AI-Driven Predictive Maintenance and Anomaly Detection for Autonomous Vehicles

1. Introduction

This document outlines the architecture and implementation of an AI-based system for predictive maintenance and anomaly detection in autonomous vehicles. The solution integrates Natural Language Processing (NLP), Computer Vision (CV), Generative AI, and Physics-Informed Neural Networks (PINNs), leveraging Google Cloud Platform (GCP) for deployment and scaling.

---

2. High-Level Design (HLD)

2.1 System Overview

The system is designed to process vehicle sensor data and diagnostic logs for predictive maintenance, detect anomalies using computer vision, generate realistic 3D scenes, and simulate vehicle dynamics.

2.2 Key Components

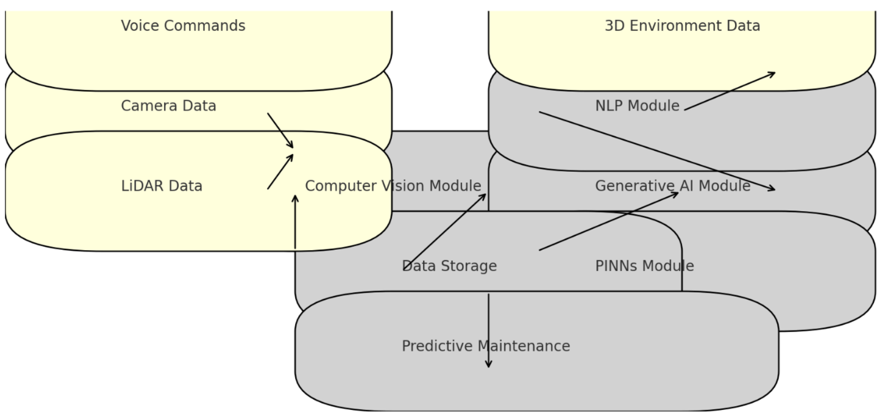
- **Data Sources**:
    - **Vehicle Sensors**: Collect real-time telemetry, such as temperature, pressure, and motion data.
    - **Diagnostic Logs**: System logs from the vehicle's onboard systems.
- **Data Ingestion Pipeline**:
    - Preprocessing and ingestion of sensor data and diagnostic logs into GCP's storage.
- **Data Processing**:
    - **NLP**: Analyzes diagnostic logs and user queries for summarization.
    - **Computer Vision (CV)**: Detects anomalies and tracks objects using multi-sensor fusion.
    - **Generative AI**: Generates 3D scenes of driving environments for simulations.
    - **PINNs**: Simulates vehicle dynamics to predict mechanical issues.
- **Data Aggregation**:

- ○ Aggregates insights from various components and sends them for display.
- **User Interface (UI)**:
  - ○ A dashboard for real-time monitoring of vehicle health and recommendations for maintenance.

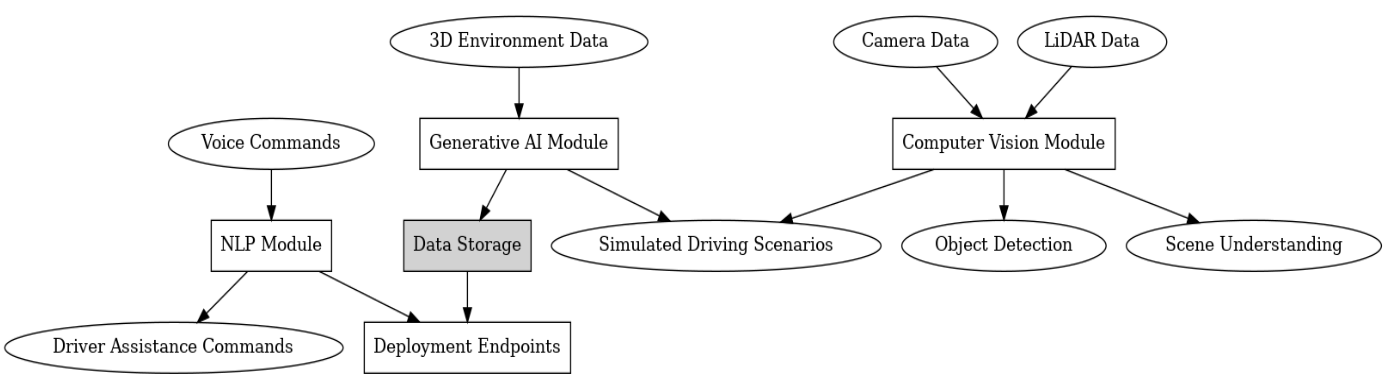
2.3 Data Flow

1. Sensors and logs capture vehicle data.
2. The ingestion pipeline processes the data.
3. Different AI models analyze and process the data, generating insights.
4. Insights are displayed on the dashboard for the user.

Architecture Diagram



Workflow Diagram



3. Low-Level Design (LLD)

3.1 Data Ingestion Pipeline

- **GCP Services Used**:
  - **Cloud Storage**: Stores raw data from sensors and logs.
  - **Pub/Sub**: Streams real-time data.
  - **Cloud Dataflow**: Processes data for NLP and CV tasks.

3.2 NLP Component

- **Model**: Fine-tune a large language model (e.g., GPT-3 or PaLM) to process diagnostic logs and user queries.
- **Process**:
  - Ingest log data and queries.
  - Run through the LLM for analysis.
  - Output diagnostic summaries and recommendations.

3.3 Computer Vision Component

- **Model**: Use YOLOv5 for object detection and anomaly detection with few-shot learning.
- **Process**:
  - Process visual data from vehicle cameras.
  - Detect anomalies in vehicle components.
  - Track objects for navigation.

3.4 Generative AI Component

- **Model**: Stable Diffusion Model for generating 3D driving environments.
- **Process**:
  - Use sensor data to simulate driving conditions.
  - Generate 3D scenes for vehicle motion planning.

3.5 Physics-Informed Neural Networks (PINNs)

- **Model**: Neural networks for simulating vehicle dynamics (e.g., CFD, FEA).
- **Process**:
  - Use historical data to predict component wear.

    - Model fluid dynamics and stress in vehicle parts for predictive maintenance.

3.6 Dashboard and User Interface

- **Technology**:
    - **Cloud Functions**: Serve data to the UI.
    - **Frontend**: Build with Angular/React for an interactive dashboard.
    - **Backend**: Node.js/Flask APIs to fetch processed data from AI models.

---

4. Tools and Technologies Used

- **Data Ingestion**: Pub/Sub, Cloud Dataflow
- **Storage**: Cloud Storage, BigQuery
- **NLP**: PaLM, GPT-3, Vertex AI
- **Computer Vision**: YOLOv5, Vertex AI Workbench
- **Generative AI**: Stable Diffusion Models
- **PINNs**: TensorFlow, Vertex AI
- **Deployment and Scalability**: GCP (Vertex AI, Cloud Functions, App Engine)
- **Frontend/UI**: Angular/React
- **Backend**: Node.js, Flask

---

5. Conclusion

This architecture provides a robust solution to autonomous vehicle maintenance, integrating advanced AI techniques to detect anomalies, predict failures, and simulate real-world conditions. GCP services ensure scalability and real-time insights.