A PRELIMENERY REPORT ON

SUMMARIZED MEDICAL REPORT GENERATION ON CHEST X-RAYS

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE
IN THE FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF ENGINEERING (COMPUTER ENGINEERING)

SUBMITTED BY

BHAVIK DIWANI	Exam No : 71900205C
AAKASH PATIL	Exam No : 71900462E
CHINMAY RAHATE	Exam No : 71900513C
SHREYAS KULKANI	Exam No : 71900621L

DEPARTMENT OF COMPUTER ENGINEERING

PUNE INSTITUTE OF COMPUTER TECHNOLOGY

DHANKAWADI, PUNE - 411043

SAVITRIBAI PHULE PUNE UNIVERSITY

2021-2022

CERTIFICATE

This is to certify that the project entitled

SUMMARIZED MEDICAL REPORT GENERATION ON CHEST X-RAYS

Submitted by

BHAVIK DIWANI	Exam No : 71900205C
AAKASH PATIL	Exam No : 71900462E
CHINMAY RAHATE	Exam No : 71900513C
SHREYAS KULKANI	Exam No : 71900621L

are bonafide students of this institute and the work has been carried out by them under the supervision of **Prof. Anupama Phakatkar**, and it is approved for the partial fulfillment of the requirements of Savitribai Phule Pune University, for the award of the degree of **Bachelor of Engineering** (Computer Engineering).

Prof. Anupama Phakatkar
Guide,
Dept. of Computer Engg.

Dr. Mukta Sunil Takalikar
Head,
Dept. of Computer Engg.

Dr. R. Sreemathy
Principal,
Pune Institute of Computer Technology

Place: Pune

Date:

ACKNOWLEDGEMENTS

It gives us great pleasure in presenting the preliminary project report on **'Summarized Medical Report Generation on Chest X-Rays '**.

We would like to take this opportunity to thank our internal guide **Prof. Anupama Phakatkar** for giving us all the help and guidance we needed. We are really grateful to them for their kind support. Their valuable suggestions were very helpful.

We are also grateful to **Dr. Mukta Sunil Takalikar**, Head of Computer Engineering Department, PICT for her indispensable support, suggestions.

We would also like to thank our family and friends, without their support this would not have been possible.

Bhavik Diwani
Aaksh Patil
Chinmay Rahate
Shreyas Kulkarni

ABSTRACT

The process of developing visual representations of the interior of a body for clinical study, as well as visual representations of the function of specific organs or tissues, is referred to as medical imaging. They're commonly utilized to diagnose fractures and disorders in hospitals and clinics. Specialized medical professionals read and analyze the medical images, and their results about each body of area inspected are provided through written Medical Reports. The time it takes to write a medical report is usually between 5 and 10 minutes. In a single day, doctors must write hundreds of medical reports, which can consume a significant amount of their time. The main motive of this case study is to create a deep learning model that can automatically write the impression section of a medical report for chest X-rays, relieving the medical professional of some of their responsibilities.

Keywords:

Deep Learning, Machine Learning, Neural Networks, LSTM, BLEU Metric Score

Contents

LIST OF ABBREVIATIONS

ABBREVATION	ILLUSTRATION
SDLC	Software Development Life Cycle
UI	User Interface
UML	Unified Modeling Language
LSTM	Long Short Term Memory
CNN	Convolutional Neural Networks
RNN	Recurrent Neural Network
BLEU	Bilingual Evaluation Understudy
COCO	Common Objects in Context Dataset

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Image captioning is a difficult artificial intelligence challenge that entails creating a textual description from an image based on its contents. "A chest x-ray" is a frequent response among "non-radiologists". They produce textual reports to describe the findings of the imaging investigation in each part of the body, including whether each area was found to be normal, abnormal, or potentially abnormal. We can extract so much useful information and create medical reports from a single photograph.

1.2 MOTIVATION

The process of developing visual representations of the interior of a body for clinical study, as well as visual representations of the function of specific organs or tissues, is referred to as medical imaging. They're commonly utilized to diagnose fractures and disorders in hospitals and clinics. Specialized medical professionals read and analyze the medical images, and their results about each body of area inspected are provided through written Medical Reports.

Producing medical-imaging reports can be hard for inexperienced radiologists and pathologists, especially those working in rural areas where healthcare quality is inadequate; on the other side, writing imaging reports can be monotonous and time consuming for experienced radiologists and pathologists.

The time it takes to write a medical report is usually between 5 and 10 minutes. In a single day, doctors must write hundreds of medical reports, which can consume a significant amount of their time. The main motive of this project is to create a deep learning model that can automatically write the impression section of a medical report for chest X-rays, relieving the medical professional of some of their responsibilities.

1.3 PROBLEM DEFINITION

To build a system that generates summarized medical report of Chest X-Rays using deep learning model. The application helps medical professionals in clinical analysis of X-Rays and to determine fractures and abnormalities. The system can be interacted with Web-app interface.

