V2LLM: A Digital-Twin-Aware Retrieval-Augmented Large Language Model for Resilient Telemetry Recovery and Fault Diagnosis in V2X

Bishmita Hazarika, *Member, IEEE*, Keshav Singh, *Member, IEEE*, Nguyen-Son Vo, *Member, IEEE,* Berk Canberk, *Senior Member, IEEE*, Simon L. Cotton, *Fellow, IEEE*, Hyundong Shin, *Fellow, IEEE*, and Trung Q. Duong, *Fellow, IEEE*

***Abstract*—Reliable telemetry is essential for fault detection and adaptive control in vehicle-to-everything (V2X) networks, yet wireless impairments often cause bursty, non-i.i.d. data loss. This paper proposes V2LLM, a digital-twin-aware, retrieval-augmented large language model (LLM) framework for resilient telemetry recovery and continual fault diagnosis. Leveraging the reasoning capabilities of large AI models (LAMs), V2LLM performs token-efficient reconstruction by combining local history, memory-guided context, and digital twin predictions. A sliding-window autoregressive scheme enables structured recovery under long gaps, while a federated continual learning module adapts fault classifiers at roadside units without raw data sharing. V2LLM tightly integrates semantic prompting, predictive modeling, and distributed adaptation to address uncertainty, heterogeneity, and spatiotemporal drift in V2X environments. Extensive evaluations show that the proposed framework consistently improves reconstruction quality and classification robustness, demonstrating the potential of LAM-powered architectures for intelligent, resilient wireless systems.**

B. Hazarika is with the Faculty of Engineering and Applied Science, Memorial University, St. John's, Canada. (Email: bhazarika@mun.ca).

K. Singh is with the Institute of Communications Engineering, National Sun Yat-sen University, Kaohsiung 80424, Taiwan. (Email: keshav.singh@mail.nsysu.edu.tw).

N.-S. Vo is with the Institute of Fundamental and Applied Sciences, Duy Tan University, Ho Chi Minh City, 70000, Vietnam, and also with the Faculty of Electrical-Electronic Engineering, Duy Tan University, Da Nang, 50000, Vietnam (e-mail: vonguyenson@duytan.edu.vn).

B. Canberk is with the School of Engineering and Built Environment, Edinburgh Napier University, Edinburgh EH10 5DT, UK, (e-mail: b.canberk@napier.ac.uk).

S. L. Cotton is with the School of Electronics, Electrical Engineering and Computer Science, Queen's University Belfast, Belfast, U.K. (Email: simon.cotton@qub.ac.uk).

H. Shin is with the Department of Electronics and Information Convergence Engineering, Kyung Hee University, 1732 Deogyeong-daero, Giheung-gu, Yongin-si, Gyeonggi-do 17104, Republic of Korea (e-mail: hshin@khu.ac.kr).

T. Q. Duong is with the Faculty of Engineering and Applied Science, Memorial University, St. John's, NL A1C 5S7, Canada, and with the School of Electronics, Electrical Engineering and Computer Science, Queen's University Belfast, Belfast, U.K, and also with the Department of Electronic Engineering, Kyung Hee University, Yongin-si, Gyeonggi-do 17104, South Korea (e-mail: tduong@mun.ca).

This paper has been accepted in part for presentation at the IEEE Global Communications Conference (Globecom), December 2025, Taipei, Taiwan.

This work was supported in part by the Canada Excellence Research Chair (CERC) Program CERC-2022-00109 and in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery Grant Program RGPIN-2025-04941. The work of B. Canberk is partially supported by The Scientific and Technological Research Council of Türkiye (TÜBİTAK) 1515 Frontier R&D Laboratories Support Program for Türk Telekom 6G R&D Lab under project number 5249902. The work of S. L. Cotton was funded in part by the U.K. Engineering and Physical Sciences Research Council (EPSRC) through the EPSRC Hub on All Spectrum Connectivity under Grant EP/X040569/1 and Grant EP/Y037197/1. The work of H. Shin was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (RS-2025-00556064), and by the Ministry of Science and ICT (MSIT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2025-RS-2021-II212046), supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation).

Corresponding authors are Trung Q. Duong and Hyundong Shin.

***Index Terms*—Vehicle-to-Everything (V2X), Large AI Models (LAMs), Large Language Models (LLMs), Retrieval-Augmented Generation (RAG), Telemetry Recovery, Digital Twin, Federated Continual Learning.**

I. Introduction

LARGE AI Models (LAMs), characterized by their capacity to learn and generalize from vast amounts of data, are increasingly transforming the landscape of intelligent wireless communication. With recent advancements in foundation models like GPT-4, Gemini, and Claude, the scope of LAMs has expanded beyond natural language processing to domains such as robotics, healthcare, and increasingly, future wireless systems [1], [2]. Their ability to perform cognitive-level reasoning, context understanding, and generative forecasting makes LAMs a compelling paradigm for next-generation vehicle-to-everything (V2X) networks, which demand predictive intelligence, semantic awareness, and decentralized adaptability in real time. In V2X networks, vehicles continuously transmit high-dimensional telemetry, capturing mobility states (e.g., position, velocity), communication metrics (e.g., received signal strength indicator (RSSI), packet loss rate (PLR)), and environmental cues—to nearby roadside units (RSUs) for real-time decision-making [3]. In contrast, due to the inherently unstable nature of wireless vehicular communication, the telemetry stream is frequently corrupted or partially lost [4]. High-speed mobility introduces rapid channel variation and Doppler shifts, while wireless impairments such as fast fading, shadowing, and multi-path interference further degrade signal integrity [5], [6]. These disruptions inject uncertainty into downstream tasks such as fault diagnosis and cooperative control. Traditional methods like interpolation or smoothing fail to capture semantic dependencies, while deep sequence models like long short-term memory (LSTM) [7] and sequence-to-sequence (Seq2Seq) architectures lack adaptability to shifting contexts and sparse inputs. These limitations call for more context-aware, memory-augmented architectures, motivating the use of LAMs, particularly large language models (LLMs), for telemetry reasoning under uncertainty [8].

To overcome these challenges, LAMs, particularly LLMs have emerged as flexible architectures capable of generalized reasoning under uncertain, multimodal conditions [9]–[11]. Their ability to model long-range dependencies and operate on serialized inputs has recently inspired a new class of retrieval-augmented generation (RAG) frameworks [12], [13]. In such systems, input sequences are supplemented with externally retrieved

context from structured memory banks, which the LLM uses during generation to ground predictions in past experience [14], [15]. Applied to telemetry recovery, this allows the system to selectively reconstruct missing data by conditioning on both local history and similar past scenarios, thereby improving generalization in non-stationary environments. However, RAG alone remains insufficient in vehicular settings where real-time adaptation to evolving traffic patterns and wireless conditions is required. Moreover, RAG models without environmental grounding may still generate contextually inconsistent sequences, particularly under severe loss or noise [16].

This limitation motivates the integration of digital twins (DTs), which are virtual replicas of physical environments that enable state estimation, forecasting, and cross-layer reasoning [17]. In vehicular networks, DTs maintained at infrastructure nodes can provide high-level summaries of local environmental dynamics, such as traffic density, signal quality distributions, and mobility trends [18]. When fused with LLM-based RAG prompting, these DTs offer contextual priors that improve the coherence and accuracy of telemetry reconstruction. Yet, even with this integration, a key challenge remains: how to adapt classification models to shifting data distributions, emerging faults, and incomplete supervision, especially in a distributed and bandwidth-limited setting.

Federated learning (FL) has traditionally addressed the challenge of distributed training in privacy-sensitive environments by enabling model optimization across multiple clients without requiring direct data sharing [19]–[21]. Early FL algorithms such as federated averaging (FedAvg) rely on periodic parameter aggregation under the assumption of independent and identically distributed (i.i.d.) data and relatively stationary client behavior [22]. In contrast, vehicular networks are inherently non-stationary, with rapidly evolving data distributions caused by environmental changes, mobility patterns, and emerging fault types. In such contexts, FedAvg and similar periodic aggregation schemes often exhibit poor generalization and suffer from catastrophic forgetting. To address these limitations, variants such as FedProx [23], which incorporates local regularization for handling system heterogeneity, and FedDyn [24], which dynamically adjusts client objectives, have been proposed. While these techniques partially mitigate convergence issues in heterogeneous settings, they are still fundamentally static in nature and ill-equipped to handle temporally evolving tasks or lifelong learning objectives. Recently, federated continual learning (FCL) has emerged as a promising direction, enabling each infrastructure node such as a roadside unit (RSU) to locally adapt to new data while maintaining alignment with a global model [25]. FCL frameworks introduce mechanisms such as episodic memory rehearsal, importance-based regularization, and selective update propagation to preserve past knowledge while integrating new experiences. This continual adaptation is especially critical in V2X scenarios, where fault patterns evolve, communication conditions shift, and data availability varies over time. Unlike traditional FL, FCL ensures that local models retain diagnostic performance across past and future tasks without incurring communication or privacy overhead, making it a better fit for sustainable and real-time vehicular intelligence.

Despite advances in LLM-based telemetry reconstruction, digital twin modeling, and federated continual learning, these capabilities remain largely disconnected in existing V2X research. Prior studies typically employ sequence modeling without environmental grounding [26], DT frameworks without generative recovery [27], or federated classification without continual adaptation [28]. Current RAG-based approaches rarely incorporate infrastructure-level environmental context, while DT-driven frameworks seldom exploit prompt-based inference for telemetry restoration. Consequently, robust telemetry recovery in burst-loss, non-i.i.d. vehicular environments remains unresolved, particularly when reconstruction, contextual reasoning, and decentralized adaptation must operate jointly.

Driven by this vision, this work proposes a unified, multi-layer framework, *V2LLM*, that leverages LAMs for structured telemetry generation, DT for context alignment, and FCL for adaptive fault classification across edge nodes. Together, these components enable a resilient and semantically enriched inference pipeline, capable of recovering and understanding vehicular telemetry in real time under uncertainty. The key contributions of this study are outlined below:

- We propose an LLM-based telemetry reconstruction framework that combines RAG with context-aware prompting to recover missing high-dimensional vehicular telemetry in non-i.i.d., burst-loss conditions using token-efficient reasoning.
- We design a DT-guided memory retrieval mechanism that ranks historical telemetry windows based on trajectory similarity and environmental embeddings, enabling semantically grounded prompt construction for improved reconstruction accuracy.
- To handle consecutive missing sequences, we implement a sliding-window autoregressive generation strategy that extends predicted telemetry within LLM token limits, and empirically show that RAG-based prompting achieves higher recovery quality with shorter prompts compared to baselines.
- We incorporate an FCL module that allows RSUs to adapt their classifiers over time while mitigating catastrophic forgetting through memory rehearsal and regularization, supporting robust long-term fault detection without raw data sharing.
- We present V2LLM, a unified five-layer framework integrating telemetry modeling, DT forecasting, memory retrieval, LLM-based recovery, and federated classification, offering scalable and context-aware V2X telemetry diagnosis.
- Extensive experiments across multiple benchmarks show that V2LLM consistently outperforms baseline approaches in both recovery accuracy and classification robustness under dynamic V2X scenarios.

Structure of the paper: The organization of this paper is as follows. Section II provides a comprehensive review of related works and positions the proposed framework within existing literature. Section III introduces the V2X system model and telemetry setup. Section IV formally defines the telemetry recovery and distributed classification problem. Section V presents the proposed V2LLM framework in detail, layer by layer. Section VI discusses memory optimization and complexity analysis. Section VII outlines the simulation setup, evaluation metrics, and baseline configurations. Section VIII presents the experimental results and insights. Finally, Section IX concludes

the paper and outlines future research directions.

II. Related Work

This section reviews prior research that provides the conceptual foundation for the proposed V2LLM framework. As discussed in Section I, V2LLM integrates digital-twin-aware reasoning, retrieval-augmented LLM, and FCL to achieve resilient telemetry recovery and fault diagnosis in vehicular networks. To position this work within existing literature, we organize related studies into five interconnected research streams:

A. Telemetry Imputation and Time-Series Recovery

Reliable reconstruction of missing telemetry data is a long-standing challenge in time-series modeling and plays a critical role in vehicular analytics. Traditional interpolation and autoregressive schemes [29] offered efficient estimation but lacked the ability to capture nonlinear motion dynamics and often failed under bursty or correlated losses. With the rise of deep learning, recurrent and variational models such as GRU-D [30], BRITS [31], and GP-VAE [32] began modeling temporal decay and uncertainty in latent space, marking a shift toward context-aware imputation. Subsequent transformer-based approaches, including SAITS [33] and TimesNet [34], extended this idea by learning long-range dependencies and irregular sampling patterns, while diffusion-based imputers [35] further improved continuity under complex loss structures.

In contrast, most existing works were developed for stationary domains such as health or finance and struggle to generalize to vehicular telemetry, where data loss correlates with motion, traffic, and wireless interference. Attempts to integrate spatial context using graph-based or Kalman-augmented recurrent networks [36]–[38] improved short-term prediction but remained deterministic and environment-agnostic. These limitations become more pronounced in vehicular telemetry, where reconstruction must account for coupled mobility and communication dynamics. This motivates retrieval-guided generative approaches capable of incorporating both historical trajectories and environmental context.

B. Retrieval-Augmented Generation for Temporal Reasoning

The emergence of LLMs has expanded sequential reasoning and generative modeling beyond natural language into structured and temporal domains. Recent advances in retrieval-augmented generation (RAG) extend this paradigm to time-series [39] and tabular modeling [40], where retrieved memory improves contextual grounding and prediction consistency. Retrieval-based transformers have demonstrated advantages over conventional autoregressive decoders by incorporating semantically related historical samples during inference [41], [42]. Subsequent developments integrate retrieval with self-attention mechanisms to jointly model local continuity and long-range temporal dependencies [43]. Domain-specific retrieval strategies have further improved sequential reasoning performance. For example, FinSrag [44] shows that retrieval aligned with task-specific temporal patterns improves forecasting accuracy compared with generic embedding-based retrieval. Related studies in memory-augmented transformers [45], retrieval-guided reasoning for large models [46], hybrid temporal attention architectures [47], and LLM-based wireless communication tasks such as channel modeling and resource allocation [48], [49] demonstrate the value of external memory and large-model reasoning for structured inference under heterogeneous and evolving conditions. Within vehicular systems, transformer-based models have also been explored for domain adaptation and trajectory prediction under incomplete observations [50].

Despite these advances, existing retrieval-driven methods primarily target forecasting, prediction, or optimization tasks and rarely address telemetry reconstruction under bursty loss, mobility-induced drift, and fluctuating wireless conditions. Moreover, the physical and environmental context is typically absent from retrieval design. In contrast, V2LLM incorporates DT-aware retrieval and structured prompting to align telemetry reconstruction with both motion dynamics and communication context in V2X environments.

