaces for file storage and retrieval.
4. User interface:The project may have an external interface for the user interface, including buttons, menus, and other graphical elements that allow users to interact with the system.
5. API: The project may offer an external interface for an API that allows other applications to access its functionality and perform image classification tasks.
6. Other libraries and frameworks: The project may depend on other libraries and frameworks for tasks such as data preprocessing, visualization, or evaluation. These external interfaces may need to be installed and configured on the system to use them in project.

3.2 Functions

- Loading and preprocessing the dataset: This function loads the Chest X-ray dataset and performs preprocessing steps such as resizing the images, converting them to grayscale, and splitting them into training and testing sets.
- Defining the ResNet-18 model: This function defines the ResNet-18 model architecture using PyTorch and specifies the number of classes for image classification.
- Training the ResNet-18 model: This function trains the ResNet-18 model on the training set and evaluates its performance on the validation set. It also saves the best performing model checkpoint for later use.
- Testing the trained model: This function loads the saved model checkpoint and evaluates its performance on the testing set, including metrics such as accuracy, precision, recall, and F1-score.
- Predicting image classification: This function takes an input Chest X-ray image and uses the trained model to predict its class as Normal, Viral Pneumonia, or COVID-19.
- Deploying the model: This function deploys the trained model on a web server or a cloud-based platform for use by external applications or users.
- Monitoring and updating the model: This function monitors the performance of the deployed model and updates it periodically with new data or model improvements to maintain optimal accuracy and reliability.

3.3 Performance Requirements

It will have following Performance requirements:

- Accuracy: The model should have an accuracy of at least 90%.
- Training and testing time: The model should be trained and tested within a reasonable time frame.
- User interface responsiveness: The user interface should be responsive and user-friendly, with fast and accurate image classification results displayed in real-time.

3.4 Software System Attributes

3.4.1 Reliability

The model will be able to accurately classify Chest X-ray images as Normal, Viral Pneumonia, or COVID-19 with a high degree of accuracy. To ensure reliability, the model will be trained on a diverse and representative dataset, validated on a separate test set, and tested on real-world data.

3.4.2 Availability

The model will be accessible to authorized users 24/7, without any interruptions or downtime. The model will be optimized for fast and efficient execution to minimize any potential delays or waiting times for the users.

3.4.3 Security

The model will protect the confidentiality, integrity, and availability of the Chest X-ray images and their associated metadata.

3.4.4 Maintainability

The Model will be easily maintainable. The model will be easy to modify, update, and extend as new data, features, or requirements emerge.

3.4.5 Portability

The model will be portable. The model will be able to run on various operating systems, hardware configurations, and cloud-based infrastructures, depending on the needs of the users.