CHAPTER 2

LITERATURE SURVEY

The research paper[1] discusses the development of a software artifact to implement the proposed model using real data.It is developed using a CNN and LSTM-based basic encoder-decoder paradigm. As a result, a data-analytic model like the well-known and widely used CRISP-DM can help this endeavour. The Cross-Industry Standard Process for Data Mining (CRISP-DM) is an open standard process model and industry-proven methodology for guiding data mining operations.

In research paper[2], basic LSTM model was trained. MSCOCO dataset has been used to train the model. A pre trained Caffe model has been used for ResNet CNN and the LSTM model was implemented in Torch. To assess the qualitative accuracy of the model, a BLEU score has been used. The architecture has been extended to use attention models, and shows some initial results from this investigation

The research paper[3] discusses three models for generating medical captions for chest-xrays. The encoder architecture is kept same for all the three models but changes are made in decoder. For encoder architecture CheXNET model is utilized.The first model discussed is simple encoder-decoder model in which simple image captioning model is explained.The second model is attention model in which global attention is considered along with contact equation.The last model discussed is a custom model where decoder is same as the one used in attention model but encoder is customized. The models are compared using two search methods i.e beam and greedy search and evaluated on basis of BLEU metric score .

The research paper[4] discusses about ChexNet, a 121-layer convolutional neural network that takes a chest X-ray image and outputs the probability of pneumonia, as well as a heatmap that shows the areas of the image that are most likely to indicate pneumonia. They have used the recently available ChestX-ray14 dataset to train CheXNet, which contains 112,120 frontal-view chest X-ray pictures that have been individually classified with up to 14 distinct thoracic illnesses, including pneumonia.

The research paper[5] discusses deep learning-based models for picture captioning that are operating on an end-to-end learning strategy. Uses the BLEU evaluation metric to generate a score that can be used to determine the quality of picture captions generated.

TITLE	DESCRIPTION	DATASET
Deep Learning for Image Captioning: An Encoder-Decoder Architecture [1]	Simple encoder decoder model using CNN and LSTM	MS-COCO, Flickr 30k, Flicker 8k
Image Captioning with Attention[2]	Attention Model Mechanism using CNN and RNN	MS-COCO
Medical Image Captioning on Chest X-Rays[3]	Encoder Decoder and Attention Model mechanism	Indiana University Dataset
CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning[4]	Detect pneumonia from chest X-rays at a level exceeding practicing radiologist	ChestX-ray14 dataset
A Survey on Various Deep Learning Models for Automatic Image Captioning[5]	Discussed various models for image captioning	MS-COCO, Flickr 30k, Flicker 8k

Table 2.1: Literature Survey

CHAPTER 3

SOFTWARE REQUIREMENTS SPECIFICATION

3.1 INTRODUCTION

This project focuses on building a web application for generating summarized medical reports for Chest X-Rays. We have achieved this using deep learning techniques like Attention mechanism, LSTM, CNN. The model generates captions for the Chest X-Ray images, then we will be converting it into a summarized report and providing it to the end user.

3.1.1 PROJECT SCOPE

This project is currently being built to automate the process of generating summarized reports for Chest X-Rays. The scope of this project is to create a deep learning model that can automatically write the impression section of a medical report for chest X-rays, relieving the medical professional of some of their responsibilities.

3.1.2 USER CLASSES AND CHARACTERISTICS

This sub-section discusses the role and privilege level enjoyed by different users of the system. The users of the system can be identified into these broad categories as follows :

- Front End User
- Back End Developer

The front-end user will be exposed only to the front UI of the system and will be able to upload Chest X-Ray images. The output produces summarized medical report for the given Chest X-Rays.

The back-end developer will implement and train the model and integrate with the front-end UI.

3.1.3 ASSUMPTIONS AND DEPENDENCIES

- All the users must have a stable internet connection for smooth and uninterrupted experience.
- All the images that are uploaded should be clearly visible and expandable.
- All the images that are uploaded should consist only of chest X-Rays for correct results.
- All the images that are uploaded should be in jpeg or png format.

3.2 FUNCTIONAL REQUIREMENTS

As a relationship between input and output, functional requirements are used to define a system function or a component of a system. This section goes over the functional requirements for the proposed system. All functional mainly revolve around these three points:

- Purpose - To compare obtained output with expected ideal output.
- Input - Images of Chest X-Rays uploaded by user.
- Output - To generate relevant summarized reports for given Chest X-Rays.

3.2.1 EXECUTING TEST CASES

The capacity to automate the testing process is one of the system's most critical functional needs. This entails running all pre-existing test cases at the same time and comparing the results to the intended outcome.

