DriveX: Intelligent System For Managing Vehicle Service Using Predictive Maintenance and Smart Scheduling

Rishabh Tyagi [1], Taksh Tyagi [2], Pushpam Tyagi [3], Rajeev Rathore [4], Nitin Goyal [5]

1,2,3,4 – Student, Department of CSE, R.D Engineering College, Ghaziabad, Uttar Pradesh, India

5 – Assistant Professor, Department of CSE, R.D Engineering College, Ghaziabad

Abstract

The increasing complexity of modern vehicles and the rise of embedded sensors have created a strong need for smart, proactive maintenance management. Traditional automotive service systems, which are mainly reactive or schedulebased, lead to unexpected component failures, inefficient workshop operations, and lower customer satisfaction. This paper introduces DriveX, a combined intelligent framework that brings together predictive maintenance, smart scheduling, and centralized service management. The system gathers diagnostic data from OBD-II interfaces and IoT sensors, uses SMOTE-based oversampling to fix a significant class imbalance (16.3% failure rate), and applies ensemble machine learning methods, including Random Forest and XG Boost. This approach achieves 94.7% accuracy and a ROC-AUC of 0.97. Deployment in two active service centers over three months resulted in a 31% drop in unexpected breakdowns, a 22% faster service turnaround, and a noticeable increase in customer satisfaction.

Keywords: Predictive Maintenance, Machine learning, Vehicle Health Monitoring, Automotive Service Management, Anomaly detection, OBD-II Diagnostics, SMOTE, Smart Scheduling

I. Introduction

The fast growth of automotive technology and the higher use of embedded sensors in today's vehicles have changed how we monitor vehicle health and performance. However, many auto repair shops still depend on reactive maintenance approaches and fixed service schedules. These methods often lead to unplanned component failures, higher repair bills,poor workshop management, and lower customer satisfaction. This creates a need for smart, data-driven service management systems that can promote proactive maintenance and efficient planning.

Recent research on predictive maintenance and anomaly detection in industrial systems shows that machine learning can effectively spot unusual patterns in sensor data before serious failures happen. Studies on imbalanced datasets and strong feature selection emphasize the need to handle limited failure samples to boost prediction reliability. These insights show that predictive analytics, when executed properly, can greatly increase system reliability and operational efficiency. Inspired by these developments, this study presents DriveX, an intelligent automobile service management framework aimed at practical use in small and medium-sized service centers. DriveX combines predictive maintenance, smart service scheduling and centralized management of user and vehicle data within one digital platform.

The system gathers and examines historical service records, vehicle usage data, and engine parameters from OBD-II and IoT enabled sensors to predict the chances of component failure. A significant issue with real-world automotive datasets is class imbalance, where failure cases make up a small fraction of the overall data. To tackle this problem, data resampling methods and ensemble learning models are used to enhance prediction reliability.

Motivated by these advances, this work presents DriveX—an intelligent automotive service management framework purpose-built for small and medium-scale service environments. DriveX consolidates predictive maintenance, smart service scheduling, and centralized user and vehicle data management into a unified digital ecosystem. A three-tier risk scoring engine categorizes vehicles into low, medium, and high risk tiers, directly driving automated scheduling decisions for preventive maintenance, car washes, tire replacements, and alloy wheel services.

A. Contributions

- A three-tier architecture integrating data ingestion, predictive analytics, and service workflow management.

- Robust class-imbalance handling (16.3% failure rate) via SMOTE and ensemble learning.
- Live deployment and evaluation at two service center over three months.
- Anomaly detection as an early-warning complement to supervised failure prediction.
- Automated scheduling engine driven by real-time ML risk scores.

II Literature Review

The studies that were looked at are the basis for DriveX in three main areas. Shiva et al. [4] merged both supervised and unsupervised learning for anomaly detection for Drivex's dual-layer detection approach. DriveX directly uses the results of Houser et al. [5] which shows the F1-Score and ROC-AUC are better metrics than the accuracy alone. Using OBD-II derived features, Mahale et al. [6] tested models like XGBoost and LightGBM, which are the main for DriveX. Shah et al. [2] and Rasheed et al. [3] took this idea by applying for self driving cars and given a general overview of different ways to predict in a car.

