AI/ML for OEMs

AI-Driven Predictive Maintenance for Connected Vehicles

—

17 November 2024

Shaurya Saxena
Atishi Agrawal

Team Name -
ACES(shauryaa2412)

Problem Statement

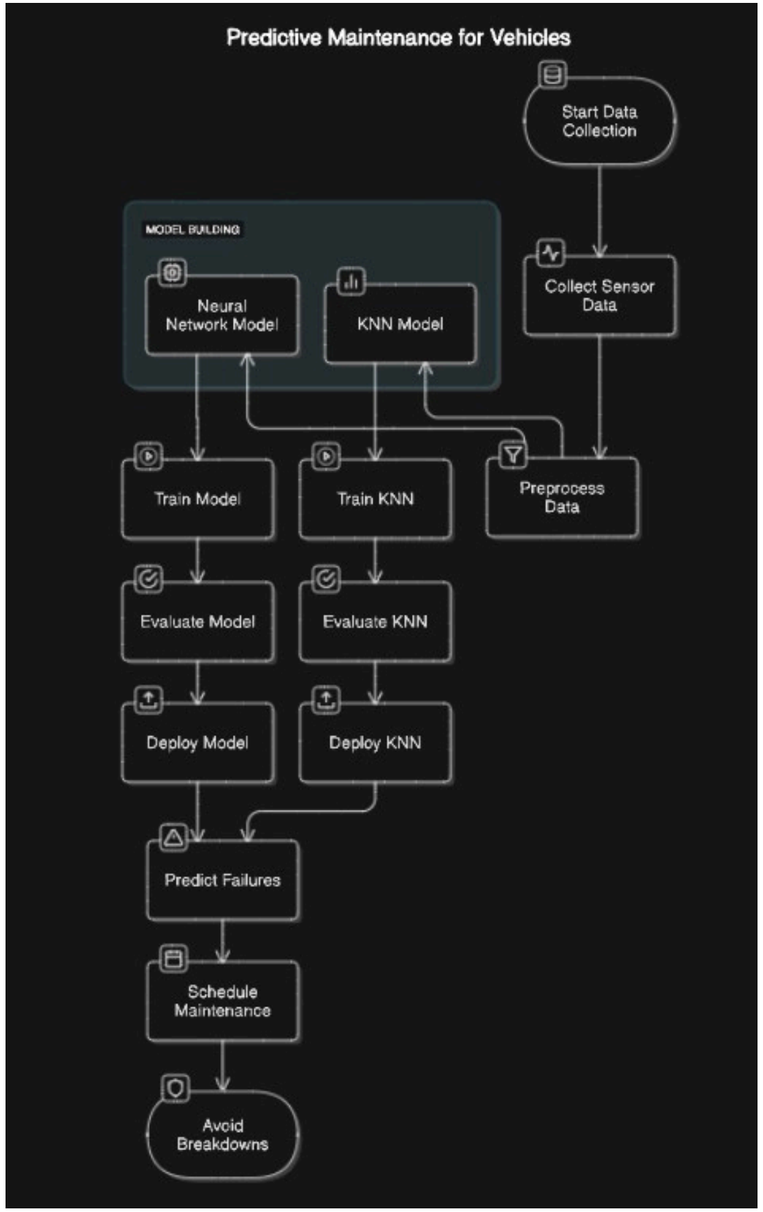
- Vehicle breakdowns are a major issue for both Original Equipment Manufacturers (OEMs) and vehicle owners.
- Unplanned downtimes result in costly repairs, reduced vehicle lifespan, and lower customer satisfaction.
- Current maintenance systems rely on scheduled checks and inspections, which do not account for real-time wear and tear or the unique usage patterns of individual vehicles.
- This leads to missed early warnings and unexpected breakdowns, affecting overall reliability and increasing costs.