3.2.2 REPORT GENERATION

The back-end developers will be able to generate a summarised report providing the specifics of each individual test case that was successfully completed when all test cases have been properly executed.

3.2.3 UPLOAD AND MODIFY TEST CASES

The back-end developer will be able to train the model with different images and reports.

3.3 EXTERNAL INTERFACE REQUIREMENTS

This section covers the requirement for an external interface to be provided in the system. User, hardware, software, and communication are the four key subsections that determine external interface requirements.

3.3.1 USER INTERFACES

To make the User Experience (UX) easy and comfortable, the User Interface (UI) should be easily navigable and secure. The user experience is one of the most crucial aspects of any program or software; if the user is unhappy with the interface or encounters any problems while using it, it leaves a negative impression, and if this happens frequently enough, the user will eventually stop using our product. While a consistent, smooth, and perfect experience distinguishes our app from the competition, allowing users to focus more on the functional features and attracting new users.

3.3.2 SOFTWARE INTERFACES

- Image Captioning model on which the test cases will be executed.
- Web interface for the front-end user to interact.
- Any command line interface to interact with the system and run test cases.

3.4 NON-FUNCTIONAL REQUIREMENTS

The quality attribute of a software system is defined by the Non-Functional Requirement (NFR). They assess the software system on the basis of its responsiveness, usability, security, portability, and other non-functional criteria that are crucial to its success.

3.4.1 PERFORMANCE REQUIREMENTS

- To give users with an uninterrupted experience, the latency should be as minimal as feasible.
- Every component should be responsive to make it compatible to all devices.
- Fetching of the data from the database should be as fast as possible. Users shouldn't be kept waiting for the data to be loaded
- Maintain the states and cookies in case of system failure to uplift the user experience

3.4.2 SAFETY REQUIREMENTS

- System should not crash due to overabundance of queries.
- Abrupt failure of system should not happen.

3.4.3 SECURITY REQUIREMENTS

To avoid falling into an aborted state, the system should keep a limit on the number of inquiries that can be handled at the same time.

3.4.4 SOFTWARE QUALITY ATTRIBUTES

The software quality attributes for the system under consideration are :

- Maintainability - Need of updation of database whenever query is entered
- Robust - The software should be able to handle multiple queries at a time
- Recoverability - The database should have replicas in case of loss of data
- Availability - User should be able to query the database anytime

3.5 SYSTEM REQUIREMENTS

This section deals with the requirements for the working of the system like Software Requirements and Hardware Requirements.

- Hardware Requirements
  - Intel i3 Processor or advanced
  - GPU for performing computations on large datasets
- Software Requirements
  - Python 3
  - Numpy
  - Pandas
  - Scikit-learn
  - Web interface - HTML, CSS, Javascript

3.6 SDLC MODEL

SDLC stands for Software Development Life Cycle and denotes the process for efficient implementation that will be followed throughout the project's completion. Different models, such as Waterfall, V-Model, and Agile approach, were considered, and Agile methodology was chosen for the project since it best suited the project's characteristics.

Planning, Analysis, Design, Implementation, Testing, Integration, and Maintenance are the six phases of the Agile paradigm.