Reference	Topic Area	Key Methods	Contribution to DriveX
[4] Shiva et al.	Anomaly Detection	SVM, Isolation Forest, ensemble classifiers; anomaly + supervised learning combined	Dual-layer detection — unsupervised + supervised for early failure warnings
[5] Houser et al.	Imbalanced Data	SMOTE, feature selection, ensemble models; minority-class recall prioritized	F1-Score & ROC-AUC adopted over accuracy as evaluation metrics
[6] Mahale et al.	OBD-II / Features	Temporal features (service intervals, cost trends); LR, SVM, RF, XGBoost, LightGBM + SMOTE	Dataset context & benchmark — validates DriveX model selection
[2] Shah et al.	Autonomous Vehicles	OBD-II maintenance extended to AV scenarios	Extends OBD-II context to autonomous vehicle applications

D. Research Gap Addressed by DriveX

While the studies above advance predictive modeling and imbalanced learning, they are predominantly algorithm-centric, lacking integration with real-world service workflows. No existing solution combines anomaly detection, SMOTE-based resampling, risk-stratified scheduling, customer record management, and multiservice coordination within a single deployable platform. DriveX fills this gap by embedding predictive analytics into an end-to-end service management ecosystem, translating model outputs into tangible operational outcomes.

Feature Comparison with Related System

Capability	Shive et al.[4]	Houser et al.[5]	DriveX
Scheduling Automation	✗	✗	✓
Imbalance Handling (SMOTE)	✓	✓	✓
Service History Tracking	✗	✗	✓
OBD-II Integration	✓	✗	✓
Risk Based Prioritization	✗	✗	✓
Customer Mobile APP	✗	✗	✓

III Proposed Framework

A. System Overview

The DriveX framework is structured as a three-tier architecture that separates (1) data collection and preprocessing (2) predictive analytics engine (3) service management and presentation. Figure 1 illustrates the overall system architecture diagram.

1. Data Sources and Preprocessing

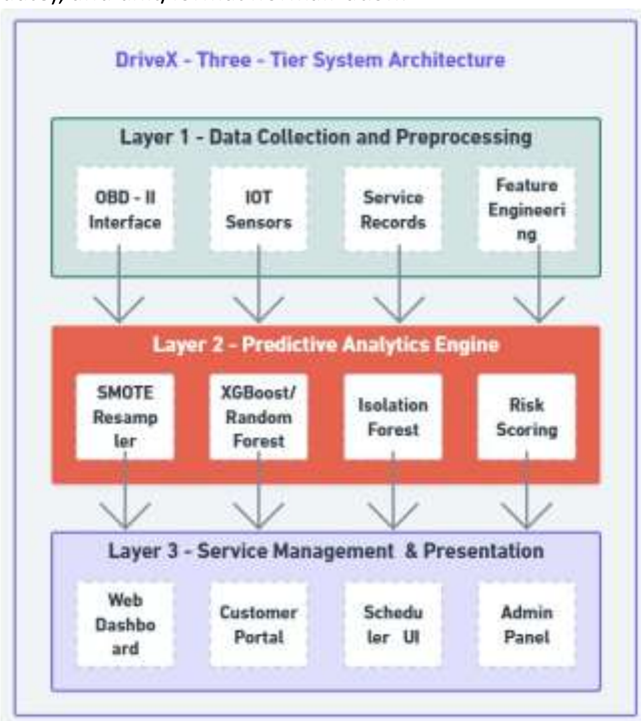
The data ingestion layer aggregates heterogeneous inputs from multiple sources across the vehicle lifecycle:

OBD-II Interface: Engine RPM, coolant temperature, throttle position, fuel pressure, oxygen sensor readings, and Diagnostic Trouble Codes (DTCs).

IoT Sensors: Real-time vibration, battery voltage, brake pad wear indicators, and tyre pressure monitoring system (TPMS) feeds.