C. Digital Twin–Driven Context Modeling

Digital twins have become integral to intelligent transportation systems by maintaining real-time virtual counterparts of vehicles and infrastructure for prediction, diagnosis, and control. Foundational works established hierarchical architectures linking physical assets, communication gateways, and cloud-based digital spaces to improve safety and mobility across large-scale networks [60], [61]. Edge-assisted DTs have further enabled efficient data exchange among connected and autonomous vehicles (CAVs), supporting cooperative sensing and learning under latency and bandwidth constraints [62]. Recent studies enhance DT fidelity and safety through delay-aware synchronization, stability analysis, and control barrier–based safety strategies [63], while others integrate AI-driven analytics for real-time vehicle performance evaluation and predictive control [64]. Beyond transportation, DT frameworks in industrial and wireless systems highlight the integration of AI, federated learning, and blockchain for secure, distributed intelligence [52], [65], [66]. Recent studies have also explored DT-assisted LLM and continual learning pipelines for industrial fault diagnosis, where DT models support abnormal data correction and retrieval-augmented LLMs compensate for missing sensor observations within federated continual settings [67]. DT-assisted federated training has also emerged as a key paradigm for optimizing vehicular learning accuracy and energy efficiency under dynamic conditions [68].

Recent advances further demonstrate the integration of digital twins with learning-based control and optimization, such as hybrid DT frameworks for 5G networks [69], which combine data reconstruction, prediction, and reinforcement learning for system-level decision-making. Similarly, hierarchical DT architectures for 6G V2X systems [70] highlight the role of DTs in scalable network management and real-time optimization.

Despite these advances, existing DT-based frameworks primarily focus on control, optimization, or system management, and do not leverage DT representations as semantic priors for generative inference. In contrast, the proposed V2LLM framework embeds DT-derived context directly into a retrieval-augmented LLM pipeline, enabling physically grounded and context-aware telemetry reconstruction in dynamic V2X environments.

TABLE I: Comparison of Key Features Across Related Works and the Proposed V2LLM Framework

Approach / Framework	Semantic-Aware	Digital Twin	RAG/ Memory Augmented	Federated Learning	Continual Adaptation	Privacy Preservation
GRU-D [30] / BRITS [31]	X	X	X	X	X	X
SAITS [33] / TimesNet [34]	✓	X	X	X	X	X
MemDeT [51] / MAuGANs [45]	✓	X	✓	X	X	X
DAG-Net [47]	✓	X	✓	X	X	X
DT-IIoT [52] / TwinFed [53]	X	✓	X	✓	X	✓
FedGau [54] / ESAFL [55]	X	X	X	✓	X	✓
Re-Fed+ [56] / F2SCIL [57]	X	X	X	✓	✓	✓
MeCo [58] / SacFL [59]	X	X	X	✓	✓	✓
V2LLM (PROPOSED)	✓	✓	✓	✓	✓	✓

D. Federated Learning for Vehicular Networks

Federated learning has become a key enabler of privacy-preserving intelligence in CAVs, allowing collaborative model training without raw data exchange. By aggregating locally trained models from vehicles and roadside units (RSUs), FL mitigates communication overhead and protects sensitive information while improving model generalization across diverse driving environments [71]. Recent advances in hierarchical and asynchronous aggregation have enhanced the scalability of vehicular FL under mobility-induced heterogeneity. Hierarchical frameworks coordinate cross-city learning with adaptive scheduling to accelerate convergence and reduce communication cost [54], while semi-asynchronous schemes address straggler effects and improve stability in dynamic networks [55], [72]. Digital-twin-assisted FL further aligns model updates with real-time virtual replicas, enabling predictive synchronization and efficient resource allocation for safety-critical tasks [73], [74]. Beyond ground vehicles, hybrid federated and centralized learning in unmanned aerial vehicle (UAV)-assisted networks demonstrates energy-efficient adaptation for aerial traffic prediction [75], and FL-reinforcement learning integration optimizes cooperative task offloading and communication efficiency in CAV systems [76].

Despite these advances, most vehicular FL frameworks remain episodic and lack continual adaptation to evolving contexts. The proposed V2LLM bridges this gap by integrating DT-aware FCL to ensure resilient, privacy-preserving, and semantically consistent fault diagnosis in dynamic V2X environments.

E. Federated Continual Learning

Traditional federated learning assumes static and stationary data distributions, which limits its adaptability to evolving vehicular environments. In contrast, FCL enables collaborative, lifelong model refinement across clients that continuously encounter new tasks or domains. By fusing local adaptation and global knowledge retention, FCL mitigates catastrophic forgetting and sustains model robustness under dynamic conditions. Recent FCL studies have introduced mechanisms such as synergistic replay and sample caching to balance old and new knowledge during domain shifts [56], [77]. Prototype-based aggregation and hierarchical parameter fusion further enhance incremental learning across mobile or resource-constrained edge devices [28], [57]. Self-adaptive frameworks like SacFL employ task-aware encoders and contrastive learning for autonomous shift detection, reducing storage overhead on end devices [59]. Similarly, industrial and IoT systems leverage meta-consolidation and task-specific transfer to achieve cost-efficient continual adaptation [58]. Hybrid architectures also integrate digital-twin-assisted coordination, where virtual replicas support synchronization and stability in edge-federated settings [53].

Comprehensive surveys on FCL highlight emerging directions in asynchronous, clustered, and spatiotemporal continual learning [25], [78]. Building upon these advances, V2LLM fuses DT-aware reasoning, retrieval augmentation, and continual adaptation to achieve resilient telemetry recovery and fault diagnosis in evolving V2X environments.

F. Comparative Summary of Related Work

To illustrate the distinct contributions of the proposed V2LLM framework, Table I summarizes the key capabilities supported by representative prior works across the five research streams. The proposed framework uniquely integrates all essential components—semantic reasoning, digital-twin grounding, retrieval augmentation, federated collaboration, and continual adaptation—within a unified architecture for resilient V2X telemetry intelligence.

III. SYSTEM MODEL

We consider a semi-urban vehicular network comprising N connected vehicles $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$ and R infrastructure-side RSUs $\mathcal{R} = \{r_1, r_2, \ldots, r_R\}$ deployed across a geographic region. Each vehicle $v_i \in \mathcal{V}$ is equipped with an onboard unit (OBU) that periodically transmits telemetry vectors $\mathbf{x}_i(t)$ to its nearest RSU r_j via a vehicle-to-infrastructure (V2I) wireless link, such as dedicated short-range communications (DSRC) or cellular V2X. The telemetry data includes both mobility features (e.g., position, velocity, acceleration) and communication-level indicators such as RSSI and PLR, which are essential for downstream tasks like fault detection, traffic state estimation, and collaborative inference. To maintain accurate environmental awareness under such uncertainty, each RSU maintains a localized DT model that estimates real-time traffic and communication dynamics. The RSU also constructs retrieval-augmented prompts using historical data and DT priors to recover missing telemetry via an LLM-based generative module. Fig. 1 summarizes telemetry transmission, DT-assisted retrieval, LLM-based reconstruction, and federated diagnosis across RSUs.

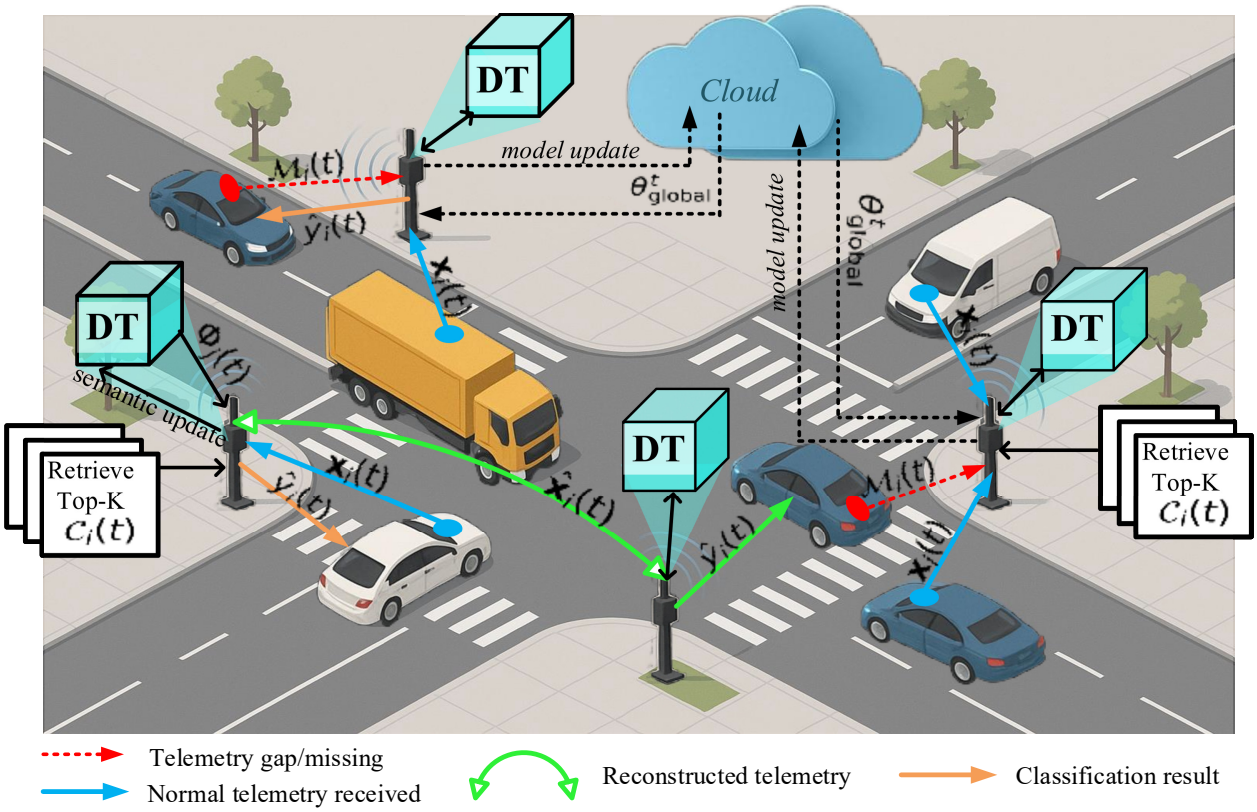


Fig. 1: Illustration of the proposed V2LLM architecture showing telemetry flow, digital twin integration, memory retrieval, and federated fault diagnosis.

A. Vehicular Telemetry and Communication Loss

Let T denote the global time horizon. At each time step $t \in \{1, 2, \ldots, T\}$, vehicle v_i generates a telemetry vector $\mathbf{x}_i(t) \in \mathbb{R}^d$ comprising motion and communication attributes [79], [80], e.g.,

$$\mathbf{x}_i(t) = [p_x(t), p_y(t), v(t), a(t), \theta(t), \text{RSSI}(t), \text{PLR}(t), \ldots], \quad (1)$$

where $p_x(t), p_y(t)$ denote position, $v(t)$ is velocity, $a(t)$ is acceleration, $\theta(t)$ is heading angle, and RSSI and PLR capture signal quality metrics. The feature dimension d may vary by application. Reception of telemetry at RSUs is indicated by

$$\delta_{i,j}(t) = \begin{cases} 1, & \text{if } \mathbf{x}_i(t) \text{ is received at RSU } r_j, \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

Each vehicle v_i is associated with its serving RSU r_j, where $j \in \{1, \ldots, R\}$ indexes the set of RSUs, based on proximity or communication range at time t. Accordingly, $\delta_{i,j}(t)$ explicitly captures the reception of telemetry from vehicle v_i at RSU r_j, enabling the system (or cloud aggregator) to distinguish observations across different RSUs. Each RSU r_j maintains a telemetry buffer $\mathcal{X}_j(t)$ consisting of all received vectors and constructs a history window per vehicle as

$$\mathcal{H}_i(t) = \{\mathbf{x}_i(t-k), \ldots, \mathbf{x}_i(t-1)\} \quad (3)$$

A telemetry gap $\mathcal{M}_i(t)$ occurs when τ consecutive vectors are missing [81], represented as

$$\delta_{i,j}(t-\tau+1) = \cdots = \delta_{i,j}(t) = 0 \quad (4)$$

The recovery objective is to approximate the missing sequence using available context as

$$\mathcal{M}_i(t) = \{\mathbf{x}_i(t-\tau+1), \ldots, \mathbf{x}_i(t)\} \quad (5)$$

B. Retrieval Memory and Digital Twin State

To support gap recovery, each RSU maintains a bounded historical memory $\mathcal{K}_j$ containing M telemetry windows and their corresponding ground truths as

$$\mathcal{K}_j = \left\{ \left(\mathcal{H}_i^{(m)}, \mathbf{x}_i^{(m)} \right) \right\}_{m=1}^{M}. \quad (6)$$

Similarity-based retrieval is used to identify relevant past trajectories for reconstruction, where a similarity score $s(\cdot, \cdot)$ is computed between the current history $\mathcal{H}_i(t)$ and each memory entry [82]. The top-K matches define the context set is given as

$$\mathcal{C}_i(t) = \text{Top-K}(\mathcal{H}_i(t), \mathcal{K}_j). \quad (7)$$

Top-K retrieval provides multiple high-similarity contexts instead of a single historical match, improving robustness to partial mismatches and noisy observations in non-i.i.d. vehicular environments. Each RSU is also equipped with a DT module that estimates macro-level traffic and wireless conditions over time. The DT state at time t is denoted as

$$\Phi_j(t) = (\mu_j(t), \sigma_j(t), \rho_j(t), \psi_j(t)), \quad (8)$$

where μ_j is average vehicle speed, σ_j is local density, ρ_j is link reliability, and ψ_j captures lane occupancy. These features are computed from recent telemetry within a sliding time window and smoothed using exponential averaging [83]. The DT state is embedded into a fixed-size vector $\mathbf{z}_j(t)$.

C. Telemetry Reconstruction Objective

When a telemetry gap $\mathcal{M}_i(t)$ is detected, the RSU constructs a structured input for LLM-based reconstruction. This input $\mathcal{Q}_i(t)$ is composed of

$$\mathcal{Q}_i(t) = \text{Prompt}\left(\mathcal{H}_i(t), \mathcal{C}_i(t), \mathbf{z}_j(t)\right) \tag{9}$$

where $\mathcal{H}_i(t)$ is the recent telemetry history, $\mathcal{C}_i(t)$ is the top-K retrieval context, $\mathbf{z}_j(t)$ is the DT embedding, and *metadata* includes auxiliary information such as vehicle identification (ID) and timestamps. The missing telemetry $\hat{\mathcal{M}}_i(t)$ is then generated autoregressively by a LLM [84], [85], which predicts each step based on the prompt context [81], as,

$$\hat{\mathcal{M}}_i(t) = \text{LLM}(\mathcal{Q}_i(t)). \tag{10}$$

This reconstructed sequence is forwarded to downstream classifiers, and optionally, the RSU contributes to a federated learning round for continual model adaptation. The operational layers and detailed mechanisms of this end-to-end process are described in Section V.