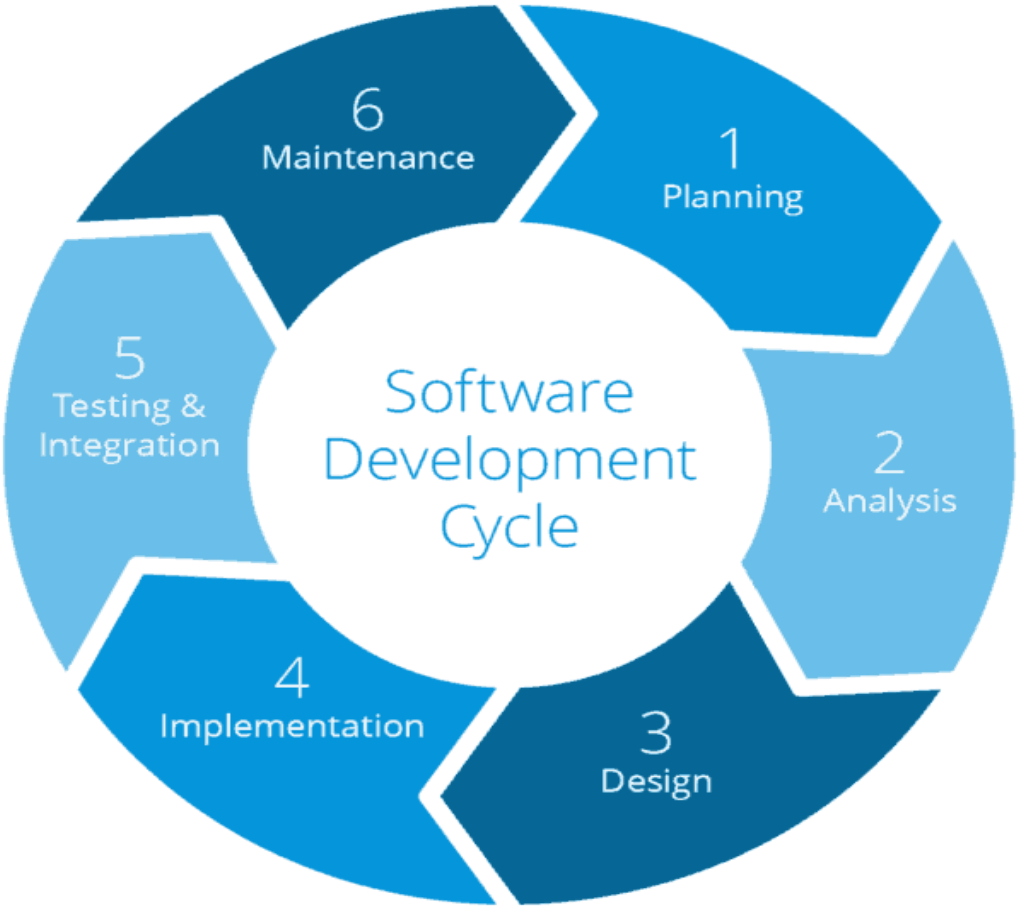

Figure 3.1: Agile Model

I. Planning - The project's outline was created together with the determination of the project's timeline.

II. Analysis - The project's outline was created together with the determination of the project's timeline.

III. Design - During this phase, the project's process and workflow were established.

IV. Implementation - The project will be programmed based on the design that has been laid out. The system's partial implementation has been completed.

V. Testing and Integration - In this phase, all of the project's components will be subjected to comprehensive unit and integration testing.

VI. Maintenance - The developed project will be maintained by conducting regular checks on its operation and ensuring that it achieves the desired results.

The Agile model allows you to analyse the current project's development at regular intervals, allowing you to judge the work's efficiency even early on. When compared to the Waterfall model and other comparable models, Agile approach also aids in preventing errors from trickling down the line. Agile methodology proves to be beneficial with respect to incorporating changes downstream.

As a result, Agile approach will be implemented throughout the whole SDLC for the system under construction.

CHAPTER 4

SYSTEM DESIGN

The graphic representation of the system as a whole is aided by system design. This chapter explains the system architecture for our regression test suite, which includes both a high-level overview and a detailed design.

4.1 SYSTEM ARCHITECTURE

The system architecture represents overall structure of the system. The system requires only one input - the set of test cases to be executed. This set will be retrieved from the database. The system runs in three primary stages. Stage I consists of running all test cases simultaneously. Stage II consists of comparing the obtained output (from the payment switch) with the expected output (retrieved from the database). Stage III involves summarizing the results of all test cases and generating a report which can be viewed by the switch developers for analysis.

Architecture for the system is discussed as per the points below:

