International Journal of Advanced Engineering, Management and Science (IJAEMS)
Peer-Reviewed Journal
ISSN: 2454-1311 | Vol-12, Issue-2; Mar-Apr, 2026
Journal Home Page: https://ijaems.com/
DOI: https://dx.doi.org/10.22161/ijaems.122.15

Integration of Artificial Intelligence Algorithms into On-Board Vehicle Diagnostic and Control Systems

Madugula Abhiram

Software Developer, Saanavi Technologies, Bloomington, Illinois, US

Received: 22 Feb 2026; Received in revised form: 21 Mar 2026; Accepted: 28 Mar 2026; Available online: 02 Apr 2026

Abstract *— The article examines architectural and engineering approaches for integrating artificial intelligence algorithms into on-board vehicle diagnostic and control systems. Relevance follows from software-defined vehicles, Electronic Control Unit (ECU) density, and the need to convert On-Board Diagnostics II (OBD-II) and Controller Area Network (CAN) telemetry into dependable decisions under real-time and safety constraints. Novelty consists of a synthesis that links data acquisition and preprocessing, model families for fault prediction, multi-fault diagnosis, and intrusion detection, and deployment patterns spanning in-vehicle compute with cloud/edge services. The objective is to systematize design decisions that yield robust diagnostics, interpretable outputs for service workflows, and controlled mitigation actions. Methods comprise comparative analysis of recent publications, taxonomy building across task types, and architectural decomposition of pipelines. The source base covers OBD-II machine learning, deep-learning predictive maintenance, attention-based fault prediction, explainable multi-fault diagnosis, CAN intrusion detection, connected-vehicle platforms, deterministic AUTOSAR Adaptive communication, and ML-aware functional-safety extensions. The article targets automotive software engineers, data scientists, and platform architects.*

Keywords *— **on-board diagnostics, OBD-II, CAN bus, ECU, fault diagnosis, predictive maintenance, intrusion detection, AUTOSAR Adaptive, functional safety, edge–cloud deployment.***

I. INTRODUCTION

Modern vehicles incorporate software functions that were traditionally implemented in dedicated hardware. This shift intensifies diagnostic complexity because faults emerge not only from component degradation but also from interactions among Electronic Control Units (ECU), network traffic, calibration states, and software updates. On-Board Diagnostics II (OBD-II) and Controller Area Network (CAN) provide standardized access to operational signals and diagnostic trouble codes, yet conventional rule-based diagnostics struggle with correlated or cascading failures and with adversarial conditions on in-vehicle networks. Recent studies show strong performance of deep learning for predictive maintenance from OBD streams, attention-based models for fault forecasting, and neural approaches for intrusion detection in CAN traffic, which motivates an integrated view that unifies diagnostic inference with control-oriented mitigation and with fleet-scale feedback loops for model governance and updates.

The purpose of the article is to develop an analytic synthesis of how AI algorithms can be embedded into on-board diagnostic and control pipelines without violating real-time, determinism, and safety expectations. The study solves three tasks:

1) to classify data sources and inference targets for on-board diagnostics, spanning Diagnostic Trouble

Codes (DTC) centric reasoning, time-series condition monitoring, and security anomaly detection;

2) to compare algorithm families and deployment patterns suitable for ECU- and vehicle-level constraints, including hybrid in-vehicle and cloud/edge arrangements;

3) to derive design recommendations that connect model outputs to actionable control or mitigation steps, while aligning the workflow with determinism and ML-related safety engineering practices.

Novelty is provided by a pipeline-centric consolidation that explicitly ties model choice, runtime placement, and safety-oriented lifecycle controls into one engineering narrative rather than treating them as separate topics.