Proposed Solution

- We propose an AI-driven predictive maintenance system that utilizes real-time data from various vehicle sensors, such as engine, transmission, brakes, and suspension.
- Our model continuously learns from each vehicle's data to predict when a component is likely to fail, enabling proactive maintenance.
- The system notifies both the vehicle owner and the OEM about upcoming service needs, preventing breakdowns, reducing repair costs, and extending the lifespan of the vehicle.
- This solution provides a more tailored approach to maintenance based on actual vehicle usage rather than fixed schedules.

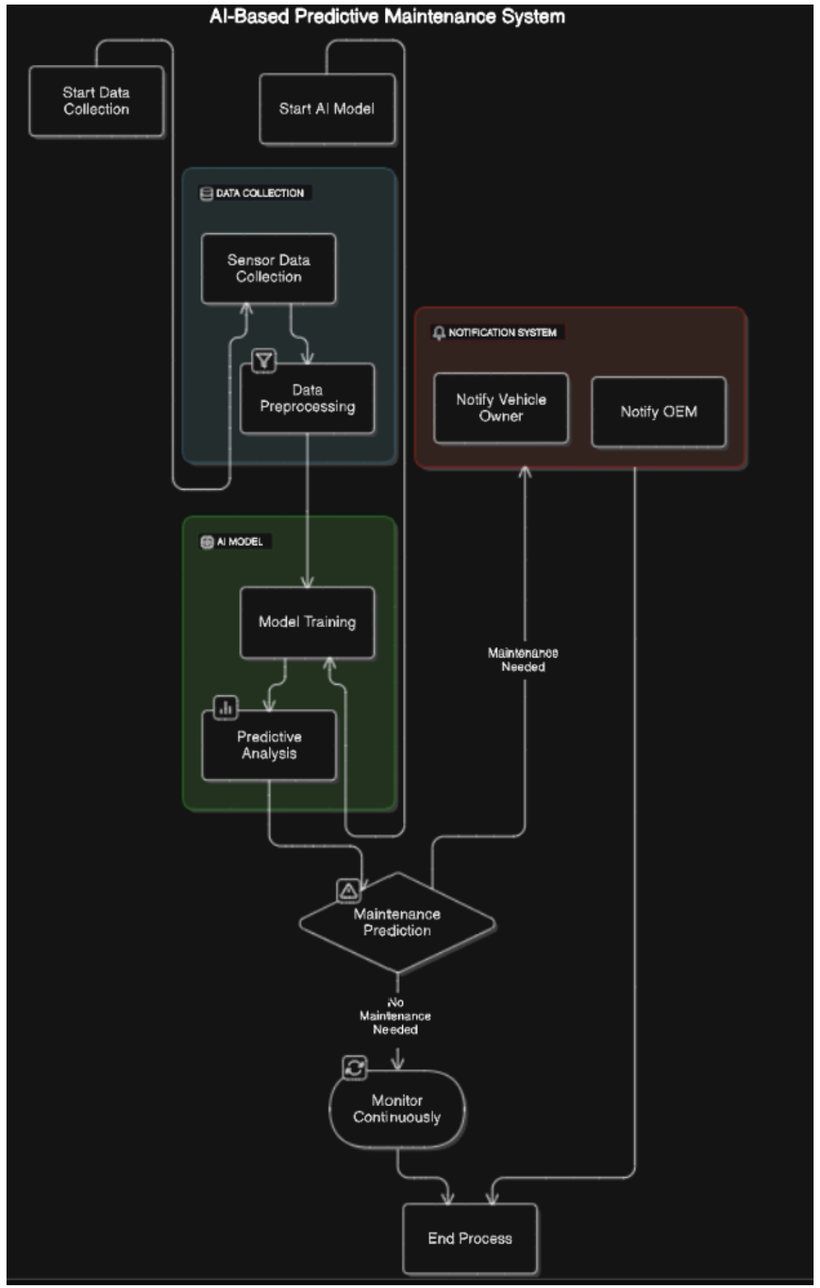
Product Features

- **Real-time Analysis**: Continuously monitors sensor data from key vehicle components.
- **AI-based Predictions**: Uses machine learning models to detect patterns and predict potential part failures before they occur.
- **User Notifications**: Alerts the vehicle owner and OEM about upcoming maintenance needs, providing actionable insights.
- **Health Dashboard**: Displays the overall condition of the vehicle, highlighting any parts that need immediate attention.
- **Customizable Alerts**: Users can set preferences for alert frequency and criticality levels based on their needs.