- System Architecture
- Data flow

4.1.1 System Architecture

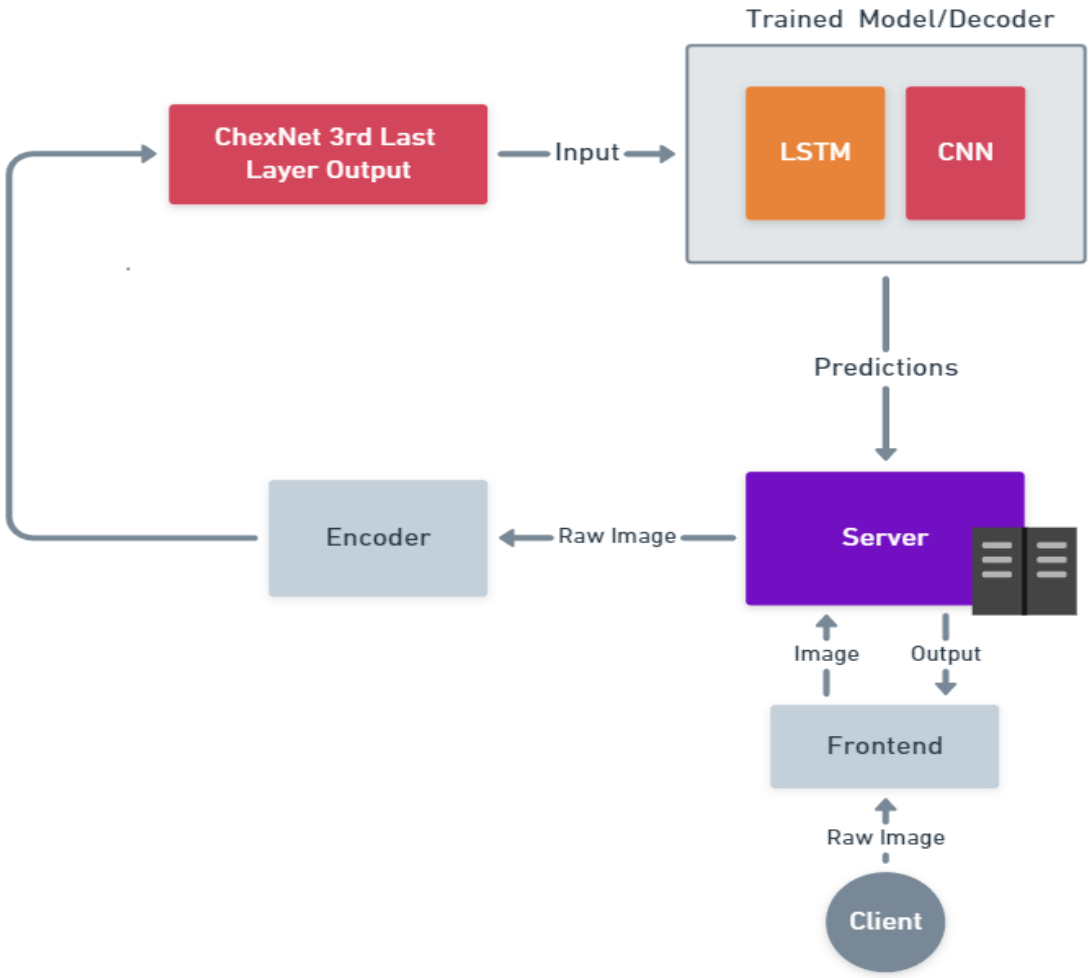

Figure 4.1: System Architecture

4.2 DATA FLOW DIAGRAMS

Data flow diagrams show how the system receives, manages, stores, and transports data. DFDs are typically discussed on three levels, and they're also important for identifying any potential dangers.

4.2.1 DFD LEVEL 0

Figure 4.2: DFD Level 0

4.2.2 DFD LEVEL 1

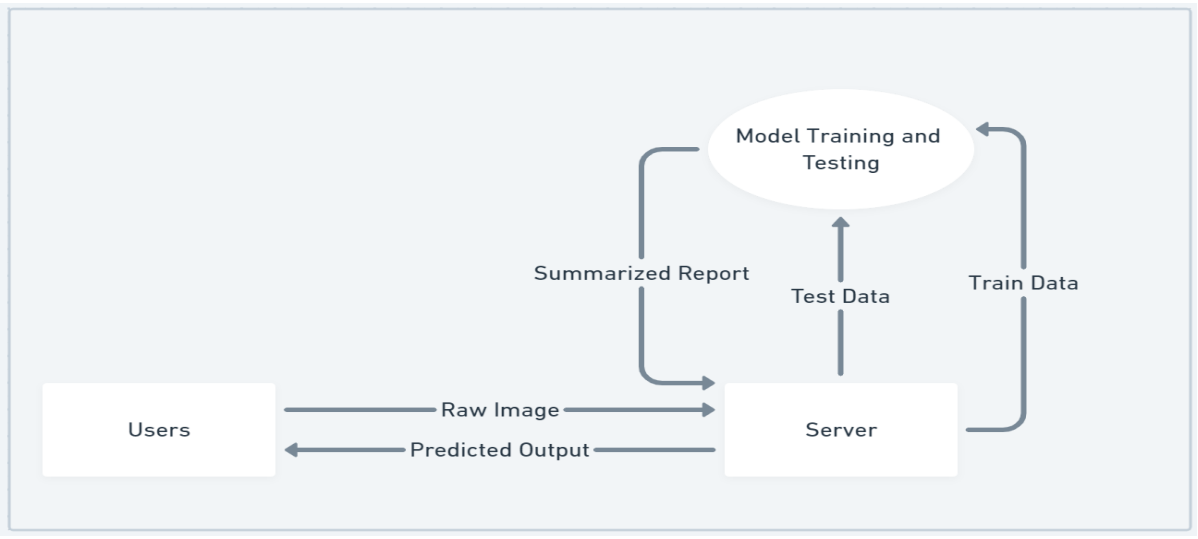

Figure 4.3: DFD Level 1

4.3 UML DIAGRAMS

The UML diagram, which stands for Unified Modelling Language, is extensively used to comprehend the interaction between external users and system components. It may also be used to understand how users interact with the system.