IV. Problem Formulation & Solution Approach

In real-world vehicular networks, the effectiveness of data-driven fault diagnosis, anomaly detection, and decision-making depends critically on the availability of complete, fine-grained telemetry streams from connected vehicles. These telemetry vectors encapsulate essential spatio-temporal features such as position, velocity, acceleration, heading, as well as communication indicators like RSSI and PLR. In contrast, due to environmental dynamics, wireless congestion, deep fading, and hardware malfunctions, telemetry reception at the infrastructure side often suffers from bursty losses. Such gaps in sensor data introduce ambiguity in downstream tasks and degrade the performance of learning-based systems responsible for fault classification or traffic-state estimation.

A. Problem definition

Let $\mathbf{x}_i(t) \in \mathbb{R}^d$ denote the telemetry vector transmitted by vehicle v_i at time t, where d is the number of telemetry features. As defined in Eq. (2), the reception of $\mathbf{x}_i(t)$ at the nearest RSU r_j is indicated by a binary mask $\delta_i(t) \in \{0, 1\}$, with $\delta_i(t) = 0$ implying packet loss. A telemetry gap $\mathcal{M}_i(t)$ of length τ is declared when

$$\delta_i(t-\tau+1) = \delta_i(t-\tau+2) = \cdots = \delta_i(t) = 0. \tag{11}$$

The objective is to reconstruct the missing segment $\mathcal{M}_i(t) = \{\mathbf{x}_i(t-\tau+1), \ldots, \mathbf{x}_i(t)\}$, as defined earlier in Eq. (5), using the information available at RSU r_j.

Let $\mathcal{H}_i(t)$ denote the history buffer of valid past telemetry received from vehicle v_i (Eq. (3)). Let $\mathcal{K}_j$ be the memory bank of trajectory-context pairs, and let $\Phi_j(t)$ denote the real-time DT state, as defined in Eq. (8).

The recovery goal is to generate a faithful approximation $\hat{\mathcal{M}}_i(t)$ of the missing telemetry, leveraging history $\mathcal{H}_i(t)$, context $\mathcal{C}_i(t) \subset \mathcal{K}_j$, and DT guidance through its embedded representation $\mathbf{z}_j(t)$. The reconstruction follows the prompt construction and autoregressive generation process defined earlier in Eq. (9), where the LLM generates the missing telemetry conditioned on historical context, retrieved memory, and DT embeddings.

To improve the contextual relevance of retrieval, the DT state is encoded into $\mathbf{z}_j(t) = f_{\text{enc}}(\Phi_j(t))$, and hybrid similarity is computed as

$$s' = \alpha \cdot \text{sim}(\mathcal{H}_i(t), \mathcal{H}_i^{(m)}) + (1-\alpha) \cdot \text{sim}(\mathbf{z}_j(t), \mathbf{z}_j^{(m)}), \tag{12}$$

where $\alpha \in [0, 1]$ balances trajectory and environmental similarity. The top-K samples $\mathcal{C}_i(t)$ are selected accordingly.

The considered problem involves: (i) recovering missing high-dimensional telemetry under bursty loss, (ii) incorporating temporal and environmental context into reconstruction under partial observability, and (iii) supporting decentralized classification under evolving non-i.i.d. data distributions.

Conventional interpolation or statistical imputation methods fall short in such settings, as they ignore the semantic, contextual, and structural aspects of telemetry data. Furthermore, traditional learning architectures struggle to generalize under heterogeneous, distributed conditions where telemetry distributions shift over time, and centralized access is restricted.

B. Solution Approach

To address the above challenges, we propose *V2LLM*, a layered and integrated framework that combines DT modeling, RAG, large autoregressive language models, and FCL for resilient and context-aware telemetry recovery and diagnosis. V2LLM operates through five tightly coupled layers:
(1) vehicle telemetry capture and loss detection;
(2) DT-based state summarization;
(3) memory-augmented prompt construction via similarity-based retrieval;
(4) transformer-driven autoregressive reconstruction of missing data; and
(5) federated continual classification of real and recovered telemetry.

V2LLM combines prompt-based semantic reasoning with transformer modeling for robust telemetry recovery, while its federated design enables adaptive, privacy-preserving fault classification. The full architecture is detailed in the following sections.

V. Proposed V2LLM Framework

V2LLM is organized into five functional layers covering telemetry preprocessing, DT estimation, retrieval-augmented prompting, autoregressive reconstruction, and federated continual classification:

A. Layer 1: Vehicular Telemetry Generation Model

This layer models how vehicular telemetry is generated, transmitted, occasionally lost, and preprocessed for downstream modules such as DT estimation, memory retrieval, and LLM-based reconstruction. As defined in Eq. (1)–(5), each vehicle v_i transmits a telemetry vector $\mathbf{x}_i(t)$ at time t, and a telemetry gap $\mathcal{M}_i(t)$ is declared when τ consecutive packets are lost, i.e., $\delta_i(t') = 0$.

Each RSU maintains a history buffer $\mathcal{H}_i(t)$ of the most recent k successfully received telemetry vectors from vehicle v_i, as in

Eq. (3). This history provides a temporally coherent window for constructing LLM prompts.

To interface with LLMs, telemetry vectors are serialized into a structured, natural language-like format. Each vector $\mathbf{x}_i(t')$ is flattened and tokenized as

$$\text{Tokenize}(\mathbf{x}_i(t')) = \texttt{[t']} : [x_1, x_2, \ldots, x_d] \tag{13}$$

Before serialization, heterogeneous telemetry features are normalized to mitigate scale imbalance across mobility and communication variables. Let $x_{i,d}(t)$, $d \in \{1, \ldots, D\}$, denote the d-th component of telemetry vector $\mathbf{x}_i(t)$. For continuous variables with broad dynamic ranges (e.g., position, velocity, acceleration, RSSI, and SNR), z-score normalization is applied as

$$\tilde{x}_{i,d}(t) = \frac{x_{i,d}(t) - \mu_d}{\sigma_d}, \tag{14}$$

where μ_d and σ_d denote feature-wise statistics computed from the training telemetry set. For bounded numerical variables such as PLR and occupancy-related indicators, min–max normalization is used as

$$\tilde{x}_{i,d}(t) = \frac{x_{i,d}(t) - x_d^{\min}}{x_d^{\max} - x_d^{\min}}. \tag{15}$$

The normalized telemetry representation

$$\tilde{\mathbf{x}}_i(t) = [\tilde{x}_{i,1}(t), \ldots, \tilde{x}_{i,D}(t)] \tag{16}$$

is subsequently serialized into the prompt token sequence used for autoregressive reconstruction.

The complete prompt is:

```
Vehicle ID: v_i
Time window: [t-k, t-1]
[t-k] : [x_1, ..., x_d]
...
[t-1] : [x_1, ..., --, --]
Predict next state:
```

Missing values are denoted with "$--$" to signal reconstruction regions to the LLM. Unlike conventional approaches that use matrix-based inputs or opaque sequence models (e.g., LSTMs, GRUs), our method reformulates telemetry recovery as a semantically guided, autoregressive generation task. The prompt format exploits LLMs' instruction-following ability and interpretability via a line-by-line, JSON-like structure. Prompt construction scales as $\mathcal{O}(k \cdot d)$. To meet the transformer's context window limit $L_{\max}$ (e.g., 1024 tokens), we ensure

$$|\mathcal{T}(\mathcal{Q}_i(t))| \leq L_{\max} \tag{17}$$

A detailed example of the prompt appears in Example V-C. **Algorithm 1** and Fig. 2 illustrate the full Layer 1 pipeline—starting from telemetry reception, through history buffer extraction, to structured prompt generation for LLM-based reconstruction.

B. Layer 2 – Digital Twin-Based State Estimation

This layer introduces a DT model at each RSU r_j to capture environmental context from vehicular and communication dynamics. The DT state $\Phi_j(t)$, defined in Eq. (8), includes moving average speed $\mu_j(t)$, normalized vehicle density $\sigma_j(t)$, link reliability $\rho_j(t)$, and lane occupancy vector $\psi_j(t)$. These features are computed from recently received telemetry within a sliding window of w seconds.

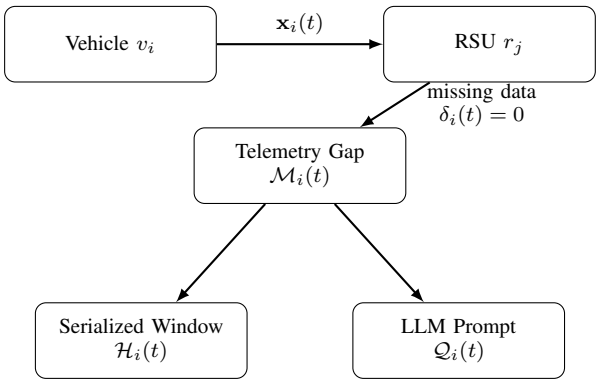


Fig. 2: *Layer 1* pipeline from raw telemetry to LLM prompt.

Algorithm 1 *Layer 1*: Telemetry-to-LLM Prompt Construction

Input: History length k, telemetry buffer $\mathcal{H}_i(t)$, feature dimension d.
Output: Prompt $\mathcal{Q}_i(t)$ for missing step t.
Initialize empty prompt $\mathcal{Q}_i(t) \leftarrow$.
Append metadata lines (vehicle ID and time window).
for $\ell = t - k$ to $t - 1$ **do**
    Serialize $\mathbf{x}_i(\ell)$ using Eq. (13).
    Append to prompt.
end for
Append "`Predict next state:`" marker.
return $\mathcal{Q}_i(t)$

Let $\mathcal{X}_j(t)$ denote the set of telemetry vectors received at RSU r_j within the most recent sliding window. The DT state $\Phi_j(t)$ is defined as a tuple of four components that jointly capture macro-level traffic and communication dynamics. Specifically, $\mu_j(t)$ denotes the average velocity of vehicles currently connected to RSU r_j, $\sigma_j(t)$ represents the vehicular density normalized over the RSU's coverage area A_j, $\rho_j(t)$ quantifies the wireless link reliability based on observed packet loss rates, and $\psi_j(t)$ is a normalized lane occupancy vector capturing the distribution of vehicles across the L lane segments. These components are computed respectively as

$$\mu_j(t) = \frac{1}{|\mathcal{X}_j(t)|} \sum_{\mathbf{x}_i \in \mathcal{X}_j(t)} v_i(t), \tag{18}$$

$$\sigma_j(t) = \frac{|\mathcal{X}_j(t)|}{A_j}, \tag{19}$$

$$\rho_j(t) = 1 - \frac{1}{|\mathcal{X}_j(t)|} \sum_{\mathbf{x}_i \in \mathcal{X}_j(t)} \text{PLR}_i(t), \tag{20}$$

$$\psi_j(t)[\ell] = \frac{N_\ell(t)}{N_{\max}}, \quad \forall \ell \in \{1, \ldots, L\}, \tag{21}$$

where A_j is the RSU's coverage area, $N_\ell(t)$ is the number of vehicles in lane ℓ, and $N_{\max}$ is the maximum observed lane occupancy used for normalization. To mitigate short-term fluctuations, each DT component is smoothed using an exponential moving average (EMA) as

$$X(t) \leftarrow \lambda X(t-1) + (1 - \lambda)\hat{X}(t), \tag{22}$$

Algorithm 2 *Layer 2*: DT Update and Embedding at RSU

1: **Input:** Recent telemetry $\mathcal{X}_j(t)$, previous state $\Phi_j(t-1)$.
2: **Output:** Updated DT embedding $\mathbf{z}_j(t)$.
3: Compute $\hat{\mu}_j(t)$, $\hat{\sigma}_j(t)$, $\hat{\rho}_j(t)$, $\hat{\psi}_j(t)$ using Eqs. (18)–(21).
4: Smooth each component using EMA in Eq. (22).
5: Form DT tuple $\Phi_j(t)$ as in Eq. (8).
6: Compute embedding $\mathbf{z}_j(t)$ using Eq. (23).
7: **return** $\mathbf{z}_j(t)$

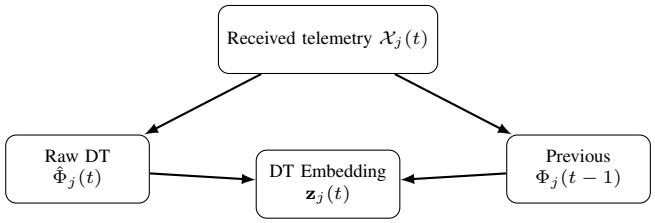


Fig. 3: *Layer 2* RSU-level DT computation and embedding.

where $\hat{X}(t)$ is the most recent raw estimate and $\lambda \in [0,1]$ is a smoothing hyperparameter. The tuple $\Phi_j(t)$ is then mapped to a dense semantic embedding $\mathbf{z}_j(t) \in \mathbb{R}^{d_z}$ using a shared encoder as

$$\mathbf{z}_j(t) = f_{\text{enc}}(\Phi_j(t)) = \text{MLP}(\text{concat}[\mu_j(t), \sigma_j(t), \rho_j(t), \psi_j(t)]). \tag{23}$$

This compact representation is incorporated into the LLM prompt as defined in Eq. (9), complementing the local telemetry history $\mathcal{H}_i(t)$ and the retrieved context $\mathcal{C}_i(t)$ with environment-aware context. The DT embedding provides environment-level context for reconstruction by encoding traffic density, communication reliability, and lane occupancy into a compact representation compatible with prompt conditioning.

Algorithm 2 outlines the computation pipeline—beginning with the extraction of raw environmental metrics from $\mathcal{X}_j(t)$, followed by exponential smoothing, and culminating in the generation of the DT embedding. Fig. 3 provides a schematic overview of this process, showing how $\Phi_j(t)$ is updated from historical and current statistics to inform the next stage of LLM-driven telemetry reconstruction.