II. MATERIALS AND METHODS

The material base for this article consists of recent peer-reviewed publications used to cover complementary layers of the problem—from embedded determinism to AI inference and connected-vehicle data infrastructure. D. Bellassai et al. analyzed deterministic publish/subscribe communication for AUTOSAR Adaptive, supporting real-time reasoning about service-oriented in-vehicle software stacks [1]. A. Errezgouny et al. studied predictive maintenance from OBD-derived time series using a hybrid Long Short-Term Memory (LSTM) and clustering strategy, informing model selection under sparse labeling [2]. M. N. Hossain et al. reviewed AI-driven vehicle fault diagnosis across subsystems, supporting the formation of a taxonomy for diagnostic targets [3]. P. Iyenghar et al. proposed ML-specific lifecycle phases and testing methods to extend ISO 26262-aligned assurance, shaping governance requirements for safety-relevant AI [4]. H. Jia et al. proposed an attention-based fault prediction model that captures fault correlations, enabling multi-fault reasoning beyond single-code diagnostics [5]. M. J. Khan et al. evaluated trade-offs in vehicle–edge–cloud deployments for deep models, informing placement decisions under latency constraints [6]. W. Lo et al. developed a hybrid deep intrusion detection approach for in-vehicle CAN traffic, supporting security integration within diagnostics [7]. E. T. Michailidis et al. reviewed OBD-II-based ML applications, consolidating signal types and use cases relevant to on-board diagnostic integration [8]. A. Mostefaoui et al. reported an in-production connected-vehicle big-data platform that informs fleet analytics and a feedback-loop architecture [9]. R. Rai et al. evaluated deep learning methods for CAN bus intrusion detection, supporting robustness considerations for cyber-physical threat detection [10].

For writing the article, comparative analysis, structured literature analysis, and architectural decomposition were applied. The approach emphasizes cross-source synthesis of constraints (latency, compute, determinism, safety assurance), mapping them to algorithm choices and to deployment topologies.

III. RESULTS

A consolidated interpretation of the literature supports viewing on-board AI diagnostics as a closed pipeline rather than as an isolated model embedded into an ECU. OBD-II and CAN telemetry jointly provide the observable layer: OBD exposes standardized parameters and DTCs, while CAN frames expose fine-grained inter-Electronic Control Unit (ECU) communication patterns. The OBD-focused review literature indicates that ML systems built on OBD signals are routinely used for anomaly detection, efficiency optimization, predictive maintenance, and broader driving support, with task feasibility depending on feature availability, sampling regimes, and data quality constraints typical for consumer-grade and service-grade telemetry streams [8]. From an integration standpoint, this implies that model design is inseparable from preprocessing choices such as synchronization, denoising, windowing, and handling of missing values, because these steps define what the embedded inference engine treats as a stable “state” for prediction.

Evidence from predictive maintenance research supports a pragmatic strategy for situations where labeled failure data are scarce. A hybrid LSTM-plus-clustering method applied to unlabeled OBD time series illustrates how temporal sequence learning can be coupled with unsupervised structure discovery to separate operational regimes and then predict condition signals with high reported accuracy, directly relevant to fleet settings where maintenance outcomes are sporadically recorded, and ground truth

arrives with a delay [2]. Attention-based fault prediction extends this idea by explicitly modeling relationships among fault events rather than treating faults as independent targets; the reported approach uses attention mechanisms (including graph attention) to capture inter-fault dependencies and forecast fault evolution from historical records [5]. When translated into on-board integration terms, these methods motivate a shift from "single DTC interpretation" to "fault graph evolution," where the AI component outputs ranked hypotheses about fault propagation and the expected following faults, enabling service workflows and on-board mitigations to prioritize interventions.

Multi-fault diagnosis in highly electronic vehicles introduces an additional integration pressure: diagnostic outputs are helpful only if they remain interpretable to technicians and consistent across operational conditions. An explainable hybrid deep model for multiple-fault diagnosis in automotive electronic systems explicitly positions interpretability as a practical requirement, pairing a high-performing fusion architecture with explainability mechanisms that render diagnostic logic in a human-consumable form [5]. This supports a design rule for on-board software teams: if diagnostic inference participates in control decisions or triggers maintenance actions, the output interface should expose not only a label, but supporting evidence (salient signals, temporal segments, confidence bands), because downstream modules—whether a service advisor interface or a safety supervisor—operate on justifications, not only predictions.

In-vehicle cybersecurity has matured into an inseparable part of diagnostics because compromised CAN traffic can masquerade as component failure and distort control loops. Deep intrusion-detection research on CAN traffic repeatedly highlights that CAN lacks built-in authentication and that attacks such as DoS, fuzzing, impersonation, and spoofing are feasible through physical or wireless entry points. A hybrid CNN–LSTM intrusion detection approach based on spatial–temporal representations reports very high detection results on benchmark car-hacking datasets, emphasizing automatic feature learning over manual engineering [7]. A complementary open-access study evaluates recurrent models (LSTM/GRU and variants). It demonstrates high detection performance across multiple datasets, reinforcing the feasibility of embedding anomaly-detection logic at the vehicle network edge [10]. For integration into diagnostic and control systems, these findings justify treating intrusion detection as a first-class diagnostic signal: a security classifier can gate, rate-limit, or quarantine specific message patterns before they contaminate control decisions or inflate false fault codes.