4.3.1 USE CASE DIAGRAM

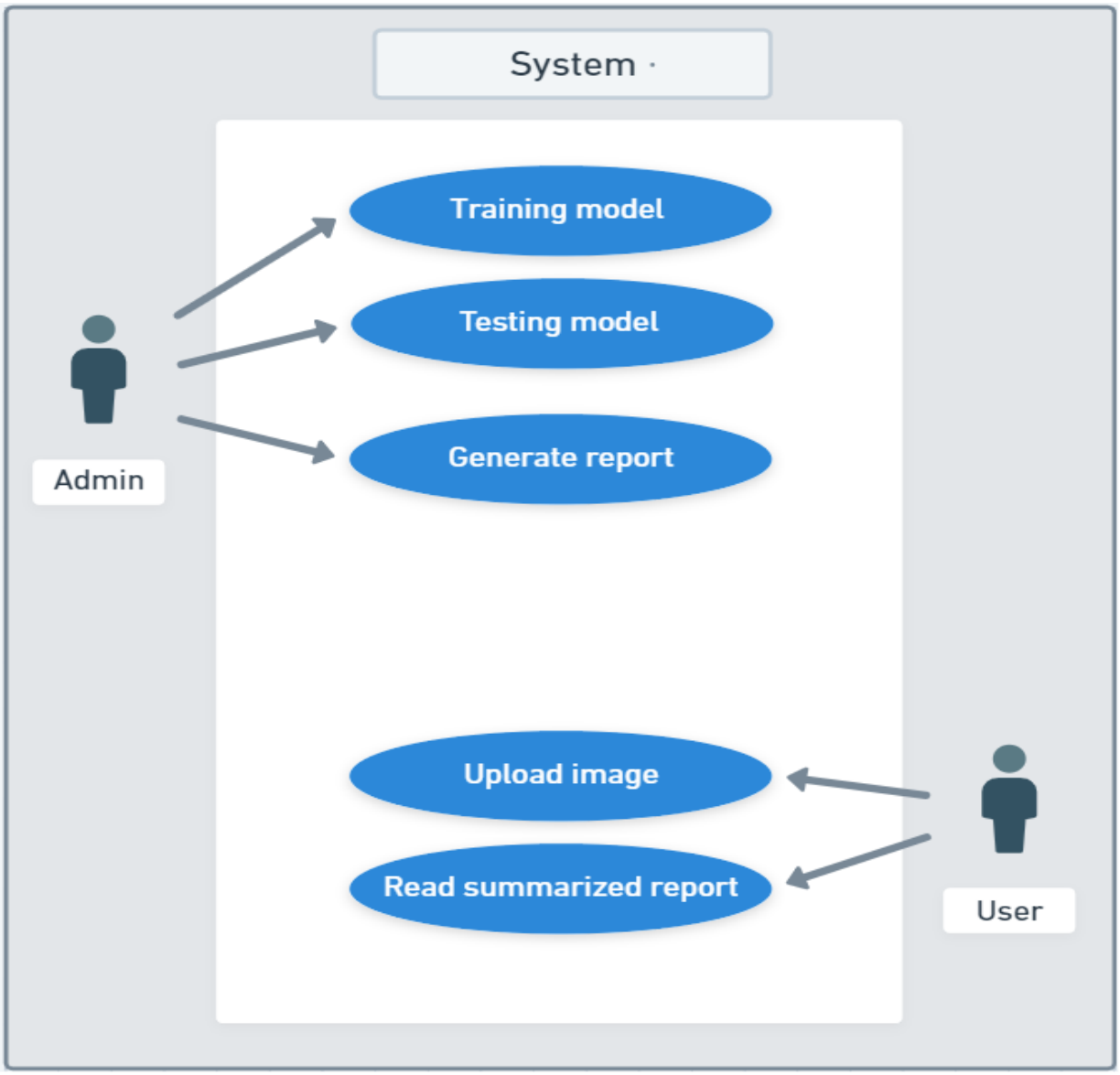

Figure 4.4: Use Case Diagram

4.3.2 ACTIVITY DIAGRAM

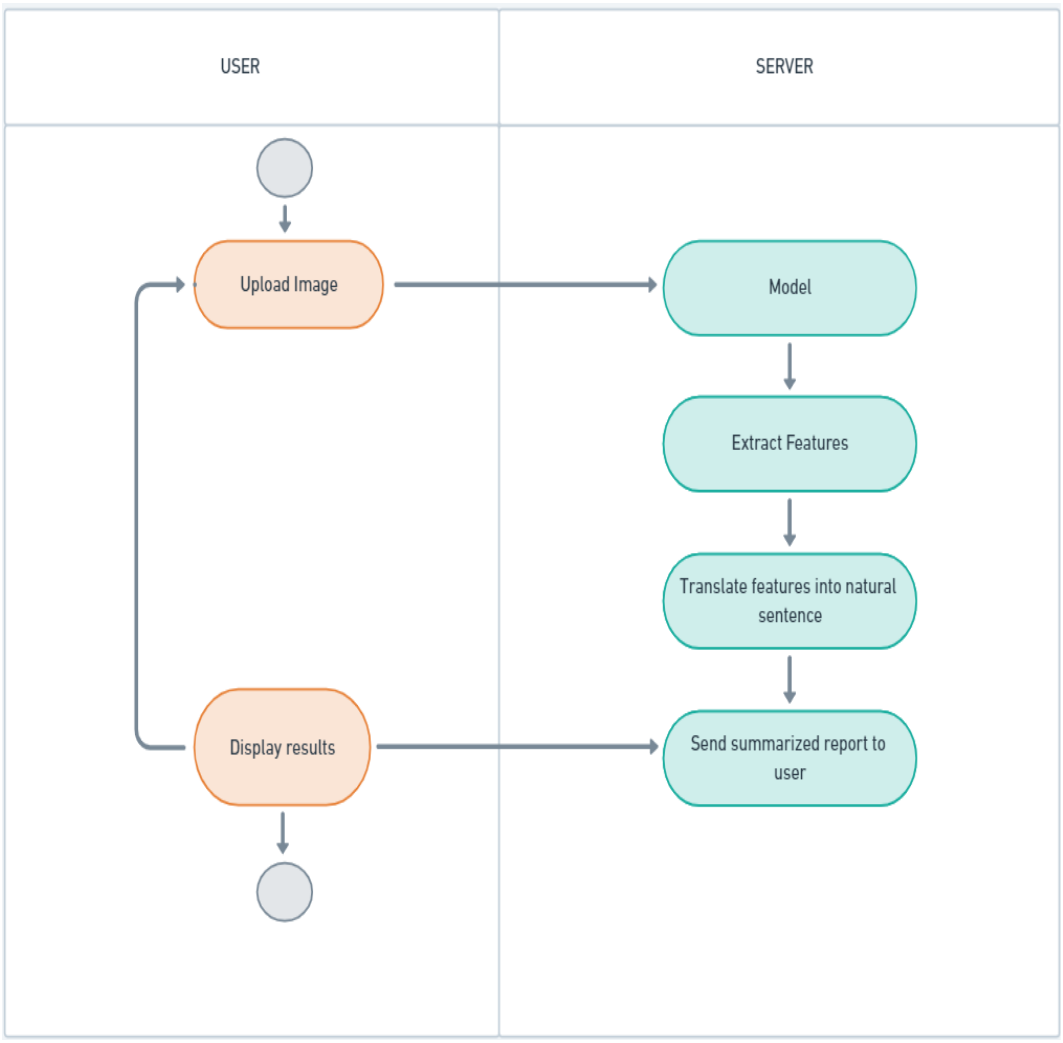

Figure 4.5: Activity Diagram

4.4 SEQUENCE DIAGRAM

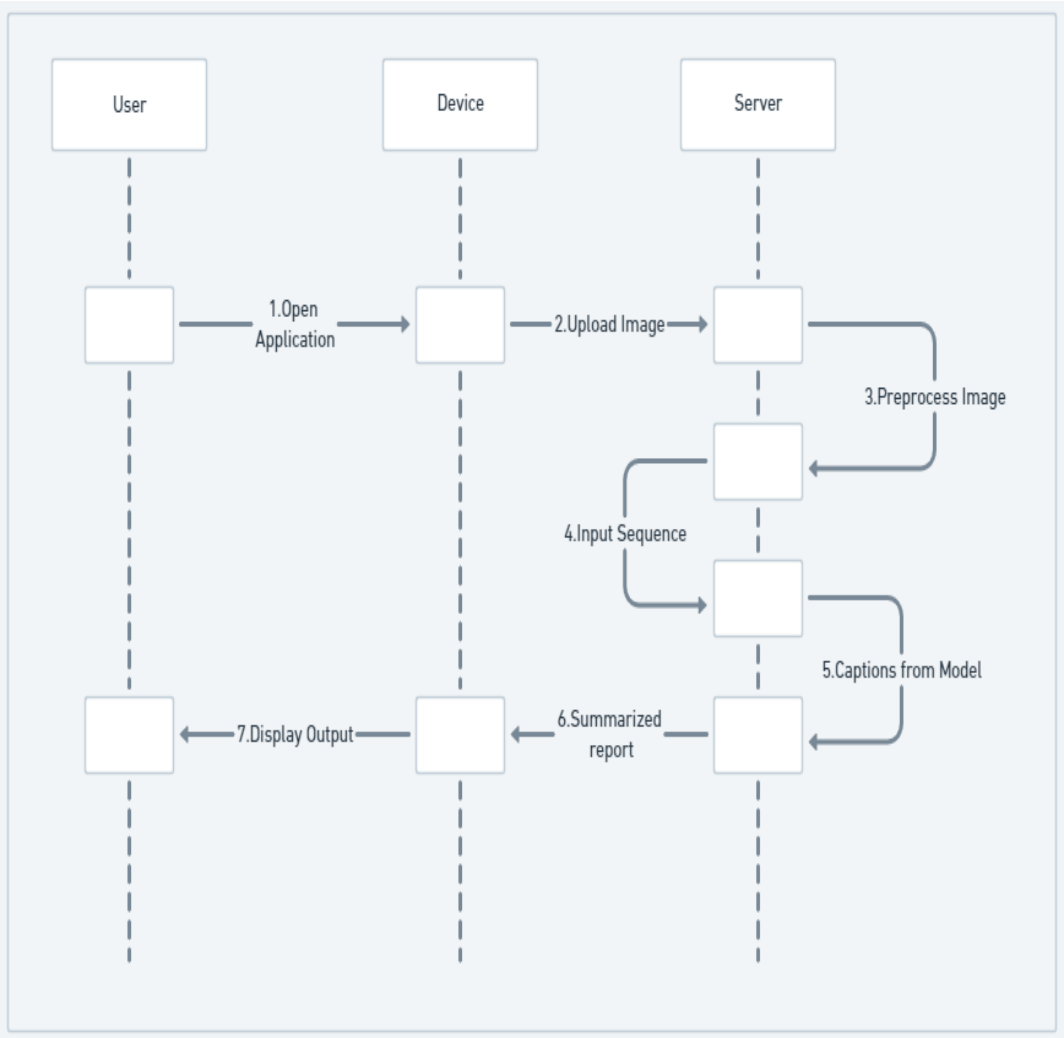

Figure 4.6: Sequence Diagram

CHAPTER 5

OTHER SPECIFICATIONS

5.1 ADVANTAGES

- The medical experts can save time by using automatic report generation for chest X-rays to detect various diseases.
- This report can be used by radiologists for cross-checking, tracking small changes, and making final decisions. It can also be employed in an emergency where experienced doctors are not immediately available.
- Low Cost Solution: Our project doesn't require much resources. We have chosen datasets that are taken from open source websites. So it is safe to say that our project operates at a lower cost as compared to the other AI techniques.
- The doctors can also use this mechanism to find any fractures in the chest.

5.2 LIMITATIONS

- There are a limited number of publicly available datasets for medical image captioning, and more large datasets are needed.

5.3 APPLICATIONS

- For medical report generation for different X-Rays
- Can be used for diagnosis of Chest diseases

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 CONCLUSION