C. Layer 3– Retrieval-Augmented Prompting (RAG)

The third layer of the V2LLM framework constructs enriched, context-aware prompts for LLM-based telemetry reconstruction by employing a RAG paradigm. It merges recent vehicle telemetry with relevant historical samples and environmental embeddings derived from the DT model (Layer 2), producing a semantically dense and temporally aligned prompt optimized for transformer-based inference. Each RSU r_j maintains a local memory bank $\mathcal{K}_j$ of size M, where

$$\mathcal{K}_j = \left\{ \left(\mathcal{H}_i^{(m)}, \mathbf{x}_i^{(m)}, \Phi_j^{(m)} \right) \mid m = 1, \ldots, M \right\} \tag{24}$$

Given a telemetry gap for vehicle v_i at time t, a query vector is formed as

$$\text{Query}_i(t) = (\mathcal{H}_i(t), \Phi_j(t)), \tag{25}$$

where $\mathcal{H}_i(t)$ is the recent history and $\Phi_j(t)$ is the current DT state. To select relevant context, a joint similarity score is computed as

$$\text{sim}\left((\mathcal{H}_i(t), \Phi_j(t)), (\mathcal{H}_i^{(m)}, \Phi_j^{(m)}) \right) = \alpha \cdot \cos\left(\mathcal{H}_i(t), \mathcal{H}_i^{(m)} \right) + (1-\alpha) \cdot \cos\left(\mathbf{z}_j(t), \mathbf{z}_j^{(m)} \right) \tag{26}$$

where $\alpha \in [0,1]$ balances trajectory and environment similarity. To enable consistent similarity computation, each trajectory segment $\mathcal{H}_i(t)$ is represented as a fixed-length vector obtained by concatenating the most recent k telemetry observations, i.e., $\mathcal{H}_i(t) \in \mathbb{R}^{k \cdot d}$, where d denotes the telemetry dimension. Prior to similarity evaluation, both trajectory vectors and DT embeddings are ℓ_2-normalized to ensure scale-invariant comparison and compatibility with cosine similarity.

The DT state $\Phi_j(t)$ is encoded into a compact embedding $\mathbf{z}_j(t)$ via a lightweight MLP encoder, and similarly normalized before computing $\cos(\cdot,\cdot)$, preventing any single feature from dominating the similarity score. The top-K entries (i.e., the K most similar past examples) define the retrieval set as

$$\mathcal{C}_i(t) = \text{Top-K}(\text{sim}, \mathcal{K}_j)\,. \tag{27}$$

The number of retrieved contexts K is selected based on empirical evaluation, as validated in the Top-K retrieval ablation study (Fig. 16), where performance improves with increasing K up to a moderate value before saturating due to prompt length constraints. In practical V2X deployments, sufficiently similar historical scenes may not always exist within the bounded memory bank $\mathcal{K}_j$, particularly under rare traffic patterns, evolving wireless conditions, or previously unseen operating scenarios. In such cases, retrieval provides weaker contextual guidance rather than acting as a hard constraint. Importantly, the proposed framework does not rely solely on retrieved memory for reconstruction. The prompt jointly conditions the LLM on recent telemetry history $\mathcal{H}_i(t)$, DT embedding $\mathbf{z}_j(t)$, and retrieved context $\mathcal{C}_i(t)$. Consequently, when memory matching is weak, generation remains grounded by local temporal continuity and real-time environmental context, reducing overdependence on imperfect retrieval and mitigating the likelihood of unconstrained telemetry recovery. The LLM prompt structure defined in Eq. (9) is implemented by serializing the history $\mathcal{H}_i(t)$, top-K retrieval context $\mathcal{C}_i(t)$, and DT embedding $\mathbf{z}_j(t)$ into a line-based, transformer-compatible format. Formally, each retrieved entry $(\mathcal{H}_i^{(m)}, \mathbf{x}_i^{(m)}, \Phi_j^{(m)}) \in \mathcal{C}_i(t)$ is serialized as a structured trajectory-response pair, where each trajectory segment is paired with its corresponding next-step telemetry target. Historical states and corresponding outputs are encoded in a line-wise format consistent with the telemetry representation. These retrieved sequences are appended in temporal order, preserving alignment between input context and predicted outputs. An illustrative example is shown below:

Serialized Prompt for Autoregressive Generation

```
Vehicle ID: v3
Time Window: [37, 38, 39]
[37] : [52.1, 13.4, 11.2, 0.3, 45.0,
-74.5, 0.02]
[38] : [52.3, 13.5, 11.4, 0.2, 44.8,
-75.1, 0.01]
[39] : [52.6, 13.8, 11.7, 0.1, 44.5,
--, --]
Context 1 : [52.9, 13.6, 11.0, 0.1,
44.9, -74.8, 0.02]
Context 2 : [53.0, 13.7, 10.8, 0.3,
44.7, -75.2, 0.03]
DT Embedding : [0.23, 0.58, 0.01,
0.76, 0.12]
Predict next state:
```

To prepare the prompt for transformer inference, we serialize the structured content into a flat, position-aware sequence, ensuring temporal coherence. Each element in $\mathcal{H}_i(t)$ and $\mathcal{C}_i(t)$ is formatted line-by-line and tokenized using byte-pair encoding (BPE). The resulting token sequence $[t_1, t_2, \ldots, t_L]$ is mapped to embeddings $\mathbf{E} = [\mathbf{e}_1, \ldots, \mathbf{e}_L] \in \mathbb{R}^{L \times d_{\text{model}}}$, which are processed by a transformer stack.

In our implementation, tokenization is performed using BPE consistent with the pretrained GPT-2 tokenizer. Telemetry values are preserved in continuous form and serialized as feature-wise normalized floating-point numbers (using z-score or min-max preprocessing, depending on the variable type) without discretization. Missing values in telemetry sequences are explicitly represented using placeholder symbols (e.g., "– "), allowing the model to distinguish between observed and unobserved entries during autoregressive generation.

The resulting prompt length depends on the history window size and the number of retrieved contexts. In our setup, the average tokenized prompt length ranges from approximately 80 to 150 tokens, remaining within the effective context window of the underlying language model while preserving sufficient contextual information for reconstruction. Within each attention layer, contextual interactions are modeled using

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right)\mathbf{V}, \tag{28}$$

where $\mathbf{Q} = \mathbf{E}\mathbf{W}^Q$, $\mathbf{K} = \mathbf{E}\mathbf{W}^K$, $\mathbf{V} = \mathbf{E}\mathbf{W}^V$ are the query, key, and value projections, and d_k is the attention dimensionality. The transformer layers model interactions among historical steps, retrieved contexts, and the masked target token(s), allowing the LLM to autoregressively predict missing entries.

Algorithm 3 outlines the process of selecting top-K samples and generating $\mathcal{Q}_i(t)$. Fig. 4 presents the corresponding schematic, showing how telemetry gaps are supplemented with historical memory and DT context to form an LLM-compatible prompt. The resulting prompt combines telemetry history, retrieved memory, and DT context into a transformer-compatible representation for autoregressive reconstruction.

Algorithm 3 *Layer 3*: Prompt Construction via RAG

1: **Input:** $\mathcal{H}_i(t)$, $\mathbf{z}_j(t)$, $\mathcal{K}_j$.
2: **Output:** Prompt $\mathcal{Q}_i(t)$.
3: **for** $m = 1$ to M **do**
4: Compute similarity using Eq.(26).
5: **end for**
6: Select top-K samples to get $\mathcal{C}_i(t)$ using Eq.(27).
7: Generate $\mathcal{Q}_i(t)$.
8: **return** $\mathcal{Q}_i(t)$

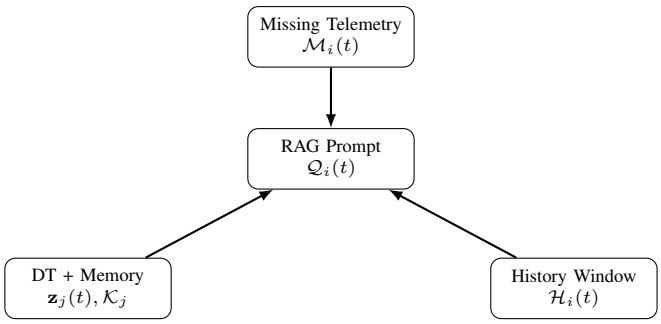


Fig. 4: *Layer 3* RAG prompt generation using historical and environmental context.

D. Layer 4 – Autoregressive Generation via LLM

This layer performs the core inference task in V2LLM: generating missing telemetry vectors autoregressively using the context-rich prompt $\mathcal{Q}_i(t)$ constructed in Layer 3. Leveraging a transformer-based decoder, this module fills the telemetry gap by sequentially predicting the missing components conditioned on the serialized history, top-K retrieved samples, and real-time DT context.

Given a prompt $\mathcal{Q}_i(t)$ structured as a token sequence $[t_1, t_2, ..., t_L]$, the LLM first tokenizes the input using byte-pair encoding (BPE) or similar segmentation into L tokens. These are embedded as

$$\mathbf{E} = [\mathbf{e}_1, \ldots, \mathbf{e}_L] \in \mathbb{R}^{L \times d_{\text{model}}}, \tag{29}$$

where d_{model} is the transformer's hidden size.

The transformer stack processes $\mathbf{E}$ through N layers, each comprising a masked multi-head self-attention block and a position-wise feedforward network. The attention computation follows the scaled dot-product formulation as in Eq.(28). For each token embedding $\mathbf{e}_i$, attention is computed as

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right)\mathbf{V}, \tag{30}$$

where $\mathbf{Q} = \mathbf{E}\mathbf{W}^Q$, $\mathbf{K} = \mathbf{E}\mathbf{W}^K$, $\mathbf{V} = \mathbf{E}\mathbf{W}^V$, and d_k is the dimensionality of queries and keys.

After each decoder block, the updated representation $\mathbf{h}_L$ for the final token is passed through a linear projection and softmax layer to obtain the probability distribution over the vocabulary as

$$P(x_d^{(t)}) = \text{softmax}(\mathbf{W}_o \cdot \mathbf{h}_L + \mathbf{b}_o). \tag{31}$$

The token with the highest probability is selected (greedy decoding) or sampled using temperature scaling or nucleus sampling.

Algorithm 4 Sliding-Window Autoregressive Generation

Input: Prompt $\mathcal{Q}_i(t)$, gap length n, LLM model f_{LLM}.
Output: Reconstructed sequence $\hat{\mathcal{M}}_i(t)$
for $\ell = 0$ to $n-1$ **do**
    Tokenize $\mathcal{Q}_i(t+\ell)$.
    $\hat{\mathbf{x}}_i(t+\ell) \leftarrow f_{\text{LLM}}(\mathcal{Q}_i(t+\ell))$.
    $\mathcal{Q}_i(t+\ell+1) \leftarrow \mathcal{Q}_i(t+\ell) \cup \text{Tokenize}(\hat{\mathbf{x}}_i(t+\ell))$.
end for
return $\{\hat{\mathbf{x}}_i(t+\ell)\}_{\ell=0}^{n-1}$.

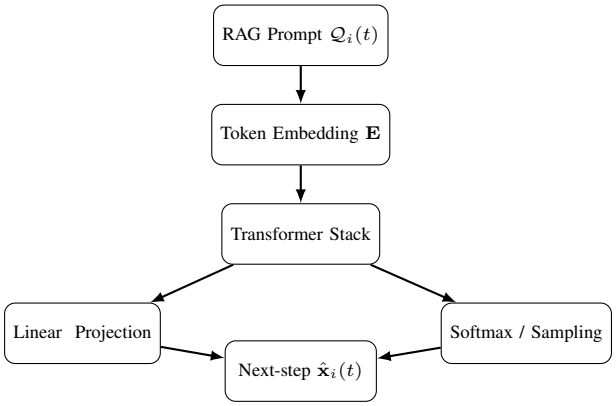


Fig. 5: *Layer 4* Autoregressive generation of telemetry vectors using the LLM decoder.

For multi-step telemetry reconstruction (i.e., filling a telemetry gap of length n), we apply a sliding-window decoding strategy. Each generated vector $\hat{\mathbf{x}}_i(t+\ell)$ is appended to the prompt, forming $\mathcal{Q}_i(t+\ell+1)$ iteratively as

$$\mathcal{Q}_i(t+\ell+1) = \mathcal{Q}_i(t+\ell) \cup \text{Tokenize}(\hat{\mathbf{x}}_i(t+\ell)). \tag{32}$$

This continues until the entire gap $\mathcal{M}_i(t)$ is filled.

The pipeline shown in Fig. 5 illustrates how each telemetry gap is resolved through token embedding, transformer decoding, and autoregressive prediction. **Algorithm 4** outlines the autoregressive decoding process, where the LLM generates missing telemetry vectors by iteratively extending the prompt with each new prediction. The decoder supports variable-length reconstruction through iterative prompt extension and context-conditioned autoregressive generation.

Physical consistency is preserved through contextual conditioning on historical telemetry, retrieved trajectories, and DT embeddings. In rare cases of invalid outputs (e.g., out-of-range values), simple range-based correction is applied to ensure physically plausible values.

E. Layer 5 – Federated Continual Learning (FCL)

The fifth and final layer of the V2LLM framework performs distributed fault classification using telemetry vectors recovered from the previous LLM generation stage. This layer enables real-time, adaptive, and decentralized learning across RSUs by employing an FCL scheme. The primary goals are: (i) to handle non-i.i.d., time-varying data distributions in vehicular networks, (ii) to avoid catastrophic forgetting as telemetry evolves, and (iii) to achieve global generalization while preserving local privacy.

Each RSU r_j maintains a local classifier $f_j(\cdot;\theta_j^t) : \mathbb{R}^d \to \mathcal{Y}$ with parameters θ_j^t, where $\mathcal{Y}$ is the set of traffic or fault condition

Algorithm 5 Layer 5: Federated Continual Learning Loop

Input: Initial global model θ^0, local datasets $\mathcal{D}_{j\,j=1}^{t\,R}$, sync interval T_{sync}, convergence threshold ΔT.
Initialize: $\theta_j^0 \leftarrow \theta^0$ for all RSUs r_j; convergence counters $c_j \leftarrow 0$.
for each round $t = 1$ to T **do**
    for each RSU r_j in parallel **do**
        Set $\theta_j^t \leftarrow \theta_{\text{global}}^{t-1}$.
        for epoch $e = 1$ to E **do**
            for mini-batch $\mathcal{B}_j^t \subset \mathcal{D}_j^t$ **do**
                Compute $\nabla_\theta \mathcal{J}_j^t(\theta_j^t)$ using Eq. (34).
                Update $\theta_j^t \leftarrow \theta_j^t - \eta \nabla_\theta \mathcal{J}_j^t(\theta_j^t)$.
            end for
        end for
        Compute local validation loss $\mathcal{L}_j^{\text{val}}(t)$.
        if $\mathcal{L}_j^{\text{val}}(t)$ not improved **then**
            Increment counter $c_j \leftarrow c_j + 1$.
        else
            Reset counter $c_j \leftarrow 0$.
        end if
        if $c_j \geq \Delta T$ **then**
            Revert $\theta_j^t \leftarrow \theta_{\text{global}}^t$.
        end if
        Update Fisher matrix $F_{j,p}^{t+1}$.
    end for
    Aggregate θ_{global}^t using Eq. (35).
    Broadcast θ_{global}^t to all RSUs.
end for

labels. RSUs receive real or LLM-reconstructed telemetry and train their models on non-i.i.d. datasets $\mathcal{D}_j^t$ representing localized conditions.