The deployment dimension links algorithm capacity to real-time constraints. Vehicle–edge–cloud analyses indicate that execution-time and accuracy trade-offs become deployment-defining variables, especially as model complexity increases and hardware diversity spans ECU microcontrollers, domain controllers, and external edge nodes [6]. In practice, the literature supports a layered placement strategy: inference steps that must act within tight latency budgets (e.g., detecting gross anomalies affecting safety-relevant functions) remain on-vehicle, while heavier analytics (fleet trend mining, model retraining, drift detection, and root-cause correlation across vehicles) sit in cloud/edge services [6, 9]. The connected-vehicle platform report, based on an in-production automotive big data stack, reinforces that large-scale ingestion, storage, and analytics can be organized into an integrated pipeline for telematics services, making fleet-scale feedback loops technically feasible rather than purely conceptual [9]. For an automotive software profile centered on Java and Google Cloud Platform, this architecture naturally maps to microservice-based ingestion, stream processing, and model serving patterns, while keeping the on-board component minimal and deterministic.

Determinism and safety assurance constrain how diagnostic outputs can influence control. Service-oriented in-vehicle platforms, especially AUTOSAR Adaptive, broaden software flexibility but introduce challenges for deterministic communication. Work on deterministic communication for AUTOSAR Adaptive proposes mechanisms to ensure determinism in publish/subscribe interactions, offering a concrete direction for integrating diagnostic messaging into safety-relevant software paths without unpredictable timing [1]. Safety engineering literature that extends ISO 26262 practices for ML emphasizes additional lifecycle phases—data preparation, ML training, and ML deployment—and

ties them to properties such as robustness, uncertainty handling, and interpretability, reflecting an assurance logic that complements classical code-centric safety processes [4]. The integration consequence is direct: an on-board AI diagnostic module cannot be treated as a static "component"; it becomes a managed asset with controlled updates, monitored drift, and evidence artifacts supporting its operational domain boundaries.

Figure 1 consolidates these results into a reference pipeline that connects in-vehicle data acquisition to AI inference, and then to control/mitigation actions and fleet-scale services. The figure is constructed as an author synthesis grounded in OBD-II ML application patterns, predictive maintenance modeling, intrusion detection in CAN, layered deployment practices, determinism in AUTOSAR Adaptive, and ML-aware safety lifecycle extensions [1, 2, 4, 6–10].

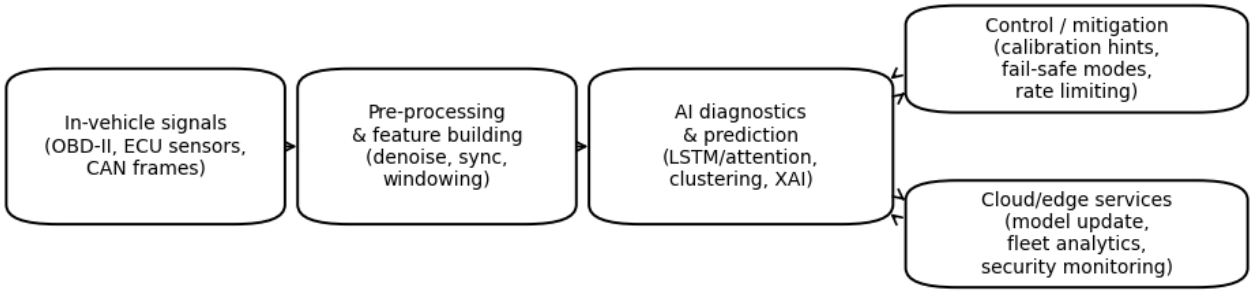

Fig.1. Reference pipeline for integrating AI diagnostics into on-board control loops

The synthesized pipeline supports a unified interpretation of AI-enabled on-board diagnostics as a closed technical loop that starts with standardized vehicle telemetry (OBD-II parameters, ECU sensor streams, CAN frames), proceeds through bounded preprocessing and feature construction, and terminates in two tightly coupled output channels: on-board mitigation and control-aligned recommendations constrained by deterministic execution and message timing, and fleet-scale cloud/edge services responsible for monitoring, model lifecycle operations, and security surveillance. This arrangement is consistent with recent evidence that deep sequential models and hybrid learning strategies can extract predictive maintenance value from OBD-derived time series under sparse labeling, that CAN-bus intrusion detection benefits from spatial–temporal deep representations suitable for near-real-time screening, and that safety-critical integration requires ML-specific lifecycle phases and testing practices layered onto functional-safety engineering, while deterministic communication patterns in AUTOSAR Adaptive provide an engineering basis for integrating diagnostic inference into service-oriented in-vehicle stacks without timing instability.