We surveyed various deep learning models for Medical Image Captioning and finalized Attention Mechanism. We will be designing a web application for generating summarized reports for given Chest X-Rays using Attention mechanism, LSTM and CNN. We will use 3rd last layer of 121-layer Chexnet model as the input to our model which will be generating captions for the given Chest X-Rays and converting them into summarized reports to make it easy for end user to understand.

6.2 FUTURE SCOPE

Additional work is needed, such as the fact that there isn't a single model that can generate a comprehensive structured medical report (equivalent to a natural language description) with various components that are typically included in hand-written medical reports. Different modalities of medical pictures are employed, and better algorithms for generating reports using these modalities must be investigated and evaluated. We plan to create our own large data set with the help of domain experts as part of our future effort. We also plan to develop an industry-standard automatic report generating system that will provide a comprehensive medical report tailored to radiologists' needs, which is currently lacking in the literature.

APPENDIX A

FEASIBILITY STUDY

Computational Neural Networks and LSTM, or Long Short-Term Memory, are the two key classification algorithms used in the suggested system. In the case of these methods, we perform a feasibility study.

Long short-term memory (LSTM) is a deep learning architecture that uses an artificial recurrent neural network architecture. LSTM has feedback connections, unlike normal feedforward neural networks. It can handle not only individual data points (such as photos), but also complete data streams.

LSTM can be used for tasks like unsegmented, connected handwriting identification, speech recognition, and anomaly detection in network traffic or IDSs, for example (intrusion detection systems). There are three sorts of layers in CNNs. Convolutional layers, pooling layers, and fully-connected layers are the three types. A CNN architecture is generated when these layers are stacked.

APPENDIX B

PLAGIARISM REPORT

CHAPTER 7

REFERENCES

[1] Gaurav and Pratistha Mathu, "A survey on various deep learning models for automatic image captioning." *International Conference on Mechatronics and Artificial Intelligence (ICMAI)*, vol. 1950, 2021.

[2] Simao Herdade, Armin Kappeler, "Image captioning: Transforming objects into words," 2019, pp. 1–10.

[3] Karpathy, Andrej, Li, Fei, "Deep visual-semantic alignments for generating image descriptions." IEEE, 2015, pp. 664–676.

[4] Mao, Junhua Xu, Wei Yang, Yi Wang, Jiang Yuille, Alan, "Deep captioning with multimodal recurrent neural networks (m-rnn)," vol. 5, 2014.

[5] Vinyals, Oriol, Toshev, Alexander, Bengio, Samy, Erhan, Dumitru, "Show and tell: A neural image caption generator." IEEE, 2015, pp. 1275–1280.