Eco-Smart Waste Management System

Project Document

Detailed Write-up of Solution

Our solution addresses the critical issue of inefficient waste collection by implementing an IoT-enabled smart bin system. Each bin is equipped with ultrasonic sensors to measure fill levels and a LoRaWAN module for data transmission. This data is sent to a cloud-based platform (AWS IoT Core) which uses a machine learning model (Random Forest Classifier) to predict optimal collection routes and schedules for waste trucks. The system also includes a mobile application for residents to report overflowing bins and view collection schedules.

Tools & Technologies

- **Hardware:** ESP32 microcontrollers, Ultrasonic sensors (HC-SR04), LoRaWAN modules (SX1276)
- **Cloud Platform:** AWS IoT Core, AWS Lambda, Amazon S3, Amazon RDS (PostgreSQL)
- **Machine Learning:** Python (Scikit-learn, Pandas, NumPy), Jupyter Notebooks for model training
- **Mobile App:** React Native
- **Version Control:** Git, GitHub

Logic

1. Sensors collect fill-level data.
2. Data is transmitted via LoRaWAN to AWS IoT Core.
3. Lambda function processes data, stores it in RDS, and triggers the ML model.
4. ML model predicts optimal routes based on fill levels, historical data, and traffic.
5. Optimized routes are displayed on a web dashboard for sanitation departments and pushed to the mobile app.

Challenges Anticipated and Mitigation Strategies

- **Challenge:** Sensor accuracy in varying weather conditions.
- **Mitigation:** Employing multiple sensors per bin and implementing kalman filters for data smoothing.
- **Challenge:** Data transmission reliability in dense urban areas.
- **Mitigation:** Deploying additional LoRaWAN gateways and utilizing adaptive data rate (ADR).

Timeline or Roadmap for Development

- **Month 1-2:** Hardware prototyping, sensor integration, initial LoRaWAN setup.
- **Month 3-4:** Cloud platform development, database design, ML model training (initial).
- **Month 5-6:** Mobile app development, system integration, pilot testing in a small community.
- **Month 7-8:** Feedback iteration, model refinement, scaling.

System Architecture and Technology Stack

- **Sensors:** Ultrasonic sensors connected to ESP32 microcontrollers.
- **Connectivity:** LoRaWAN network for long-range, low-power communication.
- **Cloud Backend:** AWS (IoT Core for ingestion, Lambda for processing, RDS for database, S3 for data lake, EC2 for ML inference).
- **Machine Learning:** Python-based ML model deployed on AWS SageMaker.
- **Frontend:** React Native for mobile, React.js for web dashboard.

Data Sources and How Data Privacy is Handled

- **Data Sources:** Ultrasonic sensor readings (fill level), GPS data from waste trucks, historical waste collection data.
- **Data Privacy:** All sensor data is anonymized and aggregated. GPS data is only used for route optimization and is not linked to individual drivers. User data from the mobile app (e.g., reported issues) is handled with strict access controls and encrypted at rest and in transit. We adhere to GDPR principles.

Main Features and User Flow

Features

- Real-time bin fill-level monitoring.
- Dynamic route optimization for waste collection.
- Resident mobile app for reporting and viewing schedules.
- Web dashboard for waste management authorities.

User Flow (Resident)

1. Opens mobile app.
2. Views nearby bin status and collection schedule.
3. Reports an overflowing bin (optional).
4. Receives notifications about updated schedules.

User Flow (Waste Management Authority)

1. Logs into web dashboard.
2. Views map of all bins and their fill levels.
3. Receives optimized collection routes.
4. Monitors truck progress and collection efficiency.