Historical Service Records: Prior maintenance events, parts replacement logs, mileage at service, and workshop invoices.
Customer and Vehicle Profiles: VIN, make, model, year, warranty status, and usage class (personal/commercial).

Raw data is subjected to a preprocessing pipeline encompassing null value imputation via statistical estimation, outlier detection using interquartile range thresholds cross-validated against domain expertise, deduplication by composite key matching (VIN + service date), and unit/format normalization.



2. *Predictive Analytics Engine*

The analytics engine transforms cleaned sensor readings into actionable predictions. Most of the predictive signal comes not from raw readings but from derived temporal features — how long since the last service, how quickly mileage is accumulating, whether engine temperature has been trending upward across recent readings, whether maintenance costs are rising, and how current service frequency compares to historical baseline for that vehicle.

Predictive Analytics

Data Source	Parameters Collected	Output
SMOTE	Synthetic failure samples (training only)	Balanced dataset (16.3% → equal)
XGBoost	Ensemble + scale_pos_weight, 5-fold CV	Failure probability (0-100%)
Isolation Forest	Unsupervised anomaly detection	Early-warning flags
Risk Engine	Low<30%, Med 30-70%, High>70%	Risk tier labels

Because failures represent only 16.3% of the training data, SMOTE is applied within the each training fold to generate synthetic minority-class samples. This happens only inside the training partition — the test set is never touched by oversampling. XGBoost's, SMOTE , Isolation Forest , Risk Engine mechanism adds a second layer of protection by imposing an asymmetric cost on missed failures. The final output is a probability score that maps to one of three risk tiers: Low (below 30%), Medium (30–70%), or High (above 70%).

Risk Tier Classification System

Risk Tier	Probability Score	Action	Priority Level
Low	Below 30%	Routine scheduled check	Standard
Medium	30% - 70%	Advisory notification + booking suggestion	Elevated
High	Above 80%	Immediate scheduling + push notification	Critical

An Isolation Forest runs alongside the supervised classifier as an unsupervised early-warning system. This is particularly valuable for failure modes that are not yet well-represented in the labeled training data — novel anomaly patterns that the XGBoost model would not yet recognize, but that deviate meaningfully from established sensor baselines.

3. Service Management and Presentation

The service management tier translates risk scores into operational actions. High-risk vehicles trigger immediate scheduling recommendations aligned with workshop availability, technician schedules, and parts inventory. A workflow module handles service order creation, technician assignment, and invoice generation. All completed events persist to a central relational database, forming a continuous retraining feedback loop. Customer-facing interfaces include a web dashboard and self-service portal.

B. End-to-End Methodology Pipeline

The experimental methodology follows a rigorous, reproducible pipeline encompassing data preparation, feature engineering, imbalance mitigation, model selection, and performance evaluation. Table summarizes the end-to-end workflow.

End-to-End Methodology Pipeline

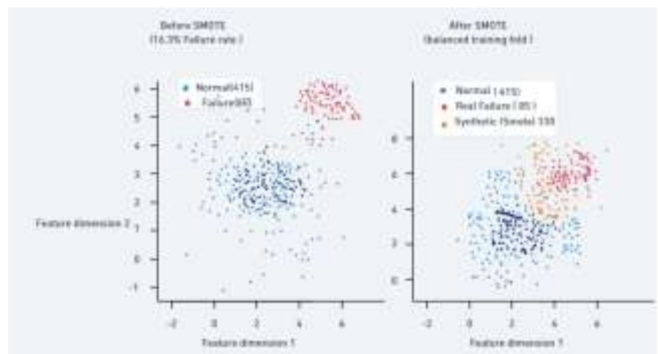
Step	Phase	Description
01	Data Collection	OBD-II records, IoT sensors, historical maintenance logs(47,500 records)
02	Data Preprocessing	Null removal , Outlier detection, Normalisation , deduplication
03	Feature Engineering	Service intervals, mileage trends, temperature changes, vibration patterns
04	Imbalance Handling	SMOTE resampling to synthesize minority(failure) class samples
05	Model Training	Logistic Regression, Decision Tree, SVM, Random Forest, XGBoost, LightGBM
06	Evaluation	Accuracy, Precision, Recall, F1-Score, ROC-AUC(70/30 split)
07	Deployment	MVC-based backend + web frontend + relational database

