MEDICAL PRESCRIPTION RECOGNITION USING MACHINE LEARNING

J. Alekhya[1], Thota Archana[2], Bandi Shivani[3], and Erukulla Reena[4]

[1]*Assistant Professor, Department of Information Technology, Gokaraju Rangaraju Institute of Engineering and Technology, JNTUH, Bachupally, Hyderabad, India.*

[2] *Student, Department of Information Technology, Gokaraju Rangaraju Institute of Engineering and Technology, JNTUH, Bachupally, Hyderabad, India.*
[3] *Student, Department of Information Technology, Gokaraju Rangaraju Institute of Engineering and Technology, JNTUH, Bachupally, Hyderabad, India.*
[4] *Student, Department of Information Technology, Gokaraju Rangaraju Institute of Engineering and Technology, JNTUH, Bachupally, Hyderabad, India*

Abstract. Illegible handwritten prescriptions by doctors pose significant challenges in healthcare, leading to misinterpretation of medication names and dosages. This project addresses the problem by leveraging machine learning technologies, particularly Convolutional Neural Networks (CNN) and Optical Character Recognition (OCR), to convert handwritten prescription details into readable digital text. The system involves pre-processing techniques like image subtraction, noise reduction, and resizing, followed by CNN for feature extraction and OCR for text conversion. Rigorous testing on real-world cases demonstrates the system's effectiveness in improving prescription legibility. This approach aims to enhance prescription accuracy, instill confidence in pharmacists and patients, and reduce misinterpretation risks.

1 Introduction

1.1 Introduction to project

In today's rapidly advancing healthcare landscape, technological innovations are essential for improving patient care, increasing efficiency, and reducing human error. A significant area where technology can make a profound impact is medical prescription handling. The proposed Medical Prescription Optical Character Recognition (OCR) system represents a ground-breaking advancement in this field, addressing critical issues and transforming the management of medical prescriptions.

Medical prescriptions are vital to healthcare, providing instructions for administering medications and outlining treatment plans. However, the current manual process of handling and interpreting prescriptions is prone to errors and inefficiencies. Misreading a prescription due to illegible handwriting or other factors can have severe consequences for patients. These challenges highlight the urgent need for a robust and reliable solution to enhance the accuracy and efficiency of prescription management.

Leveraging state-of-the-art technologies such as YOLOv5 for object detection and deep learning models for text recognition, the Medical Prescription OCR system offers a comprehensive approach to managing prescriptions. By accurately identifying and locating medicine names on prescriptions, the system minimizes the risk of misinterpretation, ensuring patients receive the correct medications and dosages. Additionally, the system enhances accessibility by making prescription information easily retrievable and interpretable, benefiting both healthcare professionals and patients.

This technology empowers patients by providing them with valuable information about their treatment regimens. Access to detailed data about medications, including their chemical composition, dosage, precautions, and potential side effects, enables patients to make informed decisions about their health. Furthermore, by automating prescription interpretation, the system significantly reduces the administrative burden on healthcare providers, allowing them to focus on more critical aspects of patient care.

In an increasingly data-driven healthcare environment, the Medical Prescription OCR system holds the potential to improve patient safety, enhance the quality of healthcare delivery, and streamline administrative processes. This project not only showcases technical expertise but also represents a meaningful contribution to the broader field of healthcare technology, with far- reaching implications for patient well-being and the efficiency of healthcare services.

1.2 Existing System

Existing systems in medical prescription recognition using machine learning leverage OCR technology and deep learning models like CNNs to digitize and interpret handwritten prescriptions accurately. These systems enhance efficiency in healthcare by speeding up prescription processing, and integrating seamlessly with electronic health records (EHR) systems.

1.3 Proposed system

The proposed system for medical prescription recognition employs LSTM and RNN architectures to better handle sequential dependencies and contextual information in handwritten text. This approach improves the accuracy of interpreting complex and varied handwriting styles, ensuring more reliable digitization and processing of medical prescription.