User Flow of the Prototype

- **Data Collection**: Sensor data from vehicle components (engine, transmission, brakes, suspension) is collected in real-time.
- **Preprocessing**: The raw data is cleaned, normalized, and prepared for analysis.
- **Prediction**: The AI model analyzes the processed data and predicts potential failures.
- **Notification**: Alerts are generated and sent to the vehicle owner and OEM with recommended maintenance actions.
- **User Interface**: The predictions and alerts are displayed on a user-friendly dashboard, allowing users to view the health status of the vehicle at a glance.

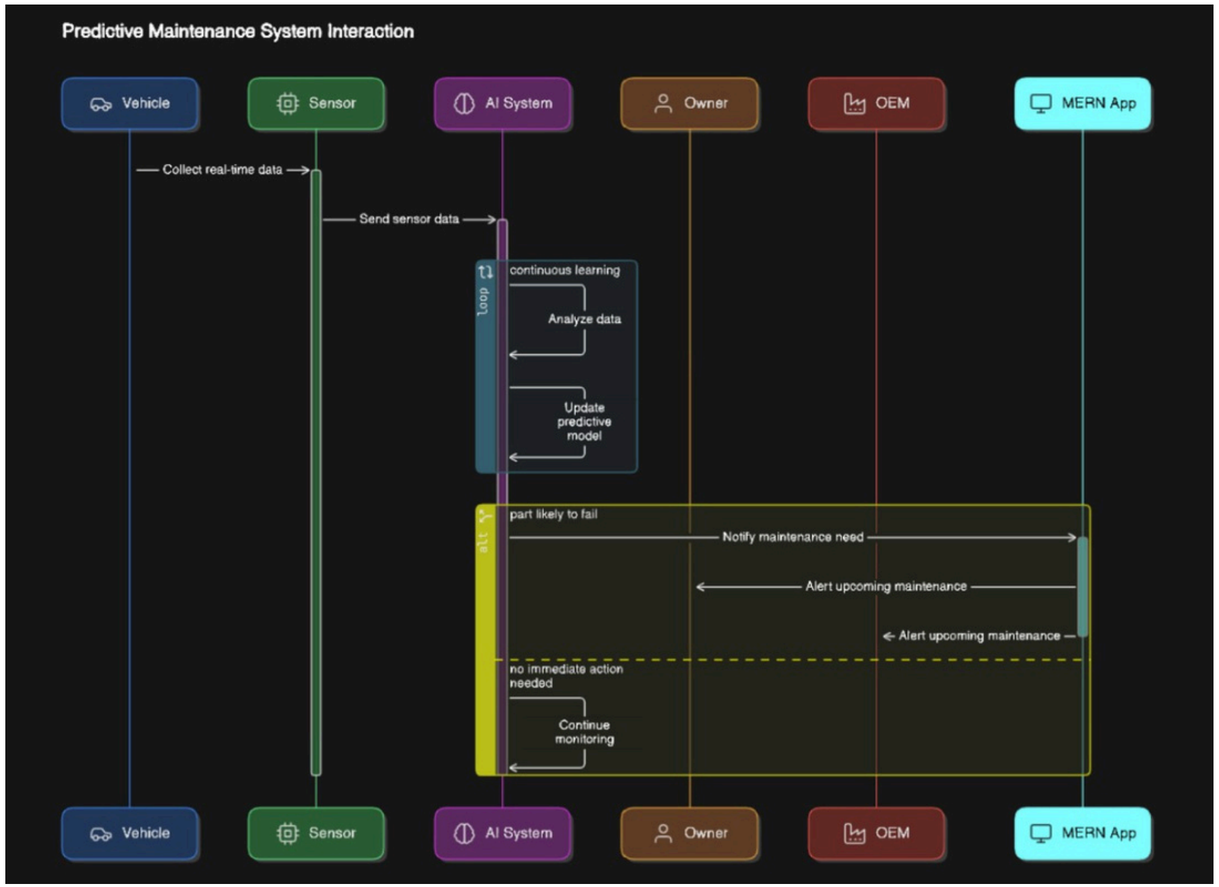


System Architecture

The architecture consists of the following components:

- **Data Sources**: Real-time sensor data from vehicle components (engine temperature, oil pressure, brake wear, etc.).
- **Data Preprocessing**: Cleans and standardizes the data for analysis.
- **AI Model**: The core predictive model, combining a neural network (for complex pattern recognition) and a K-Nearest Neighbors (KNN) model (as a baseline).
- **Alert System**: Generates notifications based on the model's predictions.
- **User Interface**: A web-based dashboard displays vehicle health, predicted failures, and recommended actions.
- **Data Storage**: Historical sensor data is stored for model training and continuous improvement.