IV. DISCUSSION

The reviewed evidence supports treating "on-board diagnostics plus AI" as a system-of-systems problem: the diagnostic function is shaped by telemetry interfaces (OBD/CAN), by algorithm families tuned to data scarcity and temporal dependence, by runtime placement across vehicle and cloud/edge, and by governance for determinism and safety assurance [1, 4, 6, 8, 9]. A practical implication for automotive software teams working primarily with Java and Google Cloud Platform is that the most stable engineering boundary is not between "embedded" and "cloud," but between "latency-critical decisions" and "fleet-scale learning and governance." The former favors compact models, bounded execution, and deterministic messaging, while the latter favors scalable data platforms and controlled model lifecycle operations [1, 6, 9].

Table 1 organizes algorithm families by diagnostic objective and integration implications, highlighting where interpretability and data labeling pressure engineering choices.

Table 1. *AI algorithm families and integration implications for on-board diagnostics and control*

Objective	Representative approaches	Typical vehicle data	Integration implication
Predictive maintenance from OBD time series	Sequence models (LSTM) combined with unsupervised clustering [2]	OBD sensor time series, derived features [8]	Supports weak-label regimes; requires stable preprocessing and drift monitoring.
Fault prediction with correlated faults	Attention-based and graph-attention formulations [5]	Fault records, event sequences	Encourages modeling fault dependencies; output interfaces should expose ranked hypotheses and their confidence levels.
Multi-fault diagnosis with interpretability	Hybrid deep fusion with explainability mechanisms [5]	Electronic subsystem signals, fault samples	Enables technician-facing evidence; reduces ambiguity when multiple simultaneous anomalies occur.
CAN intrusion detection	Hybrid CNN–LSTM spatial–temporal detectors [7]; deep learning IDS variants [10]	CAN frames and traffic patterns	Security classification serves as a gating signal for diagnostics and control, requiring low-latency inference placement.

The platform layer benefits from separating the "model execution surface" into on-vehicle inference and cloud/edge management. Vehicle–edge–cloud studies explicitly frame execution time as a deployment driver and motivate layered pipelines that shift expensive operations away from the car when safety timing allows [6]. The connected-vehicle big-data platform experience indicates that production architectures already exist for large-scale telemetry ingestion and analytics, which can be repurposed for diagnostic model monitoring, retraining triggers, and fleet-level root-cause analysis [9]. Deterministic communication mechanisms for AUTOSAR Adaptive strengthen the feasibility of routing diagnostic outputs through service-oriented stacks without compromising timing predictability in safety-sensitive paths [1]. ML-aware safety lifecycle extensions support the governance side by treating model updates and validation as explicit engineering processes rather than ad hoc operational activities [4].

Table 2 translates these discussion points into an integration matrix spanning in-vehicle, edge, and cloud responsibilities, emphasizing where Java/GCP-aligned services naturally fit.

Two engineering tensions recur across the source base. First, model accuracy gains are not automatically convertible into operational value unless the output is interpretable and action-linked; explainable multi-fault diagnosis supports technician workflows, while predictive maintenance outputs gain value only when they map to maintenance scheduling or conservative control adjustments. Second, security and fault diagnosis intersect at the signal level: a compromised network can generate symptoms that resemble mechanical or electronic faults, making intrusion detection a diagnostic prerequisite rather than a separate security feature. Both tensions reinforce the use of a pipeline architecture that treats preprocessing, inference, and mitigation as a single controlled chain with explicit governance and deterministic communication where safety constraints apply.