2 Requirement engineering

2.1 Hardware requirements

- Processor –Intel core i5 and above
- Memory – 8GB RAM (16GB recommended)
- Input devices – Keyboard Mouse
- Internet

2.2 Software requirements

- Operating System - Any OS capable of running Python and web browsers (Mac,
- Windows, Linux).
- Front-end - HTML, CSS.
- Libraries - OpenCV, Pillow, Flask.
- Languages - Python 3.x.
- Tools Required - Anaconda, Jupyter Notebook.

3 Literature Survey

The proposed model for medical prescription recognition begins by scanning the prescription with a mobile camera and proceeds through three main phases. In the pre-processing phase, the images are normalized, converted to black and white, and divided into three parts: the doctor's name, prescribed medicines and instructions, and the doctor's contact information. In the processing phase, the part containing prescribed medicines is extracted and analyzed using a Convolutional Neural Network (CNN). The CNN employs convolutional, ReLU, and max-pooling layers for feature extraction, followed by fully connected layers for classification. The convolutional layers detect features, ReLU layers introduce nonlinearity, and max-pooling reduces the input size to prevent overfitting. The pooled feature map is flattened for further processing in an Artificial Neural Network (ANN). In the post-processing phase, additional handwritten prescriptions are collected to improve training, and Optical Character Recognition (OCR) is applied if accuracy is below 50%. The OCR results are compared with a dataset of medicine names to identify the nearest match. This comprehensive approach aims to enhance the accuracy and reliability of identifying prescribed medicines and their dosages from medical prescriptions.[1]

The proposed model for medical prescription recognition involves scanning the prescription with a mobile camera and processing it through three main phases. In the pre-processing phase, images are normalized, converted to black and white, and divided into three sections: doctor's name, prescribed medicines and instructions, and doctor's contact information. The processing phase extracts the prescribed medicines and analyzes them using a Convolutional Neural Network (CNN) with convolutional, ReLU, and max-pooling layers for feature extraction, followed by fully connected layers for classification. Convolutional layers detect features, ReLU layers add nonlinearity, and max-pooling reduces input size to prevent overfitting. The pooled feature map is flattened and processed in an Artificial Neural Network (ANN). In the post-processing phase, additional handwritten prescriptions are collected to improve training, and Optical Character Recognition (OCR) is used if accuracy falls below 50%, with results compared to a medicine name dataset to find the closest match. This approach aims to improve the accuracy and reliability of identifying prescribed medicines and dosages from medical prescriptions.[2]

This research aims to develop an automatic verification system using deep learning to ensure prescription dispensing accuracy and reduce medication errors in pharmacies. The system includes two models: an image classification model using raw blister pack images processed with Histograms of Oriented Gradients (HOG) and a Convolutional Neural Network (CNN), and a text classification model that extracts and matches imprints on blister packs using CRAFT, Keras-OCR, and text correction. The dataset comprises 200 types of blister packs, each with 300 high-quality images taken in controlled conditions. The system's accuracy is determined by a majority vote between the models, achieving 95.83% accuracy for image classification and 92% for text classification, with an overall accuracy of 94.23%. [3]

Deciphering a doctor's handwritten prescription is a common challenge for patients and even some pharmacists, often leading to negative consequences due to misinterpretation. This difficulty arises partly because doctors use Latin abbreviations and medical terminology unfamiliar to most people. This paper demonstrates the development of a system that uses Artificial Neural Networks (ANN) to recognize handwritten English medical prescriptions. By employing a Deep Convolution Recurrent Neural Network for training, the supervised system segments input images and processes them to detect and classify characters into 64 predefined categories. The results indicate that the proposed system achieves high recognition rates with an accuracy of 98%.[4]

Doctors often write prescriptions hastily and in illegible handwriting, posing challenges for patients and pharmacists who must interpret them accurately. These prescriptions frequently include abbreviations, cursive writing, and even regional languages, complicating understanding further. This project aims to develop a recognition system that translates physicians' handwritten prescriptions across various languages. The system will function autonomously as an application, processing uploaded prescription images through initial image pre-processing and word segmentation. Deep learning techniques such as CNN, RNN, and LSTM will be employed to train the model for each language, utilizing Unicode for character encoding. The system will integrate fuzzy search and market basket analysis to optimize results from a pharmaceutical database, presenting structured outputs to users for clarity and accuracy in medication dispensation. [5]

4 Technology

4.1 About Python

Python's environment has evolved significantly, enhancing its capabilities for statistical analysis. It strikes a fine balance between scalability and elegance, placing a premium on efficiency and code readability. Python is renowned for its emphasis on program readability, featuring a straightforward syntax that is beginner-friendly and encourages concise code expression through indentation. Noteworthy aspects of this high-level language include dynamic system functions and automatic memory management.