The training begins with a cloud server initializing a global model θ^0, which is broadcast to all RSUs. Each RSU r_j uses this initialization to begin training on its local dataset $\mathcal{D}_j^t$ for E epochs. During local updates, stochastic gradient descent is performed over mini-batches $\mathcal{B}_j^t$ drawn from $\mathcal{D}_j^t$. The update rule follows as

$$\theta_j^t \leftarrow \theta_j^{t-1} - \eta \nabla \theta \mathcal{J}_j^t(\theta_j^t) \tag{33}$$

where η is the learning rate and $\mathcal{J}_j^t$ is the task-specific loss as

$$\begin{aligned} \mathcal{J}_j^t(\theta_j^t) = \mathcal{L}_j^t + \lambda_1 \mathcal{R}(\theta_j^t) \\ + \lambda_2 \sum_{k=1}^{t-1} \sum_p F_{j,p}^k \left(\theta_{j,p}^t - \theta_{j,p}^k\right)^2 \end{aligned} \tag{34}$$

where $\mathcal{L}_j^t$ is the supervised loss (cross-entropy), $\mathcal{R}(\cdot)$ is a regularizer (e.g., weight decay), and the final term uses Fisher information $F_{j,p}^k$ to consolidate parameters across rounds via elastic weight consolidation (EWC).

After local training, RSUs optionally compute diagonal Fisher matrices $F_{j,p}^{t+1}$ based on the gradients of the loss. Every T_{sync} steps, RSUs transmit their locally updated models to the central server. The server aggregates them via a weighted average based on the size of local datasets as

$$\theta_{\text{global}}^t = \frac{1}{\sum_{j=1}^R |\mathcal{D}_j^t|} \sum_{j=1}^R |\mathcal{D}_j^t| \cdot \theta_j^t. \tag{35}$$

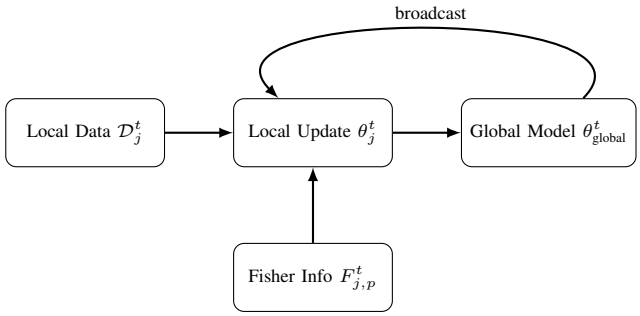


Fig. 6: *Layer 5* Federated continual learning with weighted aggregation and task consolidation.

The updated θ^tglobal is broadcast back to all RSUs for the next training round. If local models diverge significantly, RSUs can revert to the global model to prevent local overfitting or instability.

Algorithm 5 provides the FCL process. Fig. 6 illustrates this workflow, showing how local data and Fisher information are used to update RSU-specific models, which are then synchronized through a shared global model broadcast. To ensure stability and avoid overfitting, each RSU monitors local validation loss trends and halts updates if performance stagnates for ΔT rounds, reverting to the global model if necessary.

In parallel to the classification objective, the LLM used in Layer 4 is adapted from a pretrained autoregressive transformer (e.g., GPT-2) and fine-tuned for structured telemetry reconstruction. The model is trained on synthetically masked prompts, where telemetry gaps are injected into otherwise complete sequences. A masked reconstruction loss is applied only over the missing dimensions, guiding the LLM to learn context-aware prediction under partial observability as

$$\mathcal{L}_{\text{recon}} = -\sum_{t' \in \mathcal{M}_i(t)} \sum_{q \in \Omega_{\text{miss}}(t')} \log P(\hat{x}_{i,q}(t') \mid \mathcal{Q}_i(t)), \qquad (36)$$

where $\Omega_{\text{miss}}(t') \subseteq \{1, \ldots, d\}$ denotes the set of feature indices that are missing at time step t'. Thus, the loss is computed only over the unobserved entries within the corrupted telemetry segment $\mathcal{M}_i(t)$. This auxiliary training aligns with the serialized prompt format developed in earlier layers and complements the federated objective by improving the quality and reliability of reconstructed telemetry used in downstream classification.

F. V2LLM Execution Workflow

To consolidate the layer-wise components of the proposed V2LLM architecture, we present the complete system flow in **Algorithm 6**. The pipeline initiates with raw vehicular telemetry acquisition and proceeds through DT-based state estimation, retrieval-augmented prompt generation, transformer-based reconstruction, and federated continual classification. Each layer interfaces with the next through structured prompts and learned embeddings. Importantly, Layers 3 and 4 incorporate a transformer-based LAM (e.g., GPT-2), which is initialized from a pretrained language model and fine-tuned for structured telemetry reconstruction using a masked reconstruction loss over synthetically injected gaps.

Fig. 7 illustrates the block diagram of the proposed V2LLM framework. The framework begins with vehicle telemetry modeling, which provides historical state information and

Algorithm 6 End-to-End Execution Flow of V2LLM

Input: Telemetry stream $\{\mathbf{x}_i(t)\}_{i=1}^N$, RSU index r_j.
for each vehicle v_i and time step t **do**
  if Telemetry gap $\mathcal{M}_i(t)$ detected **then**
    Extract history buffer $\mathcal{H}_i(t)$ →(**Layer 1**).
    Update DT state $\Phi_j(t)$ and embed as $\mathbf{z}_j(t)$. →(**Layer 2**).
    Retrieve top-K memory $\mathcal{C}_i(t)$ from $\mathcal{K}_j$. →(**Layer 3**).
    Construct enriched prompt $\mathcal{Q}_i(t)$ using $\mathcal{H}_i(t)$, $\mathcal{C}_i(t)$, and $\mathbf{z}_j(t)$. → (**Layer 3**).
    Autoregressively reconstruct gap $\hat{\mathcal{M}}_i(t)$ using LLM. → (**Layer 4**).
    Replace missing segment in $\mathbf{x}_i(t)$ with $\hat{\mathcal{M}}_i(t)$.
  end if
  Classify $\mathbf{x}_i(t)$ using local model $f_j(\cdot; \theta_j^t)$ →(**Layer 5**).
  if $t \bmod T_{\text{sync}} = 0$ **then**
    Aggregate global model θ_{global}^t and broadcast to all RSUs.
  end if
end for

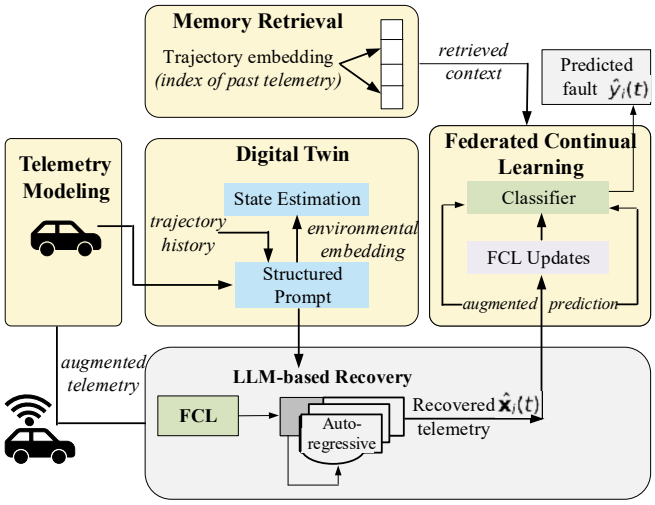


Fig. 7: V2LLM block diagram.

communication features. The digital twin module performs state estimation and generates environmental embeddings to construct structured prompts. In parallel, a local memory module retrieves top-K relevant historical trajectories based on similarity in context. These inputs are fused and passed to the LLM-based recovery module, which performs autoregressive generation to reconstruct missing telemetry segments. The recovered telemetry is then used for downstream fault classification, where FCL enables each RSU to adapt its classifier over time using local updates while mitigating catastrophic forgetting. The classifier outputs the predicted fault label $\hat{y}_i(t)$, enabling resilient and privacy-preserving diagnostics under non-i.i.d. vehicular conditions.

VI. Memory Optimization and Complexity Analysis

To ensure the deployability of V2LLM in real-time V2X environments, we now outline its memory optimization strategies and provide a detailed layer-wise complexity analysis.

A. Memory Management

The deployment of large autoregressive language models such as GPT-2 within V2X systems raises significant challenges in memory utilization and efficiency. To ensure feasibility under constrained RSU and edge resources, the proposed V2LLM framework incorporates several memory-aware design strategies.

First, the memory bank $\mathcal{K}_j$ maintained at each RSU in Layer 3 is implemented as a bounded-size FIFO queue, where the oldest entries are evicted upon insertion of new samples once the limit M is reached. This prevents unbounded growth while retaining recent and relevant context trajectories.

Second, prompt length in Layer 3 and Layer 4 is constrained by the LLM's maximum token limit L_{max}. To prevent prompt overflow, telemetry vectors are serialized using compact formats, and low-priority context lines are truncated when necessary. Token budget management is handled through soft truncation heuristics, prioritizing high-similarity retrievals and embedding vectors.

Third, during LLM fine-tuning (Layer 4), we use gradient checkpointing to reduce memory overhead by recomputing intermediate activations during backpropagation rather than storing all forward passes. This trades compute for memory, improving scalability.

Lastly, for FCL in Layer 5, each RSU retains only the current model θ_j^t and its local Fisher matrix $F_{j,p}^t$, avoiding full gradient history storage.EWC updates are computed incrementally, maintaining a constant memory footprint over time. Although the present implementation employs full-parameter fine-tuning of GPT-2 (124M), the proposed framework is compatible with parameter-efficient adaptation strategies such as LoRA, adapter tuning, or quantized fine-tuning.

B. Computational Complexity Analysis

We now analyze the layer-wise computational complexity of the proposed framework, assuming a vehicle history length k, feature dimension d, DT vector size d_z, memory bank size M, and telemetry gap length n.

1) Telemetry Tokenization: Telemetry serialization has complexity $\mathcal{O}(k \cdot d)$ per vehicle due to flattening and formatting each vector into token form.

2) Layer 2: Digital Twin Embedding.: The DT computation involves computing four scalar summaries (mean, density, reliability, occupancy), followed by an MLP encoding of length d_z, yielding a complexity of $\mathcal{O}(d_z + L)$ per RSU, where L is the number of observable lanes.

3) Retrieval-Augmented Prompting: Similarity scoring between the query and each of M memory entries incurs $\mathcal{O}(M \cdot d)$ operations (assuming normalized vector comparison). Top-K selection adds $\mathcal{O}(M \log K)$, and final prompt construction contributes an additional $\mathcal{O}(k \cdot d + K \cdot d + d_z)$.

4) LLM-Based Autoregressive Generation: Each forward pass through a transformer decoder with L tokens and N layers has complexity $\mathcal{O}(N \cdot L^2 \cdot d_{\text{model}})$. For a gap of length n with sliding-window updates, total decoding cost scales as $\mathcal{O}(n \cdot N \cdot L^2 \cdot d_{\text{model}})$.

5) Federated Continual Learning: Assuming E local epochs over batch size B, the RSU-side update has complexity $\mathcal{O}(E \cdot B \cdot d \cdot |\mathcal{Y}|)$ per round. Global aggregation is linear in the number of RSUs R and model size $|\theta|$.

C. Overall Complexity Summary

Combining the dominant components from each layer, the per-gap end-to-end complexity of V2LLM is approximately estimated as

$$\mathcal{O}\left(n \cdot N \cdot L^2 \cdot d_{\text{model}} + M \cdot d + E \cdot B \cdot d \cdot |\mathcal{Y}|\right) \tag{37}$$

In comparison with benchmark schemes, linear interpolation and KNN-based methods exhibit low computational complexity, typically scaling as $\mathcal{O}(d)$ per timestep, but cannot model temporal dependencies. Sequence-based models such as Seq2Seq and FedAvg-LSTM incur moderate complexity due to recurrent processing, with complexity approximately $\mathcal{O}(T \cdot d \cdot h)$, where T is the sequence length and h is the hidden dimension. In contrast, the proposed V2LLM framework introduces a higher computational cost due to transformer-based autoregressive decoding and retrieval operations, but provides significantly improved contextual modeling and reconstruction accuracy. This highlights a fundamental trade-off between computational cost and reconstruction performance across different approaches.

In terms of runtime, the primary latency in V2LLM arises from autoregressive decoding, while retrieval and DT embedding introduce comparatively low overhead, making the framework practical for near real-time deployment under moderate prompt lengths.

VII. Experimental Setup

To evaluate the proposed V2LLM framework, we design a simulation pipeline that integrates real-world vehicular telemetry with synthetic wireless communication and DT information.

A. Dataset and Telemetry Preparation

We utilize two publicly available datasets: HighD [79] and NGSIM [80], which provide high-resolution vehicle trajectories in highway and urban settings, respectively. Both datasets capture vehicle positions, speeds, and headings at a frequency of 10 Hz. For each vehicle v_i, we define a telemetry vector $\mathbf{x}_i(t) \in \mathbb{R}^{12}$ as: Each telemetry vector $\mathbf{x}_i(t)$ contains 12 features describing the mobility and communication state of vehicle v_i at time t. These include position, motion, and wireless attributes, given as

$$\mathbf{x}_i(t) = [p_x, p_y, v, a, \theta, j, \text{LID}, \text{RSSI}, \text{PLR}, \text{SNR}, \text{p\_size}, \text{v\_type}] \tag{38}$$

where p_x, p_y are positions, v is velocity, a is acceleration, θ is heading, j is jerk, LID denotes the lane identifier, and communication parameters such as RSSI, PLR, and SNR (signal-to-noise ratio) are synthesized to reflect realistic wireless conditions.

To augment the physical traces with wireless characteristics, we estimate the RSSI for each vehicle v_i at time t using the following log-distance path loss model, estimated as

$$\text{RSSI}_i(t) = P_{TX} - 10\gamma \log_{10}\left(\frac{d_{ij}(t)}{d_0}\right) + X_\sigma \tag{39}$$

where $d_{ij}(t)$ is the distance between vehicle v_i and its RSU r_j, P_{TX} is the transmit power, γ is the path loss exponent, and X_σ models shadow fading. PLR is estimated using congestion-aware modeling as

$$\text{PLR}_i(t) = \min\left(1, \alpha \cdot \frac{n_{\text{veh}}(t)}{n_{\text{thresh}}} + \beta \cdot \mathbb{1}_{\text{fading}}\right) \tag{40}$$

where $n_{\text{veh}}(t)$ is the number of vehicles within communication range, and $\mathbb{1}_{\text{fading}}$ indicates channel degradation due to fast fading.