Table 2. Integration matrix for AI-enabled diagnostics across vehicle, edge, and cloud layers

Layer	Primary responsibilities	Technical emphasis	Evidence basis
On-board (ECU/domain controller)	Low-latency anomaly screening; basic fault prediction; message gating/mitigation hooks	Bounded runtime, deterministic messaging, minimal model footprint	Deterministic AUTOSAR Adaptive communication [1]; CAN IDS feasibility on vehicle data [7,10]
Edge (near-vehicle compute)	Aggregation across short horizons; privacy-preserving buffering; selective offloading	Latency relief with localized compute; controlled bandwidth use	Vehicle–edge–cloud execution trade-offs [6]
Cloud (fleet platform)	Fleet analytics, drift detection, retraining pipelines, update orchestration, and historical correlation	Scalable ingestion and analytics services; lifecycle governance	Connected-vehicle significant data platform patterns [9]; ML lifecycle phases and testing methods for safety alignment [4]

V. CONCLUSION

The analysis supports three conclusions aligned with the stated tasks. First, OBD-II and CAN telemetry enable complementary diagnostic observability: OBD-centric signals and DTCs support condition monitoring and maintenance forecasting. In contrast, CAN traffic analysis supports the detection of abnormal inter-ECU communication that can corrupt diagnostic reasoning or control behavior. Second, the literature indicates that data realities and temporal structure drive algorithm selection: hybrid sequence learning with unsupervised components fits weak-label OBD regimes, attention-based modeling improves forecasting under correlated faults, and deep spatial–temporal detectors achieve strong CAN intrusion-detection performance on benchmark datasets. Third, robust integration depends on deployment and governance, not only on model choice: layered vehicle–edge–cloud placement aligns execution time and compute limits with fleet-scale learning, deterministic messaging mechanisms in AUTOSAR Adaptive support predictable diagnostic communication, and ML-aware safety lifecycle phases formalize data preparation, training, deployment, and testing activities required for safety-relevant AI functions.

REFERENCES

[1] Bellassai, D., Scordino, C., Casini, D., & Biondi, A. (2025). AP-LET: Enabling deterministic Pub/Sub communication in AUTOSAR Adaptive. *Journal of Systems Architecture, 162*, 103390. https://doi.org/10.1016/j.sysarc.2025.103390

[2] Errezgouny, A., Chater, Y., Barranco González, C. D., & Cherkaoui, A. (2025). An integrated deep learning approach for predictive vehicle maintenance. *Decision Analytics Journal, 16*, 100597. https://doi.org/10.1016/j.dajour.2025.100597

[3] Hossain, M. N., Rahman, M. M., & Ramasamy, D. (2024). Artificial intelligence-driven vehicle fault diagnosis to revolutionize automotive maintenance: A review. *CMES - Computer Modeling in Engineering and Sciences, 141*(2), 951–996. https://doi.org/10.32604/cmes.2024.056022

[4] Iyenghar, P., Gracic, E., & Pawelke, G. (2024). A systematic approach to enhancing ISO 26262 with machine learning-specific life cycle phases and testing methods. *IEEE Access, 12*, 179600–179627.

[5] Jia, H., Qian, D., Chen, F., & Zhou, W. (2025). Collaborative fusion attention mechanism for vehicle fault prediction. *Future Internet, 17*(9), 428. https://doi.org/10.3390/fi17090428

[6] Khan, M. J., Khan, M. A., Turaev, S., Malik, S., El-Sayed, H., & Ullah, F. (2024). A vehicle-edge-cloud framework for computational analysis of a fine-tuned deep learning model. *Sensors, 24*(7), 2080. https://doi.org/10.3390/s24072080

[7] Lo, W., Alqahtani, H., Thakur, K., Almadhor, A., Chander, S., & Kumar, G. (2022). A hybrid deep learning based intrusion detection system using spatial-temporal representation of in-vehicle network traffic. *Vehicular Communications, 35*, 100471. https://doi.org/10.1016/j.vehcom.2022.100471

[8] Michailidis, E. T., Panagiotopoulou, A., & Papadakis, A. (2025). A review of OBD-II-based machine learning applications for sustainable, efficient, secure, and safe

vehicle driving. *Sensors, 25*(13), 4057. https://doi.org/10.3390/s25134057

[9] Mostefaoui, A., Merzoug, M. A., Haroun, A., Nassar, A., & Dessables, F. (2022). Big data architecture for connected vehicles: Feedback and application examples from an automotive group. *Future Generation Computer Systems, 134*, 374–387. https://doi.org/10.1016/j.future.2022.04.020

[10] Rai, R., Grover, J., Sharma, P., et al. (2025). Securing the CAN bus using deep learning for intrusion detection in vehicles. *Scientific Reports, 15*, 13820. https://doi.org/10.1038/s41598-025-98433-x