Python is used for:

- Web development
- Data science and machine learning
- Artificial intelligence
- Scientific computing
- Desktop GUI applications
- Automation and scripting
- Game development
- Networking
- Healthcare

4.2 Python is widely used in Machine Learning

Python is widely favored in machine learning for its flexibility and open-source nature. It provides extensive functionality for mathematical computations and scientific operations, making it indispensable in developing and deploying machine learning models. Python's simple syntax and vast libraries accelerate the development process, reducing coding time significantly. This makes it a preferred choice for machine learning practitioners seeking efficiency and robustness in their projects.

The major Python libraries used in machine learning are as follows:

4.2.1 Pandas

Pandas is a Python library used for statistical analysis, data cleaning, exploration, and manipulation. Typically, datasets contain both useful and extraneous information. Pandas helps to make this data more readable and relevant.

4.2.2 Numpy

NumPy is a Python library utilized for numerical data reading, cleaning, exploration, and manipulation. It provides powerful data structures for efficient computation with large arrays and matrices, making the data more accessible and manageable.

4.2.3 Matplotlib

Matplotlib is a Python library for plotting graphs. Built on NumPy arrays, it allows for the creation of a wide range of graph types, from basic plots to bar graphs, histograms, scatter plots, and more.

4.2.4 Scikit-learn

Scikit-learn is a Python library for machine learning. It provides tools for machine learning and statistical modeling, including classification, regression, and clustering.

4.2.5 Tensorflow & Pytorch

Essential libraries for deep learning, used to create and deploy neural networks. They provide robust tools for developing complex models and facilitating machine learning workflows.

4.2.6 Interpreted Language

Python executes code line by line, without the need for prior compilation. This approach facilitates quicker development cycles and simplifies the debugging process. As a result, developers can iterate and test their code more efficiently.

4.2.7 Cross-Platform Compatibility

Python code runs seamlessly on multiple operating systems, including Windows, macOS, Linux, and Unix-based systems, without requiring modifications. This versatility ensures that Python applications can be deployed across diverse environments with ease.

4.2.8 YOLO v5 Model

YOLO v5 is an upgraded version of the YOLO real-time object detection system, known for its speed and accuracy in identifying objects within images and videos.

It is widely used in applications requiring real-time object detection such as autonomous vehicles, surveillance systems, image captioning, and augmented reality.

How YOLO Works:

Unlike traditional object detection methods that analyze image regions multiple times, YOLO takes a single pass through the image.

1. **Image Division:** The input image is divided into a grid of cells.
2. **Bounding Box & Class Prediction:** For each cell, YOLO predicts the probability of an object existing within that cell and its bounding box coordinates (location and size). It also predicts the class of the object (e.g., car, person, dog).

4.2.9 RNN

RNNs are a type of artificial neural network specifically designed to handle sequential data. Unlike traditional neural networks that process each piece of data independently, RNNs can utilize their internal state (memory) to process information based on preceding elements in the sequence.

How RNN Works:

1. **Input & Hidden State:** The network receives an input from the sequence and combines it with the current hidden state.
2. **Activation Function:** The combined information is passed through an activation function that determines the output of the current step.
3. **Updated Hidden State:** The hidden state is updated based on the current input, previous

hidden state, and the activation function's output.

4. **Sequence Processing:** These steps (1-3) are repeated for each element in the sequence, allowing the network to learn long-term dependencies.

4.2.10 Dataset description

The dataset selected for training our model consists of approximately 1 GB of IAM dataset for handwritten text recognition. This dataset includes 165 handwritten prescriptions. The IAM dataset is well-suited for training Recurrent Neural Network (RNN) models to accurately read and interpret text from handwritten prescriptions.

Dataset Details:

1. Size: Approximately 1 GB.
2. Number of prescriptions: 165.

Why IAM Dataset…?

