AI-Based Patient Appointment Scheduling System for Veterinary Clinics

Abstract

This project presents an AI-driven appointment scheduling system designed for veterinary clinics. It intelligently manages appointments based on patient condition urgency (emergency or normal) and doctor availability. The system prioritizes emergency cases by automatically rearranging the existing schedule and ensuring optimal use of resources. It integrates multiple modules, including AI-based emergency detection, doctor availability management, appointment rescheduling, and notification services. The project demonstrates how artificial intelligence can streamline healthcare scheduling and enhance patient response efficiency.

Problem Statement

In veterinary clinics, managing patient appointments efficiently is challenging due to unpredictable emergency cases. Traditional systems lack dynamic scheduling and often result in delays or missed treatments. This project addresses the problem by developing an intelligent appointment scheduling system that automatically prioritizes cases based on urgency and doctor availability.

Objectives

- To automate appointment scheduling using AI.
- To prioritize emergency cases dynamically.
- To suggest alternative doctors when the preferred doctor is unavailable.
- To notify both doctors and patients about schedule updates automatically.
- To maintain an optimized timetable for all appointments.

System Architecture

The system architecture follows a modular design consisting of interconnected Python modules. Each module performs a specific task in the appointment management process:

Module	Description
ai_emergency_detector.py	Analyzes patient condition data to determine if a case is an emergency.
doctor_availability_manager.py	Checks the real-time availability of doctors and suggests alternatives.
appointment_swapper.py	Reschedules appointments automatically if an emergency case occurs.
notification_service.py	Sends updates to both patients and doctors regarding appointment changes.
admin_interface.py	Allows administrators to monitor, approve, or modify the schedule.
database.py	Stores doctor details, appointments, and patient records securely.

Workflow

1. A patient (pet owner) requests an appointment through the system.
2. The AI Emergency Detector analyzes the patient’s condition to classify it as normal or emergency.
3. If the case is an emergency, the Appointment Swapper reallocates time slots, prioritizing the emergency case.
4. If a doctor is unavailable, the Doctor Availability Manager suggests other doctors.
5. Notifications are sent to both doctors and patients through the Notification Service.
6. The Admin Interface provides an overview and allows manual overrides when needed.

Implementation Details

The implementation was done using Python 3, following a modular structure. The *ai_emergency_detector.py* module uses condition keywords and symptom-based analysis to categorize cases. The *appointment_swapper.py* uses logic to identify available time slots and swap appointments when necessary. A SQLite database (handled by *database.py*) maintains persistent data for patients and doctors. The system can be deployed as a Flask web application or used as a command-line scheduler.

Results and Discussion

The AI-Based Appointment Scheduling System demonstrated efficient handling of emergency cases by automatically rescheduling appointments in real-time. Emergency patients received faster attention while maintaining minimal disruption to other appointments. Testing showed that the system effectively balanced emergency and normal cases while reducing administrative overhead.

Conclusion and Future Scope

This project successfully implements an AI-based solution for intelligent veterinary appointment scheduling. It dynamically manages priorities, reallocates appointments, and provides real-time notifications. Future improvements may include integrating predictive analytics for workload forecasting and adding a mobile app for instant patient-doctor communication.