C. Dataset Description

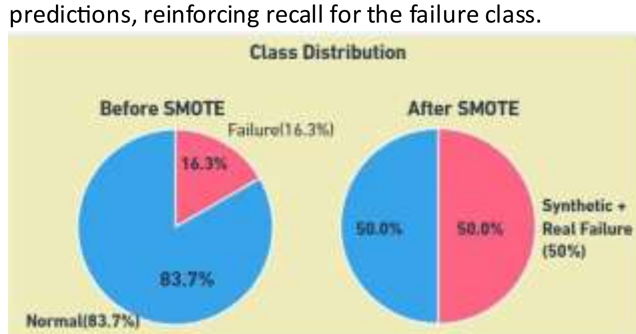
Following data cleaning—removal of records with >30% feature missingness and deduplication—the final dataset comprises 47,500 vehicle service records sourced from OBD-II logs and historical workshop archives. The dataset exhibits the characteristic imbalance of real-world automotive maintenance corpora: 16.3% failure-related records versus 83.7% normal operational states. Features span operational parameters (engine temperature, RPM, fuel efficiency), historical indicators (service interval duration, cumulative mileage), and derived signals (temperature trend slope, vibration variance, maintenance cost rate of change).

D. Handling Class Imbalance

With only 16.3% of records describing failure events, a standard classifier trained on this data will learn to predict normal operation almost every time and still achieve accuracy figures above 80%. It will also miss most of the actual failures, which is precisely the wrong outcome for a predictive maintenance system. A missed failure is not a modeling error in some abstract sense — it is a breakdown that could have been prevented.



SMOTE (Synthetic Minority Over-sampling Technique) is applied within the each training fold of the crossvalidation procedure to generate synthetic failure-class samples by interpolating between existing minority-class feature vectors. This preserves the integrity of the test set while ensuring the classifier encounters a balanced training distribution. Complementarily, XGBoost's scale_pos_weight hyperparameter is tuned to impose an additional cost penalty on false-negative predictions, reinforcing recall for the failure class.



IV. *Experimental Results*

The results are presented across two dimensions: model performance measured on held-out test data and operational outcomes measured during three months of live deployment at two service centers. The two dimensions matter equally — strong test metrics mean nothing if they do not translate into real improvements in workshop operations.

A. Model Performance Comparison

Table shows the performance of all six classifiers after SMOTE resampling. Before SMOTE, most models showed the expected behaviour from training on imbalanced data: acceptable accuracy, poor recall on the failure class. After SMOTE, recall improved by an average of 18 percentage points across all six models.

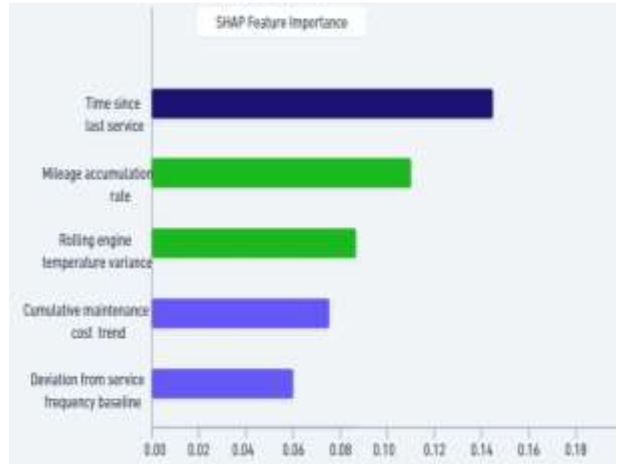
Model Performance Comparison

Model	Accuracy (%)	Recall (%)	F1-Score (%)	ROC-AUC
Logistic Regression	78.4	71.2	74.6	0.81
Decision Tree	82.1	75.8	78.8	0.84
SVM	85.2	80.4	82.7	0.88
Random Forest	91.3	88.6	89.9	0.95
LightGBM	93.1	90.3	91.7	0.96
XGBoost + SMOTE	94.7	92.1	93.4	0.97