1. The dataset provides ample data to train RNN models effectively.
2. Annotations and labels within the dataset are meticulously organized.
3. The dataset is freely accessible, facilitating cost-effective research and development.

5 Design requirement engineering

5.1 UML diagrams

The purpose of these UML-based diagrams is to visually depict the system, including its key components, roles, operations, objects, or interactions. This visual representation aims to enhance understanding, facilitate manipulation, and effectively document or manage system-related information.

The Unified Modeling Language (UML) serves as a standardized language for creating models across various domains. Its primary goal is to visually represent the structure of a system, akin to blueprints in engineering disciplines. In complex applications involving multiple teams, clear communication is crucial, especially to stakeholders who may not be familiar with programming code. UML facilitates this communication by illustrating essential system requirements, features, and processes in a visual manner. By depicting processes, user interactions, and the system's static structure, UML helps teams streamline collaboration and optimize efficiency.

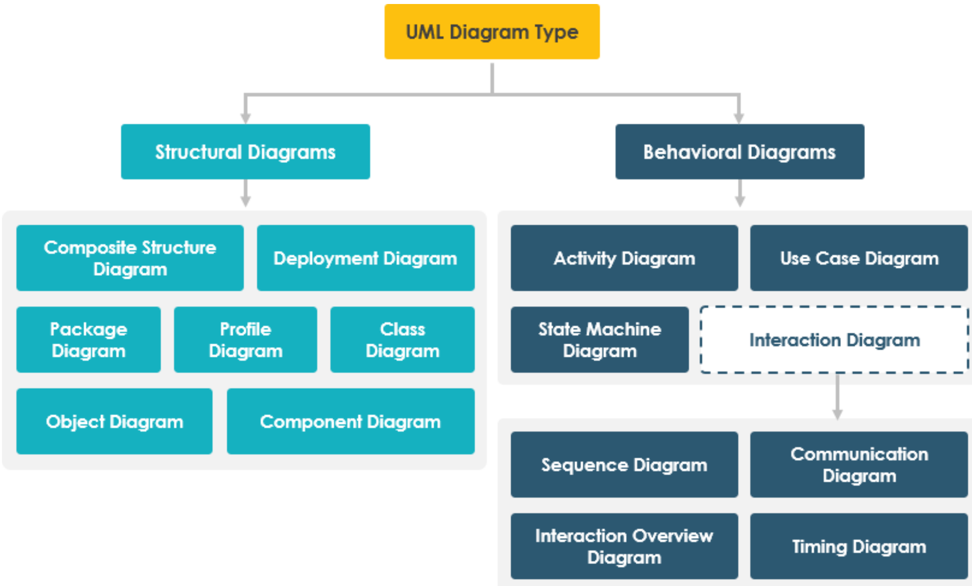


Fig. 1. Overview of types of UML diagram

5.1.1 Use case Diagram

The Use Case diagram for the Medical Prescription Recognition system illustrates the interactions between doctors, pharmacists, and the system. It highlights key functionalities such as uploading prescriptions, preprocessing images, recognizing and validating text, storing data, and retrieving and generating reports from the database.