To simulate missing telemetry, we apply a Bernoulli mask per vehicle as

$$\delta_i(t) \sim \text{Bernoulli}(1 - p_{\text{miss}}), \quad p_{\text{miss}} \sim \mathcal{U}(0.1, 0.3) \tag{41}$$

Gaps are detected where consecutive $\delta_i(t) = 0$ over a span τ. For each RSU r_j, DT states are computed using Eqs. (18)–(21), where features are estimated within a sliding window of 5 seconds and smoothed using exponential moving average as

$$X(t) \leftarrow \lambda X(t-1) + (1-\lambda)\hat{X}(t) \tag{42}$$

The resulting tuple $\Phi_j(t)$ is embedded into a latent vector $\mathbf{z}_j(t)$ for prompt augmentation.

We employ a GPT-2 (124M) transformer as the backbone LLM for telemetry reconstruction. In the current implementation, the GPT-2 backbone is adapted using full-parameter fine-tuning. This choice enables controlled evaluation of the proposed DT-aware retrieval, structured prompting, and telemetry reconstruction pipeline without introducing additional adaptation mechanisms associated with parameter-efficient tuning methods. Fine-tuning is performed offline during model preparation, while the adapted model is used for inference during deployment. Prompts are constructed as per Algorithm 1 and Algorithm 3. Synthetic masked telemetry sequences are injected during training to simulate partial observability. The model is optimized using masked reconstruction loss as

$$\mathcal{L}_{\text{recon}} = -\sum_{d \in \mathcal{M}_i(t)} \log P(\hat{x}_{i,d}(t) \mid \mathcal{Q}_i(t)) \tag{43}$$

This objective enables selective learning on missing regions while leveraging context from $\mathcal{H}_i(t)$, $\mathcal{C}_i(t)$, and $\mathbf{z}_j(t)$. To balance learning stability and communication cost, the number of local epochs per federated round is set to $E = 15$, which was empirically found sufficient to prevent local overfitting while ensuring timely synchronization.

The framework employs the GPT-2 (124M) model as the backbone LLM, consisting of 12 transformer decoder layers with multi-head self-attention, a hidden dimension of 768, and 12 attention heads. Telemetry vectors are serialized into token sequences and processed using byte-pair encoding (BPE), enabling the model to capture temporal dependencies and contextual relationships under partial observability. The model is fine-tuned using structured prompts and masked reconstruction objectives tailored to telemetry recovery.

All experiments are conducted on a workstation equipped with an NVIDIA RTX 4070 Ti GPU and 32 GB RAM, with implementation in Python using the PyTorch framework. Training is performed using the Adam optimizer, and the key hyperparameters are summarized in Table II. The reported latency accounts for both prompt construction and autoregressive decoding, reflecting practical deployment conditions.

TABLE II: Simulation Parameters

Parameter	Value	Parameter	Value
N	100	R	5
Freq	10 Hz	T	3000
d	12	d_z	8
$\lvert\mathcal{H}_i(t)\rvert$	5	M	500
K	4	n	up to 5
$L_{\max}$	1024	λ (EMA)	0.8
p_{miss}	$\mathcal{U}(0.1, 0.3)$	A_j	500×500 m^2
P_{TX}	20 dBm	γ	2.0
X_σ	$\mathcal{N}(0, 2)$ dB	d_0	1 m
L	4	n_{thresh}	50
α, β (PLR)	0.6, 0.4	α (sim)	0.5
B	16	E	15
$\lvert\mathcal{Y}\rvert$	4	ΔT	10
λ_1, λ_2	0.01, 0.05	T_{sync}	100
η	10^{-3}	LLM	GPT-2 (124M)

B. Baselines for Comparison

We evaluate the proposed V2LLM framework against the following representative baselines:

- *Linear Interpolation*: Point-wise interpolation between adjacent valid telemetry points.
- *KNN Imputation*: Trajectory reconstruction based on nearest-neighbor search in historical telemetry space.
- *Seq2Seq*: LSTM-based sequence-to-sequence predictor trained on raw trajectories.
- *FedAvg*: Federated LSTM models trained at each RSU with periodic parameter averaging.

C. Evaluation Metrics

We evaluate both telemetry reconstruction and downstream classification using the following metrics:

- **Reconstruction mean square error (MSE) (Single-step)**: Measures the average squared error between reconstructed and ground-truth telemetry, reflecting reconstruction fidelity.

$$\text{MSE}_{\text{rec}} = \frac{1}{|\mathcal{M}|} \sum_{t \in \mathcal{M}} \|\hat{\mathbf{x}}_i(t) - \mathbf{x}_i(t)\|_2^2 \tag{44}$$

- **Telemetry Cosine Similarity**: Captures directional similarity between reconstructed and true telemetry vectors, indicating structural alignment.

$$\text{Sim} = \frac{1}{|\mathcal{M}|} \sum_{t \in \mathcal{M}} \frac{\langle \hat{\mathbf{x}}_i(t), \mathbf{x}_i(t) \rangle}{\|\hat{\mathbf{x}}_i(t)\|_2 \cdot \|\mathbf{x}_i(t)\|_2} \tag{45}$$

- **Classification Accuracy**: Measures the percentage of correctly predicted fault labels, reflecting overall diagnostic performance.
- **Macro F1 Score**: Evaluates classification performance across all classes equally, accounting for class imbalance.

$$\text{F1}_{\text{macro}} = \frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}} \frac{2 \cdot \text{Precision}_y \cdot \text{Recall}_y}{\text{Precision}_y + \text{Recall}_y} \tag{46}$$

  where $\mathcal{Y}$ denotes the set of fault classes.
- **Retrieval Score**: Measures the relevance of retrieved memory contexts to the query, reflecting retrieval quality.

$$\text{Retrieval Score} = \frac{1}{K} \sum_{k=1}^{K} \text{sim}((\mathcal{H}_i, \Phi_j), (\mathcal{H}_i^{(k)}, \Phi_j^{(k)})) \tag{47}$$

  with $\text{sim}(\cdot, \cdot)$ as defined in Eq. (26).

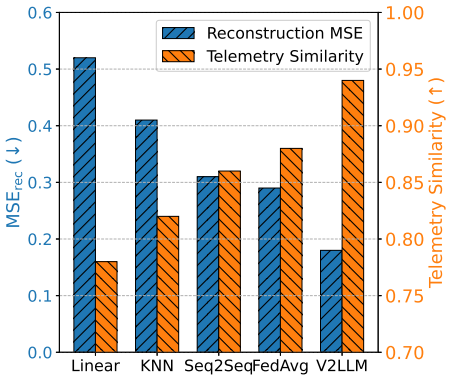


Fig. 8: MSE$_{rec}$ and Telemetry similarity.

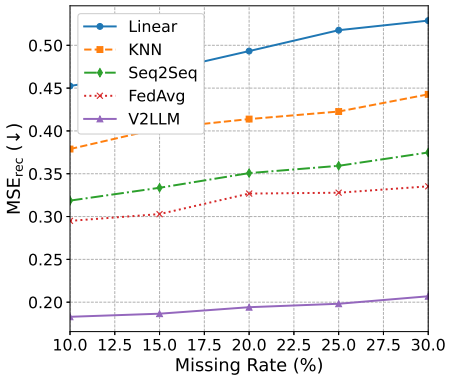


Fig. 9: MSE$_{rec}$ vs. missing rate.

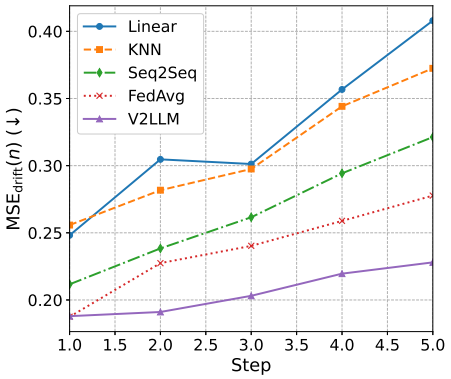


Fig. 10: MSE$_{drift}$ vs. prediction step.

- **Autoregressive Drift MSE (Multi-step)**: Measures error accumulation over multi-step predictions, capturing generation stability over time.

$$\text{MSE}_{\text{drift}}(n) = \frac{1}{n}\sum_{\ell=1}^{n} \|\hat{\mathbf{x}}_i(t+\ell) - \mathbf{x}_i(t+\ell)\|_2^2 \tag{48}$$

- **Forgetting Measure (FM)**: Quantifies performance degradation over time in continual learning, indicating knowledge retention.

$$\text{FM} = \frac{1}{T}\sum_{t=1}^{T}\left(\max_{\tau<t} \text{Acc}_\tau - \text{Acc}_t\right) \tag{49}$$

VIII. Simulation Results

We present key simulation results highlighting the effectiveness of the proposed *V2LLM* framework.

1) Reconstruction Quality: Fig. 8 compares telemetry reconstruction performance across baselines. V2LLM achieves the lowest reconstruction MSE and highest cosine similarity, indicating its effectiveness in restoring both the magnitude and directional structure of high-dimensional telemetry vectors. While Seq2Seq and FedAvg-LSTM offer moderate improvements, they struggle under non-i.i.d. and shifting contexts due to limited environmental grounding. Linear interpolation and KNN perform poorly under temporal misalignment and sparse input. By leveraging DT-informed retrieval and structured prompts, V2LLM enables environment-aligned recovery even under multi-step gaps, ensuring high-fidelity reconstruction essential for downstream V2X tasks.

2) Impact of Telemetry Loss Rate on Recovery: Fig. 9 presents reconstruction error MSE_{rec} versus telemetry missing rate. As the rate increases from 10% to 30%, all baselines show degradation. Traditional methods exhibit steep error growth due to limited generalization across long gaps. Sequence models like Seq2Seq and FedAvg are more robust but still suffer under data sparsity and distribution drift. In contrast, V2LLM consistently maintains low error, validating the effectiveness of RAG-enhanced prompting and DT-guided retrieval in handling severe loss conditions.

3) Multi-step Generation Stability: Fig. 10 shows $\text{MSE}_{\text{drift}}(n)$ under increasing prediction steps. All methods exhibit growing error due to compounding deviations from autoregressive generation. Linear and KNN degrade rapidly without feedback mechanisms, while Seq2Seq and FedAvg decline beyond 3 steps due to memory limitations and federated drift. V2LLM sustains lower drift across horizons, benefiting from DT-aware prompt construction and RAG-based conditioning, enabling coherent multi-step recovery critical for long-duration telemetry loss.

4) Sequential Task Retention and Forgetting: Fig. 11 tracks the forgetting measure $\text{FM}(t)$ across sequential learning tasks. V2LLM achieves the flattest curve, demonstrating strong retention of past knowledge and minimal degradation over time. This resilience stems from its FCL design, which integrates regularized local updates and global aggregation. In contrast, FedAvg and Seq2Seq exhibit moderate forgetting, while Linear and KNN degrade sharply due to a lack of adaptive memory. These results confirm V2LLM's ability to generalize across time-varying telemetry and sustain performance under evolving V2X conditions.

5) Effect of Prompt Length on Reconstruction Accuracy: Fig. 12 illustrates the impact of prompt length, defined as the total number of serialized input tokens on the reconstruction quality of missing telemetry in V2X networks. As LAMs like GPT-2 are sensitive to token budget constraints, this analysis highlights how different model configurations perform under varying input capacities. The full version of our proposed V2LLM framework, which includes retrieval-augmented context and DT embeddings, consistently achieves the lowest reconstruction error across all prompt lengths. In contrast, the variant without retrieval shows noticeably higher MSE, particularly when the token budget is limited, indicating the critical role of memory augmentation. We also include two truncated LSTM-based baselines, Seq2Seq and FedAvg, each restricted to use the same number of input tokens as the LLM. These models exhibit relatively flat or saturated performance, suggesting their limited capacity to benefit from increased context length. Overall, the result underscores the efficiency of prompt-based structured reasoning in V2LLM and its robustness under realistic LAM constraints, where prompt compression and context prioritization are key to maintaining accuracy.

6) Classification Accuracy and Macro F1 Comparison: Fig. 13 illustrates the comparative classification performance of the proposed V2LLM framework and baseline methods using two key metrics: accuracy and macro-averaged F1 score. Accuracy measures the proportion of correctly predicted vehicular fault states, while the macro F1 score accounts for per-class precision and recall, offering a balanced view under class imbalance. Fig. 13 shows that V2LLM achieves the highest scores on both metrics, indicating its superior capability to generalize across

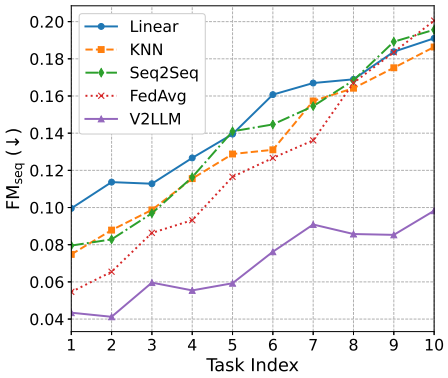


Fig. 11: FM vs. task sequence.

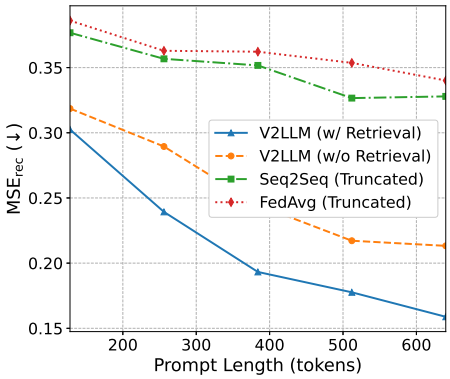


Fig. 12: Impact of prompt length on $\mathrm{MSE}_{\mathrm{rec}}$.

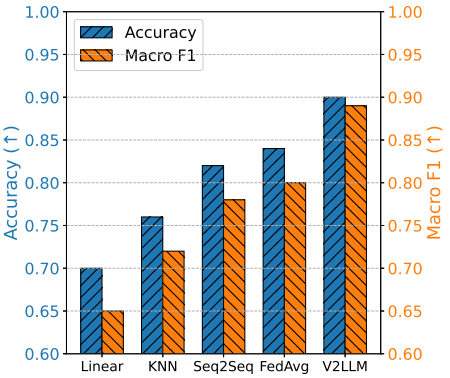


Fig. 13: Accuracy and macro F1 comparison.

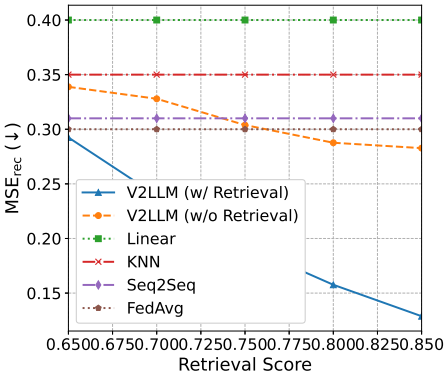


Fig. 14: Retrieval similarity vs. $\mathrm{MSE}_{\mathrm{rec}}$.