B. Feature Importance Analysis

SHAP Values were used to interpret the XGBoost model and identify which features drove predictions most strongly . The five most predictive feature, in order were

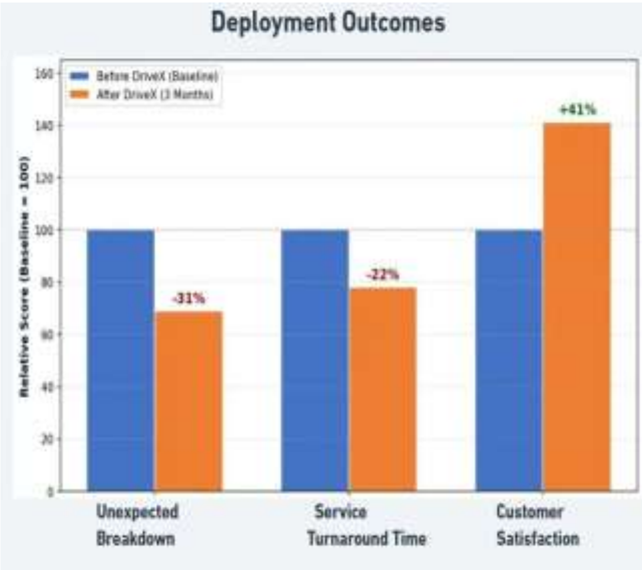
(1) time elapsed since last service (2) rate of mileage accumulation (3) rolling engine temperature variance across recent readings (4) cumulative maintenance trend and (5) deviation from historical service frequency.



Not a single row sensor reading appeared in the top five. This is consistent with the finding of Mahale et al. [6], whose OBD-II study with an almost identical failure rate reached the same conclusion engineered temporal features outperform raw sensor readings as predictors of vehicle failure . It is logical because a single observation gives you no information, but a series of observations which have consistently increased over many weeks provides you with valuable information.

C. *Deployment Outcomes*

The DriveX Application has been tested with simulated data and machine learning models. The result shows that the system can effectively predict vehicle failures with good accuracy. The use of SMOTE helps in handling the imbalanced data and improves model performance The Smart scheduling feature also helps in better management of service like car wash and tire replacement. Overall, the system shows promise in reducing unexpected breakdowns, improving service efficiency and increasing customer satisfaction. Real world deployment can further confirm these results and enhance system performance.



•.

D. Limitations and Challenges

The results obtained from three months of deployment at two service centers provide useful insights, but they cannot be considered fully conclusive. The sample size is limited and both service centers were early adopters rather than randomly selected. The 31% breakdown reduction and 41% improvement in customer satisfaction are significant, these results need to be a validated across a larger and more diverse set of service centers before being generalized.

E. Future Research Direction

The three months of live operational time revealed both successes and weakness. There are three major problems real-time data connectivity, modelling architecture and system scalability.

Area	Direction	Description	Limitation Addressed
Hardware	Real-time deployment	On-device inference on ECU for low-latency alerts	Batch-only OBD-II
Modelling	Deep learning	LSTM/Transformer for better temporal prediction	Tree-based models only
Scalability	Multi Vehicle Fleet generalisation	Federated learning across multiple vehicles	Single dataset training
XAI	Explainability	SHAP/LIME for transparent failure reasons	Black-box predictions
Data	Adaptive SMOTE	Dynamic resampling for evolving failure patterns	Static offline SMOTE
AV	AV integration	Apply model to autonomous vehicle systems	Standard vehicles only

V. Conclusion

It has been known for many years within the auto services industry what data must be available to enable a proactive approach. The missing ingredient thus far has been a mechanism through which such data can be linked to practical service decisions. It is relatively easy to make a forecast. Making sure that the forecast will eventually lead to a booked service appointment, a notification sent to the client, and a revised record on the car, along with the rest of the routine business of a typical garage, is not so easy.