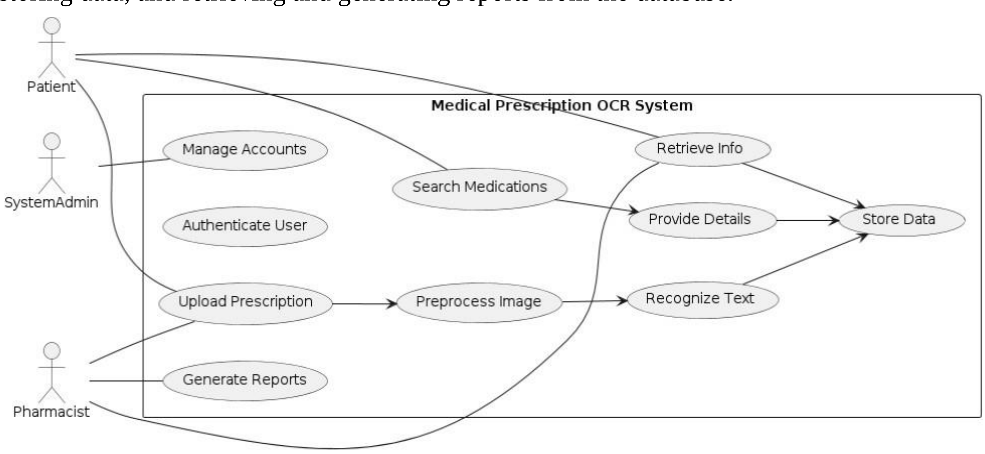


Fig. 2. Use Case Diagram

5.1.2 Class Diagram

A class diagram is a static type of structural diagram that visually represents the architecture of a system by illustrating the connections among the system's classes, attributes, operations, and relationships. It provides a blueprint of how different components of the system interact and collaborate, showcasing the structure in a clear and organized manner.

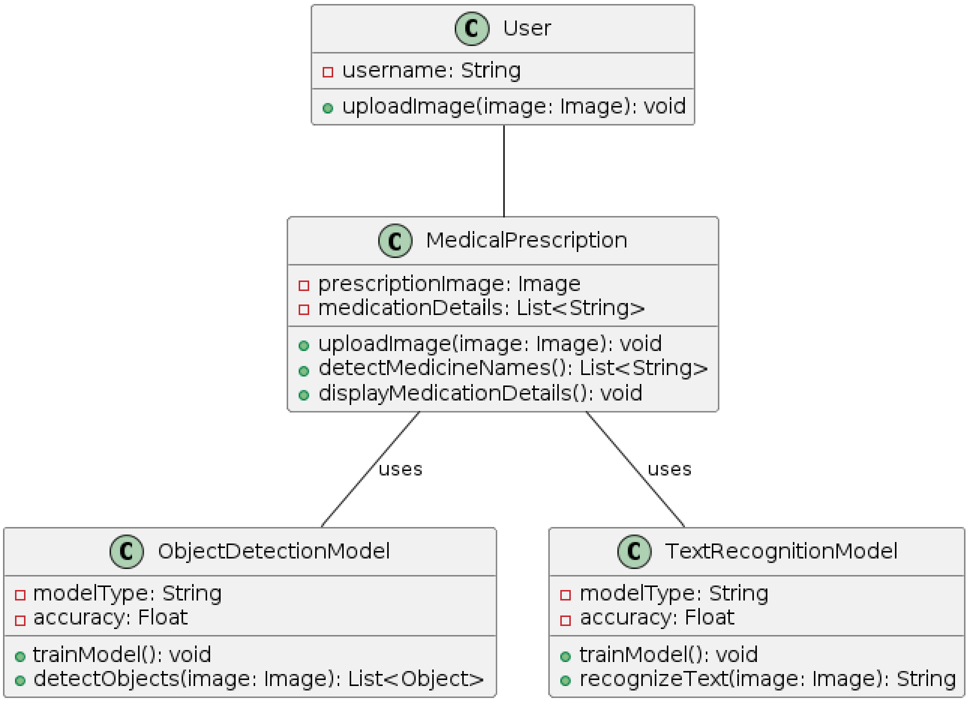


Fig. 3. Class diagram

5.1.3 Sequence diagram

A sequence diagram illustrates interactions between objects in a sequence over time, showing the messages exchanged between them. It visually represents the flow of control and data between objects in a system, emphasizing the order of events and the lifeline of each participating object.