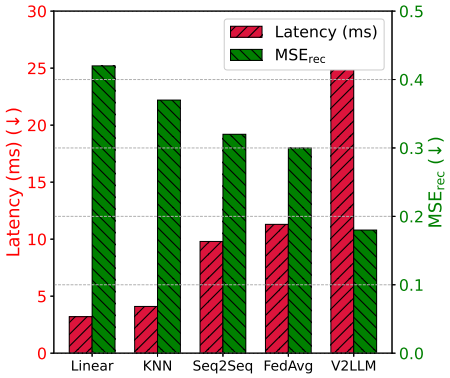


Fig. 15: Latency vs. $\mathrm{MSE}_{\mathrm{rec}}$ trade-off.

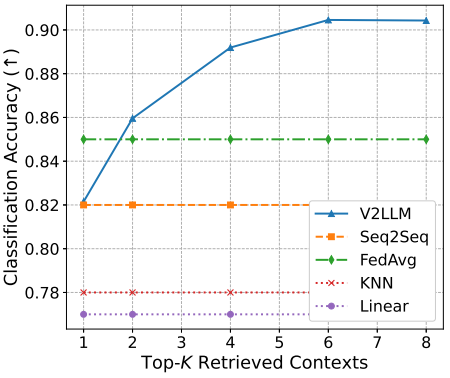


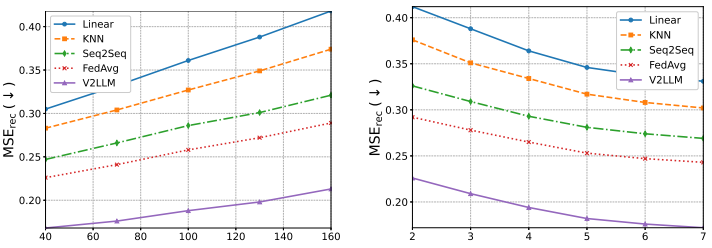
Fig. 16: Top-K retrieval ablation.

diverse fault types and maintain consistent decision quality. Unlike interpolation-based baselines (e.g., Linear, KNN) which lack temporal and semantic understanding, or sequence models (e.g., Seq2Seq, FedAvg) which struggle under data sparsity or distribution shifts, V2LLM integrates digital-twin-aware retrieval, structured prompting, and FCL. This allows it to recover more informative telemetry sequences and adapt classification boundaries over time.

7) Retrieval Similarity vs. Reconstruction Error: Fig. 14 illustrates the relationship between retrieval similarity and reconstruction accuracy, quantified via $\mathrm{MSE}_{\mathrm{rec}}$, across various methods. The x-axis represents the average similarity score between the current telemetry context and retrieved historical sequences (i.e., the retrieval score), while the y-axis indicates the reconstruction error. The V2LLM (with retrieval) curve exhibits a strong inverse correlation: as the retrieval score increases, the reconstruction MSE consistently decreases. This trend validates the effectiveness of the memory-augmented prompting strategy, where better-matched historical trajectories enable the LLM to generate more accurate telemetry completions. In contrast, V2LLM (without retrieval) shows only marginal improvement with the retrieval score since it does not condition on external memory, its MSE curve is flatter and less responsive. Baseline models such as linear interpolation, KNN, Seq2Seq, and FedAvg appear as flat lines in the plot. This is because these models do not incorporate any retrieval-augmented mechanism and therefore remain unaffected by variations in the retrieval score. Their reconstruction performance is governed solely by their internal architectures and fixed context modeling, irrespective of historical similarity. Even under lower retrieval similarity regimes, V2LLM maintains bounded reconstruction error due to the complementary conditioning provided by telemetry history and DT embeddings, illustrating robustness against weak or partially mismatched memory retrieval.

8) Latency-Accuracy Trade-off: Fig. 15 illustrates the trade-off between inference latency and reconstruction accuracy ($\mathrm{MSE}_{\mathrm{rec}}$) for various models used in vehicular telemetry recovery. Models such as linear and KNN achieve low inference latency due to their lightweight nature but exhibit high reconstruction error, indicating limited effectiveness in complex, temporally correlated data scenarios. Seq2Seq and FedAvg offer improved accuracy with moderate computational cost, benefiting from sequential modeling and federated updates, respectively. The proposed V2LLM achieves the lowest $\mathrm{MSE}_{\mathrm{rec}}$ owing to its structured prompting and digital-twin-aware context integration. Nevertheless, this comes at the expense of higher latency, attributed to the autoregressive nature and token-level processing of the transformer-based language model. This analysis highlights the core trade-off between model complexity and inference efficiency in telemetry restoration tasks.

9) Top-K Retrieval Ablation: Fig. 16 illustrates the effect of varying the number of retrieved memory contexts K on classification accuracy across different frameworks. The proposed V2LLM framework shows a clear performance gain as K increases from 1 to 4, leveraging richer contextual information from memory-augmented prompting. Beyond $K = 4$, the accuracy begins to saturate, suggesting diminishing returns as the prompt reaches token budget limits and older retrieved examples



(a) Impact of connected vehicle density. (b) Impact of roadside unit density.

Fig. 17: Impact of network density on reconstruction performance.

provide limited additional utility. In contrast, baseline models such as Seq2Seq, FedAvg, KNN, and linear remain invariant to K, as they do not utilize retrieval-based context, resulting in flat accuracy curves. This validates the design choice in V2LLM to include a moderate number of top-K memory samples, balancing prompt richness with efficiency.

10) Impact of Vehicle and Infrastructure Density: Fig. 17a evaluates reconstruction performance under varying numbers of connected vehicles. As network density increases, all methods exhibit a rise in $\mathrm{MSE}_{\mathrm{rec}}$, reflecting the increased complexity of telemetry dynamics and the higher likelihood of temporally inconsistent or incomplete observations. However, V2LLM maintains a consistently lower error across all operating points, indicating stronger robustness to dense traffic conditions. This behavior is attributed to its DT-aware retrieval mechanism and context-conditioned prompt formulation, which enable stable inference even when the diversity and variability of telemetry patterns increase.

Fig. 17b examines the effect of increasing the number of roadside units. As infrastructure density improves, all models benefit from enhanced spatial coverage and reduced telemetry discontinuities. Notably, V2LLM exhibits a more pronounced reduction in reconstruction error compared to the baselines. This gain arises from the improved quality of DT state estimation and more relevant retrieval contexts, both of which directly influence the conditioning of the LLM. The results demonstrate that V2LLM effectively leverages additional infrastructure to refine both contextual grounding and generation accuracy.

IX. Conclusion & Future Work

This paper presented V2LLM, a unified, modular framework for resilient telemetry recovery and distributed fault diagnosis in V2X networks facing bursty data loss and evolving spatiotemporal conditions. The proposed architecture integrates five key components: DT-based state estimation, hybrid memory retrieval, structured prompt construction, autoregressive LLM-based reconstruction, and FCL for classification. The framework combines DT-guided retrieval, structured prompting, autoregressive reconstruction, and federated continual learning for telemetry recovery and distributed fault diagnosis under incomplete and evolving V2X observations. The use of autoregressive prompting, memory augmentation, and continual federated updates supports scalable and privacy-preserving telemetry analysis in distributed vehicular systems.

In future work, we aim to extend V2LLM toward (i) multimodal telemetry modeling through the fusion of structured telemetry with visual sensing modalities such as camera, LiDAR, and radar data using multimodal foundation models; (ii) real-time deployment on constrained hardware via parameter-efficient, quantized, or distilled LLMs; and (iii) privacy-preserving continual learning under differential privacy and secure aggregation. Additional directions include evaluating V2LLM in 6G semantic communication testbeds and extending it to other cyber-physical domains such as UAV swarms and intelligent transportation systems.

References

[1] J. Qiu, L. Li, J. Sun, J. Peng, P. Shi, R. Zhang, Y. Dong, K. Lam, F. P.-W. Lo, B. Xiao *et al.*, "Large AI models in health informatics: Applications, challenges, and the future," *IEEE J. Biomed. Health Inform.*, vol. 27, no. 12, pp. 6074–6087, 2023.

[2] Z. Chen, Z. Zhang, and Z. Yang, "Big AI models for 6G wireless networks: Opportunities, challenges, and research directions," *IEEE Wirel. Commun.*, vol. 31, no. 5, pp. 164–172, 2024.

[3] B. Hazarika, K. Singh, S. Biswas, and C.-P. Li, "DRL-based resource allocation for computation offloading in IoV networks," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 11, pp. 8027–8038, 2022.

[4] N. H. Hussein, C. T. Yaw, S. P. Koh, S. K. Tiong, and K. H. Chong, "A comprehensive survey on vehicular networking: Communications, applications, challenges, and upcoming research directions," *IEEE Access*, vol. 10, pp. 86 127–86 180, 2022.

[5] C. A. Gómez-Vega, J. Cardenas, J. C. Ornelas-Lizcano, C. A. Gutiérrez, M. Cardenas-Juarez, J. M. Luna-Rivera, and R. M. Aguilar-Ponce, "Doppler spectrum measurement platform for narrowband V2V channels," *IEEE Access*, vol. 10, pp. 27 162–27 184, 2022.

[6] J. Ye, Y. Jiang, and X. Ge, "Deep reinforcement learning assisted hybrid precoding for V2I communications with doppler shift and delay spread," *IEEE Trans. Veh. Technol.*, vol. 73, no. 7, pp. 10 265–10 279, 2024.

[7] H. Guo, L.-l. Rui, and Z.-p. Gao, "V2V task offloading algorithm with LSTM-based spatiotemporal trajectory prediction model in SVCNs," *IEEE Trans. Veh. Technol.*, vol. 71, no. 10, pp. 11 017–11 032, 2022.

[8] X. Ouyang and M. Srivastava, "LLMSense: Harnessing llms for high-level reasoning over spatiotemporal sensor traces," in *Proc. IEEE SenSys-ML 2024*, pp. 9–14.

[9] Y. Hu, F. Wang, D. Ye, M. Wu, J. Kang, and R. Yu, "LLM-based misbehavior detection architecture for enhanced traffic safety in connected autonomous vehicles," *IEEE Trans. Veh. Technol.*, pp. 1–13, 2025.

[10] G. O. Boateng, H. Sami, A. Alagha, H. Elmekki, A. Hammoud, R. Mizouni, A. Mourad, H. Otrok, J. Bentahar, S. Muhaidat, C. Talhi, Z. Dziong, and M. Guizani, "A survey on large language models for communication, network, and service management: Application insights, challenges, and future directions," *IEEE Commun. Surv. Tutor.*, pp. 1–1, 2025.

[11] X. Cao, G. Nan, H. Guo, H. Mu, L. Wang, Y. Lin, Q. Zhou, J. Li, B. Qin, Q. Cui, X. Tao, H. Fang, H. Du, and T. Q. Quek, "Exploring LLM-based multi-agent situation awareness for zero-trust space-air-ground integrated network," *IEEE J. Sel. Areas Commun.*, pp. 1–1, 2025.

[12] X. Dai, C. Guo, Y. Tang, H. Li, Y. Wang, J. Huang, Y. Tian, X. Xia, Y. Lv, and F.-Y. Wang, "VistaRAG: Toward safe and trustworthy autonomous driving through retrieval-augmented generation," *IEEE Trans. Intell. Veh.*, vol. 9, no. 4, pp. 4579–4582, 2024.

[13] M. Hindi, L. Mohammed, O. Maaz, and A. Alwarafy, "Enhancing the precision and interpretability of retrieval-augmented generation (RAG) in legal technology: A survey," *IEEE Access*, vol. 13, pp. 46 171–46 189, 2025.

[14] Y. Pang, B. Yang, H. Tu, Y. Cao, and Z. Zhang, "Language models can see better: Visual contrastive decoding for LLM multimodal reasoning," in *Proc. IEEE ICASSP 2025*, Hyderabad, India, Apr. 2025, pp. 1–5.

[15] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat *et al.*, "GPT-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[16] Y. Wang, M. M. Afzal, Z. Li, J. Zhou, C. Feng, S. Guo, and T. Q. S. Quek, "Large language model as a catalyst: A paradigm shift in base station siting optimization," *IEEE Trans. Cogn. Commun. Netw.*, pp. 1–1, 2025.

[17] B. Hazarika, K. Singh, C.-P. Li, A. Schmeink, and K. F. Tsang, "RADiT: Resource allocation in digital twin-driven UAV-aided internet of vehicle networks," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 11, pp. 3369–3385, 2023.

[18] B. Hazarika, K. Singh, A. Paul, and T. Q. Duong, "Hybrid machine learning approach for resource allocation of digital twin in UAV-aided internet-of-vehicles networks," *IEEE Trans. Intell. Veh.*, vol. 9, no. 1, pp. 2923–2939, 2024.

[19] B. Hazarika and K. Singh, "AFL-DMAAC: Integrated resource management and cooperative caching for URLLC-IoV networks," *IEEE Trans Intell Veh*, vol. 9, no. 6, pp. 5101–5117, 2024.

[20] B. Hazarika, P. Saikia, K. Singh, and C.-P. Li, "Enhancing vehicular networks with hierarchical O-RAN slicing and federated DRL," *IEEE Trans. Green Commun. Netw.*, vol. 8, no. 3, pp. 1099–1117, 2024.

[21] P. Singh, B. Hazarika, K. Singh, W.-J. Huang, and T. Q. Duong, "GenAI-enhanced federated multiagent DRL for digital-twin-assisted IoV networks," *IEEE Internet of Things J.*, vol. 12, no. 5, pp. 4834–4851, 2025.

[22] X. Cao, G. Zhu, J. Xu, and S. Cui, "Transmission power control for over-the-air federated averaging at network edge," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 5, pp. 1571–1586, 2022.

[23] H. T. Nguyen, V. Sehwag, S. Hosseinalipour, C. G. Brinton, M. Chiang, and H. Vincent Poor, "Fast-convergent federated learning," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 1, pp. 201–218, 2021.

[24] T. Xiang, Y. Bi, X. Chen, Y. Liu, B. Wang, X. Shen, and X. Wang, "Federated learning with dynamic epoch adjustment and collaborative training in mobile edge computing," *IEEE Trans. Mob. Comput.*, vol. 23, no. 5, pp. 4092–4106, 2024.

[25] X. Yang, H. Yu, X. Gao, H. Wang, J. Zhang, and T. Li, "Federated continual learning via knowledge fusion: A survey," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 8, pp. 3832–3850, 2024.

[26] L. Specia *(*et al.*)*, "Grounded sequence to sequence transduction," *IEEE J. Sel. Top. Signal Process*, vol. 14, no. 3, pp. 577–591, 2020.

[27] S. Gebreab, A. Musamih, K. Salah, R. Jayaraman, and D. Boscovic, "Accelerating digital twin development with generative AI: A framework for 3D modeling and data integration," *IEEE Access*, vol. 12, pp. 185 918–185 936, 2024.