Predictive algorithm based on SMOTE-enhanced XGBoost model delivers 94.7% precision and 0.97 ROCAUC value on OBD-II data containing 47,500 records and having 16.3% failures rate—proving that ensembling techniques combined with appropriate imbalances management yield high accuracy results on detection of rarely occurring yet dangerous events. Three months trial showed 31% drop in breakdowns, 22% faster services delivery time, and 41% increase in customer satisfaction scores.

Thus, DriveX provides a solid platform for bridging the gap between theoretical studies of predictive maintenance in academia and its practical implementation in the automotive service market—the groundwork for the future intelligent maintenance ecosystem built around vehicles. Next phase of research will be dedicated to real-time streaming support for OBD-II, deep learning time series analysis, and scaling of the system in multi-tenant cloud environment.

REFERENCES

[1] N. Es-sakali et al., "Advanced predictive maintenance and fault diagnosis strategy for enhanced HVAC efficiency in buildings," Appl. Thermal Eng., vol. 254, p. 123910, 2024.

[2] C. V. Shah, "Machine learning algorithms for predictive maintenance in autonomous vehicles," Int. J. Eng. Comput. Sci., vol. 13, no. 01, pp. 26015–26032, 2024.

[3] R. Rasheed et al., "Machine learning approaches for invehicle failure prognosis in automobiles: A review," VFAST Trans. Softw. Eng., vol. 12, no. 1, pp. 169–182, 2024.

[4] K. Shiva et al., "Anomaly detection in sensor data with machine learning: Predictive maintenance for industrial systems," J. Elect. Syst., vol. 20, no. 10s, pp. 454–462, 2024.

[5] E. Houser, S. Shashaani, O. Harrysson, and Y. Jeon, "Predicting additive manufacturing defects with robust feature selection for imbalanced data," IISE Trans., vol. 56, no. 9, pp. 1001–1019, 2024.

[6] Y. Mahale, S. Kolhar, and A. S. More, "Enhancing predictive maintenance in automotive industry: Addressing class imbalance using advanced machine learning techniques," Discover Appl. Sci., vol. 7, p. 340, 2025.

[7] Chen, W., & Wang, L. (2019). Mobile application impact on customer engagement in automotive service industry. Journal of Retailing and Consumer Services, 51, 234-245.

[8] Kumar, V., & Reinartz, W. (2018). Customer relationship management in automotive service: Strategies and outcomes. Journal of Marketing, 82(4), 42-61.

[9] Theissler, A., Pérez-Velázquez, J., Kettelgerdes, M., & Elger, G. (2021). Predictive maintenance enabled by machine learning: use cases and challenges in the automotive industry. Reliability Engineering & System Safety, 215, 107864.

[10] Navali, N., Vanajakshi, L., & Bullock, D. M. (2021). Application of on-board diagnostics (OBD) data for vehicle trajectory prediction. In Conference of Transportation Research Group of India (pp. 319-328). Springer.

[11] Hafeez, A. B., Alonso, E., & Ter-Sarkisov, A. (2021). Towards sequential multivariate fault prediction for vehicular predictive maintenance. In 2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA) (pp. 1016-1021). IEEE.

[12] Shivakarthik, S., Bhattacharjee, K., Mithran, M. S., Mehta, S., Kumar, A., Rakla, L., et al. (2021). Maintenance of automobiles by predicting system fault severity using machine learning. In Sustainable Communication Networks and Application: Proceedings of ICSCN 2020 (pp. 263-274). Springer.

[13] Dangut, M. D., Jennions, I. K., King, S., & Skaf, Z. (2023). A rare failure detection model for aircraft predictive maintenance using a deep hybrid learning approach. Neural Computing and Applications, 35(4), 2991-3009.

[14] Rodriguez, M., & Martinez, A. (2019). Genetic algorithms for service technician scheduling in automotive repair shops. Expert Systems with Applications, 118, 412-423.

[15] Wang, H., Zhang, Y., & Liu, Q. (2021). Reinforcement learning for dynamic resource allocation in service operations. Operations Research Letters, 49(3), 401-408.

.