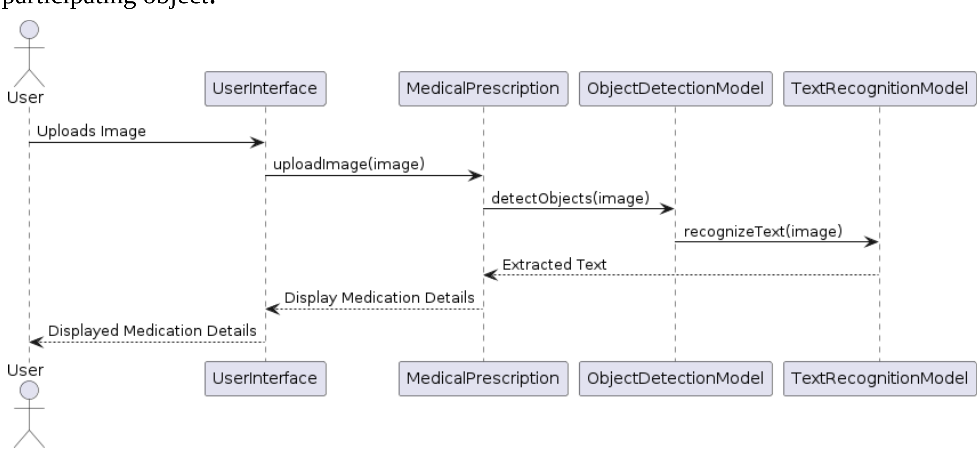


Fig. 4. Sequence diagram

5.1.4 Activity diagram

An activity diagram in UML illustrates the sequence of actions and decisions within a system or process. It shows how activities interact and flow from start to finish, making it easy to understand and analyze complex workflows and business processes.