[28] S. Iqbal, X. Zhong, M. A. Khan, Z. Wu, D. A. AlHammadi, W. Liu, and I. A. Choudhry, "Hierarchical continual learning for domain-knowledge retention in healthcare federated learning," *IEEE Trans. Consum. Electron.*, vol. 71, no. 2, pp. 5025–5035, 2025.

[29] K. Zhang, Q. Yang, C. Li, X. Sun, and J. Chen, "Missing data recovery methods on multivariate time series in IoT: A comprehensive survey," *IEEE Commun. Surv. Tutor.*, pp. 1–1, 2025.

[30] X. Ruan, S. Fu, H. Jia, K. L. Mathis, C. A. Thiels, S. M. Gavin, P. M. Wilson, C. B. Storlie, and H. Liu, "Gated recurrent unit with decay has real-time capability for postoperative ileus surveillance and offers cross-hospital transferability," *Commun. Med.*, vol. 5, no. 1, p. 331, 2025.

[31] J. Rodríguez-Ortega, R. Khaldi, D. Alcaraz-Segura, and S. Tabik, "Bidirectional recurrent imputation and abundance estimation of LULC classes with MODIS multispectral time-series and geo-topographic and climatic data," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 17, pp. 4626–4645, 2024.

[32] L. Qian, H. L. Ellis, T. Wang, J. Wang, R. Mitra, R. Dobson, and Z. Ibrahim, "How deep is your guess? a fresh perspective on deep learning for medical time-series imputation," *IEEE J. Biomed. Health Inform.*, vol. 29, no. 9, pp. 6814–6829, 2025.

[33] M. Landers, T. W. Killian, T. Hartvigsen, and A. Doryab, "SAINT: Attention-based modeling of sub-action dependencies in multi-action policies," *arXiv preprint arXiv:2505.12109*, 2025.

[34] Y. Wu, C. Chen, J. Shi, D. Yue, and C. Bo, "A hybrid framework for lithium-ion battery capacity prediction using timesnet and transformer," in *Proc. IEEE ISF 2025*, Yinchuan, China, Apr. 2025, pp. 1–7.

[35] C. Lyu and C. Antoniou, "A diffusion-based expectation–maximization framework for probabilistic traffic data imputation," *IEEE Internet Things J.*, vol. 12, no. 20, pp. 43 227–43 240, 2025.

[36] Y. He, L. Li, X. Zhu, and K. L. Tsui, "Multi-graph convolutional-recurrent neural network (MGC-RNN) for short-term forecasting of transit passenger flow," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 10, pp. 18 155–18 174, 2022.

[37] Y. Liang and Z. Zhao, "NetTraj: A network-based vehicle trajectory prediction model with directional representation and spatiotemporal attention mechanisms," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 14 470–14 481, 2022.

[38] H. Manjunatha, A. Pak, D. Filev, and P. Tsiotras, "Karnet: Kalman filter augmented recurrent neural network for learning world models in autonomous driving tasks," *arXiv preprint arXiv:2305.14644*, 2023.

[39] M. Singh, J. Challagundla, P. Karnal, G. Ganapathy, V. Shah, and R. Arora, "LLMs as master forgers: Generating synthetic time series data for manufacturing," in *Proc. IEEE ICICML 2024)*, Shenzhen, China, Nov. 2024, pp. 2053–2059.

[40] M. Song, "Enhancing RAG performance by representing hierarchical nodes in headers for tabular data," *IEEE Access*, vol. 13, pp. 85 072–85 083, 2025.

[41] Y. Ahn, S.-G. Lee, J. Shim, and J. Park, "Retrieval-augmented response generation for knowledge-grounded conversation in the wild," *IEEE Access*, vol. 10, pp. 131 374–131 385, 2022.

[42] Y. Li, X. Qiu, Y. Fu, J. Chen, T. Qian, X. Zheng, D. P. Paudel, Y. Fu, X. Huang, L. Van Gool *et al.*, "Domain-RAG: Retrieval-guided compositional image generation for cross-domain few-shot object detection," *arXiv preprint arXiv:2506.05872*, 2025.

[43] E. Zhu, S. Wang, C. Liu, and J. Wang, "Adaptive tokenization transformer: Enhancing irregularly sampled multivariate time-series analysis," *IEEE Internet Things J.*, vol. 12, no. 19, pp. 39 237–39 246, 2025.

[44] M. Xiao, Z. Jiang, L. Qian, Z. Chen, Y. He, Y. Xu, Y. Jiang, D. Li, R.-L. Weng, M. Peng *et al.*, "Retrieval-augmented large language models for financial time series forecasting," *arXiv preprint arXiv:2502.05878*, 2025.

[45] Y. Li, L. Liu, S. Deng, H. Qin, M. A. El-Yacoubi, and G. Zhou, "Memory-augmented autoencoder based continuous authentication on smartphones with conditional transformer GANs," *IEEE Trans. Mob. Comput.*, vol. 23, no. 5, pp. 4467–4482, 2024.

[46] Y. Cai, S. Chen, Y. Wu, Y. Huang, J. Feng, and Z. Ou, "The 2nd futuredial challenge: Dialog systems with retrieval augmented generation (futuredial-RAG)," in *Proc. IEEE SLT 2024*, Macau, China, Dec. 2024, pp. 1091–1098.

[47] Z. Dong, J. Kong, W. Yan, X. Wang, and H. Li, "Multivariable high-dimension time-series prediction in SIoT via adaptive dual-graph-attention encoder-decoder with global bayesian optimization," *IEEE Internet Things J.*, vol. 11, no. 20, pp. 32 956–32 968, 2024.

[48] X. Liu, S. Gao, B. Liu, X. Cheng, and L. Yang, "LLM4WM: Adapting LLM for wireless multi-tasking," *IEEE Trans. Mach. Learn. Commun. Netw.*, vol. 3, pp. 835–847, 2025.

[49] W. Lee and J. Park, "LLM-empowered resource allocation in wireless communications systems," *IEEE Access*, vol. 14, pp. 15 260–15 272, 2026.

[50] M. Geng, J. Li, C. Li, N. Xie, X. Chen, and D.-H. Lee, "Adaptive and simultaneous trajectory prediction for heterogeneous agents via transferable hierarchical transformer network," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 10, pp. 11 479–11 492, 2023.

[51] A. Azeem, Z. Li, A. Siddique, Y. Zhang, and D. Cao, "Memory-augmented detection transformer for few-shot object detection in remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, pp. 1–21, 2025.

[52] H. Xu, J. Wu, Q. Pan, X. Guan, and M. Guizani, "A survey on digital twin for industrial internet of things: Applications, technologies and tools," *IEEE Commun. Surv. Tutor.*, vol. 25, no. 4, pp. 2569–2598, 2023.

[53] L. Zhang, Z. Wu, H. Xu, D. Niyato, C. S. Hong, and Z. Han, "Digital twin-driven federated learning for converged computing and networking at the edge," *IEEE Netw.*, vol. 39, no. 2, pp. 20–28, 2025.

[54] W.-B. Kou, Q. Lin, M. Tang, R. Ye, S. Wang, G. Zhu, and Y.-C. Wu, "Fast-convergent and communication-alleviated heterogeneous hierarchical federated learning in autonomous driving," *IEEE Trans. Intel. Transp.*, vol. 26, no. 7, pp. 10 496–10 511, 2025.

[55] Z. Jin, C. Yang, Y. Ye, L. Zhang, J. Shen, and J. Su, "Mobility-aware semi-asynchronous federated learning for vehicular networks," *IEEE Trans. Veh. Technol.*, pp. 1–12, 2025.

[56] Y. Li, H. Wang, Y. Qi, W. Liu, and R. Li, "Re-Fed+: A better replay strategy for federated incremental learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 7, pp. 5489–5500, 2025.

[57] Y. Li, Y. Liu, B. Guo, D. Wang, H. Li, N. Li, Y. Wang, H. Luo, and Z. Yu, "Cross-f ^{2} scil: A federated few-shot class incremental learning method for cross mobile edge network environments," *IEEE Trans. Serv. Comput.*, pp. 1–14, 2025.

[58] J. Chen, J. He, J. Tang, W. Li, and Z. Yin, "Knowledge efficient federated continual learning for industrial edge systems," *IEEE Trans. Neural. Netw.*, vol. 12, no. 3, pp. 2107–2120, 2025.

[59] Z. Zhong, W. Bao, J. Wang, J. Chen, L. Lyu, and W. Yang Bryan Lim, "SacFL: Self-adaptive federated continual learning for resource-constrained end devices," *IEEE Trans. Neural. Netw.*, vol. 36, no. 9, pp. 17 169–17 183, 2025.

[60] M. S. Irfan, S. Dasgupta, and M. Rahman, "Toward transportation digital twin systems for traffic safety and mobility: A review," *IEEE Internet Things J.*, vol. 11, no. 14, pp. 24 581–24 603, 2024.

[61] K. Wu, P. Li, Y. Cheng, S. T. Parker, B. Ran, D. A. Noyce, and X. Ye, "A digital twin framework for physical-virtual integration in V2X-enabled connected vehicle corridors," *IEEE Trans. Intell. Transp. Syst.*, vol. 26, no. 9, pp. 14 407–14 420, 2025.

[62] C. Campolo, G. Genovese, A. Molinaro, B. Pizzimenti, G. Ruggeri, and D. M. Zappalà, "An edge-based digital twin framework for connected and autonomous vehicles: Design and evaluation," *IEEE Access*, vol. 12, pp. 46 290–46 303, 2024.

[63] X. Wang, M. Hao, M. Wu, C. Shang, R. Yu, J. Kang, Z. Xiong, and Y. Wu, "Digital-twin-assisted safety control for connected automated vehicles in mixed-autonomy traffic," *IEEE Internet Things J.*, vol. 12, no. 1, pp. 472–487, 2025.

[64] H. Kaur and M. Bhatia, "Digital-twin-driven performance assessment for IoT-integrated autonomous driving systems," *IEEE Internet Things J.*, vol. 12, no. 4, pp. 4078–4085, 2025.

[65] M. Sheraz, T. C. Chuah, Y. L. Lee, M. M. Alam, A. Al-Habashna, and Z. Han, "A comprehensive survey on revolutionizing connectivity through artificial intelligence-enabled digital twin network in 6G," *IEEE Access*, vol. 12, pp. 49 184–49 215, 2024.

[66] R. Poorzare, D. N. Kanellopoulos, V. K. Sharma, P. Dalapati, and O. P. Waldhorst, "Network digital twin toward networking, telecommunications, and traffic engineering: A survey," *IEEE Access*, vol. 13, pp. 16 489–16 538, 2025.

[67] Y. Xia, Y. Chen, Y. Zhao, L. Kuang, X. Liu, J. Hu, and Z. Liu, "FCLLM-DT: Enpowering federated continual learning with large language models for digital-twin-based industrial IoT," *IEEE Internet Things J.*, vol. 12, no. 6, pp. 6070–6081, 2025.

[68] C. Tan, P. Yu, Z. Qu, L. Zhang, W. Li, X. Qiu, and S. Guo, "Energy-efficient federated learning training optimization for digital twin driven 6G air-ground integrated vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 26, no. 10, pp. 18 116–18 128, 2025.

[69] A. Jayanthiladevi, V. P. Mishra, V. Rishiwal, S. Maurya, A. K. Sharma, and U. Agarwal, "HDTwin-E5G: Ai-driven digital twin and hybrid deep learning framework for energy-efficient 5G networks," *IEEE Trans. Consum. Electron.*, pp. 1–1, 2026.

[70] S. A. Khan, A. Kavak, K. Kucuk, and S. A. Chaudhry, "Digital twins for 6G V2X communication in ICE ecosystem," *IEEE Consum. Electron. Mag.*, pp. 1–8, 2025.

[71] V. P. Chellapandi, L. Yuan, C. G. Brinton, S. H. Żak, and Z. Wang, "Federated learning for connected and automated vehicles: A survey of existing approaches and challenges," *IEEE Trans. Intell. Veh*, vol. 9, no. 1, pp. 119–137, 2024.

[72] M. Tao, L. Liao, Y. Zhang, L. Liu, G. Min, D. Niyato, and S. Dustdar, "EDT-SaFL: Semi-asynchronous federated learning for edge digital twin in industrial internet-of-things," *IEEE Trans. Mob. Comput.*, pp. 1–17, 2025.

[73] L. Tang, M. Wen, Z. Shan, L. Li, Q. Liu, and Q. Chen, "Digital twin-enabled efficient federated learning for collision warning in intelligent driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 3, pp. 2573–2585, 2024.

[74] K. Sun, J. Wu, A. K. Bashir, J. Li, H. Xu, Q. Pan, and Y. D. Al-Otaibi, "Personalized privacy-preserving distributed artificial intelligence for digital-twin-driven vehicle road cooperation," *IEEE Internet Things J.*, vol. 11, no. 22, pp. 35 902–35 916, 2024.

[75] M. Na, S. Cho, F. Solat, T. Na, and J. Lee, "Energy-efficient hybrid federated and centralized learning for edge-based wireless traffic prediction in aerial networks," *IEEE Access*, vol. 12, pp. 130 983–130 994, 2024.

[76] Q. Zhang, H. Wen, Y. Liu, S. Chang, and Z. Han, "Federated-reinforcement-learning-enabled joint communication, sensing, and computing resources allocation in connected automated vehicles networks," *IEEE Internet Things J.*, vol. 9, no. 22, pp. 23 224–23 240, 2022.

[77] Y. Li, W. Xu, Y. Qi, H. Wang, R. Li, and S. Guo, "SR-FDIL: Synergistic replay for federated domain-incremental learning," *IEEE Transactions on Parallel and Distributed Systems*, vol. 35, no. 11, pp. 1879–1890, 2024.

[78] M. A. Helcig and S. Nastic, "FedCCL: Federated clustered continual learning framework for privacy-focused energy forecasting," in *Proc. IEEE ICFEC 2025*, Tromsø, Norway, May 2025, pp. 50–57.

[79] R. Krajewski, J. Bock, L. Kloeker, and L. Eckstein, "The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems," in *Proc. IEEE ITSC 2018*, Maui, Hawaii, Nov. 2018, pp. 2118–2125.

[80] M. Khakzar, A. Rakotonirainy, A. Bond, and S. G. Dehkordi, "A dual learning model for vehicle trajectory prediction," *IEEE Access*, vol. 8, pp. 21 897–21 908, 2020.

[81] R. J. Little and D. B. Rubin, *Statistical analysis with missing data*. John Wiley & Sons, 2019.

[82] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," *Adv. Neural Inf. Process. Syst.*, vol. 33, pp. 9459–9474, 2020.

[83] O. Davies, "Statistical forecasting for inventory control," 1960.

[84] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Adv. Neural Inf. Process. Syst.*, vol. 30, 2017.

[85] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.