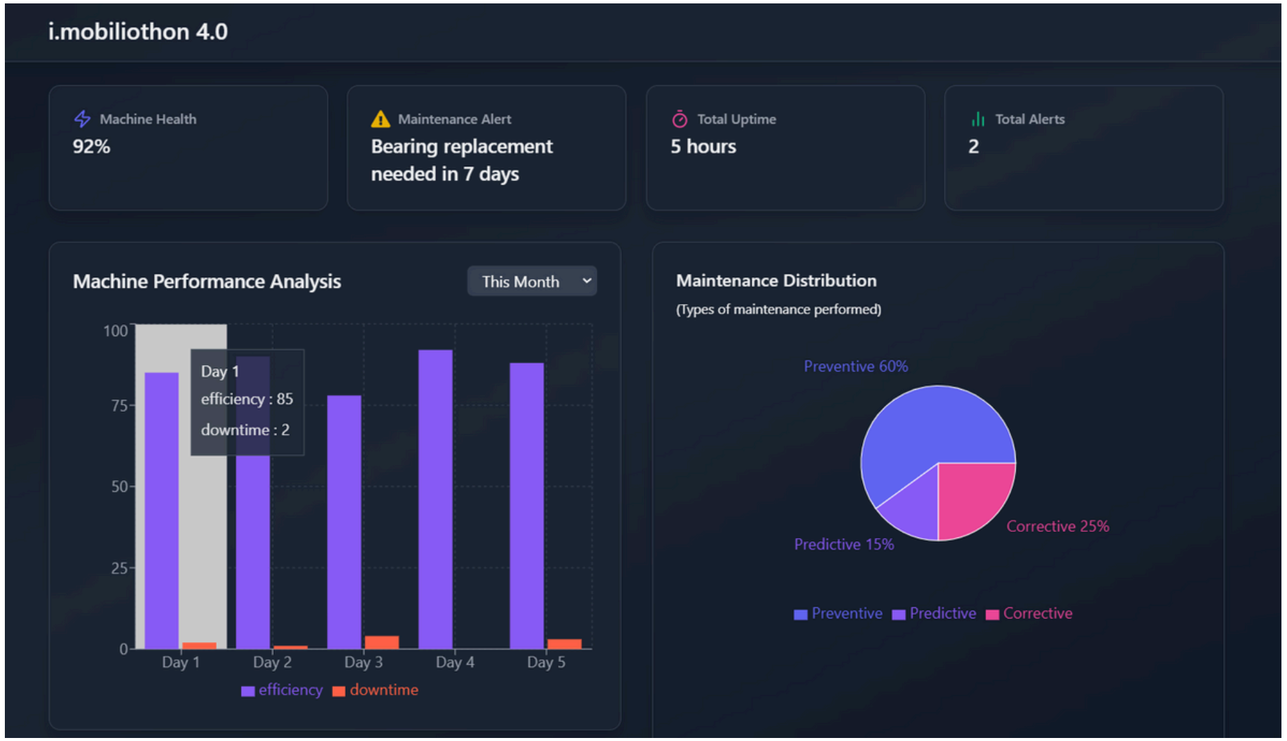
AI Models Used



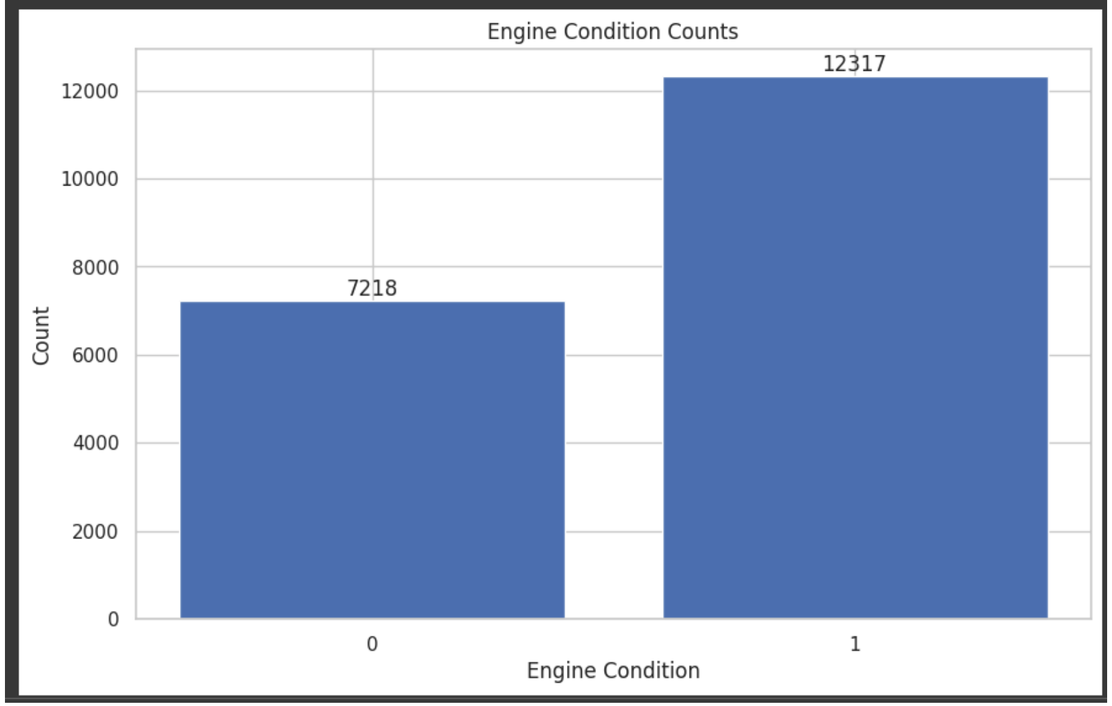
We implemented two machine learning models for predicting part failures:

1. **Neural Network Model (PyTorch)**: This deep learning model is designed to capture complex relationships in the data. It learns from historical sensor data and can generalize to different vehicle conditions.
- **Architecture**: Consists of an input layer, multiple hidden layers, and an output layer for predicting component failure.
- **Advantages**: High accuracy in recognizing complex, non-linear patterns in the data.
2. **K-Nearest Neighbors (KNN) Model (sklearn)**: This is a simpler, instance-based algorithm that classifies new data points based on their similarity to past examples.
- **Advantage**s: Provides a quick and interpretable baseline model for comparison.
- **Comparison**: The neural network outperforms KNN in handling large, diverse datasets.