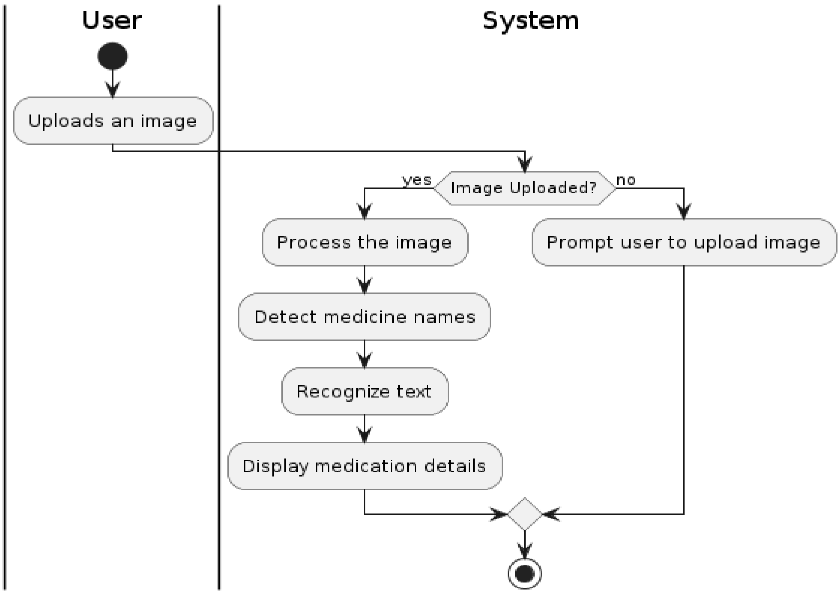


Fig. 5. Activity diagram

5.2 SYSTEM ARCHITECTURE

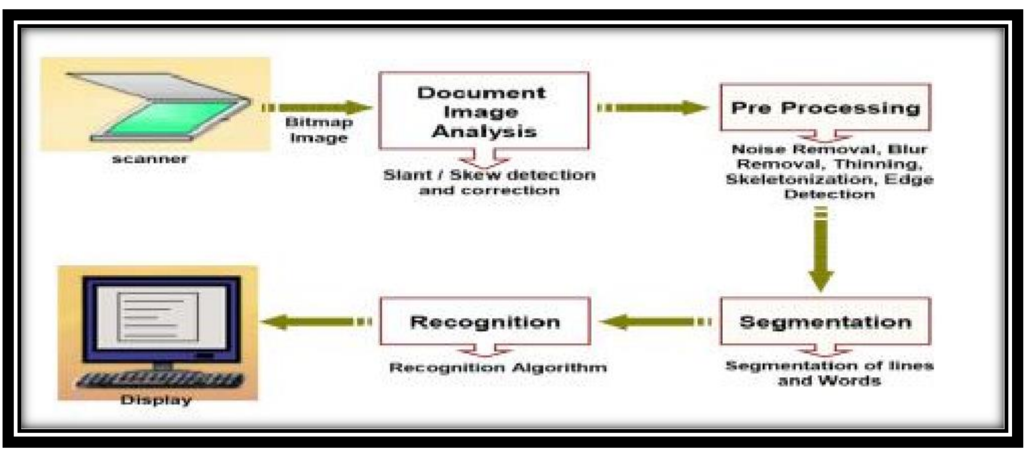
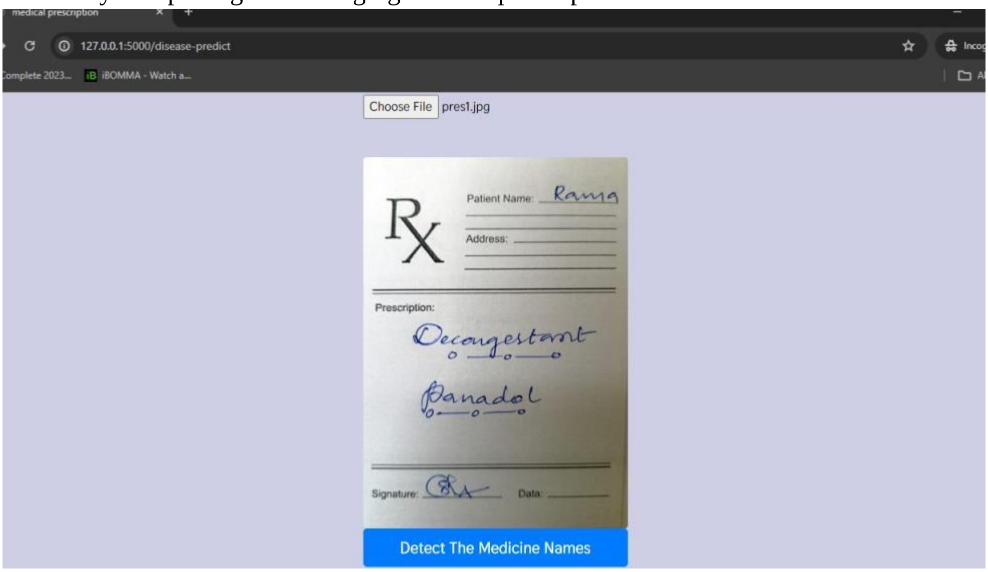
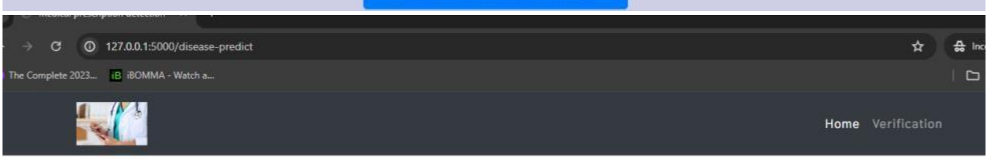


Fig. 6. System Architecture

7 RESULTS

Accuracy measures are crucial for evaluating our Medical Prescription OCR system. Key metrics include accuracy, precision, recall, F1 score, and AUC-ROC. Accuracy indicates the proportion of correctly interpreted prescriptions. Precision measures the true positives among predicted positives, while recall assesses the system's ability to identify all relevant medicine names. The F1 score balances precision and recall, and AUC-ROC evaluates the model's class differentiation ability. These metrics ensure the OCR system's effectiveness in accurately interpreting and managing medical prescriptions.





The Medicine Detected are As Follows:

Medicine Name		Company Name	Price
	medicine_name	company_name	price
0	Decongestant	Nexafed Nasal Decongestant	64.54
1	Panadol	Mersyndol Forte	20.43

8 CONCLUSION AND FUTURE ENHANCEMENTS

The Medical Prescription OCR system successfully automated prescription interpretation with high accuracy using YOLOv5 and deep learning models.

Future enhancements include refining accuracy, integrating NLP for complex instructions, enabling real-time processing, integrating with EHR systems, personalizing medication recommendations, and establishing a user feedback mechanism.

9 Reference

[1] https://ieeexplore.ieee.org/document/9609390

[2] https://www.semanticscholar.org/paper/Medical-Handwritten-Prescription-Recognition-Using-Achkar-Ghayad/3fff5b7c44431728c6cd36b7a164697c02d83acc

[3] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9509260/

[4] https://colab.ws/articles/10.1109%2FCCWC51732.2021.9376141?utm_source

[5] https://arxiv.org/abs/2210.11666?utm_source

[6] https://www.researchgate.net/publication/350150845_Medical_Prescription_Recognition_using_Machine_Learning?utm_source