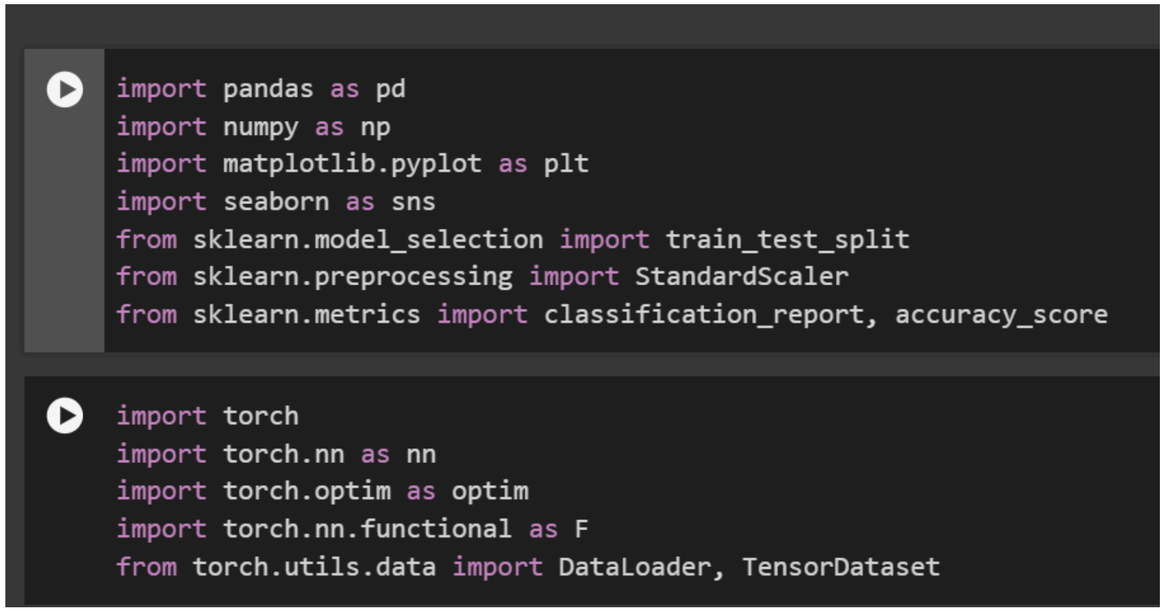
Working Prototype Demo




```python
plt.bar(conditions, counts)

plt.title('Engine Condition Counts')
plt.xlabel('Engine Condition')
plt.ylabel('Count')

# Add value labels on top of each bar
for i, count in enumerate(counts):
    plt.text(i, count, str(count), ha='center', va='bottom')

plt.show()
```



Required Sdk and Tools



```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import classification_report, accuracy_score
```

```
import torch
import torch.nn as nn
import torch.optim as optim
import torch.nn.functional as F
from torch.utils.data import DataLoader, TensorDataset
```

- **PyTorch**: For building and training the neural network model.
- **sklearn**: For implementing and evaluating the KNN model.
- **Pandas & NumPy**: For data manipulation and preprocessing.
- **MERN**: For website creation and user friendly interface
- **Matplotlib & Seaborn**: For data visualization and exploratory analysis.
- **Flask/Django**: Backend framework for handling requests and serving the user interface.
- **Plotly/Dash**: For creating an interactive dashboard to display predictions and alerts.
- **Vehicle Telematics SDK**: To simulate real-time sensor data for testing purposes.

Conclusion and Future Work

Our **AI-based predictive maintenance system** offers a proactive solution for vehicle maintenance, reducing unplanned downtimes and extending vehicle lifespan. By leveraging real-time sensor data, the system can accurately predict potential part failures, providing timely alerts to both vehicle owners and OEMs. Future enhancements include:

- **Integration with Actual Vehicle Telematics**: Real-world data integration for improved predictions.
- **Advanced Model Tuning**: Further optimization of the neural network for higher accuracy.
- **Mobile App Interface**: Development of a mobile app for easy access to maintenance alerts and dashboard insights.
- **Scalability**: Expanding the system to support a wider range of vehicle types and sensor data.

Thank You